Package ‘RCy3’

March 23, 2024

Type Package

Title Functions to Access and Control Cytoscape

Version 2.22.1

Date 2023-11-06

Imports httr, methods, RJSONIO, XML, utils, BiocGenerics, stats,
  graph, fs, uuid, stringi, glue, RCurl, base64url, base64enc,
  IRkernel, IRdisplay, RColorBrewer, gplots

Suggests BiocStyle, knitr, rmarkdown, igraph, grDevices

SystemRequirements Cytoscape (>= 3.7.1), CyREST (>= 3.8.0)

Description Vizualize, analyze and explore networks using Cytoscape via R. Anything you can do using
  the graphical user interface of Cytoscape, you can now do with a single RCy3 function.

License MIT + file LICENSE

URL https://github.com/cytoscape/RCy3

BugReports https://github.com/cytoscape/RCy3/issues

LazyLoad yes

biocViews Visualization, GraphAndNetwork, ThirdPartyClient, Network

NeedsCompilation no

VignetteBuilder knitr

RoxygenNote 7.2.3

git_url https://git.bioconductor.org/packages/RCy3

git_branch RELEASE_3_18

git_last_commit cd4cb1b

git_last_commit_date 2023-11-06

Repository Bioconductor 3.18

Date/Publication 2024-03-22

Author Alex Pico [aut, cre] (<https://orcid.org/0000-0001-5706-2163>),
  Tanja Muetze [aut],
  Paul Shannon [aut],
R topics documented:

- .exportShowImage
- .getDefaultSandbox
- .getRequester
- addAnnotationBoundedText
- addAnnotationImage
- addAnnotationShape
- addAnnotationText
- addCyEdges
- addCyNodes
- AddToGroup
- analyzeNetwork
- applyFilter
- bundleEdges
- checkNotebookIsRunning
- checkRunningRemote
- clearEdgeBends
- clearEdgePropertyBypass
- clearNetworkCenterBypass
- clearNetworkPropertyBypass
- clearNetworkZoomBypass
- clearNodeOpacityBypass
- clearNodePropertyBypass
- clearSelection
- cloneNetwork
- closeSession
- collapseGroup
- commandEcho
- commandOpenDialog
- commandPause
- commandQuit
- commandRunFile
- commandsAPI
- commandsGET
- commandsHelp
- commandSleep
- commandsPOST
- commandsRun
- copyVisualStyle
<table>
<thead>
<tr>
<th>R topics documented:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>createColumnFilter</td>
<td>43</td>
</tr>
<tr>
<td>createCompositeFilter</td>
<td>44</td>
</tr>
<tr>
<td>createCytoscapejsFromNetwork</td>
<td>45</td>
</tr>
<tr>
<td>createDegreeFilter</td>
<td>46</td>
</tr>
<tr>
<td>createGraphFromNetwork</td>
<td>47</td>
</tr>
<tr>
<td>createGroup</td>
<td>48</td>
</tr>
<tr>
<td>createGroupByColumn</td>
<td>49</td>
</tr>
<tr>
<td>createIgraphFromNetwork</td>
<td>50</td>
</tr>
<tr>
<td>createNetworkFromCytoscapejs</td>
<td>51</td>
</tr>
<tr>
<td>createNetworkFromDataFrames</td>
<td>51</td>
</tr>
<tr>
<td>createNetworkFromGraph</td>
<td>53</td>
</tr>
<tr>
<td>createNetworkFromIgraph</td>
<td>54</td>
</tr>
<tr>
<td>createSubnetwork</td>
<td>55</td>
</tr>
<tr>
<td>createView</td>
<td>56</td>
</tr>
<tr>
<td>createVisualStyle</td>
<td>57</td>
</tr>
<tr>
<td>cybrowserClose</td>
<td>58</td>
</tr>
<tr>
<td>cybrowserDialog</td>
<td>59</td>
</tr>
<tr>
<td>cybrowserHide</td>
<td>60</td>
</tr>
<tr>
<td>cybrowserList</td>
<td>60</td>
</tr>
<tr>
<td>cybrowserSend</td>
<td>61</td>
</tr>
<tr>
<td>cybrowserShow</td>
<td>62</td>
</tr>
<tr>
<td>cybrowserVersion</td>
<td>63</td>
</tr>
<tr>
<td>cyrestAPI</td>
<td>63</td>
</tr>
<tr>
<td>cyrestDELETE</td>
<td>64</td>
</tr>
<tr>
<td>cyrestGET</td>
<td>64</td>
</tr>
<tr>
<td>cyrestPOST</td>
<td>65</td>
</tr>
<tr>
<td>cyrestPUT</td>
<td>66</td>
</tr>
<tr>
<td>cytoscapeApiVersions</td>
<td>67</td>
</tr>
<tr>
<td>cytoscapeFreeMemory</td>
<td>67</td>
</tr>
<tr>
<td>cytoscapeMemoryStatus</td>
<td>68</td>
</tr>
<tr>
<td>cytoscapeNumberOfCores</td>
<td>69</td>
</tr>
<tr>
<td>cytoscapePing</td>
<td>69</td>
</tr>
<tr>
<td>cytoscapeVersionInfo</td>
<td>70</td>
</tr>
<tr>
<td>deleteAllNetworks</td>
<td>70</td>
</tr>
<tr>
<td>deleteAllVisualStyle</td>
<td>71</td>
</tr>
<tr>
<td>deleteAnnotation</td>
<td>71</td>
</tr>
<tr>
<td>deleteAnnotation</td>
<td>72</td>
</tr>
<tr>
<td>deleteDuplicateEdges</td>
<td>72</td>
</tr>
<tr>
<td>deleteGroup</td>
<td>73</td>
</tr>
<tr>
<td>deleteNetwork</td>
<td>74</td>
</tr>
<tr>
<td>deleteSelectedEdges</td>
<td>75</td>
</tr>
<tr>
<td>deleteSelectedNodes</td>
<td>75</td>
</tr>
<tr>
<td>deleteSelfLoops</td>
<td>76</td>
</tr>
<tr>
<td>deleteStyleMapping</td>
<td>77</td>
</tr>
<tr>
<td>deleteTableColumn</td>
<td>77</td>
</tr>
<tr>
<td>deleteVisualStyle</td>
<td>78</td>
</tr>
<tr>
<td>diffusionAdvanced</td>
<td>79</td>
</tr>
<tr>
<td>diffusionBasic</td>
<td>80</td>
</tr>
<tr>
<td>disableApp</td>
<td>80</td>
</tr>
</tbody>
</table>
dockPanel .................................................. 81
doInitializeSandbox ........................................ 82
doRequestRemote .......................................... 82
doSetSandbox ................................................ 83
enableApp .................................................. 84
expandGroup ................................................ 84
exportFilters .............................................. 85
exportImage ............................................... 86
exportJPG ................................................... 87
exportNetwork .............................................. 88
exportNetworkToNDEx .................................... 89
exportPDF ................................................... 90
exportPNG .................................................. 91
exportPS .................................................... 93
exportSVG ................................................... 94
exportVisualStyles ....................................... 95
findRemoteCytoscape .................................. 96
fitContent ................................................ 97
floatPanel ................................................ 97
getAbsSandboxPath ....................................... 98
getAllEdges ............................................... 99
getAllNodes ............................................... 99
getAllStyleMappings .................................... 100
getAnnotationList ....................................... 101
getAppInformation ....................................... 101
getAppStatus ............................................ 102
getAppUpdates ........................................... 103
getArrowShapes ......................................... 103
getAvailableApps ....................................... 104
getBackgroundColorDefault ............................ 104
getBrowserClientChannel .............................. 105
getBrowserClientJs ...................................... 106
getCollectionList ...................................... 106
getCollectionName ...................................... 107
getCollectionNetworks ................................ 107
getCollectionSuid ........................................ 108
getCurrentSandbox ...................................... 109
getCurrentSandboxName ................................ 109
getCurrentSandboxPath ................................ 110
getCurrentStyle ......................................... 110
defaultSandbox ......................................... 111
defaultSandboxPath ...................................... 111
defDisabledApps ......................................... 112
defEdgeColor ............................................. 112
defEdgeCount ............................................ 113
defEdgeInfo ............................................... 114
defEdgeLineStyle ........................................ 115
defEdgeLineWidth ....................................... 115
<table>
<thead>
<tr>
<th>Method</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>getEdgeProperty</td>
<td>116</td>
</tr>
<tr>
<td>getEdgeSelectionColorDefault</td>
<td>117</td>
</tr>
<tr>
<td>getEdgeTargetArrowShape</td>
<td>118</td>
</tr>
<tr>
<td>getFilterList</td>
<td>118</td>
</tr>
<tr>
<td>getFirstNeighbors</td>
<td>119</td>
</tr>
<tr>
<td>getGroupInfo</td>
<td>120</td>
</tr>
<tr>
<td>getInstalledApps</td>
<td>121</td>
</tr>
<tr>
<td>getJupyterBridgeURL</td>
<td>121</td>
</tr>
<tr>
<td>getLayoutNameMapping</td>
<td>122</td>
</tr>
<tr>
<td>getLayoutNames</td>
<td>122</td>
</tr>
<tr>
<td>getLayoutPropertyNames</td>
<td>123</td>
</tr>
<tr>
<td>getLayoutPropertyType</td>
<td>124</td>
</tr>
<tr>
<td>getLayoutPropertyValue</td>
<td>125</td>
</tr>
<tr>
<td>getLineStyles</td>
<td>126</td>
</tr>
<tr>
<td>getNetworkCenter</td>
<td>126</td>
</tr>
<tr>
<td>getNetworkCount</td>
<td>127</td>
</tr>
<tr>
<td>getNetworkList</td>
<td>128</td>
</tr>
<tr>
<td>getNetworkName</td>
<td>128</td>
</tr>
<tr>
<td>getNetworkNDExId</td>
<td>129</td>
</tr>
<tr>
<td>getNetworkProperty</td>
<td>130</td>
</tr>
<tr>
<td>getNetworkSuid</td>
<td>130</td>
</tr>
<tr>
<td>getNetworkViews</td>
<td>131</td>
</tr>
<tr>
<td>getNetworkViewSuid</td>
<td>132</td>
</tr>
<tr>
<td>getNetworkZoom</td>
<td>132</td>
</tr>
<tr>
<td>getNodeColor</td>
<td>133</td>
</tr>
<tr>
<td>getNodeCount</td>
<td>134</td>
</tr>
<tr>
<td>getNodeHeight</td>
<td>134</td>
</tr>
<tr>
<td>getNodeLabelPosition</td>
<td>135</td>
</tr>
<tr>
<td>getNodeLabelPositionDefault</td>
<td>136</td>
</tr>
<tr>
<td>getNodePosition</td>
<td>136</td>
</tr>
<tr>
<td>getNodeProperty</td>
<td>137</td>
</tr>
<tr>
<td>getNodeSelectionColorDefault</td>
<td>138</td>
</tr>
<tr>
<td>getNodeShapes</td>
<td>139</td>
</tr>
<tr>
<td>getNodeSize</td>
<td>139</td>
</tr>
<tr>
<td>getNodeWidth</td>
<td>140</td>
</tr>
<tr>
<td>getNotebookIsRunning</td>
<td>141</td>
</tr>
<tr>
<td>getSandboxReinitialize</td>
<td>141</td>
</tr>
<tr>
<td>getSelectedEdgeCount</td>
<td>142</td>
</tr>
<tr>
<td>getSelectedEdges</td>
<td>142</td>
</tr>
<tr>
<td>getSelectedNodeCount</td>
<td>143</td>
</tr>
<tr>
<td>getSelectedNodes</td>
<td>144</td>
</tr>
<tr>
<td>getStyleDependencies</td>
<td>145</td>
</tr>
<tr>
<td>getStyleMapping</td>
<td>145</td>
</tr>
<tr>
<td>getTableColumnNames</td>
<td>146</td>
</tr>
<tr>
<td>getTableColumns</td>
<td>147</td>
</tr>
<tr>
<td>getTableColumnTypes</td>
<td>148</td>
</tr>
<tr>
<td>getTableValue</td>
<td>149</td>
</tr>
<tr>
<td>getUninstalledApps</td>
<td>150</td>
</tr>
</tbody>
</table>
R topics documented:

getVisualPropertyDefault .................................................. 150
getVisualPropertyNames .................................................... 151
getVisualStyleJSON .......................................................... 152
getVisualStyleNames ........................................................ 152
groupAnnotation ............................................................... 153
hideAllPanels ................................................................. 154
hideEdges ........................................................................ 154
hideNodes ....................................................................... 155
hidePanel ....................................................................... 156
hideSelectedEdges ............................................................ 157
hideSelectedNodes ............................................................ 158
importFileFromUrl ............................................................ 159
importFilters ................................................................. 160
importNetworkFromFile ....................................................... 160
importNetworkFromNDEx ..................................................... 161
importVisualStyles ........................................................... 162
installApp ..................................................................... 163
invertEdgeSelection .......................................................... 163
invertNodeSelection .......................................................... 164
layoutCopycat ................................................................. 165
layoutNetwork ............................................................... 166
listGroups ..................................................................... 167
loadTableData ................................................................. 167
lockNodeDimensions ........................................................ 168
makeSimpleGraph ............................................................. 169
makeSimpleIgraph ............................................................ 169
mapTableColumn ............................................................. 170
mapVisualStyle ............................................................. 171
matchArrowColorToEdge ...................................................... 173
mergeNetworks ............................................................... 174
notebookExportShowImage ................................................. 175
notebookShowImage ........................................................ 177
openAppStore ................................................................. 178
openSession ................................................................. 179
paletteColorBrewerAccent .................................................. 179
paletteColorBrewerBlues .................................................... 180
paletteColorBrewerBrBG .................................................... 181
paletteColorBrewerBuGn ................................................... 181
paletteColorBrewerBuPu .................................................... 182
paletteColorBrewerDark2 ................................................... 182
paletteColorBrewerGnBu .................................................... 183
paletteColorBrewerGreens .................................................. 183
paletteColorBrewerGreys ................................................... 184
paletteColorBrewerOranges ................................................ 185
paletteColorBrewerOrRd .................................................... 185
paletteColorBrewerPaired ................................................... 186
paletteColorBrewerPastel1 .................................................. 186
paletteColorBrewerPastel2 .................................................. 187
R topics documented:

- paletteColorBrewerPiYG ........................................... 187
- paletteColorBrewerPRGn ......................................... 188
- paletteColorBrewerPuBu ......................................... 189
- paletteColorBrewerPuBuGn ....................................... 189
- paletteColorBrewerPuOr ......................................... 190
- paletteColorBrewerPuRd ......................................... 190
- paletteColorBrewerPurples ..................................... 191
- paletteColorBrewerRdBu ......................................... 191
- paletteColorBrewerRdPu ......................................... 192
- paletteColorBrewerRdYlBu ....................................... 193
- paletteColorBrewerReds ......................................... 193
- paletteColorBrewerSet1 .......................................... 194
- paletteColorBrewerSet2 .......................................... 194
- paletteColorBrewerSet3 .......................................... 195
- paletteColorBrewerYlGn ......................................... 195
- paletteColorBrewerYlGnBu ....................................... 196
- paletteColorBrewerYlOrBr ....................................... 197
- paletteColorBrewerYlOrRd ....................................... 197
- paletteColorRandom ............................................. 198
- RCy3 ...................................................................... 198
- RemoveFromGroup .................................................. 199
- removeNodeCustomGraphics ...................................... 200
- renameNetwork ..................................................... 200
- renameTableColumn ............................................... 201
- resetDefaultSandbox ............................................. 202
- rotateLayout ....................................................... 203
- runningRemoteCheck ............................................. 204
- sandboxGetFileInfo ................................................ 204
- sandboxGetFrom ................................................... 205
- sandboxInitializer ................................................ 206
- sandboxRemove ..................................................... 207
- sandboxRemoveFile ............................................... 207
- sandboxSendTo ..................................................... 208
- sandboxSet ........................................................ 209
- sandboxUrlTo ........................................................ 210
- saveSession ........................................................ 211
- scaleLayout ........................................................ 212
- selectAll ................................................................ 213
- selectAllEdges ...................................................... 213
- selectAllNodes ...................................................... 214
- selectEdges ........................................................ 215
- selectEdgesAdjacentToNodes .................................... 216
- selectEdgesAdjacentToSelectedNodes ....................... 217
- selectEdgesConnectingSelectedNodes ....................... 217
- selectFirstNeighbors ............................................. 218
- selectNodes ........................................................ 219
- selectNodesConnectedBySelectedEdges ..................... 220
- setBackgroundColorDefault .......................................... 220
setCatchupFilterSecs ...................................................... 221
setCatchupNetworkSecs .................................................. 222
setCurrentNetwork ....................................................... 223
setCurrentSandbox ....................................................... 223
setCurrentView .......................................................... 224
setDefaultSandbox ...................................................... 225
setDefaultSandboxPath .................................................. 225
setEdgeColorBypass ...................................................... 226
setEdgeColorDefault ..................................................... 227
setEdgeColorMapping .................................................... 227
setEdgeFontFaceBypass .................................................. 228
setEdgeFontFaceDefault ................................................ 229
setEdgeFontFaceMapping ............................................... 230
setEdgeFontSizeBypass .................................................. 231
setEdgeFontSizeDefault ............................................... 232
setEdgeFontSizeMapping ................................................ 233
setEdgeLabelBypass ...................................................... 234
setEdgeLabelColorBypass ............................................... 235
setEdgeLabelColorDefault ............................................. 236
setEdgeLabelColorMapping ............................................. 237
setEdgeLabelDefault .................................................... 238
setEdgeLabelMapping .................................................... 238
setEdgeLabelOpacityBypass ............................................. 239
setEdgeLabelOpacityDefault ........................................... 240
setEdgeLabelOpacityMapping ......................................... 241
setEdgeLineStyleBypass ............................................... 242
setEdgeLineStyleDefault ............................................... 243
setEdgeLineStyleMapping .............................................. 244
setEdgeLineWidthBypass ............................................... 245
setEdgeLineWidthDefault .............................................. 246
setEdgeLineWidthMapping .............................................. 247
setEdgeOpacityBypass ................................................... 248
setEdgeOpacityDefault ................................................. 249
setEdgeOpacityMapping ................................................ 250
setEdgePropertyBypass ................................................. 251
setEdgeSelectionColorDefault ....................................... 252
setEdgeSourceArrowColorBypass ..................................... 253
setEdgeSourceArrowColorDefault .................................... 254
setEdgeSourceArrowColorMapping .................................... 254
setEdgeSourceArrowMapping .......................................... 256
setEdgeSourceArrowShapeBypass ..................................... 257
setEdgeSourceArrowShapeDefault ................................... 258
setEdgeSourceArrowShapeMapping ................................... 258
setEdgeTargetArrowColorBypass ..................................... 259
setEdgeTargetArrowColorDefault .................................... 260
setEdgeTargetArrowColorMapping ................................... 261
setEdgeTargetArrowMapping .......................................... 262
setEdgeTargetArrowShapeBypass ..................................... 263
setEdgeTargetArrowShapeDefault ........................................ 264
setEdgeTargetArrowShapeMapping ...................................... 265
setEdgeTooltipBypass ......................................................... 266
setEdgeTooltipDefault ....................................................... 267
setEdgeTooltipMapping ....................................................... 268
setLayoutProperties ......................................................... 269
setModelPropagationSecs .................................................... 270
setNetworkCenterBypass ..................................................... 271
setNetworkPropertyBypass ................................................... 272
setNetworkZoomBypass ....................................................... 273
setNodeBorderColorBypass ................................................... 274
setNodeBorderColorDefault .................................................. 275
setNodeBorderColorMapping .................................................. 275
setNodeBorderOpacityBypass ................................................ 277
setNodeBorderOpacityDefault ............................................... 278
setNodeBorderOpacityMapping .............................................. 278
setNodeBorderWidthBypass ................................................... 279
setNodeBorderWidthDefault .................................................. 280
setNodeBorderWidthMapping ............................................... 280
setNodeColorBypass .......................................................... 283
setNodeColorDefault ........................................................ 284
setNodeColorMapping ........................................................ 284
setNodeComboOpacityMapping .............................................. 285
setNodeCustomBarChart ....................................................... 286
setNodeCustomBoxChart ...................................................... 288
setNodeCustomHeatMapChart ............................................... 289
setNodeCustomLinearGradient .............................................. 290
setNodeCustomLineChart ..................................................... 291
setNodeCustomPieChart ..................................................... 292
setNodeCustomPosition ....................................................... 293
setNodeCustomRadialGradient ............................................. 294
setNodeCustomRingChart .................................................... 295
setNodeFillOpacityBypass ................................................... 296
setNodeFillOpacityDefault .................................................. 298
setNodeFillOpacityMapping ............................................... 298
setNodeFontFaceBypass ....................................................... 300
setNodeFontFaceDefault ..................................................... 301
setNodeFontFaceMapping ..................................................... 301
setNodeFontSizeBypass ....................................................... 303
setNodeFontSizeDefault ..................................................... 304
setNodeFontSizeMapping ..................................................... 304
setNodeHeightBypass ........................................................ 306
setNodeHeightDefault ........................................................ 307
setNodeHeightMapping ....................................................... 307
setNodeLabelBypass .......................................................... 308
setNodeLabelColorBypass .................................................... 309
setNodeLabelColorDefault ................................................... 310
setNodeLabelColorMapping ................................................. 311
setNodeLabelDefault .............................................. 312
setNodeLabelMapping ........................................... 313
setNodeLabelOpacityBypass .................................... 313
setNodeLabelOpacityDefault ................................... 314
setNodeLabelOpacityMapping ................................... 315
setNodeLabelPositionBypass ................................... 316
setNodeLabelPositionDefault .................................. 317
setNodeOpacityBypass ............................................ 318
setNodePositionBypass .......................................... 319
setNodePropertyBypass ......................................... 320
setNodeSelectionColorDefault ................................. 321
setNodeShapeBypass .............................................. 322
setNodeShapeDefault ............................................ 323
setNodeShapeMapping ............................................ 324
setNodeSizeBypass ................................................. 325
setNodeSizeDefault .............................................. 326
setNodeSizeMapping ............................................. 326
setNodeTooltipBypass ............................................ 327
setNodeTooltipDefault .......................................... 328
setNodeTooltipMapping ......................................... 329
setNodeWidthBypass .............................................. 330
setNodeWidthDefault ............................................. 331
setNodeWidthMapping ............................................ 331
setNotebookIsRunning .......................................... 333
setSandboxReinitialize ......................................... 333
setStyleDependencies .......................................... 334
setVisualStyle .................................................. 335
spoofResponse-class ............................................. 336
syncNodeCustomGraphicsSize ................................ 336
toggleGraphicsDetails .......................................... 337
ungroupAnnotation .............................................. 338
unhideAll .......................................................... 339
unhideEdges ....................................................... 340
unhideNodes ....................................................... 341
uninstallApp ...................................................... 342
UpdateAnnotationBoundedText ................................ 342
updateAnnotationImage ......................................... 344
updateAnnotationShape ......................................... 346
updateAnnotationText .......................................... 347
updateApp ........................................................ 349
updateGroupAnnotation ........................................ 349
updateNetworkInNDEx .......................................... 350
updateStyleDefaults ............................................ 351
updateStyleMapping ............................................. 352

Index 353
Description

Show network view in notebook output.

Usage

`.exportShowImage(
  exportFirst,
  filename = "image",
  type = "PNG",
  resolution = NULL,
  units = NULL,
  height = NULL,
  width = NULL,
  zoom = NULL,
  sandboxName = NULL,
  network = NULL,
  base.url = .defaultBaseUrl,
  overwriteFile = TRUE
)

Arguments

exportFirst FALSE for notebookShowImage, TRUE for notebookExportShowImage.

filename (character) Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. If blank, the current network name will be used.

type (character) Type of image to export, e.g., PNG (default), JPEG, PDF, SVG.

resolution (numeric) The resolution of the exported image, in DPI. Valid only for bitmap formats, when the selected width and height 'units' is inches. The possible values are: 72 (default), 100, 150, 300, 600.

units (character) The units for the 'width' and 'height' values. Valid only for bitmap formats, such as PNG and JPEG. The possible values are: pixels (default), inches.

height (numeric) The height of the exported image. Valid only for bitmap formats, such as PNG and JPEG.

width (numeric) The width of the exported image. Valid only for bitmap formats, such as PNG and JPEG.

zoom (numeric) The zoom value to proportionally scale the image. The default value is 100.0. Valid only for bitmap formats, such as PNG and JPEG.

sandboxName Name of sandbox containing file. None means "the current sandbox".
Description

getDefaultSandbox

Usage

getDefaultSandbox(base.url = .defaultBaseUrl)

Arguments

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

getDefaultSandbox()
**.getRequester**

Description
getRequester

Usage
.getRequester(base.url = .defaultBaseUrl)

Arguments
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples
.getRequester()

---

**addAnnotationBoundedText**

*Add Bounded Text Annotation*

Description
Adds a bounded text annotation to a Cytoscape network view. The object will also be added to the Annotation Panel in the GUI.

Usage
addAnnotationBoundedText(
  text = NULL,
  x.pos = NULL,
  y.pos = NULL,
  fontSize = NULL,
  fontFamily = NULL,
  fontStyle = NULL,
  color = NULL,
  angle = NULL,
)
addAnnotationBoundedText

type = NULL,
customShape = NULL,
fillColor = NULL,
opacity = NULL,
borderThickness = NULL,
borderColor = NULL,
borderOpacity = NULL,
height = NULL,
width = NULL,
name = NULL,
canvas = NULL,
z.order = NULL,
network = NULL,
base.url = .defaultBaseUrl

Arguments

text (optional) The text to be displayed
x.pos (optional) X position in pixels from left; default is center of current view
y.pos (optional) Y position in pixels from top; default is center of current view
fontSize (optional) Numeric value; default is 12
fontFamily (optional) Font family; default is Arial
fontStyle (optional) Font style; default is
color (optional) Hexidecimal color; default is #000000 (black)
angle (optional) Angle of text orientation; default is 0.0 (horizontal)
type (optional) The type of the shape, default is RECTANGLE. See getNodeShapes() for valid options.
customShape (optional) If a custom shape, this is the text of the shape
fillColor (optional) Hexidecimal color; default is #000000 (black)
opacity (optional) Opacity of fill color. Must be an integer between 0 and 100; default is 100.
borderThickness (optional) Integer
borderColor (optional) Hexidecimal color; default is #000000 (black)
borderOpacity (optional) Integer between 0 and 100; default is 100.
height (optional) Height of bounding shape; default is based on text height.
width (optional) Width of bounding shape; default is based on text length.
name (optional) Name of annotation object; default is "Text"
canvas (optional) Canvas to display annotation, i.e., foreground (default) or background
z.order (optional) Arrangement order specified by number (larger values are in front of smaller values); default is 0
addAnnotationImage

Description

Adds a Image annotation to a Cytoscape network view. The object will also be added to the Annotation Panel in the GUI.

Usage

```r
addAnnotationImage(
  url = NULL,
  x.pos = NULL,
  y.pos = NULL,
  angle = NULL,
  opacity = NULL,
  brightness = NULL,
  contrast = NULL,
  borderThickness = NULL,
  borderColor = NULL,
  borderOpacity = NULL,
  height = NULL,
  width = NULL,
  name = NULL,
  canvas = NULL,
  z.order = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
Arguments

url URL or path to image file. File paths can be absolute or relative to current working directory. URLs must start with http:// or https://.
x.pos (optional) X position in pixels from left; default is center of current view
y.pos (optional) Y position in pixels from top; default is center of current view
angle (optional) Angle of text orientation; default is 0.0 (horizontal)
opacity (optional) Opacity of fill color. Must be an integer between 0 and 100; default is 100.
brightness (optional) Image brightness. Must be an integer between -100 and 100; default is 0
canvas (optional) Canvas to display annotation, i.e., foreground (default) or background
z.order (optional) Arrangement order specified by number (larger values are in front of smaller values); default is 0
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
based.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A named list of annotation properties, including UUID

Examples

addAnnotationImage("image.png")
addAnnotationImage("/Users/janedoe/Desktop/image.png", 1000, 1000, name="I2")
addAnnotationImage("https://www.example.com/image.png", 1200, 1000, 30, 40, name="I3", canvas="background", z=4)
addAnnotationShape

Description

Adds a shape annotation to a Cytoscape network view. The object will also be added to the Annotation Panel in the GUI.

Usage

addAnnotationShape(
  type = NULL,
  customShape = NULL,
  x.pos = NULL,
  y.pos = NULL,
  angle = NULL,
  fillColor = NULL,
  opacity = NULL,
  borderThickness = NULL,
  borderColor = NULL,
  borderOpacity = NULL,
  height = NULL,
  width = NULL,
  name = NULL,
  canvas = NULL,
  z.order = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

type (optional) The type of the shape, default is RECTANGLE. See getNodeShapes() for valid options.
customShape (optional) If a custom shape, this is the text of the shape
x.pos (optional) X position in pixels from left; default is center of current view
y.pos (optional) Y position in pixels from top; default is center of current view
angle (optional) Angle of text orientation; default is 0.0 (horizontal)
fillColor (optional) Hexadecimal color; default is #000000 (black)
opacity (optional) Opacity of fill color. Must be an integer between 0 and 100; default is 100.
borderThickness (optional) Integer
borderColor (optional) Hexadecimal color; default is #000000 (black)
addAnnotationShape

Description

Adds a text annotation to a Cytoscape network view. The object will also be added to the Annotation Panel in the GUI.

Usage

addAnnotationText(
  text = NULL,
  x.pos = NULL,
  y.pos = NULL,
  fontSize = NULL,
  fontFamily = NULL,
  fontStyle = NULL,
  color = NULL,
  angle = NULL,
  name = NULL,
  canvas = NULL,
)
addAnnotationText

```r
z.order = NULL,
network = NULL,
base.url = .defaultBaseUrl
)
```

**Arguments**

- **text**
  - The text to be displayed

- **x.pos**
  - (optional) X position in pixels from left; default is center of current view

- **y.pos**
  - (optional) Y position in pixels from top; default is center of current view

- **fontSize**
  - (optional) Numeric value; default is 12

- **fontFamily**
  - (optional) Font family; default is Arial

- **fontStyle**
  - (optional) Font style; default is

- **color**
  - (optional) Hexidecimal color; default is #000000 (black)

- **angle**
  - (optional) Angle of text orientation; default is 0.0 (horizontal)

- **name**
  - (optional) Name of annotation object; default is "Text"

- **canvas**
  - (optional) Canvas to display annotation, i.e., foreground (default) or background

- **z.order**
  - (optional) Arrangement order specified by number (larger values are in front of smaller values); default is 0

- **network**
  - (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

- **base.url**
  - (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

A named list of annotation properties, including UUID

**Examples**

```r
addAnnotationText("test1")
addAnnotationText("test2", 1000, 1000, name="T2")
addAnnotationText("test3", 1200, 1000, 30, "Helvetica", "bold", 
    #990000", 40, name="T3", canvas="foreground", z=4)
```
addCyEdges  

Add CyEdges

Description

Add one or more edges to a Cytoscape network by listing source and target node pairs.

Usage

```r
addCyEdges(
  source.target.list,
  edgeType = "interacts with",
  directed = FALSE,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `source.target.list`  
  A list (or list of lists) of source and target node name or SUID pairs
- `edgeType`  
  The type of interaction. Default is 'interacts with'.
- `directed`  
  boolean for whether interactions are directed. Default is FALSE.
- `network`  
  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A list of named lists of SUID, source and target for each edge added.

Examples

```r
addCyEdges(c('sourceNode', 'targetNode'))
addCyEdges(list(c('s1', 't1'), c('s2', 't2')))
```
addCyNodes

Add CyNodes

Description

Add one or more nodes to a Cytoscape network.

Usage

addCyNodes(
  node.names,
  skip.duplicate.names = TRUE,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

node.names A list of node names
skip.duplicate.names Skip adding a node if a node with the same name is already in the network. If FALSE then a duplicate node (with a unique SUID) will be added. Default is TRUE.
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A list of named lists of name and SUID for each node added.

Examples

addCyNodes(c('Node A', 'Node B', 'Node C'))
AddToGroup

Add to Group

Description

Adds the specified nodes and edges to the specified group.

Usage

AddToGroup(
  group.name,
  nodes = NULL,
  nodes.by.col = "SUID",
  edges = NULL,
  edges.by.col = "SUID",
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

group.name Specifies the name used to identify the group

nodes List of node SUIDs, names, other column values, or keyword: selected, unselected or all. Default is currently selected nodes.

nodes.by.col name of node table column corresponding to provided nodes list. Default is 'SUID'.

edges List of edge SUIDs, names, other column values, or keyword: selected, unselected or all. Default is currently selected edges.

edges.by.col name of edge table column corresponding to provided edges list. Default is 'SUID'.

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

AddToGroup('myGroup')
analyzeNetwork  

**Description**  
Calculate various network statistics.

**Usage**  
`analyzeNetwork(directed = FALSE, base.url = .defaultBaseUrl)`

**Arguments**  
- `directed` (optional) If TRUE, the network is considered a directed graph. Default is FALSE.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**  
The results are added to the Node and Edge tables and the Results Panel. The summary statistics in the Results Panel are also returned by the function as a list of named values.

**Value**  
Named list of summary statistics

**Examples**  
```r  
analyzeNetwork()  
analyzeNetwork(TRUE)  
```

---

applyFilter  

**Description**  
Run an existing filter by supplying the filter name.
Usage

```r
applyFilter(
  filter.name = "Default filter",
  hide = FALSE,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>filter.name</td>
<td>Name of filter to apply. Default is &quot;Default filter&quot;.</td>
</tr>
<tr>
<td>hide</td>
<td>Whether to hide filtered out nodes and edges. Default is FALSE. Ignored if all nodes or edges are filtered out. This is an alternative to filtering for node and edge selection.</td>
</tr>
<tr>
<td>network</td>
<td>(optional) Name or SUID of the network. Default is the &quot;current&quot; network active in Cytoscape.</td>
</tr>
<tr>
<td>base.url</td>
<td>(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://localhost:1234">http://localhost:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

Details

Known bug: selection (or hiding) of edges using edge-based column filters does not work. As a workaround, simply repeat the createColumnFilter operation to perform selection (or hiding) of edges.

Value

List of selected nodes and edges.

See Also

unhideAll

Examples

```r
applyFilter('myFilter')
applyFilter('myFilter', hide = TRUE)
```
**bundleEdges**

**Bundle Edges**

**Description**
Apply edge bundling to the network specified. Edge bundling is executed with default parameters; optional parameters are not supported.

**Usage**
```
bundleEdges(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**
- **network** (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
None

**Examples**
```
bundleEdges()
```

---

**checkNotebookIsRunning**

**checkNotebookIsRunning**

**Description**
checkNotebookIsRunning

**Usage**
```
checkNotebookIsRunning()
```

**Value**
None
clearEdgeBends

Examples

checkNotebookIsRunning()

Description

Clear all edge bends, e.g., those created from edge bundling.

Usage

clearEdgeBends(network = NULL, base.url = .defaultBaseUrl)
clearEdgePropertyBypass

Arguments

- **network** *(optional)* Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
clearEdgeBends()
```

clearEdgePropertyBypass

*Clear Edge Property Bypass*

Description

Clear bypass values for any edge property of the specified edges, effectively restoring any previously defined style defaults or mappings.

Usage

```r
clearEdgePropertyBypass(
  edge.names,
  visual.property,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- **edge.names** List of edge names or SUIDs. Pending CyREST updates, if set to 'all', then the property is cleared for all edges.
- **visual.property** Name of a visual property. See `getVisualPropertyNames`.
- **network** *(optional)* Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
clearNetworkCenterBypass

Value
None

See Also
setEdgePropertyBypass

Examples

clearEdgePropertyBypass()

clearNetworkCenterBypass

Clear Network Center Bypass

Description
Clear the bypass value for center x and y for the network, effectively restoring prior default values.

Usage

clearNetworkCenterBypass(network = NULL, base.url = .defaultBaseUrl)

Arguments

network  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples

clearNetworkCenterBypass()
clearNetworkPropertyBypass

Clear Network Property Bypass

Description

Clear bypass values for any network property, effectively restoring any previously defined style defaults or mappings.

Usage

```r
clearNetworkPropertyBypass(
  visual.property,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `visual.property`: Name of a visual property. See `getVisualPropertyNames`.
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

See Also

`setNodePropertyBypass`

Examples

```r
clearNetworkPropertyBypass()
```
clearNetworkZoomBypass

Clear Network Zoom Bypass

Description
Clear the bypass value for the scale factor for the network, effectively restoring prior default values.

Usage
clearNetworkZoomBypass(network = NULL, base.url = .defaultBaseUrl)

Arguments
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples
clearNetworkZoomBypass()
clearNodePropertyBypass

Arguments

node.names       List of node names or SUIDs
network          (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url         (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

See Also

setNodeOpacityBypass

Examples

clearNodeOpacityBypass(c('Node 1', 'Node 2'))

clearNodePropertyBypass

Clear Node Property Bypass

Description

Clear bypass values for any node property of the specified nodes, effectively restoring any previously defined style defaults or mappings.

Usage

clearNodePropertyBypass(  
  node.names,  
  visual.property,  
  network = NULL,  
  base.url = .defaultBaseUrl
)

Arguments

node.names       List of node names or SUIDs. Pending CyREST updates, if set to 'all', then the property is cleared for all nodes.
visual.property  Name of a visual property. See getVisualPropertyNames.
network          (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
clearSelection

base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

See Also
setNodePropertyBypass

Examples

clearNodePropertyBypass()

clearSelection()
cloneNetwork

Clone a Cytoscape Network

**Description**
Makes a copy of a Cytoscape Network with all of its edges and nodes.

**Usage**
cloneNetwork(network = NULL, base.url = .defaultBaseUrl)

**Arguments**
- **network** (optional) Name or SUID of the network you want to clone; default is "current" network
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
The suid of the new network

**Author(s)**
Alexander Pico, Julia Gustavsen

**Examples**
cloneNetwork("cloned network")

---

closeSession

Close Session

**Description**
Closes the current session in Cytoscape, destroying all unsaved work.

**Usage**
closeSession(save.before.closing, filename = NULL, base.url = .defaultBaseUrl)
**Arguments**

- **save.before.closing**
  - boolean: Whether to save before closing the current session. If FALSE, then all unsaved work will be lost.

- **filename**
  - (optional): If `save.before.closing` is TRUE and the session has not previously been saved, then the path and name of the session file to save should be provided. Default is NULL.

- **base.url**
  - (optional): Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

A boolean for whether to save before closing is required since you could lose data by closing without saving.

**Value**

None

**Examples**

- `closeSession(FALSE)`
- `closeSession(TRUE, '/fullpath/mySession')`
- `closeSession(TRUE)`

---

**collapseGroup**

**Collapse Group**

**Description**

Replaces the representation of all of the nodes and edges in a group with a single node.

**Usage**

```
collapseGroup(groups = NULL, network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- **groups**
  - (optional): List of group names or keywords: all, selected, unselected. Default is the currently selected group.

- **network**
  - (optional): Name or SUID of the network. Default is the "current" network active in Cytoscape.

- **base.url**
  - (optional): Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
commandEcho

Value
None

Examples

collapseGroup()

commandEcho  Command Echo

Description
The echo command will display the value of the variable specified by the variableName argument, or all variables if variableName is not provided.

Usage

commandEcho(variable.name = "*", base.url = .defaultBaseUrl)

Arguments

variable.name (optional) The name of the variable to display. Default is to display all variable values using ":*".

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
Value of variable

Examples

commandEcho()
commandOpenDialog  

**Command Open Dialog**

**Description**

The command line dialog provides a field to enter commands and view results. It also provides the help command to display namespaces, commands, and arguments.

**Usage**

```r
commandOpenDialog(base.url = .defaultBaseUrl)
```

**Arguments**

- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
commandOpenDialog()
```

commandPause  

**Command Pause**

**Description**

The pause command displays a dialog with the text provided in the message argument and waits for the user to click OK.

**Usage**

```r
commandPause(message = "", base.url = .defaultBaseUrl)
```

**Arguments**

- **message**  
  (optional) Text to display in pause dialog

- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
commandQuit

Value

None

Examples

commandPause('Please click OK to continue.')

cmd

commandQuit  Command Quit

Description

This command causes Cytoscape to exit. It is typically used at the end of a script file.

Usage

commandQuit(base.url = .defaultBaseUrl)

Arguments

base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

commandQuit()

commandRunFile  Command Run File

Description

The run command will execute a command script from the file pointed to by the file argument, which should contain Cytoscape commands, one per line. Arguments to the script are provided by the args argument.

Usage

commandRunFile(file, args = NULL, base.url = .defaultBaseUrl)
Arguments

- **file**
  Path to command script file
- **args**
  The script arguments as key: value pairs separated by commas
- **base.url**
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
commandRunFile('/path/to/my/file.txt')
```

---

**commandsAPI**

*Open Swagger docs for CyREST Commands API*

Description

Opens swagger docs in default browser for a live instance of Commands available via CyREST.

Usage

```r
commandsAPI(base.url = .defaultBaseUrl)
```

Arguments

- **base.url**
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Web page in browser

Examples

```r
commandsAPI()
```
commandsGET

Commands GET

Description

Using the same syntax as Cytoscape’s Command Line Dialog, this function converts a command string into a CyREST query URL, executes a GET request, and parses the result content into an R list object. The function check whether actual call is local or remote first. If remote, requests will go through Jupyter-Bridge.

Usage

commandsGET(cmd.string, base.url = .defaultBaseUrl)

Arguments

- cmd.string  (char) command
- base.url     (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A list, status or None.

Examples

commandsGET('layout get preferred network=current')
commandsGET('network list properties network=current')
commandsGET('layout force-directed defaultNodeMass=1')

commandsHelp

Commands Help

Description

Using the same syntax as Cytoscape’s Command Line Dialog, this function returns a list of available commands or args. The function check whether actual call is local or remote first. If remote, requests will go through Jupyter-Bridge.

Usage

commandsHelp(cmd.string = "help", base.url = .defaultBaseUrl)
Arguments

- **cmd.string** (char) command
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Works with or without 'help' command prefix. Note that if you ask about a command that doesn’t have any arguments, this function will run the command!

Value

List of available commands or args

Examples

```
commandsHelp()
commandsHelp('node')
commandsHelp('node get attribute')
```

commandSleep  

Command Sleep

Description

The sleep command will pause processing for a period of time as specified by duration seconds. It is typically used as part of a command script.

Usage

```
commandSleep(duration = NULL, base.url = .defaultBaseUrl)
```

Arguments

- **duration** (optional) The time in seconds to sleep
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 #’ port or version and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```
commandSleep(5)
```
### commandsPOST

**Commands POST**

**Description**

Using the same syntax as Cytoscape’s Command Line Dialog, this function converts a command string into a CyREST query URL, executes a POST request, and parses the result content into an R list object. The function check whether actual call is local or remote first. If remote, requests will go through Jupyter-Bridge.

**Usage**

```r
commandsPOST(cmd.string, base.url = .defaultBaseUrl)
```

**Arguments**

- `cmd.string`: (char) command
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

A list, named list, status or None.

**Examples**

```r
commandsPOST('layout get preferred')
commandsPOST('network list properties')
commandsPOST('layout force-directed defaultNodeMass=1')
```

### commandsRun

**Run a Command**

**Description**

Using the same syntax as Cytoscape’s Command Line Dialog, this function converts a command string into a CyREST query URL, executes a GET request, and parses the result content into an R list object. Same as commandsGET.

**Usage**

```r
commandsRun(cmd.string, base.url = .defaultBaseUrl)
```
Arguments

- cmd.string (char) command
- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A list, status or None.

Examples

```r
commandsRun('layout get preferred')
commandsRun('network list properties')
commandsRun('layout force-directed defaultNodeMass=1')
```

**copyVisualStyle**  
*Copy Visual Style*

Description

Create a new visual style by copying a specified style.

Usage

```r
copyVisualStyle(from.style, to.style, base.url = .defaultBaseUrl)
```

Arguments

- from.style Name of visual style to copy
- to.style Name of new visual style
- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
copyVisualStyle()
```
createColumnFilter

Create Column Filter

Description

Creates a filter to control node or edge selection. Works on columns of boolean, string, numeric and lists. Note the unique restrictions for criterion and predicate depending on the type of column being filtered.

Usage

createColumnFilter(
  filter.name,
  column,
  criterion,
  predicate,
  caseSensitive = FALSE,
  anyMatch = TRUE,
  type = "nodes",
  hide = FALSE,
  network = NULL,
  base.url = .defaultBaseUrl,
  apply = TRUE
)

Arguments

filter.name Name for filter.
column Table column to base filter upon.
criterion For boolean columns: TRUE or FALSE. For string columns: a string value, e.g., "hello". If the predicate is REGEX then this can be a regular expression as accepted by the Java Pattern class (https://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html). For numeric columns: If the predicate is BETWEEN or IS_NOT_BETWEEN then this is a two-element vector of numbers, example: c(1,5), otherwise a single number.
predicate For boolean columns: IS, IS_NOT. For string columns: IS, IS_NOT, CONTAINS, DOES_NOT_CONTAIN, REGEX. For numeric columns: IS, IS_NOT, GREATER_THAN, GREATER_THAN_OR_EQUAL, LESS_THAN, LESS_THAN_OR_EQUAL, BETWEEN, IS_NOT_BETWEEN
caseSensitive (optional) If string matching should be case sensitive. Default is FALSE.
anyMatch (optional) Only applies to List columns. If true then at least one element in the list must pass the filter, if false then all the elements in the list must pass the filter. Default is TRUE.
type (optional) Apply filter to "nodes" (default) or "edges".
createCompositeFilter

Description

Combines filters to control node and edge selection based on previously created filters.

Usage

```r
createCompositeFilter(
  filter.name,
  filter.list,
  type = "ALL",
  hide = FALSE,
  network = NULL,
  base.url = .defaultBaseUrl,
  apply = TRUE
)
```
createCytoscapejsFromNetwork

Create Cytoscape JS from Network

Description
Create a Cytoscape JS representation of a Cytoscape network

Usage
createCytoscapejsFromNetwork(network = NULL, base.url = .defaultBaseUrl)

Arguments
network (SUID or str or NULL): Name or SUID of a network or view. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

createCytoscapejsFromNetwork

Arguments

- **filter.name** Name for filter.
- **filter.list** List of filters to combine.
- **type** (optional) Type of composition, requiring ALL (default) or ANY filters to pass for final node and edge selection.
- **hide** Whether to hide filtered out nodes and edges. Default is FALSE. Ignored if all nodes or edges are filtered out. This is an alternative to filtering for node and edge selection.
- **network** (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
- **apply** (bool) True to execute filter immediately (default); False to define filter but not execute it (available in Cytoscape 3.9+).

Value
List of selected nodes and edges.

Examples

```r
createCompositeFilter("comp1", c("filter1", "filter2"))
createCompositeFilter("comp2", c("filter1", "filter2"), "ANY")
createCompositeFilter("comp3", c("comp1", "filter3"), apply=FALSE)
```
Value

(list) The Cytoscape JS object

Examples

createCytoscapejsFromNetwork()

---

createDegreeFilter  Create Degree Filter

Description

Creates a filter to control node selection based on in/out degree.

Usage

createDegreeFilter(
    filter.name,
    criterion,
    predicate = "BETWEEN",
    edgeType = "ANY",
    hide = FALSE,
    network = NULL,
    base.url = .defaultBaseUrl,
    apply = TRUE
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>filter.name</td>
<td>Name for filter.</td>
</tr>
<tr>
<td>criterion</td>
<td>A two-element vector of numbers, example: c(1,5).</td>
</tr>
<tr>
<td>predicate</td>
<td>BETWEEN (default) or IS_NOT_BETWEEN</td>
</tr>
</tbody>
</table>
| edgeType    | (optional) Type of edges to consider in degree count: ANY (default), UNDI-
              | RECTED, INCOMING, OUTGOING, DIRECTED                                        |
| hide        | Whether to hide filtered out nodes and edges. Default is FALSE. Ignored if |
              | all nodes or edges are filtered out. This is an alternative to filtering   |
              | for node and edge selection.                                               |
| network     | (optional) Name or SUID of the network. Default is the "current" network    |
              | active in Cytoscape.                                                       |
| base.url    | (optional) Ignore unless you need to specify a custom domain, port or       |
              | version to connect to the CyREST API. Default is http://localhost:1234 and |
              | the latest version of the CyREST API supported by this version of RCy3.     |
| apply       | (bool) True to execute filter immediately (default); False to define filter |
              | but not execute it (available in Cytoscape 3.9+).                         |
createGraphFromNetwork

Value

List of selected nodes and edges.

Examples

createDegreeFilter('myFilter', c(4,5))
createDegreeFilter('myFilter', c(2,5), apply=FALSE)

createGraphFromNetwork

degreeFilter

degreeFilter

Description

Returns the Cytoscape network as a Bioconductor graph.

Usage

createGraphFromNetwork(network = NULL, base.url = .defaultBaseUrl)

Arguments

network   (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A Bioconductor graph object.

Author(s)

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples

cw <- CytoscapeWindow('network', graph=make_graphnel())
g <- createGraphFromNetwork()
g <- createGraphFromNetwork('myNetwork')
createGroup  Create Group

Description

Create a group from the specified nodes.

Usage

createGroup(
  group.name,
  nodes = NULL,
  nodes.by.col = "SUID",
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

  group.name    The name used to identify and optionally label the group
  nodes         List of node SUIDs, names, other column values, or keyword: selected, unselected or all. Default is currently selected nodes.
  nodes.by.col  name of node table column corresponding to provided nodes list. Default is 'SUID'.
  network       (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
  base.url      (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

  Group SUID

Examples

createGroup(‘myGroup’)
createGroupByColumn

Create Group by Column

Description

Create a group of nodes defined by a column value.

Usage

createGroupByColumn(
  group.name,
  column = NULL,
  value = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

- **group.name**: The name used to identify and optionally label the group.
- **column**: The name or header of the Node Table column to use for selecting nodes to group.
- **value**: The value in the column to use for selecting nodes to group.
- **network** (optional): Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional): Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Group SUID

Examples

createGroupByColumn('myGroup', 'Cluster', 'A')
createIgraphFromNetwork

Create an igraph network from a Cytoscape network

Description

Takes a Cytoscape network and generates data frames for vertices and edges to send to the graph_from_data_frame function. Returns the network.suid and applies the preferred layout set in Cytoscape preferences.

Usage

createIgraphFromNetwork(network = NULL, base.url = .defaultBaseUrl)

Arguments

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Nodes and edges from the Cytoscape network will be translated into vertices and edges in igraph. Associated table columns will also be passed to igraph as vertex and edge attributes. Note: all networks are implicitly modeled as directed in Cytoscape. Round-trip conversion of an undirected network in igraph via createNetworkFromIgraph to Cytoscape and back to igraph will result in a directed network.

Value

(igraph) an igraph network

See Also

createNetworkFromDataFrames, createNetworkFromIgraph

Examples

ig <- createIgraphFromNetwork()
ig <- createIgraphFromNetwork('myNetwork')
createNetworkFromCytoscapejs

*Create a Network from Cytoscapejs*

**Description**

Create a network from CytoscapeJS JSON

**Usage**

```
createNetworkFromCytoscapejs(
    cytoscapejs,
    title = NULL,
    collection = "My CytoscapeJS Network Collection",
    base.url = .defaultBaseUrl
)
```

**Arguments**

- **cytoscapejs**: network (nodes, edges, attributes, node positions and metadata) in CytoscapeJS format
- **title**: network name (NULL means use the name in cytoscapejs)
- **collection**: collection name (NULL means create an unnamed collection)
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

The SUID of the new network

**Examples**

```
createNetworkFromCytoscapejs()
```

---

createNetworkFromDataFrames

*Create a network from data frames*

**Description**

Takes data frames for nodes and edges, as well as naming parameters to generate the JSON data format required by the “networks” POST operation via CyREST. Returns the network.suid and applies the preferred layout set in Cytoscape preferences.
Usage

createNetworkFromDataFrames(
    nodes = NULL,
    edges = NULL,
    title = "From dataframe",
    collection = "My Dataframe Network Collection",
    base.url = .defaultBaseUrl,
    ...
)

Arguments

- nodes (data.frame) see details and examples below; default NULL to derive nodes from edge sources and targets
- edges (data.frame) see details and examples below; default NULL for disconnected set of nodes
- title (char) network name
- collection (char) network collection name
- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
- ... params for nodeSet2JSON() and edgeSet2JSON()

Details

NODES should contain a column of character strings named: id. This name can be overridden by the arg: node.id.list. Additional columns are loaded as node attributes. EDGES should contain columns of character strings named: source, target and interaction. These names can be overridden by args: source.id.list, target.id.list, interaction.type.list. Additional columns are loaded as edge attributes. The 'interaction' list can contain a single value to apply to all rows; and if excluded altogether, the interaction type will be set to "interacts with". NOTE: attribute values of types (num) will be imported as (Double); (int) as (Integer); (chr) as (String); and (logical) as (Boolean). (Lists) will be imported as (Lists) in CyREST v3.9+

Value

(int) network SUID

Examples

nodes <- data.frame(id=c("node 0","node 1","node 2","node 3"),
    group=c("A","A","B","B"), # categorical strings
    score=as.integer(c(20,10,15,5))) # integers
edges <- data.frame(source=c("node 0","node 0","node 0","node 2"),
    target=c("node 1","node 2","node 3","node 3"),
    interaction=c("inhibits","interacts",
        "activates","interacts"), # optional
    weight=c(5.1,3.0,5.2,9.9)) # numeric
createNetworkFromGraph

createNetworkFromDataFrames(nodes, edges)

createNetworkFromGraph

Create Network From Graph

Description

Creates a Cytoscape network from a Bioconductor graph.

Usage

createNetworkFromGraph(
  graph,
  title = "From graph",
  collection = "My GraphNEL Network Collection",
  base.url = .defaultBaseUrl
)

Arguments

graph A GraphNEL object
title (char) network name
collection (char) network collection name
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Network SUID

Author(s)

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples

library(graph)
g <- makeSimpleGraph()
createNetworkFromGraph(g)
createNetworkFromIgraph

Create a Cytoscape network from an igraph network

Description

Takes an igraph network and generates data frames for nodes and edges to send to the createNetwork function. Returns the network.suid and applies the preferred layout set in Cytoscape preferences.

Usage

createNetworkFromIgraph(
  igraph,
  title = "From igraph",
  collection = "My Igraph Network Collection",
  base.url = .defaultBaseUrl,
  ...
)

Arguments

igraph (igraph) igraph network object

title (char) network name

collection (char) network collection name

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

... params for nodeSet2JSON() and edgeSet2JSON(); see createNetwork

Details

Vertices and edges from the igraph network will be translated into nodes and edges in Cytoscape. Associated attributes will also be passed to Cytoscape as node and edge table columns. Note: undirected networks will be implicitly modeled as directed in Cytoscape. Conversion back via createIgraphFromNetwork will result in a directed network. Also note: igraph attributes of type "other" denoted by "x" are converted to "String" in Cytoscape.

Value

(int) network SUID

See Also

createNetworkFromDataFrames, createIgraphFromNetwork
**createSubnetwork**

Create subnetwork from existing network

**Usage**

createSubnetwork(
  nodes = NULL,
  nodes.by.col = "SUID",
  edges = NULL,
  edges.by.col = "SUID",
  exclude.edges = FALSE,
  subnetwork.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)

**Arguments**

- **nodes**
  list of nodes by SUID, by specified nodes.by.col value (e.g., name) or by keyword: selected, unselected or all. Default is currently selected nodes.

- **nodes.by.col**
  name of node table column corresponding to provided nodes list; default is 'SUID'

- **edges**
  list of edges by SUID, by specified nodes.by.col value (e.g., name) or by keyword: selected, unselected or all. Default is currently selected edges.

- **edges.by.col**
  name of edge table column corresponding to provided edges list; default is 'SUID'

- **exclude.edges**
  (boolean) whether to exclude connecting edges; default is FALSE

- **subnetwork.name**
  name of new subnetwork to be created; default is to add a numbered suffix to source network name

- **network**
  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

- **base.url**
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Examples**

library(igraph)
ig <- makeSimpleIgraph()
createNetworkFromIgraph(ig)
createView

Details

If you specify both nodes and edges, the resulting subset will be the union of those sets. Typical usage only requires specifying either nodes or edges. Note that selected nodes will bring along their connecting edges by default (see exclude.edges arg) and selected edges will always bring along their source and target nodes.

Value

SUID of new subnetwork

Examples

```r
createSubnetwork()
createSubnetwork("all")
createSubnetwork(subnetwork.name="mySubnetwork")
createSubnetwork(c("node 1","node 2","node 3"),"name")
createSubnetwork(c("AKT1","TP53","PIK3CA"),"display name")
createSubnetwork(edges="all") #subnetwork of all connected nodes
```

createView

Create Network View

Description

Create a network view if one does not already exist

Usage

```r
createView(layout = TRUE, network = NULL, base.url = .defaultBaseUrl)
```

Arguments

- `layout` (optional) If TRUE (default), the preferred layout will be applied to the new view. If FALSE, no layout will be applied.
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

For networks larger than the view creation threshold, a network view is not created by default. This function creates a network view if one does not already exist.
**createVisualStyle**

Create a visual style from components

**Description**

Create a style from defaults and predefined mappings.

**Usage**

```r
createVisualStyle(style.name, defaults, mappings, base.url = .defaultBaseUrl)
```

**Arguments**

- `style.name` (char) name for style
- `defaults` (list) key-value pairs for default mappings.
- `mappings` (list) visual property mappings, see `mapVisualProperty`
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

Requires attribute mappings to be previously created, see `mapVisualProperty`.

**Value**

None

**See Also**

`applyStyle`, `mapVisualProperty`
Examples

# first there has to be a network to apply style to
example(createNetworkFromDataFrames)

# then prepare style variables
style.name = "myStyle"
defaults <- list(NODE_SHAPE="diamond",
              NODE_SIZE=30,
              EDGE_TRANSPARENCY=120,
              NODE_LABEL_POSITION="W,E,c,0.00,0.00")
nodelabels <- mapVisualProperty('node label','id','p')
nodefills <- mapVisualProperty('node fill color','group','d',c("A","B"),
                                 c("#FF9900","#66AAAA"))
arrowshapes <- mapVisualProperty('Edge Target Arrow Shape',
                                 'interaction','d',
                                 c("activates","inhibits","interacts"),
                                 c("Arrow","T","None"))
edgeWidth <- mapVisualProperty('edge width','weight','p')

# and then create the style
createVisualStyle(style.name, defaults, list(nodelabels,nodefills,arrowshapes,edgeWidth))

# finish by applying the style
setVisualStyle(style.name)

cybrowserClose

Cybrowser Close

Description

Close an internal web browser and remove all content. Provide an id for the browser you want to close.

Usage

cybrowserClose(id = NULL, base.url = .defaultBaseUrl)

Arguments

id (optional) The identifier for the browser window to close
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

cybrowserClose('554')
Description

Launch Cytoscape’s internal web browser in a separate window. Provide an id for the window if you want subsequent control of the window e.g., via `cybrowser hide`.

Usage

```r
cybrowserDialog(
  id = NULL,
  text = NULL,
  title = NULL,
  url = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `id` (optional) The identifier for the browser window
- `text` (optional) HTML text to initially load into the browser
- `title` (optional) Text to be shown in the title bar of the browser window
- `url` (optional) The URL the browser should load
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

See Also

- `cybrowserShow`
- `cybrowserHide`

Examples

```r
cybrowserDialog(url='http://cytoscape.org')
```
cybrowserHide  

**Cybrowser Hide**

**Description**

Hide an existing browser, whether it’s in the Results panel or a separate window.

**Usage**

```r
cybrowserHide(id = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `id` (optional) The identifier for the browser window to hide
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

- `None`

**See Also**

- `cybrowserShow`
- `cybrowserDialog`

**Examples**

```r
cybrowserHide()
```

cybrowserList  

**Cybrowser List**

**Description**

List all browsers that are currently open, whether as a dialog or in the results panel.

**Usage**

```r
cybrowserList(base.url = .defaultBaseUrl)
```
Arguments

**base.url**  
(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

List of open cybrowser windows

Examples

cybrowserList()

cybrowserSend

* Cybrowser Send

Description

Send the text to the browser indicated by the id and return the response, if any. Note that the JSON result field could either be a bare string or JSON formatted text.

Usage

cybrowserSend(id = NULL, script, base.url = .defaultBaseUrl)

Arguments

**id**  
(optional) The identifier for the browser window

**script**  
(optional) A string that represents a JavaScript variable, script, or call to be executed in the browser. Note that only string results are returned.

**base.url**  
(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

String result

Examples

cybrowserSend(id="Window 1", script="navigator.userAgent;")
cybrowserShow

Description

Launch Cytoscape’s internal web browser in a pane in the Result Panel. Provide an id for the window if you want subsequent control of the window via cybrowser hide.

Usage

cybrowserShow(
  id = NULL,
  text = NULL,
  title = NULL,
  url = NULL,
  base.url = .defaultBaseUrl
)

Arguments

id (optional) The identifier for the browser window
text (optional) HTML text to initially load into the browser	
title (optional) Text to be shown in the title bar of the browser window
url (optional) The URL the browser should load
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

See Also

cybrowserDialog
cybrowserHide

Examples

cybrowserShow(url='http://cytoscape.org')
cybrowserVersion  

Description 
Display the version of the CyBrowser app.

Usage 
cybrowserVersion(base.url = .defaultBaseUrl)

Arguments 
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value 
Version number

Examples 
cybrowserVersion()

cyrestAPI  

Open Swagger docs for CyREST API

Description 
Opens swagger docs in default browser for a live instance of CyREST operations.

Usage 
cyrestAPI(base.url = .defaultBaseUrl)

Arguments 
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value 
Web page in browser
Examples

cyrestAPI()

---

cyrestDELETE  CyREST DELETE

Description

Constructs the query, makes DELETE call and processes the result. The function check whether actual call is local or remote first. If remote, requests will go through Jupyter-Bridge.

Usage

cyrestDELETE(operation = NULL, parameters = NULL, base.url = .defaultBaseUrl)

Arguments

operation  A string to be converted to the REST query namespace
parameters  A named list of values to be converted to REST query parameters
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

CyREST result content

Examples

cyrestDELETE('session')

---

cyrestGET  CyREST GET

Description

Constructs the query, makes GET call and processes the result. The function check whether actual call is local or remote first. If remote, requests will go through Jupyter-Bridge.

Usage

cyrestGET(operation = NULL, parameters = NULL, base.url = .defaultBaseUrl)
cyrestPOST

Arguments

- **operation**: A string to be converted to the REST query namespace
- **parameters**: A named list of values to be converted to REST query parameters
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

CyREST result content

Examples

```r
cyrestGET('version')
```

---

**cyrestPOST**  
*CyREST POST*

Description

Constructs the query and body, makes POST call and processes the result. The function check whether actual call is local or remote first. If remote, requests will go through Jupyter-Bridge.

Usage

```r
cyrestPOST(
  operation,
  parameters = NULL,
  body = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- **operation**: A string to be converted to the REST query namespace
- **parameters**: A named list of values to be converted to REST query parameters
- **body**: A named list of values to be converted to JSON
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

CyREST result content
Examples

cyrestPOST('networks/51/views')

cyrestPUT

CyREST PUT

Description

Constructs the query and body, makes PUT call and processes the result. The function check whether actual call is local or remote first. If remote, requests will go through Jupyter-Bridge.

Usage

cyrestPUT(
  operation,
  parameters = NULL,
  body = NULL,
  base.url = .defaultBaseUrl
)

Arguments

operation A string to be converted to the REST query namespace
parameters A named list of values to be converted to REST query parameters
body A named list of values to be converted to JSON
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

CyREST result content

Examples

cyrestPUT()
cytoscapeApiVersions  

Available CyREST API Versions

Description
Get the list of available CyREST API versions

Usage
cytoscapeApiVersions(base.url = .defaultBaseUrl)

Arguments
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
A list of API versions as character strings, e.g., "v1"

Author(s)
Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples
cytoscapeApiVersions()
# [1] "v1"

---

cytoscapeFreeMemory  

Free Up Unused Memory for Cytoscape

Description
Manually call Java's garbage collection System.gc() to free up unused memory. This process happens automatically, but may be useful to call explicitly for testing or evaluation purposes.

Usage
cytoscapeFreeMemory(base.url = .defaultBaseUrl)

Arguments
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
cytoscapeMemoryStatus

Value

status message

Author(s)

Alexander Pico

Examples

cytoscapeFreeMemory()
# [1] "Unused memory freed up."

cytoscapeMemoryStatus

Description

Returns the memory resources of the server running Cytoscape

Usage

cytoscapeMemoryStatus(base.url = .defaultBaseUrl)

Arguments

- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A list of numeric values

Author(s)

Alexander Pico

Examples

cytoscapeMemoryStatus()
# usedMemory freeMemory totalMemory maxMemory
# 181 2624 2805 13653
cytoscapeNumberOfCores

Number of Processors Available to Cytoscape

Description

Returns the processor resources of the server running Cytoscape

Usage

cytoscapeNumberOfCores(base.url = .defaultBaseUrl)

Arguments

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A numeric value

Author(s)

Alexander Pico

Examples

    cytoscapeNumberOfCores()
    # [1] 8

---

cytoscapePing

Ping Cytoscape

Description

Tests the connection to Cytoscape via CyREST and verifies that supported versions of Cytoscape and CyREST API are loaded.

Usage

    cytoscapePing(base.url = .defaultBaseUrl)
Arguments

base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

status message

Author(s)

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples

cytoscapePing()
# [1] "You are connected to Cytoscape!"

cytoscapeVersionInfo

Cytoscape and CyREST API Versions

Description

Returns the versions of the current Cytoscape and CyREST API

Usage

cytoscapeVersionInfo(base.url = .defaultBaseUrl)

Arguments

base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

list of versions

Author(s)

Alexander Pico

Examples

cytoscapeVersionInfo()
# apiVersion cytoscapeVersion
#     "v1" "3.7.0-SNAPSHOT"
**deleteAllNetworks**

Delete all networks from the current Cytoscape session.

**Usage**

```r
deleteAllNetworks(base.url = .defaultBaseUrl)
```

**Arguments**

- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
deleteAllNetworks()
```

---

**deleteAllVisualStyle**

Delete all visual styles from current Cytoscape session

**Usage**

```r
deleteAllVisualStyle(base.url = .defaultBaseUrl)
```

**Arguments**

- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None
Examples

```r
deleteAllVisualStyle()
```

---

## deleteAnnotation

### Delete Annotation

**Description**

Remove an annotation from the current network view in Cytoscape

**Usage**

```r
deleteAnnotation(names = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `names`: List of annotations by UUID or Name
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
deleteAnnotation("016a4af1-69bc-4b99-8183-d6f118847f96")
deleteAnnotation(c("T2","T3"))
deleteAnnotation(sapply(getAnnotationList(), '[[', 'uuid'))
```

---

## deleteDuplicateEdges

### Delete Duplicate Edges

**Description**

Removes edges with duplicate names. Only considers cases with identical source, target, interaction and directionality.

**Usage**

```r
deleteDuplicateEdges(network = NULL, base.url = .defaultBaseUrl)
```
deleteGroup

Arguments

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Duplicate edges are first selected and then deleted. Prior edge selections will be lost; node selections will not be affected.

Value

Lists of SUIDs for selected nodes and edges

Examples

deleteDuplicateEdges()

deleGroup

Delete (or Ungroup) a Group

Description

Deletes one or more groups, while leaving member nodes intact.

Usage

deleteGroup(
    groups = NULL,
    groups.by.col = "SUID",
    network = NULL,
    base.url = .defaultBaseUrl
)

Arguments

groups (optional) List of group SUIDs, names, other column values or keywords: all, selected, unselected. Default is the currently selected group.

groups.by.col name of node table column corresponding to provided groups list. Default is 'SUID'.

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
**deleteNetwork**

**Details**

Note: group nodes are ungrouped but not deleted in Cytoscape version 3.6.1

**Value**

None

**Examples**

deleteGroup()

---

**deleteNetwork**  
*Delete Network*

**Description**

Delete a network from the current Cytoscape session.

**Usage**

```r
deleteNetwork(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
deleteNetwork()
```
**deleteSelectedEdges**  
*Delete Selected Edges*

**Description**

Delete the currently selected edges in the network.

**Usage**

```r
deleteSelectedEdges(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

list of deleted edge SUIDs

**Author(s)**

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**

```r
deleteSelectedEdges()
```

---

**deleteSelectedNodes**  
*Delete Selected Nodes*

**Description**

Delete currently selected nodes from the network.

**Usage**

```r
deleteSelectedNodes(network = NULL, base.url = .defaultBaseUrl)
```
deleteSelfLoops

Arguments

- **network** (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A named list of deleted node suids ($nodes) as well as edge suids ($edges) deleted as a result of the node deletion

Examples

```r
deleteSelfLoops()
```

---

**deleteSelfLoops**  
**Delete Self Loops**

**Description**

Removes edges that connect to a single node as both source and target.

**Usage**

```r
deleteSelfLoops(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- **network** (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

Self loop edges are first selected and then deleted. Prior edge and node selections will be lost.

**Value**

Lists of SUIDs for selected nodes and edges

**Examples**

```r
deleteSelfLoops()
```
**deleteStyleMapping**

**Delete Style Mapping**

**Description**

Deletes a specified visual style mapping from specified style.

**Usage**

```r
deleteStyleMapping(style.name, visual.prop, base.url = .defaultBaseUrl)
```

**Arguments**

- **style.name** (char) name for style
- **visual.prop** (char) name of visual property to map. See `getVisualPropertyNames()`.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is `http://localhost:1234` and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
deleteStyleMapping()
```

---

**deleteTableColumn**

**Delete a table column**

**Description**

Delete a column from node, edge or network tables.

**Usage**

```r
deleteTableColumn(
    column,
    table = "node",
    namespace = "default",
    network = NULL,
    base.url = .defaultBaseUrl
)
```
deleteVisualStyle

Delete Visual Style

Description

Deletes the specified visual style from current session.

Usage

deleteVisualStyle(style.name, base.url = .defaultBaseUrl)

Arguments

style.name (char) name of style to delete
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

deleteVisualStyle("myStyle")
**Description**

Diffusion will send the selected network view and its selected nodes to a web-based REST service to calculate network propagation. Results are returned and represented by columns in the node table. Advanced operation supports parameters.

**Usage**

```r
diffusionAdvanced(
  heat.column.name = NULL,
  time = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **heat.column.name** *(optional)* A node column name intended to override the default table column 'diffusion_input'. This represents the query vector and corresponds to h in the diffusion equation.
- **time** *(optional)* The extent of spread over the network. This corresponds to t in the diffusion equation.
- **base.url** *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

Columns are created for each execution of Diffusion and their names are returned in the response. The nodes you would like to use as input should be selected. This will be used to generate the contents of the diffusion_input column, which represents the query vector and corresponds to h in the diffusion equation.

**Value**

Version number

**Examples**

```r
diffusionAdvanced()
```
**diffusionBasic**  
*Diffusion Basic*

**Description**
Diffusion will send the selected network view and its selected nodes to a web-based REST service to calculate network propagation. Results are returned and represented by columns in the node table.

**Usage**
diffusionBasic(base.url = .defaultBaseUrl)

**Arguments**
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**
Columns are created for each execution of Diffusion and their names are returned in the response. The nodes you would like to use as input should be selected. This will be used to generate the contents of the diffusion_input column, which represents the query vector and corresponds to \( h \) in the diffusion equation.

**Value**
Version number

**Examples**
diffusionBasic()

**disableApp**  
*Disable App*

**Description**
Disable an app to effectively remove it from your Cytoscape session without having to uninstall it.

**Usage**
disableApp(app, base.url = .defaultBaseUrl)
dockPanel

Arguments

app Name of app
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

disableApp()

dockPanel('table')

dockPanel(panel.name, base.url = .defaultBaseUrl)

Description

Dock a panel back into the UI of Cytoscape.

Usage

dockPanel(panel.name, base.url = .defaultBaseUrl)

Arguments

panel.name Name of the panel. Multiple ways of referencing panels is supported: WEST, control panel, control, c SOUTH, table panel, table, ta SOUTH_WEST, tool panel, tool, to EAST, results panel, results, r base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

dockPanel('table')
doInitializeSandbox

Description

doInitializeSandbox

Usage

doInitializeSandbox(requester = NULL, base.url = .defaultBaseUrl)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>requester</td>
<td>requester</td>
</tr>
<tr>
<td>base.url</td>
<td>Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://127.0.0.1:1234">http://127.0.0.1:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

Value

None

Examples

doInitializeSandbox()

doRequestRemote

Description

Do requests remotely by connecting over Jupyter-Bridge.

Usage

doRequestRemote(method, qurl, qbody = NULL, headers = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>method</td>
<td>A string to be converted to the REST query namespace</td>
</tr>
<tr>
<td>qurl</td>
<td>A named list of values to be converted to REST query parameters</td>
</tr>
<tr>
<td>qbody</td>
<td>A named list of values to be converted to JSON</td>
</tr>
<tr>
<td>headers</td>
<td>httr headers</td>
</tr>
</tbody>
</table>
doSetSandbox

Value

httr response

Examples

doRequestRemote()

---

doSetSandbox  doSetSandbox

Description

doSetSandbox

Usage

```
doSetaSetSandbox(sandboxToSet, requester = NULL, base.url = .defaultBaseUrl)
```

Arguments

- **sandboxToSet**: sandbox to set
- **requester**: requester
- **base.url**: Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://127.0.0.1:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

doSetSandbox()
**enableApp**  

*Enable App*

**Description**

Enable a previously installed and disabled app in Cytoscape.

**Usage**

```r
enableApp(app, base.url = .defaultBaseUrl)
```

**Arguments**

- `app` Name of app
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
enableApp()
```

---

**expandGroup**  

*Expand Group*

**Description**

Replaces the group node with member nodes for a set of groups.

**Usage**

```r
expandGroup(groups = NULL, network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `groups` (optional) List of group names or keywords: all, selected, unselected. Default is the currently selected group.
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
**exportFilters**

**Value**

None

**Examples**

```
expandGroup()
```

---

**exportFilters**

**Export Filters**

**Description**

Saves filters to file in JSON format.

**Usage**

```
exportFilters(
    filename = "filters.json",
    base.url = .defaultBaseUrl,
    overwriteFile = TRUE
)
```

**Arguments**

- **filename** (char) Full path or path relative to current working directory, in addition to the name of the file. Default is "filters.json"
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
- **overwriteFile** (optional) FALSE allows an error to be generated if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.

**Details**

Unlike other export functions, Cytoscape will automatically overwrite files with the same name. You will not be prompted to confirm or reject overwrite. Use carefully!

**Value**

None

**Examples**

```
exportFilters()
```
exportImage

Description

Saves the current network view as an image file.

Usage

exportImage(
  filename = NULL,
  type = "PNG",
  resolution = NULL,
  units = NULL,
  height = NULL,
  width = NULL,
  zoom = NULL,
  network = NULL,
  base.url = .defaultBaseUrl,
  overwriteFile = TRUE
)

Arguments

filename  (character) Full path or path relative to current working directory, in addition
to the name of the file. Extension is automatically added based on the type
argument. If blank, the current network name will be used.

type      (character) Type of image to export, e.g., PNG (default), JPEG, PDF, SVG,
          PS (PostScript).

resolution  (numeric) The resolution of the exported image, in DPI. Valid only for bitmap
          formats, when the selected width and height 'units' is inches. The possible
          values are: 72 (default), 100, 150, 300, 600.

units      (character) The units for the 'width' and 'height' values. Valid only for bitmap
          formats, such as PNG and JPEG. The possible values are: pixels (default),
          inches.

height     (numeric) The height of the exported image. Valid only for bitmap formats,
          such as PNG and JPEG.

width      (numeric) The width of the exported image. Valid only for bitmap formats, such
          as PNG and JPEG.

zoom       (numeric) The zoom value to proportionally scale the image. The default value
          is 100.0. Valid only for bitmap formats, such as PNG and JPEG

network    (optional) Name or SUID of a network or view. Default is the "current" network
          active in Cytoscape. The first (presumably only) view associated a network is
          used if a specific view SUID is not provided.
exportJPG

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

overwriteFile (optional) FALSE allows Cytoscape show a message box before overwriting the file if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.

Details

The image is cropped per the current view in Cytoscape. Consider applying fitContent prior to export.

Value

server response

Examples

exportImage('/fullpath/myNetwork','PDF')

exportJPG

Export JPG

Description

Saves the current network view as an jpg file.

Usage

exportJPG(
    filename = NULL,
    allGraphicsDetails = TRUE,
    hideLabels = FALSE,
    zoom = NULL,
    network = NULL,
    base.url = .defaultBaseUrl,
    overwriteFile = TRUE
)

Arguments

filename (character) Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. If blank, the current network name will be used.

allGraphicsDetails (optional): TRUE results in image with highest detail; False allows faster image generation. The default is TRUE.
hideLabels (optional): TRUE makes node and edge labels invisible in image. False allows them to be drawn. The default is False.

zoom (numeric) The zoom value to proportionally scale the image. The default value is 100.0. Valid only for bitmap formats, such as PNG and JPEG.

network (optional) Name or SUID of a network or view. Default is the "current" network active in Cytoscape. The first (presumably only) view associated a network is used if a specific view SUID is not provided.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

overwriteFile (optional) FALSE allows Cytoscape show a message box before overwriting the file if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.

Details
Exports the current view to a graphics file and returns the path to the saved file. To speed up image export for large networks use the option 'allGraphicsDetails=FALSE'. Available for Cytoscape v3.10 or later.

Value
server response

Examples
exportJPG('/fullpath/myNetwork')

---

exportNetwork  Export Network

Description
Export a network to one of multiple file formats

Usage
exportNetwork(
  filename = NULL,
  type = "SIF",
  network = NULL,
  base.url = .defaultBaseUrl,
  overwriteFile = TRUE
)
Arguments

filename Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. If blank, then the current network name is used.

type File type. SIF (default), CX, cyjs, graphML, NNF, xGMML.

network (optional) Name or SUID of a network or view. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

overwriteFile (optional) FALSE allows Cytoscape show a message box before overwriting the file if the file already exists; TRUE. allows Cytoscape to overwrite it without asking. Default value is TRUE.

Value

None.

Examples

exportNetwork('path/filename', 'SIF')

exportNetworkToNDEx

Export Network To NDEx

Description

Send a copy of a Cytoscape network to NDEx as a new submission.

Usage

exportNetworkToNDEx(  
  username,  
  password,  
  isPublic,  
  network = NULL,  
  metadata = NULL,  
  ndex.url = "http://ndexbio.org",  
  ndex.version = "v2",  
  base.url = .defaultBaseUrl  
)
exportPDF

Arguments

username NDEx account username
password NDEx account password
isPublic (Boolean) Whether to make the network publicly accessible at NDEx.
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
metadata (optional) A list of structured information describing the network
ndex.url (optional) For alternative instances or deployments of NDEx; default is "http://ndexbio.org"
ndex.version (optional) For alternative versions of the NDEx API; default is "v2"
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

NDEx identifier (externalId) for new submission

Examples

exportNetworkToNDEx("user", "pass", TRUE)

Description

Saves the current network view as an pdf file.

Usage

```R
exportPDF(
  filename = NULL,
  exportTextAsFont = TRUE,
  hideLabels = FALSE,
  pageSize = "Letter",
  orientation = "Portrait",
  network = NULL,
  base.url = .defaultBaseUrl,
  overwriteFile = TRUE
)
```
Arguments

filename (character) Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. If blank, the current network name will be used.

exportTextAsFont (optional): If TRUE (the default value), texts will be exported as fonts.

hideLabels (optional): TRUE makes node and edge labels invisible in image. False allows them to be drawn. The default is False.

pageSize (optional): (Auto|Letter|Legal|Tabloid|A0|A1|A2|A3|A4|A5) Predefined standard page size, or choose custom. Default is 'Letter'.

orientation (optional): Page orientation, portrait or landscape.

network (optional) Name or SUID of a network or view. Default is the "current" network active in Cytoscape. The first (presumably only) view associated a network is used if a specific view SUID is not provided.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

overwriteFile (optional) FALSE allows Cytoscape to show a message box before overwriting the file if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.

Details

Exports the current view to a graphics file and returns the path to the saved file. To speed up image export for large networks use the option 'allGraphicsDetails=FALSE'. Available for Cytoscape v3.10 or later.

Value

server response

Examples

exportPDF('/fullpath/myNetwork')

Description

Saves the current network view as an png file.
Usage

```r
eexportPNG(
    filename = NULL,
    allGraphicsDetails = TRUE,
    hideLabels = FALSE,
    transparentBackground = FALSE,
    zoom = NULL,
    network = NULL,
    base.url = .defaultBaseUrl,
    overwriteFile = TRUE
)
```

Arguments

- `filename` (character): Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. If blank, the current network name will be used.

- `allGraphicsDetails` (optional): TRUE results in image with highest detail; False allows faster image generation. The default is TRUE.

- `hideLabels` (optional): TRUE makes node and edge labels invisible in image. False allows them to be drawn. The default is False.

- `transparentBackground` (optional): TRUE causes background to be transparent. The default is FALSE.

- `zoom` (numeric): The zoom value to proportionally scale the image. The default value is 100.0. Valid only for bitmap formats, such as PNG and JPEG

- `network` (optional): Name or SUID of a network or view. Default is the "current" network active in Cytoscape. The first (presumably only) view associated a network is used if a specific view SUID is not provided.

- `base.url` (optional): Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

- `overwriteFile` (optional): FALSE allows Cytoscape show a message box before overwriting the file if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.

Details

Exports the current view to a graphics file and returns the path to the saved file. To speed up image export for large networks use the option ‘allGraphicsDetails=FALSE’. Available for Cytoscape v3.10 or later.

Value

- server response
Examples

exportPNG('/fullpath/myNetwork')

Description

Saves the current network view as an ps file.

Usage

exportPS(
    filename = NULL,
    exportTextAsFont = TRUE,
    hideLabels = FALSE,
    network = NULL,
    base.url = .defaultBaseUrl,
    overwriteFile = TRUE
)

Arguments

filename (character) Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. If blank, the current network name will be used.

exportTextAsFont (optional): If TRUE (the default value), texts will be exported as fonts.

hideLabels (optional): TRUE makes node and edge labels invisible in image. False allows them to be drawn. The default is False.

network (optional) Name or SUID of a network or view. Default is the "current" network active in Cytoscape. The first (presumably only) view associated a network is used if a specific view SUID is not provided.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

overwriteFile (optional) FALSE allows Cytoscape show a message box before overwriting the file if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.

Details

Exports the current view to a graphics file and returns the path to the saved file. To speed up image export for large networks use the option 'allGraphicsDetails=FALSE'. Available for Cytoscape v3.10 or later.
**exportSVG**

**Value**

server response

**Examples**

```r
exportPS('/fullpath/myNetwork')
```

---

**exportSVG**  
*Export SVG*

**Description**

Saves the current network view as an svg file.

**Usage**

```r
exportSVG(
    filename = NULL,
    exportTextAsFont = TRUE,
    hideLabels = FALSE,
    network = NULL,
    base.url = .defaultBaseUrl,
    overwriteFile = TRUE
)
```

**Arguments**

- **filename** *(character)* Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. If blank, the current network name will be used.

- **exportTextAsFont** *(optional)*: If TRUE (the default value), texts will be exported as fonts.

- **hideLabels** *(optional)*: TRUE makes node and edge labels invisible in image. False allows them to be drawn. The default is False.

- **network** *(optional)* Name or SUID of a network or view. Default is the "current" network active in Cytoscape. The first (presumably only) view associated a network is used if a specific view SUID is not provided.

- **base.url** *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

- **overwriteFile** *(optional)* FALSE allows Cytoscape show a message box before overwriting the file if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.
Details

Exports the current view to a graphics file and returns the path to the saved file. To speed up image export for large networks use the option `allGraphicsDetails=FALSE`. Available for Cytoscape v3.10 or later.

Value

server response

Examples

```
exportSVG('/fullpath/myNetwork')
```

---

**Usage**

```r
exportVisualStyles(
  filename = NULL,
  type = "XML",
  styles = NULL,
  base.url = .defaultBaseUrl,
  overwriteFile = TRUE
)
```

**Arguments**

- **filename** (char) Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. Default is "styles.xml"
- **type** (optional) Type of data file to export, e.g., XML, JSON (case sensitive). Default is XML. Note: Only XML can be read by importVisualStyles().
- **styles** (optional) The styles to be exported, listed as a comma-separated string. If no styles are specified, only the current one is exported.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
- **overwriteFile** (optional) FALSE allows Cytoscape show a message box before overwriting the file if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.
findRemoteCytoscape

Value

Path to saved file

See Also

importVisualStyles

Examples

exportVisualStyles(’/fullpath/myStyle’)
exportVisualStyles(’/fullpath/myStyle’, type = ’JSON’)
exportVisualStyles(’/fullpath/myStyle’, style = ’Minimal,default,Directed’)

Description

Figure out whether CyREST is local or remote. If remote, we’ll want to go through Jupyter-Bridge.

Usage

findRemoteCytoscape(base.url = .defaultBaseUrl)

Arguments

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

findRemoteCytoscape()
fitContent  

Fit Content

Description

Zoom and pan network view to maximize either height or width of current network window.

Usage

```r
fitContent(selected.only = FALSE, network = NULL, base.url = .defaultBaseUrl)
```

Arguments

- `selected.only` (Boolean) Whether to fit only current selection. Default is false, i.e., to fit the entire network.
- `network` (optional) Name or SUID of a network or view. Default is the "current" network active in Cytoscape. The first (presumably only) view associated a network is used if a specific view SUID is not provided.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Takes first (presumably only) view associated with provided network

Value

None

Examples

```r
fitContent()
```

floatPanel  

Float Panel

Description

Pop out a panel from the UI of Cytoscape. Other panels will expand into the space.

Usage

```r
floatPanel(panel.name, base.url = .defaultBaseUrl)
```
getAbsSandboxPath

Arguments

panel.name  Name of the panel. Multiple ways of referencing panels is supported: WEST, control panel, control, c SOUTH, table panel, table, ta SOUTH_WEST, tool panel, tool, to EAST, results panel, results, r

base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

floatPanel('table')

gateboxPath  gateboxPath

Description

Get absolute sandbox path.

Usage

gateboxPath(fileLocation)

Arguments

fileLocation  fileLocation

Value

file location

Examples

gateboxPath()
**getAllEdges**

*Get All Edges*

**Description**

Retrieve the names of all the edges in the network.

**Usage**

```r
getAllEdges(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

list of node edges

**Examples**

```r
getAllEdges()
```

---

**getAllNodes**

*Get All Nodes*

**Description**

Retrieve the names of all the nodes in the network.

**Usage**

```r
getAllNodes(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `network` (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
getAllStyleMappings

**Value**

list of node names

**Examples**

globalsAllNodes()

globalsAllStyleMappings

---

**getDescription**

Get All Style Mappings

**Description**

Retrieve the values that define all the mappings per visual property in a given style.

**Usage**

globalsAllStyleMappings(style.name = NULL, base.url = .defaultBaseUrl)

**Arguments**

- **style.name**: Name for style; default is the 'default' style
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

List of lists with named values defining the visual property mappings

**Examples**

globalsGetStyleMapping()
getAnnotationList  Get Annotation List

Description
A list of named lists with annotation information

Usage
getAnnotationList(network = NULL, base.url = .defaultBaseUrl)

Arguments
- network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details
You can obtain a list of UUIDs by applying a subset function like so: sapply(getAnnotationList(), '[[','uuid')

Value
RETURN_DESCRIPTION

Examples
getAnnotationList()

getAppInformation  Get App Information

Description
Retrieve the name, brief description and version of a Cytoscape app.

Usage
getAppInformation(app, base.url = .defaultBaseUrl)
Arguments

app       Name of app
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

App name, brief description and version.

Examples

getAppInformation()

getAppStatus

Description

Retrieve the current status of a Cytoscape app: Installed, Uninstalled or Disabled.

Usage

getAppStatus(app, base.url = .defaultBaseUrl)

Arguments

app       Name of app
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

App name and status

Examples

getAppStatus()
getAppUpdates  List Apps With Updates

**Description**
Retrieve list of currently installed Cytoscape apps with updates available.

**Usage**
```
getAppUpdates(base.url = .defaultBaseUrl)
```

**Arguments**

- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
A list of updatable app names, versions and statuses

**Examples**
```
getAppUpdates()
```

getArrowShapes  Get Arrow Shapes

**Description**
Retrieve the names of the currently supported 'arrows' – the decorations can (optionally) appear at the ends of edges, adjacent to the nodes they connect, and conveying information about the nature of the nodes' relationship.

**Usage**
```
getArrowShapes(base.url = .defaultBaseUrl)
```

**Arguments**

- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
getBackgroundColorDefault

**Value**
A list of character strings, e.g., 'DIAMOND', 'T', 'ARROW'

**Examples**
getArrowShapes()

getAvailableApps  *List Available Apps*

**Description**
Retrieve a list of apps available for installation in Cytoscape.

**Usage**
getAvailableApps(base.url = .defaultBaseUrl)

**Arguments**
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
A list of app names and latest versions

**Examples**
getAvailableApps()

getBackgroundColorDefault  *Get Background Color Default*

**Description**
Retrieve the default background color.

**Usage**
getBackgroundColorDefault(style.name = NULL, base.url = .defaultBaseUrl)
**getBrowserClientChannel**

**Arguments**

- **style.name**  
  Name of style; default is "default" style

- **baseUrl**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

getBackgroundColorDefault()
**getBrowserClientJs**

**Description**
Prepend channel number of client Javascript so it can communicate with this process via Jupyter-bridge

**Usage**
getBrowserClientJs()

**Value**
Javascript inject code

**Examples**
getBrowserClientJs()

---

**getCollectionList**

**Get Collection List**

**Description**
FUNCTION_DESCRIPTION

**Usage**
getCollectionList(base.url = .defaultBaseUrl)

**Arguments**
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
RETURN_DESCRIPTION

**Examples**
getCollectionList()
getCollectionName  Get Collection Name

Description

FUNCTION_DESCRIPTION

Usage

getCollectionName(collection.suid = NULL, base.url = .defaultBaseUrl)

Arguments

collection.suid

description

base.url  (optional) Ignore unless you need to specify a custom domain, port or version
to connect to the CyREST API. Default is http://localhost:1234 and the latest
version of the CyREST API supported by this version of RCy3.

Value

RETURN_DESCRIPTION

Examples

getCollectionName()
getCollectionSuid

Value

RETURN_DESCRIPTION

Examples

gtcollectionNetworks()

getCollectionSuid Get Collection Suid

Description

FUNCTION_DESCRIPTION

Usage

gtcollectionSuid(network = NULL, base.url = .defaultBaseUrl)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>network</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>base.url</td>
<td>(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://localhost:1234">http://localhost:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

Value

RETURN_DESCRIPTION

Examples

gtcollectionSuid()
**getCurrentSandbox**

### Description

Return both the current sandbox name and path.

### Usage

```
getCurrentSandbox()
```

### Value

current sandbox

### Examples

```
getCurrentSandbox()
```

---

**getCurrentSandboxName**

### Description

Return the current sandbox name.

### Usage

```
getCurrentSandboxName()
```

### Value

current sandbox name

### Examples

```
getCurrentSandboxName()
```
getCurrentSandboxPath

Description
Return the current sandbox path.

Usage
gCurrentSandboxPath()

Value
current sandbox path

Examples
gCurrentSandboxPath()

getCurrentStyle

Description
Get the current visual style applied to a network.

Usage
gCurrentStyle(network = NULL, base.url = .defaultBaseUrl)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>network</td>
<td>(optional) Name or SUID of the network. Default is the &quot;current&quot; network active in Cytoscape.</td>
</tr>
<tr>
<td>base.url</td>
<td>(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://localhost:1234">http://localhost:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

Value
Name of style

Examples

gCurrentStyle()
gCurrentStyle('myNetwork')
getDefaultSandbox

Description
Return whatever is the current default sandbox properties.

Usage
getDefaultSandbox()

Value
default sandbox

Examples
getDefaultSandbox()

getDefaultSandboxPath

Description
Return the default path, which isn’t one of the properties tracked in the default_sandbox.

Usage
gDefaultSandboxPath()

Value
default sandbox path

Examples
gDefaultSandboxPath()
**getDisabledApps**  
*List Disabled Apps*

**Description**
Retrieve list of currently disabled apps in Cytoscape.

**Usage**
getDisabledApps(base.url = .defaultBaseUrl)

**Arguments**
- `base.url`  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
A list of disabled app names, versions and statuses

**Examples**
getDisabledApps()

---

**getEdgeColor**  
*Get Edge Color*

**Description**
Retrieve the actual line color of specified edges.

**Usage**
getEdgeColor(edge.names = NULL, network = NULL, base.url = .defaultBaseUrl)

**Arguments**
- `edge.names`  
  List of edge names or SUIDs. Default is NULL for all edges.
- `network`  
  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
Value

Named list of property values

Examples

getEdgeColor()

---

**getEdgeCount**  
*Get Edge Count*

**Description**

Reports the number of the edges in the network.

**Usage**

`getEdgeCount(network = NULL, base.url = .defaultBaseUrl)`

**Arguments**

- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is `http://localhost:1234` and the latest version of the CyREST API supported by this version of RCy3.

**Value**

numeric

**Author(s)**

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**

`getEdgeCount()`
getEdgeInfo

Get Edge Information

Description

Returns source, target and edge table row values.

Usage

getEdgeInfo(edges, network = NULL, base.url = .defaultBaseUrl)

Arguments

edges List of SUIDs or names of edges, i.e., values in the "name" column. Can also input a single edge.

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This function is kinda slow. It takes approximately 70ms per edge to return a result, e.g., 850 edges will take a one minute.

Value

named list of lists

Author(s)

Alexander Pico

Examples

gEDGEInfo()
getEdgeLineStyle

Description
Retrieve the actual line style of specified edges.

Usage
getEdgeLineStyle(edge.names = NULL, network = NULL, base.url = .defaultBaseUrl)

Arguments
edge.names List of edge names or SUIDs. Default is NULL for all edges.
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
Named list of property values

Examples
getEdgeLineStyle()
getEdgeProperty

Arguments

edge.names List of edge names or SUIDs. Default is NULL for all edges.

network (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Named list of property values

Examples

gEDGELineWidth()

getEdgeProperty

Get Edge Property Values

Description

Get values for any edge property of the specified edges.

Usage

gEDGEProperty(  
edge.names = NULL,  
visual.property,  
network = NULL,  
base.url = .defaultBaseUrl  
)

Arguments

edge.names List of edge names or SUIDs. Default is NULL for all edges.

visual.property Name of a visual property. See getVisualPropertyName.

network (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
getEdgeSelectionColorDefault

Details

This method retrieves the actual property of the node, given the current visual style, factoring together any default, mapping and bypass setting.

Value

Named list of property values

Examples

getEdgeProperty(c('node 0 (pp) node 1','node 0 (pp) node 2'),"EDGE_WIDTH")

getEdgeSelectionColorDefault

Get Edge Selection Color Default

Description

Retrieve the default selected edge color.

Usage

gedgeSelectionColorDefault(style.name = NULL, base.url = .defaultBaseUrl)

Arguments

style.name Name of style; default is "default" style

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

getEdgeSelectionColorDefault()
getEdgeTargetArrowShape  

*Get Edge Target Arrow Shape*

**Description**

Retrieve the actual target arrow shape of specified edges.

**Usage**

```r
getEdgeTargetArrowShape(
  edge.names = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `edge.names` List of edge names or SUIDs. Default is NULL for all edges.
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

Named list of property values

**Examples**

```r
getEdgeTargetArrowShape()
```

---

getFilterList  

*Get Filter List*

**Description**

Retrieve list of named filters in current session.

**Usage**

```r
getFilterList(base.url = .defaultBaseUrl)
```
**Arguments**

- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

List of filter names

**Examples**

`getFilterList()`

---

**getFirstNeighbors**  
*Get list of nodes neighboring provided list*

**Description**

Returns a non-redundant list of first neighbors of the supplied list of nodes or current node selection.

**Usage**

```r
getFirstNeighbors(
  node.names = NULL,
  as.nested.list = FALSE,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **node.names**  
  A list of SUIDs or names from the name column of the node table. Default is currently selected nodes.

- **as.nested.list**  
  logical Whether to return lists of neighbors per query node

- **network**  
  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

A list of unique node names, optionally nested per query node name.
Author(s)
Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

See Also
selectNodes selectFirstNeighbors

Examples
getFirstNeighbors()

getGroupInfo

Get Group Information

Description
Retrieve information about a group by name or identifier.

Usage
getGroupInfo(group, network = NULL, base.url = .defaultBaseUrl)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>group</td>
<td>Group name or SUID.</td>
</tr>
<tr>
<td>network</td>
<td>(optional) Name or SUID of the network. Default is the &quot;current&quot; network active in Cytoscape.</td>
</tr>
<tr>
<td>base.url</td>
<td>(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://localhost:1234">http://localhost:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

Value
Group information

Examples
getGroupInfo('myGroup')
getInstalledApps

List Installed Apps

Description
Retrieve list of currently installed apps in Cytoscape.

Usage
getInstalledApps(base.url = .defaultBaseUrl)

Arguments

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
A list of installed app names, versions and statuses

Examples
getInstalledApps()

getJupyterBridgeURL

Description
Get the jupyter bridge server url

Usage
getJupyterBridgeURL()

Value
jupyter bridge server url

Examples
getJupyterBridgeURL()
getLayoutNameMapping  

Get Layout Name Mapping

Description

The Cytoscape 'Layout' menu lists many layout algorithms, but the names presented there are different from the names by which these algorithms are known to layout method. This method returns a named list in which the names are from the GUI, and the values identify the names you must use to choose an algorithms in the programmatic interface.

Usage

getLayoutNameMapping(base.url = .defaultBaseUrl)

Arguments

base.url  
(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A named list of character strings

Author(s)

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples

getLayoutNameMapping()

# Degree Sorted Circle Layout  Group Attributes Layout  Edge-weighted Spring Embedded Layout
#  "degree-circle"             "attributes-layout"    "kamada-kawai"

getLayoutNames  

Get Layout Names

Description

Retrieve the names of the currently supported layout algorithms. These may be used in subsequent calls to the 'layoutNetwork' function.

Usage

getLayoutNames(base.url = .defaultBaseUrl)
getLayoutPropertyNames

Arguments

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A list of character strings, e.g., "force-directed" "circular" "grid"

Author(s)

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples

getLayoutNames()

# [1] "degree-circle" "attributes-layout" "kamada-kawai"
# [4] "force-directed" "cose" "hierarchical"
# [7] "attribute-circle" "stacked-node-layout" "circular"

getLayoutPropertyNames

Get Layout Property Names

Description

Returns a list of the tunable properties for the specified layout.

Usage

getLayoutPropertyNames(layout.name, base.url = .defaultBaseUrl)

Arguments

layout.name (character) Name of the layout
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Run getLayoutNames to list available layouts.

Value

A list of character strings
getLayoutPropertyType

Description

Returns the type of one of the tunable properties (property.name) for the specified layout.

Usage

getLayoutPropertyType(layout.name, property.name, base.url = .defaultBaseUrl)

Arguments

- layout.name (character) Name of the layout
- property.name (character) Name of the property
- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Run `getLayoutNames` to list available layouts. Run `getLayoutPropertyNames` to list properties per layout.

Value

A character string specifying the type

Author(s)

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples

getLayoutPropertyType('force-directed', 'defaultSpringLength')
# "double"
getLayoutPropertyValue

*Get Layout Property Value*

**Description**

Returns the appropriately typed value of the specified tunable property for the specified layout.

**Usage**

```r
getLayoutPropertyValue(layout.name, property.name, base.url = .defaultBaseUrl)
```

**Arguments**

- `layout.name` (character) Name of the layout
- `property.name` (character) Name of the property
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

Run `getLayoutNames` to list available layouts. Run `getLayoutPropertyNames` to list properties per layout.

**Value**

The current value set for this layout property. Typically an integer, numeric or character string value.

**Author(s)**

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**

```r
getLayoutPropertyValue('force-directed','defaultSpringLength')
# 80
```
getLineStyles     Get Line Styles

Description
Retrieve the names of the currently supported line types – values which can be used to render edges, and thus can be used in calls to ‘setEdgeLineStyleRule’.

Usage
getLineStyles(base.url = .defaultBaseUrl)

Arguments
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
A list of character strings, e.g., 'SOLID', 'DOT'

Examples
gelineStyles()

getNetworkCenter     Get Network Center

Description
Retrieve the center of specified network.

Usage
getNetworkCenter(network = NULL, base.url = .defaultBaseUrl)

Arguments
network  (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
getNetworkCount

Value

Named list of property values

Examples

getNetworkCenter()

getworkCount

Get the number of Cytoscape networks

Description

Returns the number of Cytoscape networks in the current Cytoscape session

Usage

getNetworkCount(base.url = .defaultBaseUrl)

Arguments

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

numeric

Author(s)

Alexander Pico, Tanja Muetze, Georgi Kolishtovski, Paul Shannon

Examples

getNetworkCount()
# 3
getNetworkList  Get the list of Cytoscape networks

**Description**

Returns the list of Cytoscape network names in the current Cytoscape session

**Usage**

getNetworkList(getSUIDs = FALSE, base.url = .defaultBaseUrl)

**Arguments**

- **getSUIDs**  (optional) Whether to return SUIDs instead of titles; default is FALSE.
- **base.url**  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

- **list**

**Author(s)**

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**

getNetworkList()

# 3

getNetworkName  Get the name of a network

**Description**

Retrieve the title of a network

**Usage**

getNetworkName(suid = NULL, base.url = .defaultBaseUrl)
getNetworkNDExId

Arguments

suid (optional) SUID of the network; default is current network. If a name is provided, then it is validated and returned.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

network name

Examples

getNetworkName()
getNetworkName(1111)

getNetworkNDExId Get Network NDEx Id

Description

Retrieve the NDEx externalId for a Cytoscape network, presuming it has already been exported to NDEx.

Usage

getNetworkNDExId(network = NULL, base.url = .defaultBaseUrl)

Arguments

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

If the Cytoscape network is not associated with an NDEx network, the return value will be NULL.

Value

NDEx identifier (externalId) or NULL

Examples

getNetworkNDExId()
getNetworkProperty  Get Network Property Values

Description
Get values for any network property.

Usage
getNetworkProperty(visual.property, network = NULL, base.url = .defaultBaseUrl)

Arguments
visual.property
Name of a visual property. See getVisualPropertyNames.

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details
This method retrieves the actual property of the node, given the current visual style, factoring together any default, mapping and bypass setting.

Value
Named list of property values

Examples
getNetworkProperty('NETWORK_SCALE_FACTOR')

getNetworkSuid  Get the SUID of a network

Description
Retrieve the SUID of a network

Usage
getNetworkSuid(title = NULL, base.url = .defaultBaseUrl)
getNetworkViews

Arguments

- title (optional) Name of the network; default is "current" network. If an SUID is provided, then it is validated and returned.
- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

(numeric) Network suid

Author(s)

Alexander Pico

Examples

getNetworkSuid()
getNetworkSuid("myNetwork")
# 80

getNetworkViews

Get Network Views

Description

Retrieve list of network view SUIDs

Usage

getNetworkViews(network = NULL, base.url = .defaultBaseUrl)

Arguments

- network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

List of network view SUIDs

Examples

getNetworkViews()
**getNetworkViewSuid**  
*Get the SUID of a network view*

### Description

Retrieve the SUID of a network view

### Usage

```r
getNetworkViewSuid(network = NULL, base.url = .defaultBaseUrl)
```

### Arguments

- **network** (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape. If a network view SUID is provided, then it is validated and returned.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

### Value

(numeric) Network view suid. The first (presumably only) view associated a network is returned.

### Author(s)

Alexander Pico

### Examples

```r
getNetworkViewSuid()
getNetworkViewSuid("myNetwork")
# 90
```

---

**getNetworkZoom**  
*Get Network Zoom*

### Description

Retrieve the scale factor of specified network.

### Usage

```r
getNetworkZoom(network = NULL, base.url = .defaultBaseUrl)
```
**getNodeColor**

**Arguments**

network (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

Named list of property values

**Examples**

```r
getNodeColor()
```
### getNodeCount  
*Get Node Count*

**Description**
Reports the number of nodes in the network.

**Usage**
```
getNodeCount(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**
- `network` *(optional)* Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
numeric

**Author(s)**
Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**
```
getNodeCount()
```

---

### getNodeHeight  
*Get Node Height*

**Description**
Retrieve the actual height of specified nodes.

**Usage**
```
getNodeHeight(node.names = NULL, network = NULL, base.url = .defaultBaseUrl)
```

**Author(s)**
Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**
```
getNodeHeight()
```
getNodeLabelPosition

Arguments

node.names List of node names or SUIDs. Default is NULL for all nodes.

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Named list of property values

Examples

g getNodeHeight()

------------------

getNodeLabelPosition  Get Node Label Position

Description

Retrieve the actual label position of specified nodes.

Usage

g getNodeLabelPosition(
  node.names = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

node.names List of node names or SUIDs. Default is NULL for all nodes.

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Named list of property values
Examples

getNodeLabelPosition('Node 1')

ggetNodeLabelPositionDefault

Get Node Label Position Default

Description

Retrieve the default selection node color.

Usage

getNodeLabelPositionDefault(style.name = NULL, base.url = .defaultBaseUrl)

Arguments

style.name Name of style; default is "default" style
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

getNodeLabelPositionDefault()

getNodePosition

Get Node Position

Description

Retrieve the actual x,y position of specified nodes.

Usage

getNodePosition(node.names = NULL, network = NULL, base.url = .defaultBaseUrl)
**getNodeProperty**

**Arguments**

- `node.names` List of node names or SUIDs. Default is NULL for all nodes.
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

Dataframe of x, y values

**Examples**

```r
getNodePosition()
```

---

**getNodeProperty**  
*Get Node Property Values*

**Description**

Get values for any node property of the specified nodes.

**Usage**

```r
g getNodeProperty(
  node.names = NULL,
  visual.property,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `node.names` List of node names or SUIDs. Default is NULL for all nodes.
- `visual.property` Name of a visual property. See `getVisualPropertyName`.
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
getNodeSelectionColorDefault

Get Node Selection Color Default

Details

This method retrieves the actual property of the node, given the current visual style, factoring together any default, mapping and bypass setting.

Value

Named list of property values

Examples

getNodeProperty(c('node 0','node 1'),'NODE SHAPE')

getNodeSelectionColorDefault

Get Node Selection Color Default

Description

Retrieve the default selection node color.

Usage

g getNodeSelectionColorDefault(style.name = NULL, base.url = .defaultBaseUrl)

Arguments

<table>
<thead>
<tr>
<th>argument</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>style.name</td>
<td>Name of style; default is &quot;default&quot; style</td>
</tr>
<tr>
<td>base.url</td>
<td>(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://localhost:1234">http://localhost:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

Value

None

Examples

getNodeSelectionColorDefault()
getNodeShapes

Get Node Shapes

Description
Retrieve the names of the currently supported node shapes, which can then be used in calls to
setNodeShapeRule and setDefaultVizMapValue.

Usage
getNodeShapes(base.url = .defaultBaseUrl)

Arguments
base.url (optional) Ignore unless you need to specify a custom domain, port or version
to connect to the CyREST API. Default is http://localhost:1234 and the latest
version of the CyREST API supported by this version of RCy3.

Value
A list of character strings, e.g. 'ELLIPSE', 'RECTANGLE'

Examples
getNodeShapes()

getNodeSize

Get Node Size

Description
Retrieve the actual size of specified nodes.

Usage
getNodeSize(node.names = NULL, network = NULL, base.url = .defaultBaseUrl)

Arguments
node.names List of node names or SUIDs. Default is NULL for all nodes.
network (optional) Name or SUID of the network. Default is the "current" network active
in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version
to connect to the CyREST API. Default is http://localhost:1234 and the latest
version of the CyREST API supported by this version of RCy3.
getNodeWidth

Get Node Width

Value

Named list of property values

Examples

getNodeWidth()
**getNotebookIsRunning**

**Description**

getNotebookIsRunning

**Usage**

getNotebookIsRunning()

**Value**

None

**Examples**

getNotebookIsRunning()

---

**getSandboxReinitialize**

**Description**

Return flag indicating that next command should reinitialize the sandbox according to the default_sandbox.

**Usage**

getSandboxReinitialize()

**Value**

sandbox reinitialize

**Examples**

getSandboxReinitialize()
**getSelectedEdgeCount**  
*Get Selected Edge Count*

**Description**

Returns the number of edges currently selected in the network.

**Usage**

```r
getSelectedEdgeCount(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `network` *(optional)* Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

numeric

**Author(s)**

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**

```r
getSelectedEdgeCount()
```

---

**getSelectedEdges**  
*Get Selected Edges*

**Description**

Retrieve the names of all the edges selected in the network.

**Usage**

```r
getSelectedEdges(
  edge.suids = FALSE,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
getSelectedNodeCount

Arguments

edge.suids  Whether to return edge SUIDs. Default is FALSE to return edge names.

network   (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

list of selected edge names

Examples

getSelectedEdges()

getSelectedNodeCount  Get Selected Node Count

Description

Returns the number of nodes currently selected in the network.

Usage

getSelectedNodeCount(network = NULL, base.url = .defaultBaseUrl)

Arguments

network   (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

numeric

Author(s)

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples

getSelectedNodeCount()
getSelectedNodes

Get Selected Nodes

Description

Retrieve the names of all the nodes selected in the network.

Usage

getSelectedNodes(
    node.suids = FALSE,
    network = NULL,
    base.url = .defaultBaseUrl
)

Arguments

node.suids  Whether to return node SUIDs. Default is FALSE to return node names.

network    (optional) Name or SUID of the network. Default is the "current" network active
            in Cytoscape.

base.url   (optional) Ignore unless you need to specify a custom domain, port or version
            to connect to the CyREST API. Default is http://localhost:1234 and the latest
            version of the CyREST API supported by this version of RCy3.

Value

list of selected node names

Author(s)

AlexanderPico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples

getSelectedNodes()
getStyleDependencies  Get the values of dependencies in a style

Description

Retrieves style dependency settings.

Usage

```r
g gettingeStyleDependencies(style.name = NULL, base.url = .defaultBaseUrl)
```

Arguments

- **style.name**: Name of style; default is "default" style
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is `http://localhost:1234` and the latest version of the CyREST API supported by this version of RCy3.

Value

server response

Available Dependencies

- `arrowColorMatchesEdge`
- `nodeCustomGraphicsSizeSync`
- `nodeSizeLocked`

Examples

```r
g gettingeStyleDependencies("myStyle")
```

getStyleMapping  Get Style Mapping

Description

Retrieve the values the define the mappings for a given visual property in a style.

Usage

```r
g gettingeStyleMapping(visual.prop, style.name = NULL, base.url = .defaultBaseUrl)
```
getTableColumnNames

Arguments

- `visual.prop` Name of the visual property, e.g., NODE_FILL_COLOR
- `style.name` Name for style; default is the 'default' style
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

List of named values defining the visual property mappings

See Also

getVisualPropertyNames

Examples

getTableColumnNames()

getTableColumnNames  Get Table Column Names

Description

Retrieve the names of all columns in a table

Usage

getTableColumnNames(
  table = "node",
  namespace = "default",
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

- `table` name of table, e.g., node, edge, network; default is "node"
- `namespace` namespace of table, e.g., default
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
getTableColumns

Value

list of column names

Examples

getTableColumnNames()
getTableColumnNames('edge')
getTableColumnNames('network')

description

Get table column values

Retrieve one or more columns of data from node, edge or network tables.

Usage

getTableColumns(
  table = "node",
  columns = NULL,
  namespace = "default",
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

table name of table, e.g., node (default), edge, network

columns names of columns to retrieve values from as list object or comma-separated list;
default is all columns

namespace namespace of table; default is "default"

network (optional) Name or SUID of the network. Default is the "current" network active
in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version
to connect to the CyREST API. Default is http://localhost:1234 and the latest
version of the CyREST API supported by this version of RCy3.

details

The 'SUID' column is always retrieved along with specified columns. The 'SUID' values are used
as row.names in the returned data.frame.

Value

A data.frame of column values
Examples

getTableColumns()
getTableColumns('node','group')

getTableColumnTypes Get Table Column Types

Description

Retrieve the types of all columns in a table

Usage

getTableColumnTypes(
  table = "node",
  namespace = "default",
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

table name of table, e.g., node, edge, network; default is "node"
namespace namespace of table, e.g., default
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

a named list of column types

Examples

getTableColumnTypes()
getTableColumnTypes('edge')
getTableColumnTypes('network')
getTableValue  Get table cell value

Description
Retrieve the value from a specific row and column from node, edge or network tables.

Usage
getTableValue(
  table,
  row.name,
  column,
  namespace = "default",
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments
- **table**: name of table, e.g., node, edge, network
- **row.name**: Node, edge or network name, i.e., the value in the "name" column
- **column**: name of column to retrieve values from
- **namespace**: namespace of table; default is "default"
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
A data.frame of column values

Examples
getTableValue('node','node 1', 'score')
getUninstalledApps  List Uninstalled Apps

Description
Retrieve list of apps not currently installed in Cytoscape.

Usage
getUninstalledApps(base.url = .defaultBaseUrl)

Arguments

base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
A list of uninstalled app names, versions and statuses

Examples
getUninstalledApps()

getVisualPropertyDefault  Get Visual Property Default

Description
Retrieve the default value for a visual property.

Usage
getVisualPropertyDefault(
  property,
  style.name = NULL,
  base.url = .defaultBaseUrl
)

### getVisualPropertyNames

#### Description
Retrieve the names of all possible visual properties.

#### Usage
```r
getVisualPropertyNames(base.url = .defaultBaseUrl)
```

#### Arguments
- **base.url** *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

#### Value
List of names

#### Examples
```r
getVisualPropertyNames()
```
### getVisualStyleJSON  
**Get Visual Style JSON**

**Description**
Get all defaults and mappings for a visual style.

**Usage**
```r
getVisualStyleJSON(styleName, css = FALSE, base.url = .defaultBaseUrl)
```

**Arguments**
- **styleName**: name of style
- **css**: TRUE to create a CytoscapeJS CSS style, FALSE to create a generic JSON version. Default is FALSE.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
List of visual style properties

**Examples**
```r
getVisualStyleJSON()
```

### getVisualStyleNames  
**Get Visual Style Names**

**Description**
Retrieve a list of all visual style names.

**Usage**
```r
getVisualStyleNames(base.url = .defaultBaseUrl)
```

**Arguments**
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
Value
A list of names

Examples
getVisualStyleNames()

groupAnnotation  Group Annotation

Description
Group annotation from the network view in Cytoscape

Usage
groupAnnotation(names = NULL, network = NULL, base.url = .defaultBaseUrl)

Arguments
names  Name of annotation by UUID or Name
network  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details
You can obtain a list of UUIDs by applying a subset function like so: sapply(getAnnotationList(), '[', 'uuid')

Value
UUID of group annotation

Examples
groupAnnotation(c("29ac8349-7be4-404e-8363-9537cc39d1ad","3846e949-3130-4362-83de-d02f5368e3ad"))
groupAnnotation(c("annotation1","annotation2", "annotation3"))
groupAnnotation("annotation1")
groupAnnotation(sapply(getAnnotationList(), '[', 'uuid'))
hideAllPanels  

**Description**  
Hide control, table, tool and results panels.

**Usage**  
`hideAllPanels(base.url = .defaultBaseUrl)`

**Arguments**  
- `base.url`  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**  
None

**Examples**  
`hideAllPanels()`

hideEdges  

**Description**  
Hide (but do not delete) the specified edge or edges, by setting the Visible property bypass value to false.

**Usage**  
`hideEdges(edge.names, network = NULL, base.url = .defaultBaseUrl)`

**Arguments**  
- `edge.names`  
  List of edge names or SUIDs
- `network`  
  (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
- `base.url`  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
hideNodes

Details

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `unhideNodes` or `unhideAll`.

Value

None

See Also

`setEdgePropertyBypass`, `hideSelectedEdges`, `unhideEdges`, `unhideAll`

Examples

hideEdges()

hideNodes

Description

Hide (but do not delete) the specified node or nodes, by setting the Visible property bypass value to false.

Usage

hideNodes(node.names, network = NULL, base.url = .defaultBaseUrl)

Arguments

node.names List of node names or SUIDs
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `unhideNodes` or `unhideAll`. 
Value

None

See Also

setNodePropertyBypass, hideSelectedNodes, unhideNodes, unhideAll

Examples

hideNodes()
**hideSelectedEdges**

**Description**

Hide (but do not delete) the currently selected edges, by setting the Visible property bypass value to false.

**Usage**

`hideSelectedEdges(network = NULL, base.url = .defaultBaseUrl)`

**Arguments**

- **network** (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `unhideEdges` or `unhideAll`.

**Value**

None

**See Also**

`setEdgePropertyBypass`, `hideEdges`, `unhideEdges`, `unhideAll`

**Examples**

`hideSelectedEdges()`
hideSelectedNodes  
*Hide Selected Nodes*

**Description**

Hide (but do not delete) the currently selected nodes, by setting the Visible property bypass value to false.

**Usage**

`hideSelectedNodes(network = NULL, base.url = .defaultBaseUrl)`

**Arguments**

- **network** (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `unhideNodes` or `unhideAll`.

**Value**

None

**See Also**

`setNodePropertyBypass`, `hideNodes`, `unhideNodes`, `unhideAll`

**Examples**

`hideSelectedNodes()`
**Description**

The source URL identifies a file to be transferred from a cloud resource to either the current Cytoscape directory (if executing on the Cytoscape workstation) or sandbox (if executing on a remote server or a sandbox was explicitly created). If the destination file already exists, it is overwritten. The 'destFile' can be an absolute path if the workflow is executing on the local Cytoscape workstation. Supported URLs include: Raw URL: URL directly references the file to download (e.g., http://tpsoft.com/museum_images/IBM Dropbox: Use the standard Dropbox 'Get Link' feature to create the 'sourceUrl' link in the clipboard. GDrive: Use the standard Google Drive 'Get Link' feature to create the 'sourceUrl' link in the clipboard. OneDrive: Use the OneDrive web site to right click on the file, choose the 'Embed' menu option, then copy the URL in the iframe’s “src” parameter into the clipboard. GitHub: Use the GitHub web site to show the file or a link to it, and capture the URL in the clipboard. Note that GitHub enforces a limit on the size of a file that can be stored there. We advise that you take this into account when choosing a cloud service for your files. When you capture a URL in the clipboard, you should copy it into your program for use with this function. This function is most useful for Notebooks running on the local Cytoscape workstation. For Notebooks that could run on a remote server, consider using sandboxUrlTo() and related sandbox functions.

**Usage**

```
importFileFromUrl(
  sourceURL, 
  destFile, 
  overwrite = TRUE, 
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **sourceURL**: URL addressing cloud file to download
- **destFile**: Name of file in the R workflow’s file system. If None, use file name in source_file
- **overwrite**: Name of sandbox containing file. None means "the current sandbox".
- **base.url**: Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://127.0.0.1:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

dict: 'filePath': <new file’s absolute path in Cytoscape workstation>, 'fileByteCount': number of bytes read
**importFilters**  
*Import Filters*

**Description**  
Loads filters from a file in JSON format.

**Usage**  
```
importFilters(filename, base.url = .defaultBaseUrl)
```

**Arguments**
- **filename** (char) Path and name of the filters file to load.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**  
None

**Examples**
```
importFilters()
```

---

**importNetworkFromFile**  
*Import Network From File*

**Description**  
Loads a network from specified file

**Usage**  
```
importNetworkFromFile(file = NULL, base.url = .defaultBaseUrl)
```

**Examples**
```
importNetworkFromFile()
```
importNetworkFromNDEx

Arguments

- **file**: Name of file in any of the supported formats (e.g., SIF, GML, xGMML, etc). If NULL, a demo network file in SIF format is loaded.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

- (int) network SUID

Examples

importNetworkFromFile()

---

**importNetworkFromNDEx**  
**Import Network From NDEx**

Description

Import a network from the NDEx database into Cytoscape.

Usage

```r
importNetworkFromNDEx(
  ndex.id,
  username = NULL,
  password = NULL,
  accessKey = NULL,
  ndex.url = "http://ndexbio.org",
  ndex.version = "v2",
  base.url = .defaultBaseUrl
)
```

Arguments

- **ndex.id**: Network externalId provided by NDEx. This is not the same as a Cytoscape SUID.
- **username** (optional) NDEx account username; required for private content
- **password** (optional) NDEx account password; required for private content
- **accessKey** (optional) NDEx accessKey; alternate access to private content
- **ndex.url** (optional) For alternative instances or deployments of NDEx; default is "http://ndexbio.org"
- **ndex.version** (optional) For alternative versions of the NDEx API; default is "v2"
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
importVisualStyles

Value

(integer) SUID of imported network

Examples

importNetworkFromNDEx(ndex.id)

importVisualStyles  Import Visual Styles

Description

Loads styles from an XML file and returns the names of the loaded styles.

Usage

importVisualStyles(filename = "styles.xml", base.url = .defaultBaseUrl)

Arguments

filename  (char) Name of the style file to load. Only reads XML files. Default is "styles.xml".
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

(list) Names of styles loaded

See Also

exportVisualStyles

Examples

importVisualStyles()
installApp

Description
Installs an app in Cytoscape.

Usage
installApp(app, base.url = .defaultBaseUrl)

Arguments
app Name of app
base.url (optional) Ignore unless you need to specify a custom domain, port or version
to connect to the CyREST API. Default is http://localhost:1234 and the latest
version of the CyREST API supported by this version of RCy3.

Value
None

Examples
installApp()

invertEdgeSelection

Description
Select all edges that were not selected and deselect all edges that were selected.

Usage
invertEdgeSelection(network = NULL, base.url = .defaultBaseUrl)

Arguments
network (optional) Name or SUID of the network. Default is the “current” network active
in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version
to connect to the CyREST API. Default is http://localhost:1234 and the latest
version of the CyREST API supported by this version of RCy3.
invertNodeSelection

**Value**

list of newly selected edge SUIDs

**Author(s)**

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**

invertEdgeSelection()

invertNodeSelection

**Description**

Select all nodes that were not selected and deselect all nodes that were selected.

**Usage**

invertNodeSelection(network = NULL, base.url = .defaultBaseUrl)

**Arguments**

- network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

list of newly selected node SUIDs

**Author(s)**

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**

invertNodeSelection()
**layoutCopycat**  
*Copy a layout from one network to another*

**Description**
Sets the coordinates for each node in the target network to the coordinates of a matching node in the source network.

**Usage**

```r
layoutCopycat(
  sourceNetwork,  # (character) The name of network to get node coordinates from
  targetNetwork,  # (character) The name of the network to apply coordinates to
  sourceColumn = "name",  # (optional character) The name of column in the sourceNetwork node table used to match nodes; default is 'name'
  targetColumn = "name",  # (optional character) The name of column in the targetNetwork node table used to match nodes; default is 'name'
  gridUnmapped = TRUE,  # (optional character) If this is set to true, any nodes in the target network that could not be matched to a node in the source network will be laid out in a grid; default is TRUE
  selectUnmapped = TRUE,  # (optional character) If this is set to true, any nodes in the target network that could not be matched to a node in the source network will be selected in the target network; default is TRUE
  base.url = .defaultBaseUrl  # (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
)
```

**Arguments**

- **sourceNetwork** (character) The name of network to get node coordinates from
- **targetNetwork** (character) The name of the network to apply coordinates to
- **sourceColumn** (optional character) The name of column in the sourceNetwork node table used to match nodes; default is 'name'
- **targetColumn** (optional character) The name of column in the targetNetwork node table used to match nodes; default is 'name'
- **gridUnmapped** (optional character) If this is set to true, any nodes in the target network that could not be matched to a node in the source network will be laid out in a grid; default is TRUE
- **selectUnmapped** (optional character) If this is set to true, any nodes in the target network that could not be matched to a node in the source network will be selected in the target network; default is TRUE
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**
Optional parameters such as `gridUnmapped` and `selectUnmapped` determine the behavior of target network nodes that could not be matched.

**Value**
None
Examples

layoutCopycat('network1', 'network2')

layoutNetwork

**Apply a layout to a network**

Description

Apply a layout to a network

Usage

layoutNetwork(layout.name = NULL, network = NULL, base.url = .defaultBaseUrl)

Arguments

- **layout.name** (character) Name of the layout (with optional parameters). If not specified, then the preferred layout set in the Cytoscape UI is applied.
- **network** (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Run getLayoutNames to list available layouts.

Value

None

Examples

layoutNetwork()
layoutNetwork('force-directed')
layoutNetwork('force-directed defaultSpringCoefficient=.00006 defaultSpringLength=80')
**listGroups**

List Groups

**Description**

Retrieve a list of all group SUIDs in a network.

**Usage**

`listGroups(network = NULL, base.url = .defaultBaseUrl)`

**Arguments**

- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

List of group SUIDs

**Examples**

`listGroups()`

---

**loadTableData**

*Loads data into Cytoscape tables keyed by row*

**Description**

This function loads data into Cytoscape node/edge/network tables provided a common key, e.g., name. Data.frame column names will be used to set Cytoscape table column names.

**Usage**

```r
loadTableData(
  data,
  data.key.column = "row.names",
  table = "node",
  table.key.column = "name",
  namespace = "default",
  network = NULL,
  base.url = .defaultBaseUrl
)
```
lockNodeDimensions

**Arguments**

- `data` (data.frame) each row is a node and columns contain node attributes
- `data.key.column` (char) name of data.frame column to use as key; default is "row.names"
- `table` (char) name of Cytoscape table to load data into, e.g., node, edge or network; default is "node"
- `table.key.column` (char) name of Cytoscape table column to use as key; default is "name"
- `namespace` namespace of table, e.g., default
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

Numeric values will be stored as Doubles in Cytoscape tables. Integer values will be stored as Integers. Character or mixed values will be stored as Strings. Logical values will be stored as Boolean. Lists are stored as Lists by CyREST v3.9+. Existing columns with the same names will keep original type but values will be overwritten.

**Value**

server response

---

**Description**

Set a boolean value to have node width and height fixed to a single size value.

**Usage**

lockNodeDimensions(new.state, style.name = NULL, base.url = .defaultBaseUrl)

**Arguments**

- `new.state` (Boolean) Whether to lock node width and height
- `style.name` Name of style; default is "default" style
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
**makeSimpleGraph**

**Value**

None

**Examples**

lockNodeDimensions(TRUE)

---

**Description**

This function creates a simple graphNEL object with various node and edge attribute types to help demonstrate round trip conversion with Cytoscape networks via RCy3.

**Usage**

makeSimpleGraph()

**Value**

A graphNEL object with a few nodes, edges and attributes

**See Also**

createNetworkFromGraph, createGraphFromNetwork, makeSimpleIgraph

**Examples**

```
{
  makeSimpleGraph()
}
```

---

**makeSimpleIgraph**

**Make Simple Igraph**

**Description**

This function creates a simple iGraph object with various node and edge attribute types to help demonstrate round trip conversion with Cytoscape networks via RCy3.

**Usage**

makeSimpleIgraph()
mapTableColumn

Value

A igraph object with a few nodes, edges and attributes

See Also

createNetworkFromIgraph, createIgraphFromNetwork, makeSimpleGraph

Examples

```
{
  makeSimpleIgraph()
}
```

mapTableColumn  Map Table Column

Description

Perform identifier mapping using an existing column of supported identifiers to populate a new column with identifiers mapped to the originals.

Usage

```r
mapTableColumn(
  column, 
  species, 
  map.from, 
  map.to, 
  force.single = TRUE, 
  table = "node", 
  namespace = "default", 
  network = NULL, 
  base.url = .defaultBaseUrl
)
```

Arguments

- **column**: Name of column containing identifiers of type specified by `map.from`
- **species**: Common name for species associated with identifiers, e.g., Human. See details.
- **map.from**: Type of identifier found in specified column. See details.
- **map.to**: Type of identifier to populate in new column. See details.
- **force.single**: (optional) Whether to return only first result in cases of one-to-many mappings; otherwise the new column will hold lists of identifiers. Default is TRUE.
- **table**: (optional) Name of table, e.g., node (default), edge or network
- **namespace**: (optional) Namespace of table, e.g., default (default), shared or hidden
mapVisualProperty

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Supported species: Human, Mouse, Rat, Frog, Zebrafish, Fruit fly, Mosquito, Worm, Arabidopsis thaliana, Yeast, E. coli, Tuberculosis.

Supported identifier types (depending on species): Ensembl, Entrez Gene, Uniprot-TrEMBL, miRBase, UniGene, HGNC (symbols), MGI, RGD, SGD, ZFIN, FlyBase, WormBase, TAIR.

Value
dataframe with map.from and map.to columns. Beware: if map.to is not unique, it will be suffixed with an incrementing number in parentheses, e.g., if mapIdentifiers is repeated on the same network. However, the original map.to column will be returned regardless.

Examples

mapped.cols <- mapTableColumn('name', 'Yeast', 'Ensembl', 'SGD')
#   name SGD
#17920  YER145C S000000947
#17921  YMR058W S000004662
#17922  YJL190C S000003726
#...

mapVisualProperty Creates a mapping between an attribute and a visual property

Description

Generates the appropriate data structure for the "mapping" parameter in updateStyleMapping.

Usage

mapVisualProperty(
    visual.prop,
    table.column,
    mapping.type,
    table.column.values,
    visual.prop.values,
    network = NULL,
    base.url = .defaultBaseUrl
)
Arguments

visual.prop  (char) name of visual property to map
table.column  (char) name of table column to map
mapping.type  (char) continuous, discrete or passthrough (c,d,p)
table.column.values  (list) list of values paired with visual.prop.values; skip for passthrough mapping
visual.prop.values  (list) list of values paired with table.column.values; skip for passthrough mapping
network  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

The paired list of values must be of the same length or mapping will fail. For gradient mapping, you may include two additional visual.prop.values in the first and last positions to map respectively to values less than and greater than those specified in table.column.values. Mapping will also fail if the data type of table.column.values does not match that of the existing table.column. Note that all imported numeric data are stored as Integers or Doubles in Cytoscape tables; and character or mixed data are stored as Strings.

Value

(network=network, base.url=base.url) ready to convert into JSON by style mapping operations

List of visual properties

<table>
<thead>
<tr>
<th>Node Border Stroke</th>
<th>Edge Bend</th>
<th>Network Background Paint</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node Border Paint</td>
<td>Edge Curved</td>
<td>Network Center X Location</td>
</tr>
<tr>
<td>Node Border Transparency</td>
<td>Edge Label</td>
<td>Network Center Y Location</td>
</tr>
<tr>
<td>Node Border Width</td>
<td>Edge Label Color</td>
<td>Network Center Z Location</td>
</tr>
<tr>
<td>Node CustomGraphics 1-9</td>
<td>Edge Label Font Face</td>
<td>Network Depth</td>
</tr>
<tr>
<td>Node CustomGraphics Position 1-9</td>
<td>Edge Label Font Size</td>
<td>Network Edge Selection</td>
</tr>
<tr>
<td>Node CustomGraphics Size 1-9</td>
<td>Edge Label Transparency</td>
<td>Network Height</td>
</tr>
<tr>
<td>Node CustomPaint 1-9</td>
<td>Edge Label Width</td>
<td>Network Node Selection</td>
</tr>
<tr>
<td>Node Depth</td>
<td>Edge Line Type</td>
<td>Network Scale Factor</td>
</tr>
<tr>
<td>Node Fill Color</td>
<td>Edge Paint</td>
<td>Network Size</td>
</tr>
<tr>
<td>Node Height</td>
<td>Edge Selected</td>
<td>Network Title</td>
</tr>
<tr>
<td>Node Label</td>
<td>Edge Selected Paint</td>
<td>Network Width</td>
</tr>
<tr>
<td>Node Label Color</td>
<td>Edge Source Arrow Selected Paint</td>
<td></td>
</tr>
<tr>
<td>Node Label Font Face</td>
<td>Edge Source Arrow Shape</td>
<td></td>
</tr>
<tr>
<td>Node Label Font Size</td>
<td>Edge Source Arrow Size</td>
<td></td>
</tr>
<tr>
<td>Node Label Position</td>
<td>Edge Source Arrow Unselected Paint</td>
<td></td>
</tr>
<tr>
<td>Node Label Transparency</td>
<td>Edge Stroke Selected Paint</td>
<td></td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------------------------</td>
<td></td>
</tr>
<tr>
<td>Node Label Width</td>
<td>Edge Stroke Unselected Paint</td>
<td></td>
</tr>
<tr>
<td>Node Network Image Visible</td>
<td>Edge Target Arrow Selected Paint</td>
<td></td>
</tr>
<tr>
<td>Node Paint</td>
<td>Edge Target Arrow Shape</td>
<td></td>
</tr>
<tr>
<td>Node Selected</td>
<td>Edge Target Arrow Size</td>
<td></td>
</tr>
<tr>
<td>Node Selected Paint</td>
<td>Edge Target Arrow Unselected Paint</td>
<td></td>
</tr>
<tr>
<td>Node Shape</td>
<td>Edge Tooltip</td>
<td></td>
</tr>
<tr>
<td>Node Size</td>
<td>Edge Transparency</td>
<td></td>
</tr>
<tr>
<td>Node Tooltip</td>
<td>Edge Unselected Paint</td>
<td></td>
</tr>
<tr>
<td>Node Transparency</td>
<td>Edge Visible</td>
<td></td>
</tr>
<tr>
<td>Node Visible</td>
<td>Edge Visual Property</td>
<td></td>
</tr>
<tr>
<td>Node Width</td>
<td>Edge Width</td>
<td></td>
</tr>
<tr>
<td>Node X Location</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Node Y Location</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Node Z Location</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**See Also**

updateStyleMapping getVisualPropertyNames

**Examples**

```r
mapVisualProperty('node fill color','score','c',c(-4.0,0.0,9.0),c('#99CCFF','#FFFFFF','#FF7777'))
mapVisualProperty('node shape','type','d',c('protein','metabolite'),c('ellipse','rectangle'))
mapVisualProperty('node label','alias','p')
```

---

**matchArrowColorToEdge**  
*Match Arrow Color To Edge*

**Description**

Set a boolean value to have arrow shapes share the same color as the edge.

**Usage**

```r
matchArrowColorToEdge(new.state, style.name = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `new.state` (Boolean) Whether to match arrow color to edge.
- `style.name` Name of style; default is "default" style
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
mergeNetworks

Value
None

Examples
matchArrowColorToEdge(TRUE)

mergeNetworks  Merge Networks

Description
Combine networks via union, intersection, or difference operations. Lots of optional parameters choose from!

Usage
mergeNetworks(
  sources = NULL,
  title = NULL,
  operation = "union",
  nodeKeys = NULL,
  nodeMergeMap = NULL,
  nodesOnly = FALSE,
  edgeKeys = NULL,
  edgeMergeMap = NULL,
  networkMergeMap = NULL,
  inNetworkMerge = TRUE,
  base.url = .defaultBaseUrl
)

Arguments

  sources     List of network names (max:2) to be merged.
  title       (optional) Title of the resulting merged network. Default is a concatenation of operation and source network titles.
  operation   (optional) Type of merge: union (default), intersection or difference.
  nodeKeys    (optional) An order-dependent list of columns to match nodes across source networks. Default is "name" column for all sources.
  nodeMergeMap (optional) A list of column merge records specifying how to merge node table data. Each record should be of the form: c("network1 column", "network2 column", "merged column", "type"), where column names are provided and type is String, Integer, Double or List.
  nodesOnly   (optional) If TRUE, this will merge the node tables and ignore edge and network table data. Default is FALSE.
edgeKeys (optional) An order-dependent list of columns to match edges across source networks. Default is "name" column for all sources.

edgeMergeMap (optional) A list of column merge records specifying how to merge edge table data. Each record should be of the form: c("network1 column", "network2 column", "merged column", "type"), where column names are provided and type is String, Integer, Double or List.

networkMergeMap (optional) A list of column merge records specifying how to merge network table data. Each record should be of the form: c("network1 column", "network2 column", "merged column", "type"), where column names are provided and type is String, Integer, Double or List.

inNetworkMerge (optional) If TRUE (default), nodes and edges with matching attributes in the same network will be merged.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
SUID of resulting merged network

Examples
mergeNetworks(c("Network 1", "Network 2"), "Merged Network")
mergeNetworks(c("my network","string network"), "Merged Network", nodeKeys=c("HGNC","query term"))

Description
Show network view in notebook output.

Usage

notebookExportShowImage(
  filename = "image",
  type = "PNG",
  resolution = NULL,
  units = NULL,
  height = NULL,
  width = NULL,
  zoom = NULL,
sandboxName = NULL,
network = NULL,
base.url = .defaultBaseUrl,
overwriteFile = TRUE
)

Arguments

filename (character) Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. If blank, the current network name will be used.

type (character) Type of image to export, e.g., PNG (default), JPEG, PDF, SVG.

resolution (numeric) The resolution of the exported image, in DPI. Valid only for bitmap formats, when the selected width and height 'units' is inches. The possible values are: 72 (default), 100, 150, 300, 600.

units (character) The units for the 'width' and 'height' values. Valid only for bitmap formats, such as PNG and JPEG. The possible values are: pixels (default), inches.

height (numeric) The height of the exported image. Valid only for bitmap formats, such as PNG and JPEG.

width (numeric) The width of the exported image. Valid only for bitmap formats, such as PNG and JPEG.

zoom (numeric) The zoom value to proportionally scale the image. The default value is 100.0. Valid only for bitmap formats, such as PNG and JPEG

sandboxName (optional) Name of sandbox containing file. None means "the current sandbox".

network (optional) Name or SUID of a network or view. Default is the "current" network active in Cytoscape. The first (presumably only) view associated a network is used if a specific view SUID is not provided.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

overwriteFile (optional) FALSE allows Cytoscape show a message box before overwriting the file if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.

Details

Show network view in notebook output.

Value

display image

Examples

notebookExportShowImage()
Description

Show network view in notebook output.

Usage

```r
notebookShowImage(
    filename = "image",
    type = "PNG",
    resolution = NULL,
    units = NULL,
    height = NULL,
    width = NULL,
    zoom = NULL,
    sandboxName = NULL,
    network = NULL,
    base.url = .defaultBaseUrl,
    overwriteFile = TRUE
)
```

Arguments

- **filename** (character) Full path or path relative to current working directory, in addition to the name of the file. Extension is automatically added based on the type argument. If blank, the current network name will be used.
- **type** (character) Type of image to export, e.g., PNG (default), JPEG, PDF, SVG.
- **resolution** (numeric) The resolution of the exported image, in DPI. Valid only for bitmap formats, when the selected width and height 'units' is inches. The possible values are: 72 (default), 100, 150, 300, 600.
- **units** (character) The units for the 'width' and 'height' values. Valid only for bitmap formats, such as PNG and JPEG. The possible values are: pixels (default), inches.
- **height** (numeric) The height of the exported image. Valid only for bitmap formats, such as PNG and JPEG.
- **width** (numeric) The width of the exported image. Valid only for bitmap formats, such as PNG and JPEG.
- **zoom** (numeric) The zoom value to proportionally scale the image. The default value is 100.0. Valid only for bitmap formats, such as PNG and JPEG
- **sandboxName** Name of sandbox containing file. None means "the current sandbox".
- **network** (optional) Name or SUID of a network or view. Default is the "current" network active in Cytoscape. The first (presumably only) view associated a network is used if a specific view SUID is not provided.
openAppStore

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

overwriteFile (optional) FALSE allows Cytoscape show a message box before overwriting the file if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.

Details
Show network view in notebook output.

Value
display image

Examples
notebookShowImage()

openAppStore Open App Store Page

Description
Opens the Cytoscape App Store in a new tab in your default browser.

Usage
openAppStore(app, base.url = .defaultBaseUrl)

Arguments
app Name of app
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples
openAppStore()
openSession

openSession  

Open Session File or URL

Description

Open a session file or URL. This will clear all networks, tables and styles associated with current session. Be sure to saveSession first.

Usage

openSession(file.location = NULL, base.url = .defaultBaseUrl)

Arguments

file.location  File path or URL (with 'http' or 'https' prefix). Default is a sample session file.
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

See Also

saveSession

Examples

openSession(’/fullpath/mySession.CYS’)

paletteColorBrewerAccent

paletteColorBrewerAccent  Qualitative

Description

Generate a qualitative Accent Brewer palette of a given size

Usage

paletteColorBrewerAccent(value.count = 3)
Arguments

- `value.count` Number of colors to generate; min is 3 (default); max is 8. See RColorBrewer::display.brewer.all()

Value

List of palette colors

Examples

```r
paletteColorBrewerBlues()
```
### paletteColorBrewerBrBG

**paletteColorBrewerBrBG Divergent**

**Description**

Generate a divergent BrBG Brewer palette of a given size

**Usage**

```
paletteColorBrewerBrBG(value.count = 3)
```

**Arguments**

- `value.count`: Number of colors to generate; min is 3 (default); max is 9. See `RColorBrewer::display.brewer.all()`

**Value**

List of palette colors

**Examples**

```
paletteColorBrewerBrBG()
```

---

### paletteColorBrewerBuGn

**paletteColorBrewerBuGn Sequential**

**Description**

Generate a sequential BuGn Brewer palette of a given size

**Usage**

```
paletteColorBrewerBuGn(value.count = 3)
```

**Arguments**

- `value.count`: Number of colors to generate; min is 3 (default); max is 9. See `RColorBrewer::display.brewer.all()`

**Value**

List of palette colors
Examples

```r
paletteColorBrewerBuGn()
```

----------

```r
paletteColorBrewerBuPu

paletteColorBrewerBuPu Sequential
```

Description

Generate a sequential BuPu Brewer palette of a given size

Usage

```r
paletteColorBrewerBuPu(value.count = 3)
```

Arguments

- `value.count`: Number of colors to generate; min is 3 (default); max is 9. See `RColorBrewer::display.brewer.all()`

Value

List of palette colors

Examples

```r
paletteColorBrewerBuPu()
```

----------

```r
paletteColorBrewerDark2

paletteColorBrewerDark2 Qualitative
```

Description

Generate a qualitative Dark2 Brewer palette of a given size

Usage

```r
paletteColorBrewerDark2(value.count = 3)
```

Arguments

- `value.count`: Number of colors to generate; min is 3 (default); max is 8. See `RColorBrewer::display.brewer.all()`
### `paletteColorBrewerGnBu`

**Value**

List of palette colors

**Examples**

`paletteColorBrewerDark2()`

---

#### `paletteColorBrewerGnBu`

---

**Description**

Generate a sequential GnBu Brewer palette of a given size

**Usage**

`paletteColorBrewerGnBu(value.count = 3)`

**Arguments**

- `value.count` Number of colors to generate; min is 3 (default); max is 9. See `RColorBrewer::display.brewer.all()`

**Value**

List of palette colors

**Examples**

`paletteColorBrewerGnBu()`

---

#### `paletteColorBrewerGreens`

---

**Description**

Generate a sequential Greens Brewer palette of a given size

**Usage**

`paletteColorBrewerGreens(value.count = 3)`
Arguments

value.count  Number of colors to generate; min is 3 (default); max is 9. See \texttt{RColorBrewer::display.brewer.all()}

Value

List of palette colors

Examples

\texttt{paletteColorBrewerGreys()}

---

\texttt{paletteColorBrewerGreys}

\textit{paletteColorBrewerGreys Sequential}

Description

Generate a sequential Greys Brewer palette of a given size

Usage

\texttt{paletteColorBrewerGreys(value.count = 3)}

Arguments

value.count  Number of colors to generate; min is 3 (default); max is 9. See \texttt{RColorBrewer::display.brewer.all()}

Value

List of palette colors

Examples

\texttt{paletteColorBrewerGreys()}
paletteColorBrewerOranges

**Description**
Generate a sequential Oranges Brewer palette of a given size

**Usage**
```
paletteColorBrewerOranges(value.count = 3)
```

**Arguments**
- `value.count` Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

**Value**
List of palette colors

**Examples**
```
paletteColorBrewerOranges()
```

---

paletteColorBrewerOrRd

**Description**
Generate a sequential OrRd Brewer palette of a given size

**Usage**
```
paletteColorBrewerOrRd(value.count = 3)
```

**Arguments**
- `value.count` Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

**Value**
List of palette colors
Examples

```r
paletteColorBrewerOrRd()
```

---

```r
paletteColorBrewerPaired
```

**paletteColorBrewerPaired Qualitative**

**Description**
Generate a qualitative Paired Brewer palette of a given size

**Usage**
```r
paletteColorBrewerPaired(value.count = 3)
```

**Arguments**
- `value.count` Number of colors to generate; min is 3 (default); max is 12. See RColorBrewer::display.brewer.all()

**Value**
List of palette colors

**Examples**
```r
paletteColorBrewerPaired()
```

---

```r
paletteColorBrewerPastel1
```

**paletteColorBrewerPastel1 Qualitative**

**Description**
Generate a qualitative Pastel1 Brewer palette of a given size

**Usage**
```r
paletteColorBrewerPastel1(value.count = 3)
```

**Arguments**
- `value.count` Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()
paletteColorBrewerPastel2

Value
   List of palette colors

Examples
   paletteColorBrewerPastel2()

---

paletteColorBrewerPastel2
   `paletteColorBrewerPastel2` Qualitative

Description
   Generate a qualitative Pastel2 Brewer palette of a given size

Usage
   paletteColorBrewerPastel2(value.count = 3)

Arguments
   value.count   Number of colors to generate; min is 3 (default); max is 8. See RColorBrewer::display.brewer.all()

Value
   List of palette colors

Examples
   paletteColorBrewerPastel2()

---

paletteColorBrewerPiYG
   `paletteColorBrewerPiYG` Divergent

Description
   Generate a divergent PiYG Brewer palette of a given size

Usage
   paletteColorBrewerPiYG(value.count = 3)
Arguments

value.count  Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

Value

List of palette colors

Examples

paletteColorBrewerPiYG()

paletteColorBrewerPRGn

paletteColorBrewerPRGn Divergent

Description

Generate a divergent PRGn Brewer palette of a given size

Usage

paletteColorBrewerPRGn(value.count = 3)

Arguments

value.count  Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

Value

List of palette colors

Examples

paletteColorBrewerPRGn()
paletteColorBrewerPuBu

polluteColorBrewerPuBu Sequential

Description
Generate a sequential PuBu Brewer palette of a given size

Usage
paletteColorBrewerPuBu(value.count = 3)

Arguments
value.count Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

Value
List of palette colors

Examples
paletteColorBrewerPuBu()

paletteColorBrewerPuBuGn

polluteColorBrewerPuBuGn Sequential

Description
Generate a sequential PuBuGn Brewer palette of a given size

Usage
paletteColorBrewerPuBuGn(value.count = 3)

Arguments
value.count Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

Value
List of palette colors
Examples

`paletteColorBrewerPuBuGn()`

---

`paletteColorBrewerPuOr`

`paletteColorBrewerPuOr Divergent`

Description

Generate a divergent PuOr Brewer palette of a given size

Usage

`paletteColorBrewerPuOr(value.count = 3)`

Arguments

value.count Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

Value

List of palette colors

Examples

`paletteColorBrewerPuOr()`

---

`paletteColorBrewerPuRd`

`paletteColorBrewerPuRd Sequential`

Description

Generate a sequential PuRd Brewer palette of a given size

Usage

`paletteColorBrewerPuRd(value.count = 3)`

Arguments

value.count Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()`
Value

List of palette colors

Examples

paletteColorBrewerPuRd()

paletteColorBrewerPurples

Description

Generate a sequential Purples Brewer palette of a given size

Usage

paletteColorBrewerPurples(value.count = 3)

Arguments

value.count Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

Value

List of palette colors

Examples

paletteColorBrewerPurples()

paletteColorBrewerRdBu

Description

Generate a divergent RdBu Brewer palette of a given size

Usage

paletteColorBrewerRdBu(value.count = 3)
paletteColorBrewerRdPu

Arguments

value.count  Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

Value

List of palette colors

Examples

paletteColorBrewerRdPu()
Description
Generate a divergent RdYlBu Brewer palette of a given size

Usage
```r
paletteColorBrewerRdYlBu(value.count = 3)
```

Arguments
```r
value.count Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()
```

Value
List of palette colors

Examples
```r
paletteColorBrewerRdYlBu()
```

Description
Generate a sequential Reds Brewer palette of a given size

Usage
```r
paletteColorBrewerReds(value.count = 3)
```

Arguments
```r
value.count Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()
```

Value
List of palette colors
Examples

```r
paletteColorBrewerReds()
```

---

`paletteColorBrewerSet1`  
**paletteColorBrewerSet1 Qualitative**

**Description**

Generate a qualitative Set1 Brewer palette of a given size

**Usage**

```r
paletteColorBrewerSet1(value.count = 3)
```

**Arguments**

- `value.count`: Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

**Value**

List of palette colors

**Examples**

```r
paletteColorBrewerSet1()
```

---

`paletteColorBrewerSet2`  
**paletteColorBrewerSet2 Qualitative**

**Description**

Generate a qualitative Set2 Brewer palette of a given size

**Usage**

```r
paletteColorBrewerSet2(value.count = 3)
```

**Arguments**

- `value.count`: Number of colors to generate; min is 3 (default); max is 8. See RColorBrewer::display.brewer.all()
paletteColorBrewerSet3

Value
List of palette colors

Examples
paletteColorBrewerSet2()

paletteColorBrewerSet3
paletteColorBrewerSet3 Qualitative

Description
Generate a qualitative Set3 Brewer palette of a given size

Usage
paletteColorBrewerSet3(value.count = 3)

Arguments
value.count Number of colors to generate; min is 3 (default); max is 12. See RColorBrewer::display.brewer.all()

Value
List of palette colors

Examples
paletteColorBrewerSet3()

paletteColorBrewerYlGn
paletteColorBrewerYlGn Sequential

Description
Generate a sequential YlGn Brewer palette of a given size

Usage
paletteColorBrewerYlGn(value.count = 3)
Arguments

value.count  Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

Value

List of palette colors

Examples

paletteColorBrewerYlGnBu()
**paletteColorBrewerYlOrBr**

### Sequential

**Description**
Generate a sequential YlOrBr Brewer palette of a given size

**Usage**
```
paletteColorBrewerYlOrBr(value.count = 3)
```

**Arguments**
- `value.count` Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

**Value**
List of palette colors

**Examples**
```
paletteColorBrewerYlOrBr()
```

---

**paletteColorBrewerYlOrRd**

### Sequential

**Description**
Generate a sequential YlOrRd Brewer palette of a given size

**Usage**
```
paletteColorBrewerYlOrRd(value.count = 3)
```

**Arguments**
- `value.count` Number of colors to generate; min is 3 (default); max is 9. See RColorBrewer::display.brewer.all()

**Value**
List of palette colors
Examples

paletteColorBrewerYlOrRd()

paletteColorRandom  

paletteColorRandom Qualitative

Description

Generate a qualitative random color map of a given size

Usage

paletteColorRandom(value.count = 1)

Arguments

value.count  
Number of colors to generate; default is 1

Value

List of random colors

See Also

genNodeColorMap genEdgeColorMap

Examples

paletteColorRandom()

RCy3  

RCy3: Functions to Access and Control Cytoscape

Description

Vizualize, analyze and explore networks using Cytoscape via R.

Details

To learn more about RCy3, start with the vignettes: browseVignettes("RCy3")
**RemoveFromGroup**

**Description**

Removes the specified nodes and edges from the specified group.

**Usage**

```r
RemoveFromGroup(
  group.name,  # Specifies the name used to identify the group
  nodes = NULL,   # List of node SUIDs, names, other column values, or keyword: selected, unselected or all. Default is currently selected nodes.
  nodes.by.col = "SUID",  # name of node table column corresponding to provided nodes list. Default is 'SUID'.
  edges = NULL,  # List of edge SUIDs, names, other column values, or keyword: selected, unselected or all. Default is currently selected edges.
  edges.by.col = "SUID",  # name of edge table column corresponding to provided edges list. Default is 'SUID'.
  network = NULL,  # (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
  base.url = .defaultBaseUrl  # (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
)
```

**Arguments**

- **group.name**: Specifies the name used to identify the group
- **nodes**: List of node SUIDs, names, other column values, or keyword: selected, unselected or all. Default is currently selected nodes.
- **nodes.by.col**: name of node table column corresponding to provided nodes list. Default is 'SUID'.
- **edges**: List of edge SUIDs, names, other column values, or keyword: selected, unselected or all. Default is currently selected edges.
- **edges.by.col**: name of edge table column corresponding to provided edges list. Default is 'SUID'.
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
RemoveFromGroup('myGroup')
```
renameNetwork

Remove a network

Description
Sets a new name for this network

Usage
renameNetwork(title, network = NULL, base.url = .defaultBaseUrl)

renameNodeCustomGraphics

Remove Node Custom Graphics

Description
Remove the default custom charts, images and gradients.

Usage
renameNodeCustomGraphics(
  slot = 1,
  style.name = NULL,
  base.url = .defaultBaseUrl
)

Arguments
- slot (optional) Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.
- style.name (optional) Name of style; default is "default" style
- base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples
renameNodeCustomGraphics()
renameTableColumn

Arguments

title New name for the network

network (optional) Name or suid of the network that you want to rename; default is "current" network

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Duplicate network names are not allowed

Value

None

Author(s)

Alexander Pico, Julia Gustavsen

Examples

renameNetwork("renamed network")

renameTableColumn  Rename Table Column

Description

Sets a new name for a column.

Usage

renameTableColumn(
  column, 
  new.name, 
  table = "node", 
  namespace = "default", 
  network = NULL, 
  base.url = .defaultBaseUrl
)
resetDefaultSandbox

Arguments

column          Name of the column to rename
new.name        New name for the specified column
table           (optional) Name of table, e.g., node (default), edge or network
namespace       (optional) Namespace of table, e.g., default (default), shared or hidden
network         (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url        (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

renameTableColumn('exp','log2FC')

resetDefaultSandbox resetDefaultSandbox

Description

Reset the entire state of the sandbox system.

Usage

resetDefaultSandbox()

Value

None

Examples

resetDefaultSandbox()
rotateLayout  

Description

Rotate the layout.

Usage

rotateLayout(
angle,
selected.only = FALSE,
network = NULL,
base.url = .defaultBaseUrl
)

Arguments

angle  The angle (in degrees) to rotate the network. From -180 to 180
selected.only  (Boolean) Whether to rotate only current selection. Default is false.
network  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Author(s)

Yihang Xin

Examples

rotateLayout(90, 'current', selected.only=FALSE)
**runningRemoteCheck**

**Description**
runningRemoteCheck

**Usage**
runningRemoteCheck(base.url = .defaultBaseUrl, newState = NULL)

**Arguments**
- **base.url** *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
- **newState** new state of running remote

**Value**
oldState

**Examples**
runningRemoteCheck()

---

**sandboxGetFileInfo**

**Description**
Get metadata on file in sandbox (or entire sandbox). If the current sandbox is the entire file system on a Cytoscape workstation, trying to delete it is an error. Otherwise, deleting the current sandbox results in the default sandbox becoming the new current sandbox. When running standalone on the same workstation as Cytoscape, the default sandbox is the entire file system on the Cytoscape workstation. When running in a Notebook or remote server, the default sandbox is the 'default_sandbox' created automatically under the filetransfer directory in the CytoscapeConfiguration directory. If that sandbox is deleted, it will be re-created so that subsequent file operations can complete successfully. Note that this function can be used to query either a file or a directory.

**Usage**
sandboxGetFileInfo(fileName, sandboxName = NULL, base.url = .defaultBaseUrl)
sandboxGetFrom

Arguments

fileName  Name of file whose metadata to return ... can be sandbox-relative path ... "." returns metadata on sandbox itself
sandboxName Name of sandbox containing file. None means "the current sandbox".
base.url Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://127.0.0.1:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
dict: 'filePath': <full path on Cytoscape workstation>, 'modifiedTime': <last changed time, "" if file doesn't exist>, 'isFile': <True if file, False if directory>

Examples
sandboxGetFileInfo()

Description
Transfer a file from a sandbox. The source file is transferred from the named (or current) sandbox to the R workflow's file system, overwriting an existing file if one already exists. The sourceFile can be an absolute path if the sandbox is the entire file system (i.e., for standalone R execution) or a path relative to the sandbox (i.e., for Notebook or remote execution or if a sandbox was explicitly created).

Usage
sandboxGetFrom(
  sourceFile,
  destFile = NULL,
  overwrite = TRUE,
  sandboxName = NULL,
  base.url = .defaultBaseUrl
)

Arguments

sourceFile Name of file to read (as absolute path or sandbox-relative path)
destFile Name of file in the R workflow's file system ... if None, use file name in source_file
overwrite Name of sandbox containing file. None means "the current sandbox".
sandboxName Name of sandbox containing file. None means "the current sandbox".
sandboxInitializer

**base.url**

Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://127.0.0.1:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

sandboxGetFrom

**Examples**

sandboxGetFrom()

**sandboxInitializer**

**Description**

Start with a sandbox template and update properties using whatever is found in the new_sandbox.

**Usage**

sandboxInitializer(newSandbox = NULL, ...)

**Arguments**

newSandbox

...  ...

**Value**

sandbox

**Examples**

sandboxInitializer()
sandboxRemove

Description

Delete sandbox contents and remove its directory. If the current sandbox is the entire file system on a Cytoscape workstation, trying to delete it is an error. Otherwise, deleting the current sandbox results in the default sandbox becoming the new current sandbox. When running standalone on the same workstation as Cytoscape, the default sandbox is the entire file system on the Cytoscape workstation. When running in a Notebook or remote server, the default sandbox is the 'default_sandbox' created automatically under the under the filetransfer directory in the CytoscapeConfiguration directory. If that sandbox is deleted, it will be re-created so that subsequent file operations can complete successfully.

Usage

sandboxRemove(sandboxName = NULL, base.url = .defaultBaseUrl)

Arguments

sandboxName Name of sandbox to delete. None means to delete the current sandbox. If that sandbox is the default sandbox, it is automatically re-created.

base.url Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://127.0.0.1:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

dict: 'sandboxPath': <directory on Cytoscape workstation>, 'existed': <True if sandbox existed>

Examples

sandboxRemove()

sandboxRemoveFile

Description

Remove a file from a sandbox. The named file is removed from the named sandbox. If the sandbox is the entire file system (i.e., for standalone R execution), the file name can be an absolute path. Otherwise, it is a path relative to the named sandbox. Note that there is no function that deletes a directory, except for sandboxRemove(), which deletes a sandbox and all of its contents.
sandboxSendTo

Usage

sandboxRemoveFile(fileName, sandboxName = NULL, base.url = .defaultBaseUrl)

Arguments

fileName Name of file to delete (as absolute path or sandbox-relative path)
sandboxName Name of sandbox containing file. None means "the current sandbox".
base.url Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://127.0.0.1:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

dict: 'filePath': <file’s absolute path in Cytoscape workstation>, 'existed': True if file existed before being deleted

Examples

sandboxRemoveFile()

sandboxSendTo

Description

Transfer a file to a sandbox. The source file is transferred to the named (or current) sandbox, overwriting an existing file if one already exists. The destFile can be an absolute path if the sandbox is the entire file system (i.e., for standalone R execution) or a path relative to the sandbox (i.e., for Notebook or remote execution or if a sandbox was explicitly created). Note that there is no function that transfers an entire directory. Note, though, that when using sandboxSet() to make a sandbox current, it is possible to copy the Cytoscape sample data directories into to the sandbox at the same time.

Usage

sandboxSendTo(
  sourceFile,
  destFile = NULL,
  overwrite = TRUE,
  sandboxName = NULL,
  base.url = .defaultBaseUrl
)
sandboxSet

Arguments

- **sourceFile**: Name of file to read (as absolute path or sandbox-relative path)
- **destFile**: Name of file in the R workflow’s file system ... if None, use file name in source_file
- **overwrite**: Name of sandbox containing file. None means "the current sandbox".
- **sandboxName**: Name of sandbox containing file. None means "the current sandbox".
- **base.url**: Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://127.0.0.1:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

sandboxSendTo

Examples

sandboxSendTo()

sandboxSet

**Description**

Set a new default sandbox, creating it if necessary. A sandbox is the root for the file system used for all file operations. When running standalone on the same workstation as Cytoscape, the default sandbox is the directory that’s current for the R kernel. When running in a Notebook or remote server, the default sandbox is the ‘default_sandbox’ created automatically under the under the filetransfer directory in the CytoscapeConfiguration directory. Naming a sandbox with this function creates a new sub-directory as a sibling to ’default_sandbox’ and uses it for subsequent file operations. Setting a None sandbox uses the default sandbox instead. Sandboxes are highly recommended as an aid to creating workflows that can be shared with others.

**Usage**

```
sandboxSet(
    sandboxName,  
    copySamples = TRUE,  
    reinitialize = TRUE, 
    base.url = .defaultBaseUrl
)
```
sandboxUrlTo

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sandboxName</td>
<td>Name of new default sandbox. None means to use the original default sandbox</td>
</tr>
<tr>
<td>copySamples</td>
<td>True to copy the Cytoscape sampleData into the sandbox</td>
</tr>
<tr>
<td>reinitialize</td>
<td>True to delete sandbox contents (if any) if sandbox already exists</td>
</tr>
<tr>
<td>base.url</td>
<td>Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://127.0.0.1:1234">http://127.0.0.1:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

**Value**

sandbox path in Cytoscape workstation’s file system

**Examples**

sandboxSet()

---

**Description**

Transfer a cloud-based file to a sandbox. The source URL identifies a file to be transferred to the named (or current) sandbox, overwriting an existing file if one already exists. The destFile can be an absolute path if the sandbox is the entire file system (i.e., for standalone R execution), or it can be a path relative to the sandbox (i.e., for Notebook or remote execution or if a sandbox was explicitly created).

**Usage**

sandboxUrlTo(
  sourceURL,
  destFile,
  overwrite = TRUE,
  sandboxName = NULL,
  base.url = .defaultBaseUrl
)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sourceURL</td>
<td>URL addressing cloud file to download</td>
</tr>
<tr>
<td>destFile</td>
<td>Name of file in the R workflow’s file system ... if None, use file name in source_file</td>
</tr>
<tr>
<td>overwrite</td>
<td>Name of sandbox containing file. None means &quot;the current sandbox&quot;.</td>
</tr>
<tr>
<td>sandboxName</td>
<td>Name of sandbox containing file. None means &quot;the current sandbox&quot;.</td>
</tr>
<tr>
<td>base.url</td>
<td>Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://127.0.0.1:1234">http://127.0.0.1:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>
saveSession

Value

dict: 'filePath': <new file’s absolute path in Cytoscape workstation>, 'fileByteCount': number of bytes read

Examples

sandboxUrlTo()

saveSession

Save Session to File

Description

Saves the current Cytoscape session as a CYS file.

Usage

saveSession(filename = NULL, base.url = .defaultBaseUrl, overwriteFile = TRUE)

Arguments

filename Full path or path relative to current working directory, in addition to the name of the file. The .cys extension is automatically added. Leave blank to update previously saved session file.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

overwriteFile (optional) FALSE allows an error to be generated if the file already exists; TRUE allows Cytoscape to overwrite it without asking. Default value is TRUE.

Details

If no filename is provided, then it attempts to save to an existing CYS file associated with the session. If filename already exists, then it is overwritten.

Unlike most export functions in RCy3, Cytoscape will automatically overwrite CYS session files with the same name. You will not be prompted to confirm or reject overwrite. Use carefully!

Value

server response

Examples

saveSession('/fullpath/mySession')
saveSession()
scaleLayout  
Scale Layout

Description

Scale the layout in either the X, Y, or both directions.

Usage

scaleLayout(
  axis,  
scaleFactor,  
selected.only = FALSE,  
network = NULL,  
base.url = .defaultBaseUrl
)

Arguments

axis  The axis to scale. Options: "X Axis", "Y Axis", "Both Axes".
scaleFactor  The axis The scale factor to apply to the network.
selected.only  (Boolean) Whether to rotate only current selection. Default is false.
network  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Author(s)

Yihang Xin

Examples

scaleLayout('X Axis', 2, 'current', selected.only=FALSE)
selectAll

Select all nodes and edges

Description

Selects all nodes and edges in a Cytoscape Network

Usage

selectAll(network = NULL, base.url = .defaultBaseUrl)

Arguments

network (optional) Name or SUID of the network into which you want to select; default is "current" network
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Selects all nodes and edges in a specified network.

Author(s)

Alexander Pico, Julia Gustavsen

See Also

selectNodes

Examples

selectAllNodes()

selectAllEdges

Select all edges

Description

Selects all edges in a Cytoscape Network

Usage

selectAllEdges(network = NULL, base.url = .defaultBaseUrl)
Arguments

network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Selects all edges in a specified network.

Author(s)

Alexander Pico, Julia Gustavsen

Examples

```r
cw <- CytoscapeWindow('new.demo', new('graphNEL'))
selectAllEdges(cw)
```

**selectAllNodes**

Select all nodes

Arguments

network (optional) Name or SUID of the network into which you want to select; default is "current" network

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Selects all nodes in a specified network.

Author(s)

Alexander Pico, Julia Gustavsen
selectEdges

See Also

selectNodes

Examples

selectAllNodes()

---

selectEdges  Select Edges

Description

Select edges in the network by SUID, name or other column values.

Usage

selectEdges(
  edges, 
  by.col = "SUID", 
  preserve.current.selection = TRUE, 
  network = NULL, 
  base.url = .defaultBaseUrl
)

Arguments

edges List of edge SUIDs, names or other column values
by.col Edge table column to lookup up provide edge values. Default is 'SUID'.
preserve.current.selection boolean Whether to maintain previously selected edges.
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

list of newly selected edge SUIDs

Author(s)

AlexanderPico, Tanja Muetze, Georgi Kolishovski, Paul Shannon
selectEdgesAdjacentToNodes

Select Edges Adjacent To Nodes

Description

Takes list of node SUIDs, names or other column values and adds to the selection all edges connected to those nodes, regardless of directionality.

Usage

```r
selectEdgesAdjacentToNodes(
  nodes,
  by.col = "name",
  selectNodes = TRUE,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `nodes`: List of node SUIDs, names or other column values
- `by.col`: Node table column to lookup up provide node values. Default is 'name'.
- `selectNodes`: The default is TRUE. If TRUE, nodes will also be selected.
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Lists of SUIDs for selected nodes and edges

Examples

```r
selectEdgesAdjacentToNodes()
```
**selectEdgesAdjacentToSelectedNodes**

*Select Edges Adjacent To Selected Nodes*

**Description**

Takes currently selected nodes and adds to the selection all edges connected to those nodes, regardless of directionality.

**Usage**

```r
call = selectEdgesAdjacentToSelectedNodes(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

Lists of SUIDs for selected nodes and edges

**Examples**

```r
selectEdgesAdjacentToSelectedNodes()
```

---

**selectEdgesConnectingSelectedNodes**

*Select the edges connecting selected nodes in Cytoscape Network*

**Description**

Selects edges in a Cytoscape Network connecting the selected nodes, including self loops connecting single nodes.

**Usage**

```r
call = selectEdgesConnectingSelectedNodes(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

Lists of SUIDs for selected nodes and edges

**Examples**

```r
selectEdgesConnectingSelectedNodes()
```
**selectFirstNeighbors**

---

### Description

Select nodes directly connected to currently selected nodes. Can specify connection directionality using the direction param.

### Usage

```r
selectFirstNeighbors(
  direction = "any",
  network = NULL,
  base.url = .defaultBaseUrl
)
```

### Arguments

- **direction** (optional) direction of connections to neighbors to follow, e.g., incoming, outgoing, undirected, or any (default)
- **network** (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
**selectNodes**

**Value**

list of suids of selected nodes, including original selection

**Examples**

```r
selectFirstNeighbors()
selectFirstNeighbors('outgoing')
selectFirstNeighbors('incoming')
```

---

**selectNodes**  
*Select Nodes*

**Description**

Select nodes in the network by SUID, name or other column values.

**Usage**

```r
selectNodes(
  nodes,
  by.col = "SUID",
  preserve.current.selection = TRUE,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **nodes** List of node SUIDs, names or other column values
- **by.col** Node table column to lookup up provide node values. Default is 'SUID'.
- **preserve.current.selection** boolean Whether to maintain previously selected nodes.
- **network** (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

list of newly selected node SUIDs

**Author(s)**

AlexanderPico, Tanja Muetze, Georgi Kolishovski, Paul Shannon
Examples

selectNodes()

selectNodesConnectedBySelectedEdges

Select Nodes Connected By Selected Edges

Description

Takes currently selected edges and extends the selection to connected nodes, regardless of directionality.

Usage

selectNodesConnectedBySelectedEdges(network = NULL, base.url = .defaultBaseUrl)

Arguments

network (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

Lists of SUIDs for currently selected nodes and edges

Examples

selectNodesConnectedBySelectedEdges()

setBackgroundColorDefault

Set Background Color Default

Description

Set the default background color.
setCatchupFilterSecs

Usage

setBackgroundColorDefault(
    new.color,
    style.name = NULL,
    base.url = .defaultBaseUrl
)

Arguments

new.color  Color as hex code, e.g., #FD5903
style.name  Name of style; default is "default" style
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

setBackgroundColorDefault('#888888')

________________________________________

setCatchupFilterSecs  Set Catchup Filter Delay

Description

This function sets an internal delay variable that allows Cytoscape to "catchup" prior to subsequent functions. Call without specifying secs to restore default value.

Usage

setCatchupFilterSecs(secs = 1)

Arguments

secs  Number of seconds to delay.

Details

This delay is only necessary while concurrency bugs exist in the Cytoscape application. This delay may need to be increased from the default value in certain use cases, e.g., larger networks.

Value

None
setCatchupNetworkSecs

See Also

setModelPropagationSecs, setCatchupNetworkSecs

Examples

```
{  
  setCatchupFilterSecs(2)
  setCatchupFilterSecs() #restores default delay
}
```

```
setCatchupNetworkSecs

Set Catchup Network Delay

Description

This function sets an internal delay variable that allows Cytoscape to "catchup" prior to subsequent functions. Call without specifying secs to restore default value.

Usage

setCatchupNetworkSecs(secs = 2)

Arguments

secs Number of seconds to delay.

Details

This delay is only necessary while concurrency bugs exist in the Cytoscape application. This delay may need to be increased from the default value in certain use cases, e.g., larger networks.

Value

None

See Also

setModelPropagationSecs, setCatchupFilterSecs

Examples

```
{  
  setCatchupNetworkSecs(2)
  setCatchupNetworkSecs() #restores default delay
}
```
**setCurrentNetwork**

*Set current network*

**Description**

Selects the given network as "current"

**Usage**

```r
setCurrentNetwork(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `network` *(optional)* Name or suid of the network that you want set as current
- `base.url` *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

server response

**Examples**

```r
setCurrentNetwork('MyNetwork')
```

**setCurrentSandbox**

*setCurrentSandbox*

**Description**

Set and return the current sandbox name and path.

**Usage**

```r
setCurrentSandbox(sandboxName, sandboxPath)
```

**Arguments**

- `sandboxName`  
- `sandboxPath`  

**Value**

current sandbox
**setCurrentView**

*Examples*

```r
currentSandbox()
```

---

**Description**

Set which network view is "current".

**Usage**

```r
setCurrentView(network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `network` *(optional)* Name or SUID of a network or view. The first (presumably only) view associated a network is used if a specific view SUID is not provided.

- `base.url` *(optional)* Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

Takes first (presumably only) view associated with provided network

**Value**

None

**Examples**

```r
setCurrentView()
```
**setDefaultSandbox**

**Description**
Set and return the sandbox properties to be used as a default, probably based on whether running remote.

**Usage**
```r
defaultSandbox(newSandbox = NULL, ...)
```

**Arguments**
- `newSandbox`: new sandbox
- `...`: ...

**Value**
default sandbox

**Examples**
```r
defaultSandbox()
```

**setDefaultSandboxPath**

**Description**
Set and return the default path, which isn’t one of the properties tracked in the default_sandbox.

**Usage**
```r
defaultSandboxPath(newPath)
```

**Arguments**
- `newPath`: new path of default sandbox

**Value**
default sandbox path

**Examples**
```r
defaultSandboxPath()
```
setEdgeColorBypass  

**Description**
Override the color for particular edges.

**Usage**

```r
setEdgeColorBypass(
  edge.names,
  new.colors,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `edge.names` List of edge names or SUIDs
- `new.colors` List of hex colors, or single value
- `network` (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**
This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearEdgePropertyBypass`.

**Value**
None

**See Also**

`setEdgePropertyBypass`, `clearEdgePropertyBypass`

**Examples**

```r
setEdgeColorBypass()
```
**setEdgeColorDefault**  
*Set Edge Color Default*

**Description**
Set the default edge color.

**Usage**
```
setEdgeColorDefault(new.color, style.name = NULL, base.url = .defaultBaseUrl)
```

**Arguments**
- **new.color**  
  Color as hex code, e.g., #FD5903
- **style.name**  
  Name of style; default is "default" style
- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
None

**Examples**
```
setEdgeColorDefault('#FD5903')
```

**setEdgeColorMapping**  
*Set Edge Color Mapping*

**Description**
Map table column values to colors to set the edge color.

**Usage**
```
setEdgeColorMapping(
  table.column,
  table.column.values = NULL,
  colors = NULL,
  mapping.type = "c",
  default.color = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
Arguments

- **table.column**: Name of Cytoscape table column to map values from
- **table.column.values**: List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- **colors**: List of hex colors to map to table.column.values or a color palette function, e.g., `paletteColorBrewerSet3` (without quotes). See RColorBrewer::display.brewer.all()
- **mapping.type**: (char) continuous, discrete or passthrough (c,d,p); default is continuous
- **default.color**: Hex color to set as default
- **style.name**: Name of style; default is "default" style
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setEdgeColorMapping('score', c(0,5), c('#FFFFFF', '#FF7755'))
setEdgeColorMapping('score', colors=paletteColorBrewerRdBu)
setEdgeColorMapping('score', colors=paletteColorBrewerSet3, mapping.type='d')
```

setEdgeFontFaceBypass

Set Edge Font Face Bypass

Description

Override the font face for particular edges.

Usage

```r
setEdgeFontFaceBypass(
  edge.names,  # list of edge names
  new.fonts,   # list of font names
  network = NULL,  # name or SUID of the network
  base.url = .defaultBaseUrl  # base url
)
```
Arguments

edge.names List of edge names or SUIDs
new.fonts List of font faces, or single value
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearEdgePropertyBypass`.

Value

None

See Also

`setEdgePropertyBypass`, `clearEdgePropertyBypass`

Examples

```r
setEdgeFontFaceBypass()
```

```
setEdgeFontFaceDefault

*Set Edge Font Face Default*

Description

Set the default edge font.

Usage

```r
setEdgeFontFaceDefault(new.font, style.name = NULL, base.url = .defaultBaseUrl)
```
Arguments

- **new.font** String specification of font face, style and size, e.g., "SansSerif,plain,12" or "Dialog,plain,10"
- **style.name** Name of style; default is "default" style
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```
setEdgeFontFaceDefault("Dialog,plain,10")
```

Description

Sets font face for edge labels.

Usage

```
setEdgeFontFaceMapping(
  table.column,  # Name of Cytoscape table column to map values from
  table.column.values,  # List of values from Cytoscape table to be used in mapping
  fonts,  # List of string specifications of font face, style and size, e.g., c("SansSerif,plain,12", "Dialog,plain,10")
  mapping.type = "d",  # (char) discrete or passthrough (d,p); default is discrete
  default.font = NULL,  # Name of default font
  style.name = NULL,  # Name of style; default is "default" style
  network = NULL,  # Name of Cytoscape network
  base.url = .defaultBaseUrl  # URL to connect to the CyREST API
)
```
setEdgeFontSizeBypass

Description

Override the font size for particular edges.

Usage

```r
setEdgeFontSizeBypass(
  edge.names,  
  new.sizes,  
  network = NULL,  
  base.url = .defaultBaseUrl  
)
```

Arguments

- `edge.names`: List of edge names or SUIDs
- `new.sizes`: List of size values, or single value
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setEdgeFontFaceMapping("myfonts", c("normal","small"),
  c("SansSerif,plain,12", "Dialog,plain,10"))
```
setEdgeFontSizeDefault

Details

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, setEdgePropertyBypass, which can be used to set any visual property. To restore defaults and mappings, use clearEdgePropertyBypass.

Value

None

See Also

setEdgePropertyBypass, clearEdgePropertyBypass

Examples

setEdgeFontSizeBypass()

---

setEdgeFontSizeDefault

Set Edge Font Size Default

Description

Set the default edge font size.

Usage

setEdgeFontSizeDefault(new.size, style.name = NULL, base.url = .defaultBaseUrl)

Arguments

- **new.size**: Numeric value for size
- **style.name**: Name of style; default is "default" style
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

setEdgeFontSizeDefault(12)
**setEdgeFontSizeMapping**

*Set Edge Font Size Mapping*

**Description**

Map table column values to sizes to set the edge size.

**Usage**

```r
defaultSetEdgeFontSizeMapping(
  table.column,
  table.column.values = NULL,
  sizes = NULL,
  mapping.type = "c",
  default.size = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `table.column`: Name of Cytoscape table column to map values from
- `table.column.values`: List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `sizes`: List of sizes to map to `table.column.values`. A range of 10 to 100 is used by default for automatic mapping.
- `mapping.type`: (char) continuous, discrete or passthrough (c,d,p); default is continuous
- `default.size`: Size value to set as default
- `style.name`: Name of style; default is "default" style
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None
Examples

setEdgeFontSizeMapping('score')
setEdgeFontSizeMapping('score', sizes=c(6,24))
setEdgeFontSizeMapping('score', c(0,30), c(35,55))

setEdgeLabelBypass Set Edge Label Bypass

Description

Override the label for particular edges.

Usage

setEdgeLabelBypass(
  edge.names,
  new.labels,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

edge.names List of edge names or SUIDs
new.labels List of labels, or single value
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, setEdgePropertyBypass, which can be used to set any visual property. To restore defaults and mappings, use clearEdgePropertyBypass.

Value

None

See Also

setEdgePropertyBypass, clearEdgePropertyBypass
**Examples**

```r
setEdgeLabelBypass()
```

---

**setEdgeLabelColorBypass**  

*Set Edge Label Color Bypass*

---

**Description**

Override the label color for particular edges.

**Usage**

```r
setEdgeLabelColorBypass(
  edge.names,
  new.colors,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `edge.names`: List of edge names or SUIDs
- `new.colors`: List of hex colors, or single value
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearEdgePropertyBypass`.

**Value**

None

**See Also**

`setEdgePropertyBypass, clearEdgePropertyBypass`
setEdgeLabelColorDefault

Set Edge Label Color Default

Description

Set the default edge label color.

Usage

```r
setEdgeLabelColorDefault(
  new.color,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `new.color`: Color as hex code, e.g., #FD5903
- `style.name`: Name of style; default is "default" style
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setEdgeLabelColorDefault("#FD5903")
```
setEdgeLabelColorMapping

*Set Edge Label Color Mapping*

**Description**

Map table column values to colors to set the edge border color.

**Usage**

```r
table.column, table.column.values = NULL,
colors = NULL,
mapping.type = "c",
default.color = NULL,
style.name = NULL,
network = NULL,
base.url = .defaultBaseUrl
)
```

**Arguments**

- `table.column` Name of Cytoscape table column to map values from
- `table.column.values` List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `colors` List of hex colors to map to `table.column.values` or a color palette function, e.g., `paletteColorBrewerSet3` (without quotes). See `RColorBrewer::display.brewer.all()`.
- `mapping.type` (char) continuous, discrete or passthrough (c,d,p); default is continuous
- `default.color` Hex color to set as default
- `style.name` Name of style; default is "default" style
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None
setEdgeLabelMapping

Examples

setEdgeLabelColorMapping('score', c(0, 5), c('#FFFFFF', '#FF7755'))
setEdgeLabelColorMapping('score', colors=paletteColorBrewerRdBu)
setEdgeLabelColorMapping('score', colors=paletteColorBrewerSet3, mapping.type='d')

setEdgeLabelDefault  Set Edge Label Default

Description

Set the default edge label.

Usage

setEdgeLabelDefault(new.label, style.name = NULL, base.url = .defaultBaseUrl)

Arguments

new.label  String label for unmapped edges.
style.name  Name of style; default is "default" style
base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

setEdgeLabelDefault('unknown')

setEdgeLabelMapping  Set Edge Label Mapping

Description

Pass the values from a table column to display as edge labels.
Usage

```
setEdgeLabelMapping(
  table.column,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- **table.column**: Name of Cytoscape table column to map values from.
- **style.name**: Name of style; default is "default" style.
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```
setEdgeLabelMapping('label')
```

---

**setEdgeLabelOpacityBypass**

*Set Edge Label Opacity Bypass*

Description

Override the label opacity for particular edges.

Usage

```
setEdgeLabelOpacityBypass(
  edge.names,
  new.value,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
setEdgeLabelOpacityDefault

Arguments

- **edge.names**: List of edge names or SUIDs
- **new.value**: List of opacity values, or single value
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearEdgePropertyBypass`.

Value

None

See Also

- `setEdgePropertyBypass`, `clearEdgePropertyBypass`

Examples

```r
setEdgeLabelOpacityBypass()
```

---

**setEdgeLabelOpacityDefault**

*Set Edge Label Opacity Default*

Description

Set default opacity value for all unmapped edge labels.

Usage

```r
setEdgeLabelOpacityDefault(
  new.opacity,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```
Arguments

- **new.opacity** numeric values between 0 and 255; 0 is invisible.
- **style.name** name of style; default is "default" style.
- **base.url** (optional) ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setEdgeLabelOpacityDefault(50)
```

Description

Sets opacity for edge label only.

Usage

```r
setEdgeLabelOpacityMapping(
  table.column, 
  table.column.values = NULL, 
  opacities = NULL, 
  mapping.type = "c", 
  default.opacity = NULL, 
  style.name = NULL, 
  network = NULL, 
  base.url = .defaultBaseUrl
)
```

Arguments

- **table.column** Name of Cytoscape table column to map values from
- **table.column.values** list of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- **opacities** (integer) values between 0 and 255; 0 is invisible. A range of 50 to 255 is used by default for automatic mapping.
- **mapping.type** (char) continuous, discrete or passthrough (c, d, p); default is continuous
setEdgeLineStyleBypass

**Description**

Override the style for particular edges.

**Usage**

```r
setEdgeLineStyleBypass(
  edge.names,
  new.styles,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **edge.names**: List of edge names or SUIDs
- **new.styles**: List of style values, or single value
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Examples**

```r
setEdgeLabelOpacityMapping('weight')
setEdgeLabelOpacityMapping('weight', opacities=c(0,100))
setEdgeLabelOpacityMapping('weight', c(1,10), c(50,255))
```
setEdgeLineStyleDefault

Details

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, setEdgePropertyBypass, which can be used to set any visual property. To restore defaults and mappings, use clearEdgePropertyBypass.

Value

None

See Also

setEdgePropertyBypass, clearEdgePropertyBypass

Examples

setEdgeLineStyleBypass()

Description

Set the default edge style.

Usage

setEdgeLineStyleDefault(
  new.line.style,
  style.name = NULL,
  base.url = .defaultBaseUrl
)

Arguments

new.line.style Name of line style, e.g., SOLID, LONG_DASH, etc (see getLineStyles)
stylename Name of style; default is "default" style
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None
setEdgeLineStyleMapping

*Set Edge Line Style Mapping*

**Description**

Map table column values to styles to set the edge style.

**Usage**

```r
setEdgeLineStyleMapping(
  table.column,
  table.column.values = NULL,
  line.styles = NULL,
  default.line.style = "SOLID",
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `table.column` Name of Cytoscape table column to map values from
- `table.column.values` List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `line.styles` List of line styles. Leave NULL to perform an automatic mapping to available line styles. See `getLineStyles`.
- `default.line.style` Style to set as default. See `getLineStyles`.
- `style.name` Name of style; default is "default" style
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setEdgeLineStyleDefault(’LONG_DASH’)
```
Examples

```r
setEdgeLineStyleMapping('type')
setEdgeLineStyleMapping('type', c('pp', 'pd'), c('SOLID', 'LONG_DASH'))
```

**setEdgeLineWidthBypass**

*Set Edge Line Width Bypass*

**Description**

Override the width for particular edges.

**Usage**

```r
setEdgeLineWidthBypass(
  edge.names,  
  new.widths,  
  network = NULL,  
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `edge.names`: List of edge names or SUIDs
- `new.widths`: List of width values, or single value
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setPropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearEdgePropertyBypass`.

**Value**

None

**See Also**

`setPropertyBypass, clearEdgePropertyBypass`
Examples

```
setEdgeLineWidthDefault()
```

---

**Set Edge Line Width Default**

**Description**

Set the default edge width.

**Usage**

```
setEdgeLineWidthDefault(
  new.width,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `new.width` Numeric value for width
- `style.name` Name of style; default is "default" style
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```
setEdgeLineWidthDefault(3)
```
**setEdgeLineWidthMapping**

*Set Edge Line Width Mapping*

**Description**

Map table column values to widths to set the edge line width.

**Usage**

```r
setEdgeLineWidthMapping(
  table.column,
  table.column.values = NULL,
  widths = NULL,
  mapping.type = "c",
  default.width = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **table.column**  
  Name of Cytoscape table column to map values from

- **table.column.values**  
  List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.

- **widths**  
  List of width values to map to `table.column.values`. A range of 10 to 100 is used by default for automatic mapping.

- **mapping.type**  
  (char) continuous, discrete or passthrough (c,d,p); default is continuous

- **default.width**  
  Width value to set as default for all unmapped values for all unmapped values.

- **style.name**  
  Name of style; default is "default" style

- **network**  
  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None
Examples

```r
setEdgeLineWidthMapping('score')
setEdgeLineWidthMapping('score', widths=c(1,10))
setEdgeLineWidthMapping('score', c(0,30), c(1,5))
```

---

### setEdgeOpacityBypass

**Set Edge Opacity Bypass**

#### Description

Override the opacity for particular edges.

#### Usage

```r
setEdgeOpacityBypass(
  edge.names,
  new.values,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

#### Arguments

- `edge.names` List of edge names or SUIDs
- `new.values` List of values to set, or single value
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

#### Details

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearEdgePropertyBypass`.

#### Value

None

#### See Also

`setEdgePropertyBypass`, `clearEdgePropertyBypass`
**setEdgeOpacityDefault**

**Examples**

```r
setEdgeOpacityBypass()
```

---

**setEdgeOpacityDefault  Set Edge Opacity Default**

**Description**

Set default opacity value for all unmapped edges.

**Usage**

```r
setEdgeOpacityDefault(
  new.opacity,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `new.opacity`: Numeric values between 0 and 255; 0 is invisible.
- `style.name`: Name of style; default is "default" style.
- `base.url` (optional): Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setEdgeOpacityDefault(50)
```
setEdgeOpacityMapping  Set Edge Opacity Mapping

Description
Map table column values to opacities to set the edge opacity.

Usage
setEdgeOpacityMapping(
  table.column,
  table.column.values = NULL,
  opacities = NULL,
  mapping.type = "c",
  default.opacity = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

  table.column  Name of Cytoscape table column to map values from
  table.column.values
    List of values from Cytoscape table to be used in mapping. Leave NULL to
    perform an automatic mapping to all column values.
  opacities     (integer) values between 0 and 255; 0 is invisible. A range of 50 to 255 is used
                by default for automatic mapping.
  mapping.type  (char) continuous, discrete or passthrough (c,d,p); default is continuous
  default.opacity
    Opacity value to set as default for all unmapped values
  style.name    Name of style; default is "default" style
  network       (optional) Name or SUID of the network. Default is the "current" network active
                 in Cytoscape.
  base.url      (optional) Ignore unless you need to specify a custom domain, port or version
                 to connect to the CyREST API. Default is http://localhost:1234 and the latest
                 version of the CyREST API supported by this version of RCy3.

Value
None
setEdgePropertyBypass

Examples

setEdgeOpacityMapping('weight')
setEdgeOpacityMapping('weight', opacities=c(0,100))
setEdgeOpacityMapping('weight', c(10), c(50,255))

Description

Set bypass values for any edge property of the specified edges, overriding default values and mappings defined by any visual style.

Usage

setEdgePropertyBypass(
  edge.names,
  new.values,
  visual.property,
  bypass = TRUE,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

edge.names List of edge names or SUIDs
new.values List of values to set, or single value
visual.property Name of a visual property. See getVisualPropertyNames.
bypass Whether to set permanent bypass value. Default is TRUE.
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for the visual properties of the edge or edges specified. To restore defaults and mappings, use clearEdgePropertyBypass.

Value

None
setEdgeSelectionColorDefault

Set Edge Selection Color Default

Description

Set the default selected edge color.

Usage

setEdgeSelectionColorDefault(
  new.color,
  style.name = NULL,
  base.url = .defaultBaseUrl
)

Arguments

new.color      Color as hex code, e.g., #FD5903
style.name     Name of style; default is "default" style
base.url       (optional) Ignore unless you need to specify a custom domain, port or version
to connect to the CyREST API. Default is http://localhost:1234 and the latest
version of the CyREST API supported by this version of RCy3.

Value

None

Examples

setEdgeSelectionColorDefault('#FD5903')
**setEdgeSourceArrowColorBypass**

*Set Edge Source Arrow Color Bypass*

**Description**

Override the source arrow color for particular edges.

**Usage**

```r
setEdgeSourceArrowColorBypass(
  edge.names,  
  new.colors,  
  network = NULL,  
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `edge.names`: List of edge names or SUIDs
- `new.colors`: List of hex colors, or single value
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearEdgePropertyBypass`.

**Value**

None

**See Also**

`setEdgePropertyBypass`, `clearEdgePropertyBypass`

**Examples**

```r
setEdgeSourceArrowColorBypass()
```
setEdgeSourceArrowColorMapping

Set Edge Source Arrow Color Mapping

Description

Map table column values to colors to set the source arrow color.

setEdgeSourceArrowColorDefault

Set Edge Source Arrow Color Default

Description

Set the default edge source arrow color.

Usage

setEdgeSourceArrowColorDefault(
  new.color,
  style.name = NULL,
  base.url = .defaultBaseUrl
)

Arguments

new.color Color as hex code, e.g., #FD5903
style.name Name of style; default is "default" style
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

setEdgeSourceArrowColorDefault('#FD5903')
**setEdgeSourceArrowColorMapping**

**Usage**

```r
setEdgeSourceArrowColorMapping(
  table.column,
  table.column.values = NULL,
  colors = NULL,
  mapping.type = "c",
  default.color = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `table.column` Name of Cytoscape table column to map values from
- `table.column.values` List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `colors` List of hex colors to map to `table.column.values` or a color palette function, e.g., `paletteColorBrewerSet3` (without quotes). See `RColorBrewer::display.brewer.all()`.
- `mapping.type` (char) continuous, discrete or passthrough (c,d,p); default is continuous
- `default.color` Hex color to set as default
- `style.name` Name of style; default is "default" style
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setEdgeSourceArrowColorMapping('score', c(0,5), c('#FFFFFF','#FF7755'))
setEdgeSourceArrowColorMapping('score', colors=paletteColorBrewerRdBu)
setEdgeSourceArrowColorMapping('score', colors=paletteColorBrewerSet3, mapping.type='d')
```
setEdgeSourceArrowMapping

Set Edge Source Arrow Mapping

Description
Map table column values to shapes to set the source arrow shape.

Usage
setEdgeSourceArrowMapping(
  table.column,
  table.column.values = NULL,
  shapes = NULL,
  default.shape = "ARROW",
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments
  table.column  Name of Cytoscape table column to map values from
  table.column.values  List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
  shapes  List of shapes to map to table.column.values. Leave NULL to perform an automatic mapping to available shapes. See getArrowShapes
  default.shape  Shape to set as default. See getArrowShapes
  style.name  Name of style; default is "default" style
  network  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
  base.url  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples
setEdgeSourceArrowMapping('type')
setEdgeSourceArrowMapping('type',c('activation','inhibition'),c('ARROW','T'))
**setEdgeSourceArrowShapeBypass**

*Set Edge Source Arrow Shape Bypass*

**Description**

Override the source arrow shape for particular edges.

**Usage**

```r
setEdgeSourceArrowShapeBypass(
  edge.names,  # List of edge names or SUIDs
  new.shapes,  # List of shapes, or single value. See `getArrowShapes`
  network = NULL,  # optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
  base.url = .defaultBaseUrl  # optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
)
```

**Arguments**

- `edge.names`: List of edge names or SUIDs.
- `new.shapes`: List of shapes, or single value. See `getArrowShapes`.
- `network` (optional): Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional): Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearEdgePropertyBypass`.

**Value**

None

**See Also**

`setEdgePropertyBypass`, `clearEdgePropertyBypass`

**Examples**

```r
setEdgeSourceArrowShapeBypass()
```
**setEdgeSourceArrowShapeDefault**

*Set Edge Source Arrow Shape Default*

**Description**

Set the default edge source arrow shape.

**Usage**

```r
setEdgeSourceArrowShapeDefault(
  new.shape,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **new.shape**
  Name of shape, e.g., ARROW, T, etc (see `getArrowShapes`)
- **style.name**
  Name of style; default is "default" style
- **base.url**
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is `http://localhost:1234` and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setEdgeSourceArrowShapeDefault('ARROW')
```

---

**setEdgeSourceArrowShapeMapping**

*Set Edge Source Arrow Shape Mapping*

**Description**

Map table column values to shapes to set the source arrow shape.
Usage

```
setEdgeSourceArrowShapeMapping(
  table.column,
  table.column.values = NULL,
  shapes = NULL,
  default.shape = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `table.column` Name of Cytoscape table column to map values from
- `table.column.values` List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `shapes` List of shapes to map to `table.column.values`. Leave NULL to perform an automatic mapping to available shapes. See `getArrowShapes`
- `default.shape` Shape to set as default. See `getArrowShapes`
- `style.name` Name of style; default is "default" style
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```
setEdgeSourceArrowShapeMapping('type')
setEdgeSourceArrowShapeMapping('type', c('activation', 'inhibition'), c('ARROW', 'T'))
```

Description

Override the target arrow color for particular edges.
setEdgeTargetArrowColorDefault

Usage

```r
setEdgeTargetArrowColorBypass(
  edge.names,
  new.colors,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `edge.names`: List of edge names or SUIDs
- `new.colors`: List of hex colors, or single value
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, `setEdgePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearEdgePropertyBypass`.

Value

None

See Also

`setEdgePropertyBypass`, `clearEdgePropertyBypass`

Examples

```r
setEdgeTargetArrowColorBypass()
```

Description

Set the default edge target arrow color.
Usage

```
setEdgeTargetArrowColorDefault(
  new.color,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- **new.color**  
  Color as hex code, e.g., #FD5903
- **style.name**  
  Name of style; default is "default" style
- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```
setEdgeTargetArrowColorDefault('#FD5903')
```

---

**setEdgeTargetArrowColorMapping**

*Set Edge Target Arrow Color Mapping*

Description

Map table column values to colors to set the target arrow color.

Usage

```
setEdgeTargetArrowColorMapping(
  table.column,
  table.column.values = NULL,
  colors = NULL,
  mapping.type = "c",
  default.color = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
setEdgeTargetArrowMapping

Set Edge Target Arrow Mapping

Description

Map table column values to shapes to set the target arrow shape.

Usage

```r
setEdgeTargetArrowMapping(
  table.column,
  table.column.values = NULL,
  shapes = NULL,
  default.shape = "ARROW",
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
**Arguments**

- **table.column**: Name of Cytoscape table column to map values from
- **table.column.values**: List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- **shapes**: List of shapes to map to table.column.values. Leave NULL to perform an automatic mapping to available shapes. See `getArrowShapes`.
- **default.shape**: Shape to set as default. See `getArrowShapes`.
- **style.name**: Name of style; default is "default" style.
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setEdgeTargetArrowMapping('type')
setEdgeTargetArrowMapping('type',c('activation','inhibition'),c('ARROW','T'))
```

---

**setEdgeTargetArrowShapeBypass**

*Set Edge Target Arrow Shape Bypass*

**Description**

Override the target arrow shape for particular edges.

**Usage**

```r
setEdgeTargetArrowShapeBypass(
  edge.names,
  new.shapes,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
setEdgeTargetArrowShapeDefault

Description

Set the default edge target arrow shape.

Usage

```r
setEdgeTargetArrowShapeDefault(
  new.shape,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```
**setEdgeTargetArrowShapeMapping**

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new.shape</td>
<td>Name of shape, e.g., ARROW, T, etc (see <code>getArrowShapes</code> )</td>
</tr>
<tr>
<td>style.name</td>
<td>Name of style; default is &quot;default&quot; style</td>
</tr>
<tr>
<td>base.url</td>
<td>(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://localhost:1234">http://localhost:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

**Value**

None

**Examples**

```r
setEdgeTargetArrowShapeDefault('ARROW')
```

---

**Description**

Map table column values to shapes to set the target arrow shape.

**Usage**

```r
setEdgeTargetArrowShapeMapping(
  table.column,  # Name of Cytoscape table column to map values from
  table.column.values = NULL,  # List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
  shapes = NULL,  # List of shapes to map to table.column.values. Leave NULL to perform an automatic mapping to available shapes. See `getArrowShapes`
  default.shape = NULL,  # Shape to set as default. See `getArrowShapes`
  style.name = NULL,  # Name of style; default is "default" style
  network = NULL,  # Network to perform mapping on
  base.url = .defaultBaseUrl  # (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
)
```
**setEdgeTooltipBypass**  

**Set Edge Tooltip Bypass**

**Description**
Override the tooltip for particular edges.

**Usage**
```r
setEdgeTooltipBypass(
  edge.names,
  new.values,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**
- `edge.names`  List of edge names or SUIDs
- `new.values`  List of tooltip values, or single value
- `network`  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
Details

This method permanently overrides any default values or mappings defined for this visual property of the edge or edges specified. This method ultimately calls the generic function, setEdgePropertyBypass, which can be used to set any visual property. To restore defaults and mappings, use clearEdgePropertyBypass.

Value

None

See Also

setEdgePropertyBypass, clearEdgePropertyBypass

Examples

setEdgeTooltipBypass()

setEdgeTooltipDefault  Set Edge Tooltip Default

Description

Set the default edge tooltip

Usage

setEdgeTooltipDefault(
  new.tooltip,
  style.name = NULL,
  base.url = .defaultBaseUrl
)

Arguments

new.tooltip  String tooltip for unmapped edges.
style.name   Name of style; default is "default" style
base.url     (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None
**Examples**

```r
cy.setEdgeTooltipDefault('unknown')
```

---

**Description**

Pass the values from a table column to display as edge tooltips.

**Usage**

```r
cy.setEdgeTooltipMapping(
  table.column, 
  style.name = NULL, 
  network = NULL, 
  base.url = .defaultBaseUrl
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>table.column</code></td>
<td>Name of Cytoscape table column to map values from</td>
</tr>
<tr>
<td><code>style.name</code></td>
<td>Name of style; default is &quot;default&quot; style</td>
</tr>
<tr>
<td><code>network</code></td>
<td>(optional) Name or SUID of the network. Default is the &quot;current&quot; network active in Cytoscape.</td>
</tr>
<tr>
<td><code>base.url</code></td>
<td>(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://localhost:1234">http://localhost:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

**Value**

None

**Examples**

```r
cy.setEdgeTooltipMapping('description')
```
**setLayoutProperties**  

**Set Layout Properties**

**Description**
Sets the specified properties for the specified layout. Unmentioned properties are left unchanged.

**Usage**

```r
setLayoutProperties(layout.name, properties.list, base.url = .defaultBaseUrl)
```

**Arguments**

- `layout.name`  
  - (character) Name of the layout

- `properties.list`  
  - (list) List of one or more property=value pairs

- `base.url`  
  - (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

Run `getLayoutNames` to list available layouts. Run `getLayoutPropertyNames` to list properties per layout.

**Value**

None

**Author(s)**

Alexander Pico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**

```r
setLayoutProperties('force-directed', list(defaultSpringLength=50, defaultSpringCoefficient=6E-04))
# Successfully updated the property 'defaultSpringLength'.
# Successfully updated the property 'defaultSpringCoefficient'.
```
**setModelPropagationSecs**

*Set Model Propagation Delay*

**Description**

This function sets an internal delay variable that allows Cytoscape to "catchup" prior to subsequent functions. Call without specifying `secs` to restore default value.

**Usage**

```
setModelPropagationSecs(secs = 5)
```

**Arguments**

- `secs` Number of seconds to delay.

**Details**

This delay is only necessary while concurrency bugs exist in the Cytoscape application. This delay may need to be increased from the default value in certain use cases, e.g., larger networks.

**Value**

None

**See Also**

`setCatchupFilterSecs`, `setCatchupNetworkSecs`

**Examples**

```r
{
    setModelPropagationSecs(2)
    setModelPropagationSecs()  #restores default delay
}
```
Description

Set the bypass value for center x and y for the network. This function could be used to pan and scroll the Cytoscape canvas.

Usage

```
setNetworkCenterBypass(
  x,
  y,
  bypass = FALSE,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

x Coordinate value, increases going to the right.
y Coordinate value, increase going down.
bypass Whether to set permanent bypass value. Default is FALSE per common use of temporary center settings.
network (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values for this visual property. This method ultimately calls the generic function, `setNetworkPropertyBypass`, which can be used to set any visual property. To restore defaults, use `clearNetworkPropertyBypass`.

Value

None

See Also

`setNetworkPropertyBypass`, `clearNetworkPropertyBypass`
**Examples**

```r
setNetworkCenterBypass()
```

---

**setNetworkPropertyBypass**

*Set Network Property Bypass*

---

**Description**

Set bypass values for any network property, overriding default values defined by any visual style.

**Usage**

```r
setNetworkPropertyBypass(
  new.value,
  visual.property,
  bypass = TRUE,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `new.value` Value to set
- `visual.property` Name of a visual property. See `getVisualPropertyNames`.
- `bypass` Whether to set permanent bypass value. Default is `TRUE`.
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is `http://localhost:1234` and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for the visual properties of the node or nodes specified. To restore defaults and mappings, use `clearNodePropertyBypass`.

**Value**

None

**See Also**

`clearNetworkPropertyBypass`
setNetworkZoomBypass

Examples

setNetworkPropertyBypass()

setNetworkZoomBypass

Set Network Zoom Bypass

Description

Set the bypass value for scale factor for the network.

Usage

setNetworkZoomBypass(
  new.value,
  bypass = FALSE,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

new.value  Zoom factor
bypass     Whether to set permanent bypass value. Default is FALSE per common use of temporary zoom settings.
network    (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url   (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values for this visual property. This method ultimately calls the generic function, setNetworkPropertyBypass, which can be used to set any visual property. To restore defaults, use clearNetworkPropertyBypass.

Value

None

See Also

setNetworkPropertyBypass, clearNetworkPropertyBypass

Examples

setNetworkZoomBypass()
setNodeBorderColorBypass

Set Node Border Color Bypass

Description
Override the border color for particular nodes.

Usage
setNodeBorderColorBypass(
  node.names,
  new.colors,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments
node.names   List of node names or SUIDs
new.colors   List of hex colors, or single value
network      (optional) Name or SUID of the network. Default is the "current" network active
              in Cytoscape.
base.url     (optional) Ignore unless you need to specify a custom domain, port or version
t              to connect to the CyREST API. Default is http://localhost:1234 and the latest
              version of the CyREST API supported by this version of RCy3.

Details
This method permanently overrides any default values or mappings defined for this visual property
of the node or nodes specified. This method ultimately calls the generic function, setNodeProp-
ertyBypass, which can be used to set any visual property. To restore defaults and mappings, use
clearNodePropertyBypass, see examples.

Value
None

See Also
setNodePropertyBypass, clearNodePropertyBypass

Examples
setNodeBorderColorBypass('Node 1', '#FF55AA')
setNodeBorderColorBypass(c('Node 1','Node 2'), '#FF55AA')
clearNodePropertyBypass(c('Node 1','Node 2'), 'NODE_BORDER_PAINT')
setNodeBorderColorDefault

*Set Node Border Color Default*

**Description**

Set the default node border color.

**Usage**

```r
defaultNodeBorderColorDefault(
  new.color,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `new.color`: Color as hex code, e.g., #FD5903
- `style.name`: Name of style; default is "default" style
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
defaultNodeBorderColorDefault('#FD5903')
```

---

setNodeBorderColorMapping

*Set Node Border Color Mapping*

**Description**

Map table column values to colors to set the node border color.
Usage

```r
setNodeBorderColorMapping(
  table.column,
  table.column.values = NULL,
  colors = NULL,
  mapping.type = "c",
  default.color = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `table.column` Name of Cytoscape table column to map values from
- `table.column.values` List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `colors` List of hex colors to map to table.column.values or a color palette function, e.g., `paletteColorBrewerSet3` (without quotes). See RColorBrewer::display.brewer.all()
- `mapping.type` (char) continuous, discrete or passthrough (c,d,p); default is continuous
- `default.color` Hex color to set as default
- `style.name` Name of style; default is "default" style
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is `http://localhost:1234` and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeBorderColorMapping('score', c(0.5), c('#FFFFFF', '#FF7755'))
setNodeBorderColorMapping('score', colors=paletteColorBrewerRdBu)
setNodeBorderColorMapping('score', colors=paletteColorBrewerSet3, mapping.type='d')
```
setNodeBorderOpacityBypass

Set Node Border Opacity Bypass

Description
Override the border opacity for particular nodes.

Usage
```
setNodeBorderOpacityBypass(
    node.names,  
    new.values,  
    network = NULL,  
    base.url = .defaultBaseUrl
)
```

Arguments
- `node.names`: List of node names or SUIDs
- `new.values`: List of values to set, or single value
- `network` (optional): Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional): Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details
This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`.

Value
None

See Also
- `setNodePropertyBypass`, `clearNodePropertyBypass`

Examples
```
setNodeBorderOpacityBypass()
```
setNodeBorderOpacityDefault

*Set Node Border Opacity Default*

**Description**
Set defaults opacity value for all unmapped node borders.

**Usage**

```r
setNodeBorderOpacityDefault(
  new.opacity,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **new.opacity**: Numeric values between 0 and 255; 0 is invisible.
- **style.name**: Name of style; default is "default" style.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**
None

**Examples**

```r
setNodeBorderOpacityDefault(50)
```

---

setNodeBorderOpacityMapping

*Set Node Border Opacity Mapping*

**Description**
Sets opacity for node border only.

Sets opacity for node border only.
setNodeBorderOpacityMapping

Usage

setNodeBorderOpacityMapping(
    table.column,
    table.column.values = NULL,
    opacities = NULL,
    mapping.type = "c",
    default.opacity = NULL,
    style.name = NULL,
    network = NULL,
    base.url = .defaultBaseUrl
)

Arguments

table.column Name of Cytoscape table column to map values from
table.column.values List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
opacities (integer) values between 0 and 255; 0 is invisible. A range of 50 to 255 is used by default for automatic mapping.
mapping.type (char) continuous, discrete or passthrough (c,d,p); default is continuous
default.opacity Opacity value to set as default for all unmapped values
style.name Name of style; default is "default" style
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

setNodeBorderOpacityMapping('score')
setNodeBorderOpacityMapping('score', opacities=c(0,100))
setNodeBorderOpacityMapping('score', c(-5,5), c(50,255))
setNodeBorderWidthBypass

Set Node Border Width Bypass

Description

Override the border width for particular nodes.

Usage

```r
setNodeBorderWidthBypass(
  node.names,
  new.sizes,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `node.names` : List of node names or SUIDs
- `new.sizes` : List of size values, or single value
- `network` : (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` : (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`, see examples.

Value

None

See Also

`setNodePropertyBypass`, `clearNodePropertyBypass`

Examples

```r
setNodeBorderWidthBypass('Node 1', 5)
setNodeBorderWidthBypass(c('Node 1','Node 2'), 5)
clearNodePropertyBypass(c('Node 1','Node 2'), 'NODE_BORDER_WIDTH')
```
**Description**

Set the default node border width.

**Usage**

```r
setNodeBorderWidthDefault(
  new.width,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `new.width` Numeric value for width
- `style.name` Name of style; default is "default" style.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setNodeBorderWidthDefault(2)
```

---

**Description**

Map table column values to widths to set the node border width.
Usage

```
setNodeBorderWidthMapping(
    table.column,
    table.column.values = NULL,
    widths = NULL,
    mapping.type = "c",
    default.width = NULL,
    style.name = NULL,
    network = NULL,
    base.url = .defaultBaseUrl
)
```

Arguments

- **table.column**: Name of Cytoscape table column to map values from
- **table.column.values**: List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- **widths**: List of width values to map to table.column.values. A range of 10 to 100 is used by default for automatic mapping.
- **mapping.type**: (char) continuous, discrete or passthrough (c,d,p); default is continuous
- **default.width**: Width value to set as default for all unmapped values
- **style.name**: Name of style; default is "default" style
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```
setNodeBorderWidthMapping('score')
setNodeBorderWidthMapping('score', widths=c(1,10))
setNodeBorderWidthMapping('score', c(0,30), c(1,5))
```
setNodeColorBypass

**Description**
Set the bypass value for fill color for the specified node or nodes.

**Usage**
```r
setNodeColorBypass(
  node.names,
  new.colors,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**
- `node.names` List of node names or SUIDs
- `new.colors` List of hex colors, or single value
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**
This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`, see examples.

**Value**
None

**See Also**
- `setNodePropertyBypass`, `clearNodePropertyBypass`

**Examples**
```r
setNodeColorBypass ('node1', '#FF0088')
setNodeColorBypass (c('node1', 'node2'), c('#88FF88', '#FF0088'))
clearNodePropertyBypass(c('Node 1', 'Node 2'), 'NODE_FILL_COLOR')
```
setNodeColorDefault  
Set Node Color Default

Description
Set the default node color.

Usage
setNodeColorDefault(new.color, style.name = NULL, base.url = .defaultBaseUrl)

Arguments
- new.color: Color as hex code, e.g., #FD5903
- style.name: Name of style; default is "default" style
- base.url: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples
setNodeColorDefault('#FD5903')

setNodeColorMapping  
Set Node Color Mapping

Description
Map table column values to colors to set the node fill color.

Usage
setNodeColorMapping(
  table.column,
  table.column.values = NULL,
  colors = NULL,
  mapping.type = "c",
  default.color = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
setNodeComboOpacityMapping

Description

Sets opacity for node fill, border and label all together.

Usage

```r
setNodeComboOpacityMapping(
  table.column,
  table.column.values = NULL,
  opacities = NULL,
  mapping.type = "c",
  default.opacity = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `table.column`: Name of Cytoscape table column to map values from.
- `table.column.values`: List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `colors`: List of hex colors to map to `table.column.values` or a color palette function, e.g., `paletteColorBrewerSet3` (without quotes). See `RColorBrewer::display.brewer.all()`.
- `mapping.type`: (char) continuous, discrete or passthrough (c,d,p); default is continuous.
- `default.color`: Hex color to set as default.
- `style.name`: Name of style; default is "default" style.
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is `http://localhost:1234` and the latest version of the CyREST API supported by this version of RCy3.

Examples

```r
setNodeColorMapping('score', c(-5,0,5), c('#5577FF','#FFFFFF','#FF7755'))
setNodeColorMapping('score', colors=paletteColorBrewerRdBu)
setNodeColorMapping('score', colors=paletteColorBrewerSet3, mapping.type='d')
```
setNodeCustomBarChart

Description

Makes a bar chart per node using specified node table columns by setting a default custom graphic style.

Usage

setNodeCustomBarChart(
  columns,
  type = "GROUPED",
  colors = NULL,
  range = NULL,
  orientation = "VERTICAL",
  colAxis = FALSE,
  table.column, table.column.values, opacities, mapping.type, default.opacity, style.name, network, base.url
)
setNodeCustomBarChart

rangeAxis = FALSE,
zeroLine = FALSE,
axisWidth = 0.25,
axisColor = "#000000",
axisFontSize = 1,
separation = 0,
slot = 1,
style.name = NULL,
base.url = .defaultBaseUrl
)

Arguments

columns List of node column names to be displayed, in order.
type Type of bar chart: GROUPED (default), STACKED, HEAT_STRIPs, or UP_DOWN

colors (optional) List of colors to be matched with columns or with range, depending on type. Default is a set of colors from an appropriate Brewer palette.

range (optional) Min and max values of chart. Default to use min and max from specified data columns.

orientation (optional) HORIZONTAL or VERTICAL (default).

colAxis (optional) Show axis with column labels. Default is FALSE.

rangeAxis (optional) Show axis with range of values. Default is FALSE.

zeroLine (optional) Show a line at zero. Default is FALSE.

axisWidth (optional) Width of axis lines, if shown. Default is 0.25.

axisColor (optional) Color of axis lines, if shown. Default is black.

axisFontSize (optional) Font size of axis labels, if shown. Default is 1.

separation (optional) Distance between bars. Default is 0.0.

slot (optional) Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.

style.name (optional) Name of style; default is "default" style

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

See Also

setNodeCustomPosition, removeNodeCustomGraphics

Examples

setNodeCustomBarChart(c("data1","data2","data3"))
Description

Makes a box chart per node using specified node table columns by setting a default custom graphic style.

Usage

```r
setNodeCustomBoxChart(
  columns,
  colors = NULL,
  range = NULL,
  orientation = "VERTICAL",
  rangeAxis = FALSE,
  zeroLine = FALSE,
  axisWidth = 0.25,
  axisColor = "#000000",
  axisFontSize = 1,
  slot = 1,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- **columns**: List of node column names to be displayed.
- **colors**: (optional) List of colors to be matched with columns or with range, depending on type. Default is a set of colors from an appropriate Brewer palette.
- **range**: (optional) Min and max values of chart. Default is to use min and max from specified data columns.
- **orientation**: (optional) HORIZONTAL or VERTICAL (default).
- **rangeAxis**: (optional) Show axis with range of values. Default is FALSE.
- **zeroLine**: (optional) Show a line at zero. Default is FALSE.
- **axisWidth**: (optional) Width of axis lines, if shown. Default is 0.25.
- **axisColor**: (optional) Color of axis lines, if shown. Default is black.
- **axisFontSize**: (optional) Font size of axis labels, if shown. Default is 1.
- **slot**: (optional) Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.
- **style.name**: (optional) Name of style; default is "default" style.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
setNodeCustomHeatMapChart

Value

None

See Also

setNodeCustomPosition, removeNodeCustomGraphics

Examples

setNodeCustomBoxChart(c("data1","data2","data3"))

---

setNodeCustomHeatMapChart

**Set Node Custom HeatMap Chart**

### Description

Makes a heatmap chart per node using specified node table columns by setting a default custom graphic style.

### Usage

```r
setNodeCustomHeatMapChart(
  columns,
  colors = NULL,
  range = NULL,
  orientation = "HORIZONTAL",
  rangeAxis = FALSE,
  zeroLine = FALSE,
  axisWidth = 0.25,
  axisColor = "#000000",
  axisFontSize = 1,
  slot = 1,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

### Arguments

- **columns**: List of node column names to be displayed.
- **colors**: (optional) List of colors to be matched with columns or with range, depending on type. Default is a set of colors from an appropriate Brewer palette.
- **range**: (optional) Min and max values of chart. Default is to use min and max from specified data columns.
- **orientation**: (optional) VERTICAL or HORIZONTAL (default).
setNodeCustomLinearGradient

Set Node Custom Linear Gradient

Description

Makes a gradient fill per node by setting a default custom graphic style.

Usage

```r
setNodeCustomLinearGradient(
  colors = c("#DDDDDD", "#888888"),
  anchors = c(0, 1),
  angle = 0,
  slot = 1,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

Value

None

See Also

setNodeCustomPosition, removeNodeCustomGraphics

Examples

```r
setNodeCustomHeatMapChart(c("data1","data2","data3"))
```
setNodeCustomLineChart

Arguments

- **colors** (optional) List of colors to define gradient
- **anchors** (optional) Position of colors from 0.0 to 1.0.
- **angle** (optional) Angle of gradient. Default is 0 (left-to-right).
- **slot** (optional) Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.
- **style.name** (optional) Name of style; default is "default" style
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeCustomLinearGradient()
```

---

**setNodeCustomLineChart**  
*Set Node Custom Line Chart*

Description

Makes a line chart per node using specified node table columns by setting a default custom graphic style.

Usage

```r
setNodeCustomLineChart(
  columns,
  colors = NULL,
  range = NULL,
  lineWidth = 1,
  rangeAxis = FALSE,
  zeroLine = FALSE,
  axisWidth = 0.25,
  axisColor = "#000000",
  axisFontSize = 1,
  slot = 1,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```
setNodeCustomPieChart

Arguments

- **columns**: List of node column names to be displayed.
- **colors** *(optional)*: List of colors to be matched with columns or with range, depending on type. Default is a set of colors from an appropriate Brewer palette.
- **range** *(optional)*: Min and max values of chart. Default is to use min and max from specified data columns.
- **lineWidth** *(optional)*: Width of chart line. Default is 1.0.
- **rangeAxis** *(optional)*: Show axis with range of values. Default is FALSE.
- **zeroLine** *(optional)*: Show a line at zero. Default is FALSE.
- **axisWidth** *(optional)*: Width of axis lines, if shown. Default is 0.25.
- **axisColor** *(optional)*: Color of axis lines, if shown. Default is black.
- **axisFontSize** *(optional)*: Font size of axis labels, if shown. Default is 1.
- **slot** *(optional)*: Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.
- **style.name** *(optional)*: Name of style; default is "default" style.
- **baseUrl** *(optional)*: Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

See Also

setNodeCustomPosition, removeNodeCustomGraphics

Examples

```
setNodeCustomLineChart(c("data1","data2","data3"))
```

Description

Makes a pie chart per node using specified node table columns by setting a default custom graphic style.
Usage

```r
setNodeCustomPieChart(
  columns,
  colors = NULL,
  startAngle = 0,
  slot = 1,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `columns`: List of node column names to be displayed.
- `colors`: (optional) List of colors to be matched with columns or with range, depending on type. Default is a set of colors from an appropriate Brewer palette.
- `startAngle`: (optional) Angle to start filling pie. Default is 0.0.
- `slot`: (optional) Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.
- `style.name`: (optional) Name of style; default is "default" style.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

See Also

- `setNodeCustomPosition`, `removeNodeCustomGraphics`

Examples

```r
setNodeCustomPieChart(c("data1","data2","data3"))
```

Description

Adjust the position of a custom graphic relative to its node.
Usage

```r
setNodeCustomPosition(
  nodeAnchor = "C",
  graphicAnchor = "C",
  justification = "c",
  xOffset = 0,
  yOffset = 0,
  slot = 1,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `nodeAnchor` Position on node to place the graphic: NW,N,NE,E,SE,S,SW,W or C for center (default)
- `graphicAnchor` Position on graphic to place on node: NW,N,NE,E,SE,S,SW,W or C for center (default)
- `justification` Positioning of content within graphic: l,r,c (default)
- `xOffset` Additional offset in the x direction
- `yOffset` Additional offset in the y direction
- `slot` (optional) Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.
- `style.name` (optional) Name of style; default is "default" style
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeCustomPosition()
```

Description

Makes a gradient fill per node by setting a default custom graphic style.
Usage

```r
setNodeCustomRadialGradient(
  colors = c("#DDDDDD", "#888888"),
  anchors = c(0, 1),
  xCenter = 0.5,
  yCenter = 0.5,
  slot = 1,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `colors` (optional) List of colors to define gradient.
- `anchors` (optional) Position of colors from 0.0 to 1.0.
- `xCenter` (optional) X position for center of radial effect from 0.0 to 1.0. Default is 0.5.
- `yCenter` (optional) Y position for center of radial effect from 0.0 to 1.0. Default is 0.5.
- `slot` (optional) Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.
- `style.name` (optional) Name of style; default is "default" style.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeCustomRadialGradient()
```

Description

Makes a ring chart per node using specified node table columns by setting a default custom graphic style.
Usage

setNodeCustomRingChart(
  columns,
  colors = NULL,
  startAngle = 0,
  holeSize = 0.5,
  slot = 1,
  style.name = NULL,
  base.url = .defaultBaseUrl
)

Arguments

columns List of node column names to be displayed.

colors (optional) List of colors to be matched with columns or with range, depending on type. Default is a set of colors from an appropriate Brewer palette.

startAngle (optional) Angle to start filling ring Default is 0.0.

holeSize (optional) Size of hole in ring. Ranges 0-1. Default is 0.5.

slot (optional) Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.

style.name (optional) Name of style; default is "default" style

base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

See Also

setNodeCustomPosition, removeNodeCustomGraphics

Examples

setNodeCustomRingChart(c("data1","data2","data3"))
setNodeFillOpacityBypass

Usage

setNodeFillOpacityBypass(
    node.names,
    new.values,
    network = NULL,
    base.url = .defaultBaseUrl
)

Arguments

node.names      List of node names or SUIDs
new.values      List of values to set, or single value
network         (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
base.url        (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, setNodePropertyBypass, which can be used to set any visual property. To restore defaults and mappings, use clearNodePropertyBypass, see examples.

Value

None

See Also

setNodePropertyBypass, clearNodePropertyBypass

Examples

setNodeFillOpacityBypass('Node 1', 100)
setNodeFillOpacityBypass(c('Node 1','Node 2'), 100)
clearNodePropertyBypass(c('Node 1','Node 2'), 'NODE_TRANSPARENCY')
**setNodeFillOpacityDefault**

*Set Node Fill Opacity Default*

**Description**

Set default opacity value for all unmapped nodes.

**Usage**

```r
setNodeFillOpacityDefault(
  new.opacity,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **new.opacity**  
  Numeric values between 0 and 255; 0 is invisible.

- **style.name**  
  Name of style; default is "default" style.

- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setNodeFillOpacityDefault(50)
```

---

**setNodeFillOpacityMapping**

*Set Node Fill Opacity Mapping*

**Description**

Sets opacity for node fill only.
**setNodeFillOpacityMapping**

**Usage**

```r
setNodeFillOpacityMapping(
  table.column,
  table.column.values = NULL,
  opacities = NULL,
  mapping.type = "c",
  default.opacity = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `table.column` Name of Cytoscape table column to map values from
- `table.column.values` List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `opacities` (integer) values between 0 and 255; 0 is invisible. A range of 50 to 255 is used by default for automatic mapping.
- `mapping.type` (char) continuous, discrete or passthrough (c,d,p); default is continuous
- `default.opacity` Opacity value to set as default for all unmapped values
- `style.name` Name of style; default is "default" style
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setNodeFillOpacityMapping('score')
setNodeFillOpacityMapping('score', opacities=c(0,100))
setNodeFillOpacityMapping('score', c(-5,5), c(50,255))
```
**setNodeFontFaceBypass**  
*Set Node Font Face Bypass*

**Description**

Override the font face for particular nodes.

**Usage**

```r
setNodeFontFaceBypass(
  node.names,
  new.fonts,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `node.names`: List of node names or SUIDs
- `new.fonts`: List of font faces, or single value
- `network`: (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`.

**Value**

None

**See Also**

`setNodePropertyBypass`, `clearNodePropertyBypass`

**Examples**

```r
setNodeFontFaceBypass()
```
**setNodeFontFaceDefault**  
*Set Node Font Face Default*

**Description**

Set the default node font.

**Usage**

```r
code: setNodeFontFaceDefault(new.font, style.name = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- **new.font**: String specification of font face, style and size, e.g., "SansSerif,plain,12" or "Dialog,plain,10"
- **style.name**: Name of style; default is "default" style
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
code: setNodeFontFaceDefault("Dialog,plain,10")
```

---

**setNodeFontFaceMapping**  
*Set Node Font Face Mapping*

**Description**

Sets font face for node labels.
Usage

```r
setNodeFontFaceMapping(  
  table.column,  
  table.column.values,  
  fonts,  
  mapping.type = "d",  
  default.font = NULL,  
  style.name = NULL,  
  network = NULL,  
  base.url = .defaultBaseUrl  
)
```

Arguments

table.column Name of Cytoscape table column to map values from
table.column.values List of values from Cytoscape table to be used in mapping
fonts List of string specifications of font face, style and size, e.g., c("SansSerif,plain,12", "Dialog,plain,10")
mapping.type (char) discrete or passthrough (d,p); default is discrete
default.font String specification of font face, style and size, e.g., "SansSerif,plain,12" or "Dialog,plain,10"
style.name Name of style; default is "default" style
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeFontFaceMapping("myfonts", c("normal","small"),  
c("SansSerif,plain,12", "Dialog,plain,10"))
```
**setNodeFontSizeBypass**  
*Set Node Font Size Bypass*

**Description**

Override the font size for particular nodes.

**Usage**

```r
setNodeFontSizeBypass(
  node.names,
  new.sizes,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `node.names` List of node names or SUIDs
- `new.sizes` List of size values, or single value
- `network` (optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`, see examples.

**Value**

None

**See Also**

`setNodePropertyBypass`, `clearNodePropertyBypass`

**Examples**

```r
setNodeFontSizeBypass('Node 1', 5)
setNodeFontSizeBypass(c('Node 1', 'Node 2'), 5)
clearNodePropertyBypass(c('Node 1', 'Node 2'), 'NODE_LABEL_FONT_SIZE')
```
setNodeFontSizeDefault

*Set Node Font Size Default*

**Description**

Set the default node font size.

**Usage**

```r
setNodeFontSizeDefault(new.size, style.name = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `new.size`: Numeric value for size
- `style.name`: Name of style; default is "default" style
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setNodeFontSizeDefault(12)
```

setNodeFontSizeMapping

*Set Node Font Size Mapping*

**Description**

Map table column values to sizes to set the node size.
setNodeFontSizeMapping

Usage

```r
setNodeFontSizeMapping(
  table.column,
  table.column.values = NULL,
  sizes = NULL,
  mapping.type = "c",
  default.size = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `table.column` Name of Cytoscape table column to map values from
- `table.column.values` List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `sizes` List of sizes to map to `table.column.values`. A range of 10 to 100 is used by default for automatic mapping.
- `mapping.type` (char) continuous, discrete or passthrough (c,d,p); default is continuous
- `default.size` Size value to set as default
- `style.name` Name of style; default is "default" style
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeFontSizeMapping('score')
setNodeFontSizeMapping('score', sizes=c(6,24))
setNodeFontSizeMapping('score', c(0,30), c(35,55))
```
setNodeHeightBypass  Set Node Height Bypass

Description

Override the height for particular nodes.

Usage

```r
setNodeHeightBypass(
  node.names,
  new.heights,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- `node.names`: List of node names or SUIDs
- `new.heights`: List of height values, or single value
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`, see examples.

Value

None

See Also

`setNodePropertyBypass`, `clearNodePropertyBypass`

Examples

```r
setNodeHeightBypass('Node 1', 35)
setNodeHeightBypass(c('Node 1', 'Node 2'), 35)
clearNodePropertyBypass(c('Node 1', 'Node 2'), 'NODE_HEIGHT')
```
setNodeHeightDefault  Set Node Height Default

Description
Set the default node height.

Usage
setNodeHeightDefault(new.height, style.name = NULL, base.url = .defaultBaseUrl)

Arguments
- new.height: Numeric value for height.
- style.name: Name of style; default is "default" style.
- base.url: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples
setNodeHeightDefault(35)

setNodeHeightMapping  Set Node Height Mapping

Description
Map table column values to the node heights.

Usage
setNodeHeightMapping(
  table.column,
  table.column.values = NULL,
  heights = NULL,
  mapping.type = "c",
  default.height = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
setNodeLabelBypass

Arguments

- **table.column**: Name of Cytoscape table column to map values from
- **table.column.values**: List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- **heights**: List of height values to map to table.column.values. A range of 10 to 100 is used by default for automatic mapping.
- **mapping.type**: (char) continuous, discrete or passthrough (c,d,p); default is continuous
- **default.height**: Size value to set as default
- **style.name**: Name of style; default is "default" style
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

Using this function will unlock node width and height to use separate values.

Value

None

Examples

```r
setNodeHeightMapping('score')
setNodeHeightMapping('score', heights=c(30,80))
setNodeHeightMapping('score', c(0,30), c(35,55))
```

setNodeLabelBypass

Set Node Label Bypass

Description

Override the label for particular nodes.

Usage

```r
setNodeLabelBypass(
  node.names,
  new.labels,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
Arguments

node.names List of node names or SUIDs
new.labels List of labels, or single value
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, setNodePropertyBypass, which can be used to set any visual property. To restore defaults and mappings, use clearNodePropertyBypass, see examples.

Value

None

See Also

setNodePropertyBypass, clearNodePropertyBypass

Examples

setNodeLabelBypass('Node 1', 'Custom Label')
setNodeLabelBypass(c('Node 1', 'Node 2'), 'Custom Label')
clearNodePropertyBypass(c('Node 1', 'Node 2'), 'NODE_LABEL')

setNodeLabelColorBypass

Set Node Label Color Bypass

Description

Override the label color for particular nodes.

Usage

setNodeLabelColorBypass(
  node.names,
  new.colors,
  network = NULL,
  base.url = .defaultBaseUrl
)
setNodeLabelColorDefault

Set Node Label Color Default

Description

Set the default node label color.

Usage

setNodeLabelColorDefault(
    new.color,
    style.name = NULL,
    base.url = .defaultBaseUrl
)
**Arguments**

- **new.color**  
  Color as hex code, e.g., #FD5903
- **style.name**  
  Name of style; default is "default" style
- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setNodeLabelColorDefault('FD5903')
```

---

**Description**

Map table column values to colors to set the node border color.

**Usage**

```r
setNodeLabelColorMapping(
  table.column,
  table.column.values = NULL,
  colors = NULL,
  mapping.type = "c",
  default.color = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **table.column**  
  Name of Cytoscape table column to map values from
- **table.column.values**  
  List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- **colors**  
  List of hex colors to map to table.column.values or a color palette function, e.g., paletteColorBrewerSet3 (without quotes). See RColorBrewer::display.brewer.all()
- **mapping.type**  
  (char) continuous, discrete or passthrough (c,d,p); default is continuous
**setNodeLabelDefault**

**Description**
Set the default node label.

**Usage**
```r
default.color  Hex color to set as default
style.name     Name of style; default is "default" style
network        (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url       (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
```

**Value**
None

**Examples**
```
setNodeLabelColorMapping('score', c(0,5), c('#FFFFFF', '#FF7755'))
setNodeLabelColorMapping('score', colors=paletteColorBrewerRdBu)
setNodeLabelColorMapping('score', colors=paletteColorBrewerSet3, mapping.type='d')
```

```r
setNodeLabelDefault('unknown')
```
setNodeLabelMapping

Description

Pass the values from a table column to display as node labels.

Usage

```r
setNodeLabelMapping(
  table.column,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- **table.column**: Name of Cytoscape table column to map values from
- **style.name**: Name of style; default is "default" style
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeLabelMapping('name')
```

setNodeLabelOpacityBypass

Description

Override the label opacity for particular nodes.
Usage

setNodeLabelOpacityBypass(
  node.names,
  new.values,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

node.names  List of node names or SUIDs
new.values  List of values to set, or single value
network     (optional) Name or SUID of the network. Default is the "current" network active
             in Cytoscape.
base.url    (optional) Ignore unless you need to specify a custom domain, port or version
             to connect to the CyREST API. Default is http://localhost:1234 and the latest
             version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property
of the node or nodes specified. This method ultimately calls the generic function, setNodePropertyBypass,
which can be used to set any visual property. To restore defaults and mappings, use
clearNodePropertyBypass.

Value

None

See Also

setNodePropertyBypass, clearNodePropertyBypass

Examples

setNodeLabelOpacityBypass()

---

setNodeLabelOpacityDefault

Set Node Label Opacity Default

Description

Set default opacity value for all unmapped node labels.
**setNodeLabelOpacityDefault**

**Usage**

```r
setNodeLabelOpacityDefault(
  new.opacity,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `new.opacity`: Numeric values between 0 and 255; 0 is invisible.
- `style.name`: Name of style; default is “default” style.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setNodeLabelOpacityDefault(50)
```

---

**setNodeLabelOpacityMapping**

*Set Node Label Opacity Mapping*

**Description**

Sets opacity for node label only.

**Usage**

```r
setNodeLabelOpacityMapping(
  table.column,
  table.column.values = NULL,
  opacities = NULL,
  mapping.type = "c",
  default.opacity = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
Arguments

- **table.column** Name of Cytoscape table column to map values from
- **table.column.values** List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- **opacities** (integer) values between 0 and 255; 0 is invisible. A range of 50 to 255 is used by default for automatic mapping.
- **mapping.type** (char) continuous, discrete or passthrough (c,d,p); default is continuous
- **default.opacity** Opacity value to set as default for all unmapped values
- **style.name** Name of style; default is "default" style
- **network** (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeLabelOpacityMapping('score')
setNodeLabelOpacityMapping('score', opacities=c(0,100))
setNodeLabelOpacityMapping('score', c(-5,5), c(50,255))
```

setNodeLabelPositionBypass

*Set Node Label Position Bypass*

Description

Override the label position for particular nodes.

Usage

```r
setNodeLabelPositionBypass(
  node.names,
  new.positions,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
setNodeLabelPositionDefault

**Arguments**

- `node.names`: List of node names or SUIDs
- `new.positions`: List of label positions, or single value
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`, see examples.

**Value**

None

**See Also**

`setNodePropertyBypass`, `clearNodePropertyBypass`

**Examples**

```r
setNodeLabelPositionBypass('Node 1','E,S,c,0.00,0.00')
```

---

**Description**

Set the default node label position

**Usage**

```r
setNodeLabelPositionDefault(
  new.nodeAnchor,
  new.graphicAnchor,
  new.justification,
  new.xOffset,
  new.yOffset,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```
Arguments

- **new.nodeAnchor**
  - New position on node to place the graphic: C,NW,N,NE,E,SE,S,SW,W
- **new.graphicAnchor**
  - New position on graphic to place on node: C,NW,N,NE,E,SE,S,SW,W
- **new.justification**
  - New positioning of content within graphic: l,r,c
- **new.xOffset**
  - New additional offset in the x direction
- **new.yOffset**
  - New additional offset in the y direction
- **style.name**
  - Name of style; default is "default" style.
- **base.url**
  - (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeLabelPositionDefault("S","C","c",0.00,0.00)
```

setNodeOpacityBypass  Set Node Opacity Bypass

Description

Set the bypass value for node fill, label and border opacity for the specified node or nodes.

Usage

```r
setNodeOpacityBypass(
  node.names,
  new.values,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- **node.names**
  - List of node names or SUIDs
- **new.values**
  - List of values to set, or single value
- **network**
  - (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**
  - (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
Details

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodeOpacityBypass`, see examples.

Value

None

See Also

`setNodePropertyBypass`, `clearNodeOpacityBypass`

Examples

```r
setNodeOpacityBypass('Node 1', 100)
setNodeOpacityBypass(c('Node 1', 'Node 2'), 100)
clearNodeOpacityBypass(c('Node 1', 'Node 2'))
```

---

### setNodePositionBypass

**Set Node Position Bypass**

Description

Sets the bypass value of node position for one or more nodes. Only applicable if node dimensions are locked. See `lockNodeDimensions`.

Usage

```r
setNodePositionBypass(
  node.names,
  new.x.locations = NULL,
  new.y.locations = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

Arguments

- **node.names**: List of node names or SUIDs
- **new.x.locations**: List of x position values, or single value, default is current x position
- **new.y.locations**: List of y position values, or single value, default is current y position
### setNodePropertyBypass

**Description**

Set bypass values for any node property of the specified nodes, overriding default values and mappings defined by any visual style.

**Usage**

```r
setNodePropertyBypass(
  node.names,
  new.values,
  visual.property,
  bypass = TRUE,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**network**

(optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

**base.url**

(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`, see examples.

**Value**

None

**See Also**

`setNodePropertyBypass`, `clearNodePropertyBypass`

**Examples**

```r
setNodePositionBypass('Node 1', 35)
```
Arguments

- **node.names**: List of node names or SUIDs
- **new.values**: List of values to set, or single value
- **visual.property**: Name of a visual property. See `getVisualPropertyNames`.
- **bypass**: Whether to set permanent bypass value. Default is `TRUE`.
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is `http://localhost:1234` and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for the visual properties of the node or nodes specified. To restore defaults and mappings, use `clearNodePropertyBypass`.

Value

- None

See Also

- `clearNodePropertyBypass`

Examples

```r
setNodePropertyBypass()
```

---

**setNodeSelectionColorDefault**

*Set Node Selection Color Default*

Description

Set the default selection node color.

Usage

```r
setNodeSelectionColorDefault(
    new.color,
    style.name = NULL,
    base.url = .defaultBaseUrl
)
```
setNodeShapeBypass

**Arguments**

- **new.color**  
  Color as hex code, e.g., #FD5903
- **style.name**  
  (optional) Name of style; default is "default" style
- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setNodeSelectionColorDefault('#FD5903')
```

**setNodeShapeBypass**  
*Set Node Shape Bypass*

**Description**

Override the shape for particular nodes.

**Usage**

```r
setNodeShapeBypass(
  node.names,
  new.shapes,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **node.names**  
  List of node names or SUIDs
- **new.shapes**  
  List of shapes, or single value. See **getNodeShapes**.
- **network**  
  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, **setNodePropertyBypass**, which can be used to set any visual property. To restore defaults and mappings, use **clearNodePropertyBypass**, see examples.
setNodeShapeDefault

Value
None

See Also
setNodePropertyBypass, clearNodePropertyBypass

Examples
setNodeShapeBypass('Node 1', 'ROUND_RECTANGLE')
setNodeShapeBypass(c('Node 1', 'Node 2'), 'ROUND_RECTANGLE')
clearNodePropertyBypass(c('Node 1', 'Node 2'), 'NODE_SHAPE')

---

setNodeShapeDefault  Set Node Shape Default

Description
Set the default node shape.

Usage
setNodeShapeDefault(new.shape, style.name = NULL, base.url = .defaultBaseUrl)

Arguments
- new.shape: Name of shape, e.g., ELLIPSE, RECTANGLE, etc (see getNodeShapes)
- style.name: Name of style; default is "default" style
- base.url: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples
setNodeShapeDefault('ELLIPSE')
setNodeShapeMapping  

Set Node Shape Mapping

**Description**

Map table column values to shapes to set the node shape.

**Usage**

```r
setNodeShapeMapping(
  table.column,
  table.column.values = NULL,
  shapes = NULL,
  default.shape = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- `table.column`  Name of Cytoscape table column to map values from
- `table.column.values`  List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `shapes`  List of shapes to map to `table.column.values`. Leave NULL to perform an automatic mapping to available shapes. See `getNodeShapes`
- `default.shape`  Shape to set as default. See `getNodeShapes`
- `style.name`  Name of style; default is "default" style
- `network`  (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setNodeShapeMapping('type')
setNodeShapeMapping('type',c('protein','dna'),c('ELLIPSE','RECTANGLE'))
```
setNodeSizeBypass

**Description**
Sets the bypass value of node size for one or more nodes. Only applicable if node dimensions are locked. See `lockNodeDimensions`.

**Usage**
```r
def setNodeSizeBypass(
    node.names,  # List of node names or SUIDs
    new.sizes,  # List of size values, or single value
    network = NULL,  # (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
    base.url = .defaultBaseUrl  # (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
)
```

**Arguments**
- `node.names`: List of node names or SUIDs
- `new.sizes`: List of size values, or single value
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**
This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`, see examples.

**Value**
None

**See Also**
- `setNodePropertyBypass`
- `clearNodePropertyBypass`

**Examples**
```r
def setNodeSizeBypass('Node 1', 35)
def setNodeSizeBypass(c('Node 1', 'Node 2'), 35)
def clearNodePropertyBypass(c('Node 1', 'Node 2'), 'NODE_SIZE')
```
setNodeSizeDefault  Set Node Size Default

Description

Set the default node size.

Usage

```r
setNodeSizeDefault(new.size, style.name = NULL, base.url = .defaultBaseUrl)
```

Arguments

- `new.size`: Numeric value for size
- `style.name`: Name of style; default is "default" style.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeSizeDefault(35)
```

setNodeSizeMapping  Set Node Size Mapping

Description

Map table column values to node font sizes.

Usage

```r
setNodeSizeMapping(
  table.column,  # Required
  table.column.values = NULL,  # Optional
  sizes = NULL,  # Optional
  mapping.type = "c",  # Optional
  default.size = NULL,  # Optional
  style.name = NULL,  # Optional
  network = NULL,  # Optional
  base.url = .defaultBaseUrl  # Optional
)
```
**setNodeTooltipBypass**

**Arguments**

- `table.column` Name of Cytoscape table column to map values from
- `table.column.values` List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
- `sizes` List of sizes to map to `table.column.values`. A range of 10 to 100 is used by default for automatic mapping.
- `mapping.type` (char) continuous, discrete or passthrough (c,d,p); default is continuous
- `default.size` Size value to set as default
- `style.name` Name of style; default is "default" style
- `network` (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

Using this function will lock node width and height to use a singular "size" value.

**Value**

None

**Examples**

```
setNodeSizeMapping('score')
setNodeSizeMapping('score', sizes=c(30,80))
setNodeSizeMapping('score', c(0,30), c(35,55))
```

---

**setNodeTooltipBypass**  Set Node Tooltip Bypass

**Description**

Sets a bypass tooltip for one or more nodes

**Usage**

```
setNodeTooltipBypass(
  node.names, 
  new.tooltip, 
  network = NULL, 
  base.url = .defaultBaseUrl
)
```
Arguments

- **node.names**: List of node names or SUIDs
- **new.tooltip**: List of tooltips, or a single tooltip
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`, see examples.

Value

None

See Also

- `setNodePropertyBypass`, `clearNodePropertyBypass`

Examples

```r
setNodeTooltipBypass('Node 1', 'This is an important node.')
```

---

**setNodeTooltipDefault**  Set Node Tooltip Default

Description

Set the default node tooltip

Usage

```r
setNodeTooltipDefault(
  new.tooltip,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```
Arguments

- **new.tooltip**: String tooltip for unmapped nodes.
- **style.name**: Name of style; default is "default" style
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setNodeTooltipDefault('unknown')
```

---

**setNodeTooltipMapping**  
*Set Node Tooltip Mapping*

**Description**

Pass the values from a table column to display as node tooltips.

**Usage**

```r
setNodeTooltipMapping(
  table.column,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **table.column**: Name of Cytoscape table column to map values from
- **style.name**: Name of style; default is "default" style
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None
Examples

```r
setNodeTooltipMapping('description')
```

Description

Override the width for particular nodes.

Usage

```r
def setNodeWidthBypass(
  node.names,  # List of node names or SUIDs
  new.widths,  # List of width values, or single value
  network = NULL,  # (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
  base.url = .defaultBaseUrl  # (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
)
```

Arguments

- `node.names`: List of node names or SUIDs
- `new.widths`: List of width values, or single value
- `network`: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- `base.url`: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method permanently overrides any default values or mappings defined for this visual property of the node or nodes specified. This method ultimately calls the generic function, `setNodePropertyBypass`, which can be used to set any visual property. To restore defaults and mappings, use `clearNodePropertyBypass`, see examples.

Value

None

See Also

`setNodePropertyBypass`, `clearNodePropertyBypass`
Examples

setNodeWidthBypass('Node 1', 35)
setNodeWidthBypass(c('Node 1', 'Node 2'), 35)
clearNodePropertyBypass(c('Node 1', 'Node 2'), 'NODE_WIDTH')

---

setNodeWidthDefault  
\textit{Set Node Width Default}

Description

Set the default node width.

Usage

setNodeWidthDefault(new.width, style.name = NULL, base.url = .defaultBaseUrl)

Arguments

\begin{itemize}
\item new.width \hspace{1cm} \text{Numeric value for width.}
\item style.name \hspace{1cm} \text{Name of style; default is "default" style.}
\item base.url \hspace{1cm} \text{(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.}
\end{itemize}

Value

None

Examples

setNodeWidthDefault(35)

---

setNodeWidthMapping  
\textit{Set Node Width Mapping}

Description

Map table column values to the node widths.
Usage

```r
setNodeWidthMapping(
  table.column,  # Name of Cytoscape table column to map values from
  table.column.values = NULL,  # List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
  widths = NULL,  # List of width values to map to table.column.values. A range of 10 to 100 is used by default for automatic mapping.
  mapping.type = "c",  # (char) continuous, discrete or passthrough (c,d,p); default is continuous
  default.width = NULL,  # Size value to set as default
  style.name = NULL,  # Name of style; default is "default" style
  network = NULL,  # (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
  base.url = .defaultBaseUrl  # (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
)
```

Arguments

table.column Name of Cytoscape table column to map values from
table.column.values List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.
widths List of width values to map to table.column.values. A range of 10 to 100 is used by default for automatic mapping.
mapping.type (char) continuous, discrete or passthrough (c,d,p); default is continuous
default.width Size value to set as default
style.name Name of style; default is "default" style
network (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
based url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None # details Using this function will unlock node width and height to use separate values.

Examples

```r
setNodeWidthMapping('score')
setNodeWidthMapping('score', widths=c(30,80))
setNodeWidthMapping('score', c(0,30), c(35,55))
```
### setNotebookIsRunning

**Description**

setNotebookIsRunning

**Usage**

```r
setNotebookIsRunning(newState = NULL)
```

**Arguments**

- `newState` new state of running remote

**Value**

`oldState`

**Examples**

```r
setNotebookIsRunning()
```

---

### setSandboxReinitialize

**Description**

Set and return flag indicating that next command should reinitialize the sandbox according to the default_sandbox.

**Usage**

```r
setSandboxReinitialize(doReinitialize = TRUE)
```

**Arguments**

- `doReinitialize` default is TRUE

**Value**

sandbox reinitialize

**Examples**

```r
setCurrentSandbox()
```
**setStyleDependencies**  
*Set Style Dependencies*

### Description

Sets the values of dependencies in a style, overriding any prior settings.

### Usage

```r
setStyleDependencies(
  style.name = NULL,
  dependencies,
  base.url = .defaultBaseUrl
)
```

### Arguments

- **style.name**  
  Name of style; default is "default" style

- **dependencies**  
  A list of style dependencies, see Available Dependencies below. Note: each dependency is set by a boolean, TRUE or FALSE (T or F)

- **base.url**  
  (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

### Value

server response

### Available Dependencies

- arrowColorMatchesEdge
- nodeCustomGraphicsSizeSync
- nodeSizeLocked

### Examples

```r
setStyleDependencies("myStyle", list(nodeSizeLocked=TRUE))
```
### setVisualPropertyDefault

**Set Visual Property Default**

**Description**

Set the default value for a visual property.

**Usage**

```r
setVisualPropertyDefault(
  style.string,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

- **style.string**: A named list including "visualProperty" and "value"
- **style.name**: Name of style; default is "default" style
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Value**

None

**Examples**

```r
setVisualPropertyDefault(list(visualProperty = "NODE_SIZE", value = 35))
```

---

### setVisualStyle

**Set Visual Style**

**Description**

Apply a visual style to a network.

**Usage**

```r
setVisualStyle(style.name, network = NULL, base.url = .defaultBaseUrl)
```
Arguments

- **style.name**: Name of a visual style (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **network**: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

None

Examples

```r
setVisualStyle()
```

---

**spoofResponse-class**  
**spoofResponse**

Description

Call CyREST as a remote service via Jupyter-bridge

---

**syncNodeCustomGraphicsSize**  
**Sync Node Custom Graphics Size**

Description

Set a boolean value to have the size of custom graphics match that of the node.

Usage

```r
syncNodeCustomGraphicsSize(
  new.state,  
  style.name = NULL,  
  base.url = .defaultBaseUrl  
)
```

Arguments

- **new.state**: (Boolean) Whether to sync node custom graphics size
- **style.name**: Name of style; default is "default" style
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
**toggleGraphicsDetails**  
*Toggle Graphics Details*

**Description**

Regardless of the current zoom level and network size, show (or hide) graphics details, e.g., node labels.

**Usage**

```r
toggleGraphicsDetails(base.url = .defaultBaseUrl)
```

**Arguments**

- `base.url` (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is `http://localhost:1234` and the latest version of the CyREST API supported by this version of RCy3.

**Details**

Displaying graphics details on a very large network will affect pan and zoom performance, depending on your available RAM. See `cytoscapeMemoryStatus`.

**Value**

None

**Examples**

```r
showGraphicsDetails(TRUE)
```
ungroupAnnotation  

Ungroup Annotation Group

---

### Description

Ungroup annotation group from the network view in Cytoscape

### Usage

```r
ungroupAnnotation(names = NULL, network = NULL, base.url = .defaultBaseUrl)
```

### Arguments

- **names**: Name of annotation group by UUID or Name  
- **network** (optional): Name or SUID of the network. Default is the "current" network active in Cytoscape.  
- **base.url** (optional): Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

### Details

You can obtain a list of UUIDs by applying a subset function like so: `sapply(getAnnotationList(), '[[', 'uuid')`

### Value

None

### Examples

```r
ungroupAnnotation("016a4af1-69bc-4b99-8183-d6f118847f96")
ungen(groupAnnotation(c("316869a4-39fc-4731-8f45-199dec9af10d","c3621eb4-4687-490f-9396-b829dd8767d5"))
ungroupAnnotation("Group 1")
ungroupAnnotation(c("Group1","Group2", "Group3"))
```
Unhide all previously hidden nodes and edges, by clearing the Visible property bypass value.

Usage

unhideAll(network = NULL, base.url = .defaultBaseUrl)

Arguments

- network: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- base.url: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method sets node and edge visibility bypass to true, overriding any defaults or mappings. Pending CyREST updates, this method will ultimately call the generic function, clearEdgePropertyBypass, which can be used to clear any visual property.

Value

None

See Also

clearEdgePropertyBypass, unhideNodes unhideEdges

Examples

unhideAll()
unhideEdges  

Unhide Edges

Description

Unhide specified edges that were previously hidden, by clearing the Visible property bypass value.

Usage

unhideEdges(edge.names, network = NULL, base.url = .defaultBaseUrl)

Arguments

edge.names  
List of edge names or SUIDs

network  
(optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

base.url  
(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details

This method ultimately calls the generic function, clearEdgePropertyBypass, which can be used to clear any visual property.

Value

None

See Also

clearEdgePropertyBypass, unhideAll

Examples

unhideEdges()
unhideNodes  

**Description**

Unhide specified nodes that were previously hidden, by clearing the Node Visible property bypass value.

**Usage**

```r
unhideNodes(node.names, network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- **node.names**  List of node names or SUIDs
- **network**     (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**    (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

**Details**

This method ultimately calls the generic function, `clearNodePropertyBypass`, which can be used to clear any visual property.

**Value**

None

**See Also**

`clearNodePropertyBypass`, `unhideAll`

**Examples**

```r
unhideNodes()
```
uninstallApp  Uninstall App

Description
Uninstall an app from Cytoscape.

Usage
uninstallApp(app, base.url = .defaultBaseUrl)

Arguments

app       Name of app
base.url  (optional) Ignore unless you need to specify a custom domain, port or version
to connect to the CyREST API. Default is http://localhost:1234 and the latest
version of the CyREST API supported by this version of RCy3.

Value
None

Examples
uninstallApp()

UpdateAnnotationBoundedText  Update Bounded Text Annotation

Description
Adds a bounded text annotation to a Cytoscape network view. The object will also be added to the
Annotation Panel in the GUI.

Usage
UpdateAnnotationBoundedText(
  text = NULL,
  annotationName = NULL,
  x.pos = NULL,
  y.pos = NULL,
  fontSize = NULL,
  fontFamily = NULL,
  fontStyle = NULL,
color = NULL,
angle = NULL,
type = NULL,
customShape = NULL,
fillColor = NULL,
opacity = NULL,
borderThickness = NULL,
borderColor = NULL,
borderOpacity = NULL,
height = NULL,
width = NULL,
name = NULL,
canvas = NULL,
z.order = NULL,
network = NULL,
base.url = .defaultBaseUrl
)

Arguments

text The text to be displayed
annotationName Name of annotation by UUID or Name
x.pos (optional) X position in pixels from left; default is center of current view
y.pos (optional) Y position in pixels from top; default is center of current view
fontSize (optional) Numeric value; default is 12
fontFamily (optional) Font family; default is Arial
fontStyle (optional) Font style; default is
color (optional) Hexidecimal color; default is #000000 (black)
angle (optional) Angle of text orientation; default is 0.0 (horizontal)
type (optional) The type of the shape, default is RECTANGLE. See getNodeShapes() for valid options.
customShape (optional) If a custom shape, this is the text of the shape
fillColor (optional) Hexidecimal color; default is #000000 (black)
opacity (optional) Opacity of fill color. Must be an integer between 0 and 100; default is 100.
borderThickness (optional) Integer
borderColor (optional) Hexidecimal color; default is #000000 (black)
borderOpacity (optional) Integer between 0 and 100; default is 100.
height (optional) Height of bounding shape; default is based on text height.
width (optional) Width of bounding shape; default is based on text length.
name (optional) Name of annotation object; default is "Text"
canvas (optional) Canvas to display annotation, i.e., foreground (default) or background
updateAnnotationImage

**Value**

A named list of annotation properties, including UUID

**Examples**

```r
UpdateAnnotationBoundedText("test1", "annotationName")
UpdateAnnotationBoundedText("test2", "annotationName", 1000, 1000, name="B2")
UpdateAnnotationBoundedText("test3", "annotationName", 1200, 1000, 30, "Helvetica", "bold", "#990000", 40, name="B3", canvas="foreground",z=4)
```

**Description**

Updates a Image annotation to a Cytoscape network view. The object will also be added to the Annotation Panel in the GUI.

**Usage**

```r
updateAnnotationImage(
  url = NULL,
  annotationName = NULL,
  x.pos = NULL,
  y.pos = NULL,
  angle = NULL,
  opacity = NULL,
  brightness = NULL,
  contrast = NULL,
  borderThickness = NULL,
  borderColor = NULL,
  borderOpacity = NULL,
  height = NULL,
  width = NULL,
  name = NULL,
  canvas = NULL,
  z.order = NULL,
)```
updateAnnotationImage

```
network = NULL,
base.url = .defaultBaseUrl
```

### Arguments

**url**
- URL or path to image file. File paths can be absolute or relative to current working directory. URLs must start with http:// or https://.

**annotationName**
- Name of annotation by UUID or Name

**x.pos**
- (optional) X position in pixels from left; default is center of current view

**y.pos**
- (optional) Y position in pixels from top; default is center of current view

**angle**
- (optional) Angle of text orientation; default is 0.0 (horizontal)

**opacity**
- (optional) Opacity of fill color. Must be an integer between 0 and 100; default is 100.

**brightness**
- (optional) Image brightness. Must be an integer between -100 and 100; default is 0

**contrast**
- (optional) Image contrast. Must be an integer between -100 and 100; default is 0

**borderThickness**
- (optional) Integer

**borderColor**
- (optional) Hexidecimal color; default is #000000 (black)

**borderOpacity**
- (optional) Integer between 0 and 100; default is 100.

**height**
- (optional) Height of image; default is based on text height.

**width**
- (optional) Width of image; default is based on text length.

**name**
- (optional) Name of annotation object; default is "Image"

**canvas**
- (optional) Canvas to display annotation, i.e., foreground (default) or background

**z.order**
- (optional) Arrangement order specified by number (larger values are in front of smaller values); default is 0

**network**
- (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

**base.url**
- (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

### Value

A named list of annotation properties, including UUID

### Examples

updateAnnotationImage("image.png")
updateAnnotationImage("/Users/janedoe/Desktop/image.png", 1000, 1000, name="I2")
updateAnnotationImage("https://www.example.com/image.png", 1200, 1000, 30, 40, name="I3", canvas="background",z=4)
updateAnnotationShape  Update Shape Annotation

Description

Updates a shape annotation to a Cytoscape network view. The object will also be added to the Annotation Panel in the GUI.

Usage

updateAnnotationShape(
  type = NULL,
  customShape = NULL,
  annotationName = NULL,
  x.pos = NULL,
  y.pos = NULL,
  angle = NULL,
  fillColor = NULL,
  opacity = NULL,
  borderThickness = NULL,
  borderColor = NULL,
  borderOpacity = NULL,
  height = NULL,
  width = NULL,
  name = NULL,
  canvas = NULL,
  z.order = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)

Arguments

type  (optional) The type of the shape, default is RECTANGLE. See getNodeShapes() for valid options.
customShape  (optional) If a custom shape, this is the text of the shape
annotationName  Name of annotation by UUID or Name
x.pos  (optional) X position in pixels from left; default is center of current view
y.pos  (optional) Y position in pixels from top; default is center of current view
angle  (optional) Angle of text orientation; default is 0.0 (horizontal)
fillColor  (optional) Hexidecimal color; default is #000000 (black)
opacity  (optional) Opacity of fill color. Must be an integer between 0 and 100; default is 100.
borderThickness  (optional) Integer
### borderTopColor
(optional) Hexidecimal color; default is #000000 (black)

### borderTopOpacity
(optional) Integer between 0 and 100; default is 100.

### height
(optional) Height of shape; default is based on text height.

### width
(optional) Width of shape; default is based on text length.

### name
(optional) Name of annotation object; default is "Shape"

### canvas
(optional) Canvas to display annotation, i.e., foreground (default) or background

### z.order
(optional) Arrangement order specified by number (larger values are in front of smaller values); default is 0

### network
(optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.

### base.url
(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

### Value
A named list of annotation properties, including UUID

### Examples
```
updateAnnotationShape("rectangle", "annotationName")
updateAnnotationShape("rectangle", "annotationName", 1000, 1000, name="S2")
updateAnnotationShape("rectangle", "annotationName", 1200, 1000, 30, "#990000", 40,name="S3", canvas="background",z=4)
```

---

### updateAnnotationText  Update Text Annotation

### Description
Updates a text annotation to a Cytoscape network view. The object will also be added to the Annotation Panel in the GUI.

### Usage
```
updateAnnotationText(
  text = NULL,
  annotationName = NULL,
  x.pos = NULL,
  y.pos = NULL,
  fontSize = NULL,
  fontFamily = NULL,
  fontStyle = NULL,
  color = NULL,
```
Arguments

text: The text to be displayed
annotationName: Name of annotation by UUID or Name
x.pos: (optional) X position in pixels from left; default is center of current view
y.pos: (optional) Y position in pixels from top; default is center of current view
fontSize: (optional) Numeric value; default is 12
fontFamily: (optional) Font family; default is Arial
fontStyle: (optional) Font style; default is
color: (optional) Hexidecimal color; default is #000000 (black)
angle: (optional) Angle of text orientation; default is 0.0 (horizontal)
name: (optional) Name of annotation object; default is "Text"
canvas: (optional) Canvas to display annotation, i.e., foreground (default) or background
z.order: (optional) Arrangement order specified by number (larger values are in front of smaller values); default is 0
network: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
base.url: (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A named list of annotation properties, including UUID

Examples

```
updateAnnotationText("test1", "annotationName")
updateAnnotationText("test2", "annotationName", 1000, 1000, name="T2")
updateAnnotationText("test3", "annotationName", 1200, 1000, 30, "Helvetica", "bold", "#990000", 40,name="T3", canvas="foreground",z=4)
```
updateApp

Description
Update a Cytoscape app to the latest available version.

Usage
updateApp(app, base.url = .defaultBaseUrl)

Arguments
app
Name of app
base.url
(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value
None

Examples
updateApp()

updateGroupAnnotation

Description
Updates a group annotation, changing the given properties.

Usage
updateGroupAnnotation(
  name = NULL,
  annotationName = NULL,
  x.pos = NULL,
  y.pos = NULL,
  angle = NULL,
  canvas = NULL,
  z.order = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)

updateNetworkInNDEx

Arguments

- **name** (optional) Name of annotation object
- **annotationName** Name of annotation by UUID or Name
- **x.pos** (optional) X position in pixels from left; default is center of current view
- **y.pos** (optional) Y position in pixels from top; default is center of current view
- **angle** (optional) Angle of text orientation; default is 0.0 (horizontal)
- **canvas** (optional) Canvas to display annotation, i.e., foreground (default) or background
- **z.order** (optional) Arrangement order specified by number (larger values are in front of smaller values); default is 0
- **network** (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url** (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Value

A named list of annotation properties, including UID

Examples

```r
updateGroupAnnotation("test1", "annotationName")
```

updateNetworkInNDEx

Update Network In NDEx

Description

Update an existing network in NDEx, given a previously associated Cytoscape network, e.g., previously exported to NDEx or imported from NDEx.

Usage

```r
updateNetworkInNDEx(
  username,
  password,
  isPublic,
  network = NULL,
  metadata = NULL,
  base.url = .defaultBaseUrl
)
```
updateStyleDefaults

Updates the default values of visual properties in a style

Description

Updates visual property defaults, overriding any prior settings. See mapVisualProperty for the list of visual properties.

Usage

updateStyleDefaults(style.name, defaults, base.url = .defaultBaseUrl)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>style.name</td>
<td>(char) name for style</td>
</tr>
<tr>
<td>defaults</td>
<td>(list) a list of visual property default settings</td>
</tr>
<tr>
<td>base.url</td>
<td>(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://localhost:1234">http://localhost:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

Value

server response

updateNetworkInNDEx

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>username</td>
<td>NDEx account username</td>
</tr>
<tr>
<td>password</td>
<td>NDEx account password</td>
</tr>
<tr>
<td>isPublic</td>
<td>(Boolean) Whether to make the network publicly accessible at NDEx.</td>
</tr>
<tr>
<td>network</td>
<td>(optional) Name or SUID of the network. Default is the “current” network active in Cytoscape.</td>
</tr>
<tr>
<td>metadata</td>
<td>(optional) A list of structured information describing the network</td>
</tr>
<tr>
<td>base.url</td>
<td>(optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is <a href="http://localhost:1234">http://localhost:1234</a> and the latest version of the CyREST API supported by this version of RCy3.</td>
</tr>
</tbody>
</table>

Value

NDEx identifier (externalId) for the updated submission

Examples

updateNetworkInNDEx("user", "pass", TRUE)
updateStyleMapping

See Also
mapVisualProperty

Examples
updateStyleDefaults('myStyle',list('node fill color'='#0000FF','node size'=50))

updateStyleMapping(style.name, mapping, base.url = .defaultBaseUrl)

Arguments
style.name (char) name for style
mapping a single visual property mapping, see mapVisualProperty
base.url (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.

Details
Requires visual property mappings to be previously created, see mapVisualProperty.

Value
server response

See Also
mapVisualProperty

Examples
updateStyleMapping('myStyle',mapVisualProperty('node label','name','p'))
Index

.addAnnotationBoundedText, 13
.addAnnotationImage, 15
.addAnnotationShape, 17
.addAnnotationText, 18
.addCyEdges, 20
.addCyNodes, 21
.AddToGroup, 22
.analyzeNetwork, 23
.applyFilter, 23

.bundleEdges, 25

.checkNotebookIsRunning, 25
.checkRunningRemote, 26

.clearEdgeBends, 26
.clearNetworkCenterBypass, 28
.clearNetworkPropertyBypass, 29, 271–273
.clearNetworkZoomBypass, 30
.clearNodeOpacityBypass, 30, 319
.clearSelection, 32
.cloneNetwork, 33
.closeSession, 33
.collapseGroup, 34
.commandEcho, 35
.commandOpenDialog, 36
.commandPause, 36
.commandQuit, 37
.commandRunFile, 37

.commandsAPI, 38
.commandsGET, 39
.commandsHelp, 39
.commandSleep, 40
.commandsPOST, 41
.commandsRun, 41
.copyVisualStyle, 42
.createColumnFilter, 43
.createCompositeFilter, 44
.createCytoscapejsFromNetwork, 45
.createDegreeFilter, 46
.createGraphFromNetwork, 47
.createGroup, 48
.createGroupByColumn, 49
.createIgraphFromNetwork, 50
.createNetworkFromCytoscapejs, 51
.createNetworkFromDataFrames, 51
.createNetworkFromGraph, 53
.createNetworkFromIgraph, 54
.createSubnetwork, 55
.createView, 56
.createVisualStyle, 57
.cybrowserClose, 58
.cybrowserDialog, 59, 60, 62
.cybrowserHide, 59, 60, 62
.cybrowserList, 60
.cybrowserSend, 61
.cybrowserShow, 59, 60, 62
.cybrowserVersion, 63
.cyrestAPI, 63
.cyrestDELETE, 64
.cyrestGET, 64
.cyrestPOST, 65
.cyrestPUT, 66
.cytoscapeApiVersions, 67
.cytoscapeFreeMemory, 67
.cytoscapeMemoryStatus, 68, 337
.cytoscapeNumberOfCores, 69
.cytoscapePing, 69

353
cytoscapeVersionInfo, 70
deleteAllNetworks, 71
deleteAllVisualStyle, 71
deleteAnnotation, 72
deleteDuplicateEdges, 72
deleteGroup, 73
deleteNetwork, 74
deleteSelectedEdges, 75
deleteSelectedNodes, 75
deleteSelfLoops, 76
deleteVisualStyle, 77
deleteTableColumn, 77
deleteVisualStyle, 78
diffusionAdvanced, 79
diffusionBasic, 80
disableApp, 80
dockPanel, 81
doInitializeSandbox, 82
doRequestRemote, 82
doSetSandbox, 83
enableApp, 84
expandGroup, 84
exportFilters, 85
exportImage, 86
exportJPG, 87
exportNetwork, 88
exportNetworkToNDEx, 89
exportPDF, 90
exportPNG, 91
exportPS, 93
exportSVG, 94
exportVisualStyle, 95
findRemoteCytoscape, 96
fitContent, 87, 97
floatPanel, 97
getAbsSandboxPath, 98
getAllEdges, 99
getAllNodes, 99
getAllStyleMappings, 100
getAnnotationList, 101
getAppInformation, 101
getAppStatus, 102
getAppUpdates, 103
getArrowShapes, 103, 256–259, 263–265
getAvailableApps, 104
getBackgroundColorDefault, 112
getEdgeColor, 112
getEdgeCount, 113
getEdgeInfo, 114
getEdgeLineStyle, 115
getEdgeLineWidth, 115
getEdgeProperty, 116
getEdgeSelectionColorDefault, 117
getEdgeTargetArrowShape, 118
getFilterList, 118
getFirstNeighbors, 119
getGroupInfo, 120
getInstalledApps, 121
getJupyterBridgeURL, 121
getLayoutNameMapping, 122
getLayoutNames, 122, 123–125, 166, 269
getLayoutPropertyNames, 123, 124, 125, 269
getLayoutPropertyType, 124
getLayoutPropertyValue, 125
getLineStyles, 126, 243, 244
getNetworkCenter, 126
getNetworkCount, 127
getNetworkList, 128
getNetworkName, 128
getNetworkNDExId, 129
getNetworkProperty, 130
getNetworkSuid, 130
getNetworkViews, 131
getNetworkViewSuid, 132
getNetworkZoom, 132
getNodeColor, 133
getNodeCount, 134
getNodeHeight, 134
ggetNodeLabelPosition, 135
renameNetwork, 200
renameTableColumn, 201
resetDefaultSandbox, 202
rotateLayout, 203
runningRemoteCheck, 204
sandboxGetFileInfo, 204
sandboxGetFrom, 205
sandboxInitializer, 206
sandboxRemove, 207
sandboxRemoveFile, 207
sandboxSendTo, 208
sandboxSet, 209
sandboxUrlTo, 210
saveSession, 179, 211
scaleLayout, 212
selectAll, 213
selectAllEdges, 213
selectAllNodes, 214
selectEdges, 215
selectEdgesAdjacentToNodes, 216
selectEdgesAdjacentToSelectedNodes, 217
selectEdgesConnectingSelectedNodes, 217
selectFirstNeighbors, 218
selectNodes, 213, 215, 219
selectNodesConnectedBySelectedEdges, 220
setBackgroundColorDefault, 220
setCatchupFilterSecs, 221
setCatchupNetworkSecs, 222
setCurrentNetwork, 223
setCurrentSandbox, 223
setCurrentView, 224
setDefaultSandbox, 225
setDefaultSandboxPath, 225
setEdgeColorBypass, 226
setEdgeColorDefault, 227
setEdgeColorMapping, 227
setEdgeFontFaceBypass, 228
setEdgeFontFaceDefault, 229
setEdgeFontFaceMapping, 230
setEdgeFontSizeBypass, 231
setEdgeFontSizeDefault, 232
setEdgeFontSizeMapping, 233
setEdgeLabelBypass, 234
setEdgeLabelColorBypass, 235
setEdgeLabelColorDefault, 236
setEdgeLabelColorMapping, 237
setEdgeLabelDefault, 238
setEdgeLabelMapping, 238
setEdgeOpacityBypass, 239
setEdgeOpacityDefault, 240
setEdgeOpacityMapping, 241
setEdgeLineStyleBypass, 242
setEdgeLineStyleDefault, 243
setEdgeLineStyleMapping, 244
setEdgeLineWidthBypass, 245
setEdgeLineWidthDefault, 246
setEdgeLineWidthMapping, 247
setEdgeOpacityBypass, 248
setEdgeOpacityDefault, 249
setEdgeOpacityMapping, 250
setEdgeSelectionColorDefault, 252
setEdgeSourceArrowColorBypass, 253
setEdgeSourceArrowColorDefault, 254
setEdgeSourceArrowColorMapping, 254
setEdgeSourceArrowMapping, 256
setEdgeSourceArrowShapeBypass, 257
setEdgeSourceArrowShapeDefault, 258
setEdgeSourceArrowShapeMapping, 258
setEdgeTargetArrowColorBypass, 259
setEdgeTargetArrowColorDefault, 260
setEdgeTargetArrowColorMapping, 261
setEdgeTargetArrowMapping, 262
setEdgeTargetArrowShapeBypass, 263
setEdgeTargetArrowShapeDefault, 264
setEdgeTargetArrowShapeMapping, 265
setEdgeTooltipBypass, 266
setEdgeTooltipDefault, 267
setEdgeTooltipMapping, 268
setLayoutProperties, 269
setModelPropagationSecs, 270
setNetworkCenterBypass, 271
setNetworkPropertyBypass, 271, 272, 273
setNetworkZoomBypass, 273
setNodeBorderColorBypass, 274
setNodeBorderColorDefault, 275
setNodeBorderColorMapping, 275
setNodeBorderOpacityBypass, 277
setNodeBorderOpacityDefault, 278
setNodeBorderOpacityMapping, 278
setNodeBorderWidthBypass, 280
INDEX

setNodeBorderWidthDefault, 281
setNodeBorderWidthMapping, 281
setNodeBorderColorBypass, 283
setNodeBorderColorDefault, 284
setNodeBorderColorMapping, 284
setNodeComboBoxOpacityMapping, 285
setNodeCustomBarChart, 286
setNodeCustomBoxChart, 288
setNodeCustomHeatMapChart, 289
setNodeCustomLineGradient, 290
setNodeCustomLineChart, 291
setNodeCustomPieChart, 292
setNodeCustomPosition, 293
setNodeCustomRadialGradient, 294
setNodeCustomRingChart, 295
setNodeFillOpacityBypass, 296
setNodeFillOpacityDefault, 298
setNodeFillOpacityMapping, 298
setNodeFontSizeBypass, 300
setNodeFontSizeDefault, 301
setNodeFontSizeMapping, 301
setNodeHeightBypass, 303
setNodeHeightDefault, 304
setNodeHeightMapping, 304
setNodeLabelBypass, 308
setNodeLabelColorBypass, 309
setNodeLabelColorDefault, 310
setNodeLabelColorMapping, 311
setNodeLabelDefault, 312
setNodeLabelMapping, 313
setNodeLabelOpacityBypass, 313
setNodeLabelOpacityDefault, 314
setNodeLabelOpacityMapping, 315
setNodeLabelPositionBypass, 316
setNodeLabelPositionDefault, 317
setNodeOpacityBypass, 318
setNodePositionBypass, 319
setNodeSelectionColorDefault, 321
setNodeShapeBypass, 322
setNodeShapeDefault, 323
setNodeShapeMapping, 324

synNodeSizeBypass, 325
setNodeSizeDefault, 326
setNodeSizeMapping, 326
setNodeTooltipBypass, 327
setNodeTooltipDefault, 328
setNodeTooltipMapping, 329
setNodeWidthBypass, 330
setNodeWidthDefault, 331
setNodeWidthMapping, 331
setNotebookIsRunning, 333
setSandboxReinitialize, 333
setStyleDependencies, 334
setVisualStyle, 335
spoofResponse (spoofResponse-class), 336
spoofResponse-class, 336
syncNodeCustomGraphicsSize, 336
toggleGraphicsDetails, 337
ungroupAnnotation, 338
unhideAll, 155–158, 339, 340, 341
unhideEdges, 155, 157, 339, 340
unhideNodes, 155, 156, 158, 339, 341
uninstallApp, 342
UpdateAnnotationBoundedText, 342
updateAnnotationImage, 344
updateAnnotationShape, 346
updateAnnotationText, 347
updateApp, 349
updateGroupAnnotation, 349
updateNetworkInNDEex, 350
updateStyleDefaults, 351
updateStyleMapping, 352