Package ‘RCy3’

March 19, 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 R Cy3 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 cd4eb1b
git_last_commit_date 2023-11-06
Repository Bioconductor 3.18
Date/Publication 2024-03-18
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
R topics documented:

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
cybrowserHide ............................................... 60
cybrowserList ............................................... 60
cybrowserSend .............................................. 61
cybrowserShow .............................................. 62
cybrowserVersion ............................................ 63
cyrestAPI ....................................................... 63
cyrestDELETE .................................................. 64
cyrestGET ....................................................... 64
cyrestPOST ..................................................... 65
cyrestPUT ....................................................... 66
cytoscapeApiVersions ...................................... 67
cytoscapeFreeMemory ....................................... 67
cytoscapeMemoryStatus .................................... 68
cytoscapeNumberOfCores ................................... 69
cytoscapePing ................................................. 69
cytoscapeVersionInfo ....................................... 70
deleteAllNetworks .......................................... 70
deleteAllVisualStyle ...................................... 71
deleteAnnotation ............................................. 72
deleteDuplicateEdges ....................................... 72
deleteGroup ................................................... 73
deleteNetwork ............................................... 74
deleteSelectedEdges ....................................... 75
deleteSelectedNodes ....................................... 75
deleteSelfLoops .............................................. 76
deleteStyleMapping ......................................... 77
deleteTableColumn .......................................... 77
deleteVisualStyle ............................................ 78
diffusionAdvanced .......................................... 79
diffusionBasic ............................................... 80
disableApp ..................................................... 80
R topics documented:

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
getDefaultSandbox ............................................. 111
getDefaultSandboxPath ....................................... 111
getDisabledApps ............................................... 112
getEdgeColor ..................................................... 112
getEdgeCount ..................................................... 113
getEdgeInfo ....................................................... 114
getEdgeLineStyle ............................................... 115
getEdgeLineWidth ............................................... 115
R topics documented:

getEdgeProperty ..................................................... 116
getEdgeSelectionColorDefault .................................. 117
getEdgeTargetArrowShape ......................................... 118
getFilterList ......................................................... 118
getFirstNeighbors .................................................. 119
getGroupInfo ......................................................... 120
getInstalledApps .................................................... 121
getJupyterBridgeURL ................................................ 121
getLayoutNameMapping ............................................. 122
getLayoutNames ...................................................... 122
getLayoutPropertyNames .......................................... 123
getLayoutPropertyType ............................................. 124
getLayoutPropertyValue ............................................ 125
getLineStyles ......................................................... 126
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
getNodeLabelPosition .............................................. 135
getNodeLabelPositionDefault ...................................... 136
getNodePosition ..................................................... 136
getNodeProperty ..................................................... 137
getNodeSelectionColorDefault .................................... 138
getNodeShapes ....................................................... 139
getNodeSize ........................................................ 139
getNodeWidth ....................................................... 140
getNotebookIsRunning ............................................. 141
getSandboxReinitialize ............................................. 141
gSelectedEdgeCount ................................................ 142
getSelectedEdges ................................................... 142
getSelectedNodeCount .............................................. 143
getSelectedNodes ................................................... 144
getStyleDependencies ............................................. 145
getStyleMapping ..................................................... 145
gTableColumnNames ................................................ 146
gTableColumns ....................................................... 147
gTableColumnTypes ................................................ 148
gTableValue ........................................................ 149
gUninstalledApps .................................................... 150
<table>
<thead>
<tr>
<th>R topics documented:</th>
</tr>
</thead>
<tbody>
<tr>
<td>getVisualPropertyDefault</td>
</tr>
<tr>
<td>getVisualPropertyNames</td>
</tr>
<tr>
<td>getVisualStyleJSON</td>
</tr>
<tr>
<td>getVisualStyleNames</td>
</tr>
<tr>
<td>groupAnnotation</td>
</tr>
<tr>
<td>hideAllPanels</td>
</tr>
<tr>
<td>hideEdges</td>
</tr>
<tr>
<td>hideNodes</td>
</tr>
<tr>
<td>hidePanel</td>
</tr>
<tr>
<td>hideSelectedEdges</td>
</tr>
<tr>
<td>hideSelectedNodes</td>
</tr>
<tr>
<td>importFileFromUrl</td>
</tr>
<tr>
<td>importFilters</td>
</tr>
<tr>
<td>importNetworkFromFile</td>
</tr>
<tr>
<td>importNetworkFromNDEx</td>
</tr>
<tr>
<td>importVisualStyles</td>
</tr>
<tr>
<td>installApp</td>
</tr>
<tr>
<td>invertEdgeSelection</td>
</tr>
<tr>
<td>invertNodeSelection</td>
</tr>
<tr>
<td>layoutCopycat</td>
</tr>
<tr>
<td>layoutNetwork</td>
</tr>
<tr>
<td>listGroups</td>
</tr>
<tr>
<td>loadTableData</td>
</tr>
<tr>
<td>lockNodeDimensions</td>
</tr>
<tr>
<td>makeSimpleGraph</td>
</tr>
<tr>
<td>makeSimpleIgraph</td>
</tr>
<tr>
<td>mapTableColumn</td>
</tr>
<tr>
<td>mapVisualStyle</td>
</tr>
<tr>
<td>matchArrowColorToEdge</td>
</tr>
<tr>
<td>mergeNetworks</td>
</tr>
<tr>
<td>notebookExportShowImage</td>
</tr>
<tr>
<td>notebookShowImage</td>
</tr>
<tr>
<td>openAppStore</td>
</tr>
<tr>
<td>openSession</td>
</tr>
<tr>
<td>paletteColorBrewerAccent</td>
</tr>
<tr>
<td>paletteColorBrewerBlues</td>
</tr>
<tr>
<td>paletteColorBrewerBrBG</td>
</tr>
<tr>
<td>paletteColorBrewerBuGn</td>
</tr>
<tr>
<td>paletteColorBrewerBuPu</td>
</tr>
<tr>
<td>paletteColorBrewerDark2</td>
</tr>
<tr>
<td>paletteColorBrewerGnBu</td>
</tr>
<tr>
<td>paletteColorBrewerGreens</td>
</tr>
<tr>
<td>paletteColorBrewerGreys</td>
</tr>
<tr>
<td>paletteColorBrewerOranges</td>
</tr>
<tr>
<td>paletteColorBrewerOrRd</td>
</tr>
<tr>
<td>paletteColorBrewerPaired</td>
</tr>
<tr>
<td>paletteColorBrewerPastel1</td>
</tr>
<tr>
<td>paletteColorBrewerPastel2</td>
</tr>
</tbody>
</table>
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 ............................................. 280
setNodeBorderWidthDefault ......................................... 281
setNodeBorderWidthMapping ......................................... 281
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
setVisualStyleDefault ..................................... 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

```r
.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".
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
 exportedShowImage()
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
  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

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
)

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

addAnnotationBoundedText("test1")
addAnnotationBoundedText("test2", 1000, 1000, name="B2")
addAnnotationBoundedText("test3", 1200, 1000, 30, "Helvetica", "bold", "#990000", 40, name="B3", canvas="foreground", z=4)
addAnnotationImage

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
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

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**  

**Add Shape Annotation**

**Description**

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

**Usage**

```r
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) 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)
addAnnotationShape

borderOpacity (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

```
addAnnotationShape("rectangle")
addAnnotationShape("rectangle", 1000, 1000, name="S2")
addAnnotationShape("rectangle", 1200, 1000, 30, "#990000", 40,name="S3", canvas="background",z=4)
```

addAnnotationText

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

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

value

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

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

Description

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

Usage

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

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

Add CyEdges
addCyNodes  

Add CyNodes

Description

Add one or more nodes to a Cytoscape network.

Usage

```r
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

```r
addCyNodes(c('Node A', 'Node B', 'Node C'))
```
**AddToGroup**

**Description**

Adds the specified nodes and edges to the specified group.

**Usage**

```r
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**

```r
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

analyzeNetwork()
analyzeNetwork(TRUE)

applyFilter

Description

Run an existing filter by supplying the filter name.
applyFilter

Usage

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

Arguments

filter.name  Name of filter to apply. Default is "Default filter".
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.

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

applyFilter('myFilter')
applyFilter('myFilter', hide = TRUE)
### bundleEdges

**Bundle Edges**

<table>
<thead>
<tr>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply edge bundling to the network specified. Edge bundling is executed with default parameters; optional parameters are not supported.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Usage</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>bundleEdges(network = NULL, base.url = .defaultBaseUrl)</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Arguments</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>network</code> (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> (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>

<table>
<thead>
<tr>
<th><strong>Value</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Examples</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>bundleEdges()</code></td>
</tr>
</tbody>
</table>

---

### checkNotebookIsRunning

**checkNotebookIsRunning**

<table>
<thead>
<tr>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>checkNotebookIsRunning</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Usage</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>checkNotebookIsRunning()</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Value</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
</tr>
</tbody>
</table>
clearEdgeBends

Examples

```r
clearEdgeBends()
```

description

Clear Edge Bends

Description

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

Usage

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

description

Determine whether we’re running locally or on a remote server. If locally (either via raw R or via a locally installed Notebook), we prefer to connect to Cytoscape over a local socket. If remote, we have to connect over Jupyter-Bridge. Either way, we can determine which by whether Cytoscape answers to a version check. If Cytoscape doesn’t answer, we have no information ... and we have to wait until Cytoscape is started and becomes reachable before we can determine local vs remote.

Usage

```r
checkRunningRemote(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
checkRunningRemote()
```
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

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

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()

---

clearNodeOpacityBypass

Clear Node Opacity Bypass

Description

Clear the bypass value for node fill, label and border opacity for the specified node or nodes, effectively restoring any previously defined style defaults or mappings.

Usage

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

clearNodePropertyBypass(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**

```r
clearNodePropertyBypass()
```

---

**clearSelection**  
**Clear Selection**

**Description**

If any nodes are selected in the network, they will be unselected.

**Usage**

```r
clearSelection(type = "both", network = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- **type**  
  'nodes', 'edges' or 'both' (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**

None

**Author(s)**

AlexanderPico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

**Examples**

```r
clearSelection()
```
cloneNetwork

**Description**

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

**Usage**

```r
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**

```r
cloneNetwork("cloned network")
```

---

closeSession

**Description**

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

**Usage**

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

- **save.before.closing**
  - Definition: Whether to save before closing the current session. If FALSE, then all unsaved work will be lost.
- **filename**
  - Definition: (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**
  - Definition: (optional) Ignore unless you need to specify a custom domain, port or version to 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**

**Collapsible 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**
  - Definition: (optional) List of group names or keywords: all, selected, unselected. Default is the currently selected group.
- **network**
  - Definition: (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- **base.url**
  - Definition: (optional) Ignore unless you need to specify a custom domain, port or version to 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
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
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
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.')

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)
commandsAPI

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

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

---

Open Swagger docs for CyREST Commands API

Description

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

Usage

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

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)
commandSleep

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 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
cmdsPOST('layout get preferred')
cmdsPOST('network list properties')
cmdsPOST('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

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmd.string</td>
<td>(char) command</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

A list, status or None.

Examples

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

---

**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

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>from.style</td>
<td>Name of visual style to copy</td>
</tr>
<tr>
<td>to.style</td>
<td>Name of new visual 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
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 CytoscapeJS 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.

Examples

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

createDegreeFilter

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

filter.name Name for filter.
criterion A two-element vector of numbers, example: c(1,5).
predicate BETWEEN (default) or IS_NOT_BETWEEN
edgeType (optional) Type of edges to consider in degree count: ANY (default), UNDIRECTED, 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+).
Value

List of selected nodes and edges.

Examples

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

desc

createGraphFromNetwork

desc

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

```r
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

```r
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.
createNetworkFromDataFrames

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.
...

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 0"),
  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

Examples

```r
library(igraph)
ig <- makeSimpleIgraph()
createNetworkFromIgraph(ig)
```

createSubnetwork  Create subnetwork from existing network

Description

Copies a subset of nodes and edges into a newly created subnetwork.

Usage

```r
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.
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

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

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

Value

Network view SUID

Examples

generateNetworkViews()

createVisualStyle  Create a visual style from components

Description

Create a style from defaults and predefined mappings.

Usage

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')
cybrowserDialog

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

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

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.

**Value**

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)
```
cybrowserSend

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;")
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

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

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)
**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')
```

---

**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

cyrestPUT('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**

```r
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**

```r
cyrestPUT()
```
cytoscapeApiVersions  

Available CyREST API Versions

Description
Get the list of available CyREST API versions

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

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)
cytoscapeVersionInfo

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!"

---

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**

**Description**
Delete all networks from the current Cytoscape session.

**Usage**
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**
deleteAllNetworks()

---

deleteAllVisualStyle  

**Delete All Visual Style**

**Description**
Delete all visual styles from current Cytoscape session

**Usage**
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
### deleteAnnotation

**Delete Annotation**

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**

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

```r
deleteDuplicateEdges()
```

---

**deleteGroup**  
*Delete (or Ungroup) a Group*

Description

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

Usage

```r
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

deleGroup()
deleteSelectedEdges  \hspace{1cm} \textit{Delete Selected Edges}

**Description**

Delete the currently selected edges in the network.

**Usage**

\[\text{deleteSelectedEdges}(\text{network} = \text{NULL}, \text{base.url} = \text{.defaultBaseUrl})\]

**Arguments**

- \textit{network} (optional) Name or SUID of the network. Default is the "current" network active in Cytoscape.
- \textit{base.url} (optional) Ignore unless you need to specify a custom domain, port or version to 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**

\[\text{deleteSelectedEdges}()\]

-----

deleteSelectedNodes  \hspace{1cm} \textit{Delete Selected Nodes}

**Description**

Delete currently selected nodes from the network.

**Usage**

\[\text{deleteSelectedNodes}(\text{network} = \text{NULL}, \text{base.url} = \text{.defaultBaseUrl})\]

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

Examples

deleteSelectedNodes()

deleSelfLoops

Delete Self Loops

Description

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

Usage

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

deleteSelfLoops()
deleteStyleMapping  Delete Style Mapping

Description

Deletes a specified visual style mapping from specified style.

Usage

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

deleteStyleMapping()

deleteTableColumn  Delete a table column

Description

Delete a column from node, edge or network tables.

Usage

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")
diffusionAdvanced

**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**

```r
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**

```r
diffusionBasic()
```

---

**disableApp**  
*Disable App*

**Description**

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

**Usage**

```r
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**  
**Dock Panel**

**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**
- requester
- base.url

**Value**
None

**Examples**
doInitializeSandbox()

---

doRequestRemote

**Description**
Do requests remotely by connecting over Jupyter-Bridge.

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

**Arguments**
- method
- qurl
- qbody
- headers

httr headers
doSetSandbox

Value

httr response

Examples

doSandbox()
enableApp  

**Enable App**

**Description**

Enable a previously installed and disabled app in Cytoscape.

**Usage**

```
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**

```
enableApp()
```

---

expandGroup  

**Expand Group**

**Description**

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

**Usage**

```
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**

```r
eexpandGroup()
```

---

**Description**

Saves filters to file in JSON format.

**Usage**

```r
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**

```r
exportFilters()
```
**Description**

Saves the current network view as an image file.

**Usage**

```r
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 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**

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

**Value**

- server response

**Examples**

```r
exportImage("/fullpath/myNetwork","PDF")
```

---

**Description**

Saves the current network view as an jpg file.

**Usage**

```r
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 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

exportJPG('/fullpath/myNetwork')

---

exportNetwork Export Network

Description

Export a network to one of multiple file formats

Usage

```r
eexportNetwork(
  filename = NULL,
  type = "SIF",
  network = NULL,
  base.url = .defaultBaseUrl,
  overwriteFile = TRUE
)
```
exportNetworkToNDEx

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
)
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

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

---

**exportPDF**  
*Export PDF*

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 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**

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

**Description**

Saves the current network view as an png file.
Usage

exportPNG(
    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
exportPS

Examples

exportPNG('/fullpath/myNetwork')

---

exportPS  

Export PS

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.
**Value**

server response

**Examples**

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

---

### `exportSVG`

**Export SVG**

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.
exportVisualStyles

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')

exportVisualStyles  Export Visual Styles

Description

Save one or more visual styles to file.

Usage

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.
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**

`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**

`fitContent()`

---

floatPanel

**Float Panel**

**Description**

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

**Usage**

`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')

g AbsSandboxPath  getAbsSandboxPath

Description
Get absolute sandbox path.

Usage

g AbsSandboxPath(fileLocation)

Arguments

fileLocation  fileLocation

Value
file location

Examples

g AbsSandboxPath()
getAllEdges

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

Usage
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
getAllEdges()

getAllNodes

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

Usage
ggetAllNodes(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

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

Usage
getAllStyleMappings(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
ggetStyleMapping()
**getAnnotationList**

*Get Annotation List*

**Description**

A list of named lists with annotation information

**Usage**

```r
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**

```r
getAnnotationList()
```

---

**getAppInformation**

*Get App Information*

**Description**

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

**Usage**

```r
getAppInformation(app, base.url = .defaultBaseUrl)
```
**getAppStatus**

**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**

```r
getAppInformation()
```

---

**getAppStatus**

**App Status**

**Description**

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

**Usage**

```r
g.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**

```r
g.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**

Get Background Color Default

**Description**

Retrieve the default background color.

**Usage**

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

**Arguments**

- `style.name` (optional) Ignore unless you need to specify a custom style 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**

A list of the character strings, e.g., DIAMOND, T, ARROW

**Examples**

```r
getBackgroundColorDefault()
```

---

**getAvailableApps**

List Available Apps

**Description**

Retrieve a list of apps available for installation in Cytoscape.

**Usage**

```r
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**

```r
getAvailableApps()
```

---

**getArrowShapes**

**Description**

A list of character strings, e.g., DIAMOND, T, ARROW

**Usage**

```r
getArrowShapes()
```
getBrowserClientChannel

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

getBackgroundColorDefault()

---

getBrowserClientChannel

Description

Get the unique channel

Usage

getBrowserClientChannel()

Value

client channel

Examples

getBrowserClientChannel()
**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**

**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**  \hspace{5pt} \textit{Get Collection Name}

**Description**

FUNCTION_DESCRIPTION

**Usage**

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

**Arguments**

- collection.suid

- \textbf{base.url} \hspace{5pt} (optional) Ignore unless you need to specify a custom domain, port or version to 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()
```

---

**getCollectionNetworks**  \hspace{5pt} \textit{Get Collection Networks}

**Description**

FUNCTION_DESCRIPTION

**Usage**

```
getCollectionNetworks(collection.suid = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- \textbf{collection.suid}

- \textbf{base.url} \hspace{5pt} (optional) Ignore unless you need to specify a custom domain, port or version to connect to the CyREST API. Default is http://localhost:1234 and the latest version of the CyREST API supported by this version of RCy3.
**getCollectionSuid**

**Description**

FUNCTION_DESCRIPTION

**Usage**

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

**Arguments**

- **network**
  - 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**

```r
collectionSuid()
```
getDescription

Description
Return both the current sandbox name and path.

Usage
generateDescription()

Value
current sandbox

Examples
generateDescription()

getDescriptionName

Description
Return the current sandbox name.

Usage
generateDescriptionName()

Value
current sandbox name

Examples
generateDescriptionName()
**getCurrentSandboxPath**  

*Description*

Return the current sandbox path.

*Usage*

```r
getCurrentSandboxPath()
```

*Value*

current sandbox path

*Examples*

```r
getCurrentSandboxPath()
```

---

**getCurrentStyle**  

*Description*

Get the current visual style applied to a network.

*Usage*

```r
ggetCurrentStyle(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*

Name of style

*Examples*

```r
ggetCurrentStyle()
ggetCurrentStyle('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**  
```
getDefaultSandboxPath()
```

**Value**  
default sandbox path

**Examples**  
```
getDefaultSandboxPath()
```
getDisabledApps  

**List Disabled Apps**

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

**Usage**
```r
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**
```r
getDisabledApps()
```

getEdgeColor  

**Get Edge Color**

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

**Usage**
```r
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.
**getEdgeCount**

**Value**

Named list of property values

**Examples**

```r
getEdgeColor()
```

---

### Description

Reports the number of the edges in the network.

### Usage

```r
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**

```r
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

ggetEdgeInfo()
**getEdgeLineStyle**  
*Get Edge Line Style*

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

**Usage**

```r
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**

```r
getEdgeLineStyle()
```

**getEdgeLineWidth**  
*Get Edge Line Width*

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

**Usage**

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

**Value**
Named list of property values

**Examples**

```r
getEdgeLineWidth()
```
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

getEdgeLineWidth()
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

```r
getEdgeProperty(c('node 0 (pp) node 1','node 0 (pp) node 2'),'EDGE_WIDTH')
```
### 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)
```
**getFirstNeighbors**

**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**

```r
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.
**getGroupInfo**

**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**

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

**Arguments**

- `group`  
  Group name or 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 information

**Examples**

```r
getGroupInfo('myGroup')
```
**getInstalledApps**  

*List Installed Apps*

**Description**

Retrieve list of currently installed apps in Cytoscape.

**Usage**

```r
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**

```r
getInstalledApps()
```

---

**getJupyterBridgeURL**  

*getJupyterBridgeURL*

**Description**

Get the jupyter bridge server url

**Usage**

```r
getJupyterBridgeURL()
```

**Value**

jupyter bridge server url

**Examples**

```r
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** \( \text{Get Line Styles} \)

**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**

```r
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**

```r
gelineStyles()
```

**getNetworkCenter** \( \text{Get Network Center} \)

**Description**

Retrieve the center of specified network.

**Usage**

```r
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.
Value

Named list of property values

Examples

getNetworkCenter()

getNetworkCount  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 Kolishovski, 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**

```r
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**

```r
getNetworkList()
# 3
```

---

getNetworkName  
*Get the name of a network*

**Description**

Retrieve the title of a network

**Usage**

```r
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

```r
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

```r
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

```r
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
getchainNetworkSuid(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  \hspace{1cm} \textit{Get the SUID of a network view}

\section*{Description}
Retrieve the SUID of a network view

\section*{Usage}
getNetworkViewSuid(network = NULL, base.url = .defaultBaseUrl)

\section*{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.

\section*{Value}
(numeric) Network view suid. The first (presumably only) view associated a network is returned.

\section*{Author(s)}
Alexander Pico

\section*{Examples}
- getNetworkViewSuid()
- getNetworkViewSuid("myNetwork")
  # 90

---

getNetworkZoom  \hspace{1cm} \textit{Get Network Zoom}

\section*{Description}
Retrieve the scale factor of specified network.

\section*{Usage}
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()
```

**Description**

Retrieve the actual fill color of specified nodes.

**Usage**

```r
getNodeColor(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**

```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)
**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**

```r
getNodeHeight()
```

---

**Description**

Retrieve the actual label position of specified nodes.

**Usage**

```r
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
getNodeLabelPosition('Node 1')

cgetNodeLabelPositionDefault

Get Node Label Position Default

Description
Retrieve the default selection node color.

Usage
cgetNodeLabelPositionDefault(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
cgetNodeLabelPositionDefault()

cgetNodePosition

Get Node Position

Description
Retrieve the actual x,y position of specified nodes.

Usage
cgetNodePosition(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()
```

**Description**

Get values for any node property of the specified nodes.

**Usage**

```r
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 `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.
getNodeSelectionColorDefault

Description
Retrieve the default selection node color.

Usage
getNodeSelectionColorDefault(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
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.
getValue

Named list of property values

Examples

giveValue()

---

getNodeType

Get Node Type

Description

Retrieve the actual width of specified nodes.

Usage

giveNodeWidth(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

giveNodeWidth()
**getNotebookIsRunning**

| getNotebookIsRunning | getNotebookIsRunning |

**Description**
getNotebookIsRunning

**Usage**
getNotebookIsRunning()

**Value**
None

**Examples**
getNotebookIsRunning()

---

**getSandboxReinitialize**

| getSandboxReinitialize | 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

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

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

gSelectedNodes()
getStyleDependencies

*Get the values of dependencies in a style*

**Description**

Retrieves style dependency settings.

**Usage**

```r
getStyleDependencies(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
getStyleDependencies("myStyle")
```

---

getStyleMapping

*Get Style Mapping*

**Description**

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

**Usage**

```r
getStyleMapping(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

getStyleMapping()

---

tableColumnNames  Get Table Column Names

Description

Retrieve the names of all columns in a table

Usage

getableColumnNames(
  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**

list of column names

**Examples**

```r
getTableColumnNames()
getTableColumnNames('edge')
getTableColumnNames('network')
```

---

**getDescription**

*Get table column values*

**Description**

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

**Usage**

```r
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
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>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>table</td>
<td>name of table, e.g., node, edge, network; default is &quot;node&quot;</td>
</tr>
<tr>
<td>namespace</td>
<td>namespace of table, e.g., default</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

a named list of column types

Examples

getTableColumnTypes()
ggetTableColumnTypes('edge')
ggetTableColumnTypes('network')
**Description**

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

**Usage**

```r
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**

```r
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()

goingVisualPropertyDefault  Get Visual Property Default

Description
Retrieve the default value for a visual property.

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

Arguments

- **property**: Name of property, e.g., NODE_FILL_COLOR (see `getVisualPropertyNames`)
- **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
getVisualPropertyDefault('NODE_SIZE')
```

---

getVisualPropertyNames

Get Visual Property Names

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**  \hspace{1cm} *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** \hspace{1cm} *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.
**groupAnnotation**

**Value**

A list of names

**Examples**

```r
getVisualStyleNames()
```

---

**Description**

Group annotation from the network view in Cytoscape

**Usage**

```r
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**

```r
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 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`, `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`. 
hidePanel

Value

None

See Also

setNodePropertyBypass, hideSelectedNodes, unhideNodes, unhideAll

Examples

hideNodes()

Description

Hide a panel in the UI of Cytoscape. Other panels will expand into the space.

Usage

hidePanel(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

hidePanel('table')
**hideSelectedEdges**

**Description**

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

**Usage**

```r
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**

```r
hideSelectedEdges()
```
### hideSelectedNodes

**Description**

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

**Usage**

```r
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**

```r
hideSelectedNodes()
```
importFileFromUrl

**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 (e.g., https://www.dropbox.com/sr15azh0xb53smul/GDS112_full.soft?dl=0) GDrive: Use the standard Google Drive ‘Get Link’ feature to create the ‘sourceUrl’ link in the clipboard (e.g., https://drive.google.com/file/d/12sJaKQQbesF10xrbgiNtUcqCQYY1YI3/view?usp=sharing) 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 (e.g., https://onedrive.live.com/embed?cid=C357475E90DD89C4&resid=C357475E90DD89C4 GitHub: Use the GitHub web site to show the file or a link to it, capture the URL in the clipboard (e.g., https://github.com/cytoscape/file-transfer-app/blob/master/test_data/GDS112_full.soft) 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**

```r
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
importNetworkFromFile

Examples

importNetworkFromFile()

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()

importNetworkFromUrl

Import Network From File

Description

Loads a network from specified file

Usage

importNetworkFromUrl(file = NULL, base.url = .defaultBaseUrl)
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

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  

Install App

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  
Invert Edge Selection

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

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)
AlexanderPico, Tanja Muetze, Georgi Kolishovski, Paul Shannon

Examples
invertNodeSelection()
Description

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

Usage

`layoutCopycat(
  sourceNetwork, targetNetwork,
  sourceColumn = "name",
  targetColumn = "name",
  gridUnmapped = TRUE,
  selectUnmapped = TRUE,
  base.url = .defaultBaseUrl
)

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

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**

```r
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**

```r
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

**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)

---

**makeSimpleGraph**  

*Make Simple Graph*

**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()}

Description

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

Usage

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>
</tbody>
</table>
matchArrowColorToEdge

Node Label Transparency  Edge Stroke Selected Paint
Node Label Width        Edge Stroke Unselected Paint
Node Network Image Visible Edge Target Arrow Selected Paint
Node Paint              Edge Target Arrow Shape
Node Selected           Edge Target Arrow Size
Node Selected Paint     Edge Target Arrow Unselected Paint
Node Shape              Edge Tooltip
Node Size               Edge Transparency
Node Tooltip            Edge Unselected Paint
Node Transparency       Edge Visible
Node Visible            Edge Visual Property
Node Width              Edge Width
Node X Location         
Node Y Location         
Node Z Location         

See Also
updateStyleMapping getVisualPropertyNames

Examples
mapVisualProperty('node fill color','score',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
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**

```r
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**

```r
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

```
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

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

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

paletteColorBrewerBlues()

Description

Generate a sequential Blues Brewer palette of a given size

Usage

paletteColorBrewerBlues(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

paletteColorBrewerBlues()
paletteColorBrewerBrBG

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

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

paletteColorBrewerBuGn()

paletteColorBrewerBuPu

*paletteColorBrewerBuPu Sequential*

Description

Generate a sequential BuPu Brewer palette of a given size

Usage

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

paletteColorBrewerBuPu()

paletteColorBrewerDark2

*paletteColorBrewerDark2 Qualitative*

Description

Generate a qualitative Dark2 Brewer palette of a given size

Usage

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

\textit{paletteColorBrewerGnBu} \textit{Sequential}

Description
Generate a sequential GnBu Brewer palette of a given size

Usage
\texttt{paletteColorBrewerGnBu(value.count = 3)}

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

Value
List of palette colors

Examples
\texttt{paletteColorBrewerGnBu()}

---

paletteColorBrewerGreens

\textit{paletteColorBrewerGreens} \textit{Sequential}

Description
Generate a sequential Greens Brewer palette of a given size

Usage
\texttt{paletteColorBrewerGreens(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

paletteColorBrewerGreys()
paletteColorBrewerOranges

**paletteColorBrewerOranges Sequential**

**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

**paletteColorBrewerOrRd Sequential**

**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

paletteColorBrewerOrRd()

paletteColorBrewerPaired

**paletteColorBrewerPaired** Qualitative

Description

Generate a qualitative Paired Brewer palette of a given size

Usage

paletteColorBrewerPaired(value.count = 3)

Arguments

text **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

paletteColorBrewerPaired()

paletteColorBrewerPastel1

**paletteColorBrewerPastel1** Qualitative

Description

Generate a qualitative Pastel1 Brewer palette of a given size

Usage

paletteColorBrewerPastel1(value.count = 3)

Arguments

text **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

Value
List of palette colors

Examples
paletteColorBrewerPastel2()

paletteColorBrewerPiYG

Value
List of palette colors

Examples
paletteColorBrewerPiYG()
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

*paletteColorBrewerPuBu Sequential*

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

**Usage**

```r
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**

```r
paletteColorBrewerPuBu()
```

---

paletteColorBrewerPuBuGn

*paletteColorBrewerPuBuGn Sequential*

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

**Usage**

```r
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
\begin{verbatim}
paletteColorBrewerPuRd()
\end{verbatim}

\begin{verbatim}
paletteColorBrewerPurples
\end{verbatim}

\textit{paletteColorBrewerPurples Sequential}

Description
Generate a sequential Purples Brewer palette of a given size

Usage
\begin{verbatim}
paletteColorBrewerPurples(value.count = 3)
\end{verbatim}

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

Value
List of palette colors

Examples
\begin{verbatim}
paletteColorBrewerPurples()
\end{verbatim}

\begin{verbatim}
paletteColorBrewerRdBu
\end{verbatim}

\textit{paletteColorBrewerRdBu Divergent}

Description
Generate a divergent RdBu Brewer palette of a given size

Usage
\begin{verbatim}
paletteColorBrewerRdBu(value.count = 3)
\end{verbatim}
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 sequential RdPu Brewer palette of a given size

Usage

paletteColorBrewerRdPu(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

paletteColorBrewerRdPu()
paletteColorBrewerRdYlBu

paletteColorBrewerRdYlBu Divergent

Description
Generate a divergent RdYlBu Brewer palette of a given size

Usage
paletteColorBrewerRdYlBu(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
paletteColorBrewerRdYlBu()

paletteColorBrewerReds

paletteColorBrewerReds Sequential

Description
Generate a sequential Reds Brewer palette of a given size

Usage
paletteColorBrewerReds(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
paletteColorBrewerSet2

Examples

paletteColorBrewerSet1

paletteColorBrewerSet1 Qualitative

Description
Generate a qualitative Set1 Brewer palette of a given size

Usage

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

paletteColorBrewerSet1()

paletteColorBrewerSet2

paletteColorBrewerSet2 Qualitative

Description
Generate a qualitative Set2 Brewer palette of a given size

Usage

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)
```
paletteColorBrewerYlGnBu

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()

paletteColorBrewerYlGnBu Sequential

Description

Generate a sequential YlGnBu Brewer palette of a given size

Usage

paletteColorBrewerYlGnBu(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**

*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**

*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

```r
paletteColorBrewerYlOrRd()
```

---

```r
paletteColorRandom
```

**paletteColorRandom Qualitative**

**Description**

Generate a qualitative random color map of a given size

**Usage**

```r
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**

```r
paletteColorRandom()
```

---

**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

RemoveFromGroup(
  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

RemoveFromGroup('myGroup')
removeNodeCustomGraphics

*Remove Node Custom Graphics*

**Description**
Remove the default custom charts, images and gradients.

**Usage**
```r
removeNodeCustomGraphics(
  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**
```r
removeNodeCustomGraphics()
```

renameNetwork

*Rename a network*

**Description**
Sets a new name for this network

**Usage**
```r
renameNetwork(title, network = NULL, base.url = .defaultBaseUrl)
```
renameTableColumn

Arguments

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

Details

Duplicate network names are not allowed

Value

None

Author(s)

Alexander Pico, Julia Gustavsen

Examples

renameNetwork("renamed network")

renameTableColumn

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**

```r
renameTableColumn('exp','log2FC')
```

---

resetDefaultSandbox  
resetDefaultSandbox

**Description**

Reset the entire state of the sandbox system.

**Usage**

```r
resetDefaultSandbox()
```

**Value**

None

**Examples**

```r
resetDefaultSandbox()
```
**rotateLayout**  

**Rotate Layout**

**Description**

Rotate the layout.

**Usage**

```r
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**

```r
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 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()

---

**sandboxGetFrom**

**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**

```r
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

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 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(fileName, sandboxName = NULL, base.url = .defaultBaseUrl)

Usage

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(sourceFile, destFile = NULL, overwrite = TRUE, sandboxName = NULL, base.url = .defaultBaseUrl)

Usage

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.
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
)
Arguments

sandboxName  Name of new default sandbox. None means to use the original default sandbox
copySamples  True to copy the Cytoscape sampleData into the sandbox
reinitialize  True to delete sandbox contents (if any) if sandbox already exists
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

sandbox path in Cytoscape workstation’s file system

Examples

sandboxSet()

sandboxUrlTo(sourceURL, destFile, overwrite = TRUE, sandboxName = NULL, base.url = .defaultBaseUrl)

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

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".
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.
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.

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.

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 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**

```r
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**

```r
selectAllNodes()
```

---

### selectAllEdges

**Select all edges**

**Description**

Selects all edges in a Cytoscape Network

**Usage**

```r
selectAllEdges(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 edges in a specified network.
selectAllNodes

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

cw <- CytoscapeWindow('new.demo', new('graphNEL'))
selectAllEdges(cw)

selectAllNodes

Select all nodes

Description

Selects all nodes in a Cytoscape Network

Usage

selectAllNodes(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 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
Examples

```r
selectEdges()
```

---

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
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
selectEdgesConnectingSelectedNodes(network = NULL, base.url = .defaultBaseUrl)
```
selectFirstNeighbors

Select first neighbor nodes

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

- **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

Author(s)

Alexander Pico, Julia Gustavsen

Examples

```r
selectEdgesConnectingSelectedNodes()
```
selectNodes

Value
list of suids of selected nodes, including original selection

Examples
selectFirstNeighbors()
selectFirstNeighbors('outgoing')
selectFirstNeighbors('incoming')

selectNodes Select Nodes

Description
Select nodes in the network by SUID, name or other column values.

Usage
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**
```
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**
```
setCurrentNetwork('MyNetwork')
```

setCurrentSandbox  

**setCurrentSandbox**

**Description**
Set and return the current sandbox name and path.

**Usage**
```
setCurrentSandbox(sandboxName, sandboxPath)
```

**Arguments**
- `sandboxName`  
- `sandboxPath`  

**Value**
current sandbox
**Examples**

```
setCurrentSandbox()
```

---

**setCurrentView**  
*Set Current View*

**Description**

Set which network view is "current".

**Usage**

```
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**

```
setCurrentView()
```
setDefaultSandbox

Description
Set and return the sandbox properties to be used as a default, probably based on whether running remote.

Usage
setDefaultSandbox(newSandbox = NULL, ...)

Arguments
newSandbox newSandbox
... ... 

Value
default sandbox

Examples
setDefaultSandbox()

setDefaultSandboxPath

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

Usage
setDefaultSandboxPath(newPath)

Arguments
newPath new path of default sandbox

Value
default sandbox path

Examples
setDefaultSandboxPath()
setEdgeColorBypass  Set Edge Color Bypass

Description
Override the color for particular edges.

Usage
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
setEdgeColorBypass()
### SetEdgeColorDefault

**Description**

Set the default edge color.

**Usage**

```r
setEdgeColorDefault(new.color, style.name = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- `new.color`: Color as hex code, e.g., 
- `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
setEdgeColorDefault('#FD5903')
```

### SetEdgeColorMapping

**Description**

Map table column values to colors to set the edge color.

**Usage**

```r
setEdgeColorMapping( 
  table.column,
  table.column.values = NULL,
  colors = NULL,
  mapping.type = "c",
  default.color = NULL,
  style.name = NULL,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
**setEdgeFontFaceBypass**

**Description**

Override the font face for particular edges.

**Usage**

```r
setEdgeFontFaceBypass(
  edge.names,
  new.fonts,
  network = NULL,
  base.url = .defaultBaseUrl
)
```
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()
```

```r
setEdgeFontFaceDefault()
```

Description

Set the default edge font.

Usage

```r
setEdgeFontFaceDefault(new.font, style.name = NULL, base.url = .defaultBaseUrl)
```
setEdgeFontFaceMapping

Sets font face for edge labels.

Usage

```r
setEdgeFontFaceMapping(
  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
**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, \texttt{setEdgePropertyBypass}, which can be used to set any visual property. To restore defaults and mappings, use \texttt{clearEdgePropertyBypass}.

Value

None

See Also

\texttt{setEdgePropertyBypass, clearEdgePropertyBypass}

Examples

\begin{verbatim}
setEdgeFontSizeBypass()
\end{verbatim}

\begin{verbatim}
setEdgeFontSizeDefault(12)
\end{verbatim}

---

\textit{Set Edge Font Size Default}

Description

Set the default edge font size.

Usage

\begin{verbatim}
setEdgeFontSizeDefault(new.size, style.name = NULL, base.url = .defaultBaseUrl)
\end{verbatim}

Arguments

\begin{itemize}
  \item \texttt{new.size} \hspace{1cm} Numeric value for size
  \item \texttt{style.name} \hspace{1cm} Name of style; default is "default" style
  \item \texttt{base.url} \hspace{1cm} (optional) Ignore unless you need to specify a custom domain, port or version to 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

\begin{verbatim}
setEdgeFontSizeDefault(12)
\end{verbatim}
setEdgeFontSizeMapping

Set Edge Font Size Mapping

Description

Map table column values to sizes to set the edge size.

Usage

```r
setEdgeFontSizeMapping(
  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
setEdgeLabelBypass

**Examples**

```r
setEdgeFontSizeMapping('score')
setEdgeFontSizeMapping('score', sizes=c(6,24))
setEdgeFontSizeMapping('score', c(0,30), c(35,55))
```

**Description**

Override the label for particular edges.

**Usage**

```r
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**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>edge.names</code></td>
<td>List of edge names or SUIDs</td>
</tr>
<tr>
<td><code>new.colors</code></td>
<td>List of hex colors, or single value</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>

**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

setEdgeLabelColorDefault("#FD5903")
setEdgeLabelColorMapping

Set Edge Label Color Mapping

Description

Map table column values to colors to set the edge border color.

Usage

```
setEdgeLabelColorMapping(
  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

setEdgeLabelColorMapping('score', c(0.5), c('#FFFFFF', '#FF7755'))
setEdgeLabelColorMapping('score', colors=paletteColorBrewerRdBu)
setEdgeLabelColorMapping('score', colors=paletteColorBrewerSet3, mapping.type='d')

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')

Description

Pass the values from a table column to display as edge labels.
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
)
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
)
```
setEdgeLabelOpacityMapping

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

setEdgeLabelOpacityDefault(50)

---

setEdgeLabelOpacityMapping

*Set Edge Label Opacity Mapping*

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
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

setEdgeLabelOpacityMapping('weight')
setEdgeLabelOpacityMapping('weight', opacities=c(0,100))
setEdgeLabelOpacityMapping('weight', c(1,10), c(50,255))

setEdgeLineStyleBypass

_set Edge Line Style Bypass_

Description

Override the style for particular edges.

Usage

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.
**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()
```

---

**setEdgeLineStyleDefault**

*Set Edge Line Style Default*

**Description**

Set the default edge style.

**Usage**

```
setEdgeLineStyleDefault(
  new.line.style,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>new.line.style</code></td>
<td>Name of line style, e.g., SOLID, LONG_DASH, etc (see <code>getLineStyles</code>)</td>
</tr>
<tr>
<td><code>style.name</code></td>
<td>Name of style; default is &quot;default&quot; style</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
setEdgeLineStyleMapping

Set Edge Line Style Mapping

Description

Map table column values to styles to set the edge style.

Usage

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
setEdgeLineWidthBypass

setEdgeLineWidthBypass

Set Edge Line Width Bypass

Description

Override the width for particular edges.

Usage

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, setEdgePropertyBypass, which can be used to set any visual property. To restore defaults and mappings, use clearEdgePropertyBypass.

Value

None

See Also

setEdgePropertyBypass, clearEdgePropertyBypass

Examples

setEdgeLineStyleMapping('type')
setEdgeLineStyleMapping('type', c('pp', 'pd'), c('SOLID', 'LONG_DASH'))
**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**  

**Set Edge Opacity Default**

**Description**

Set default opacity value for all unmapped edges.

**Usage**

```r
defaultOpacityDefault(
    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**

```r
code
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

```r
setEdgeOpacityMapping('weight')
setEdgeOpacityMapping('weight', opacities=c(0,100))
setEdgeOpacityMapping('weight', c(1,10), 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

```r
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
See Also

clearEdgePropertyBypass

Examples

setEdgePropertyBypass()

---

setEdgeSelectionColorDefault

*Set Edge Selection Color Default*

Description

Set the default selected edge color.

Usage

```r
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

```r
setEdgeSelectionColorDefault('#FD5903')
```
setEdgeSourceArrowColorBypass

**Set Edge Source Arrow Color Bypass**

**Description**

Override the source arrow color for particular edges.

**Usage**

```r
setEdgeSourceArrowColorBypass(
  edge.names,  # List of edge names or SUIDs
  new.colors,  # List of hex colors, 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**

- `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()
```
**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**

*Set Edge Source Arrow Color Mapping*

**Description**

Map table column values to colors to set the source arrow color.
**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., `paletteColorBrewerRdBu` (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'))
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

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

setEdgeSourceArrowShapeDefault('ARROW')

setEdgeSourceArrowShapeMapping

Set Edge Source Arrow Shape Mapping

Description

Map table column values to shapes to set the source arrow shape.
Usage

```r
setEdgeTargetArrowColorBypass()
```

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
setEdgeSourceArrowShapeMapping('type')
setEdgeSourceArrowShapeMapping('type', c('activation', 'inhibition'), c('ARROW', 'T'))
```

Description

Override the target arrow color for particular edges.
Usage

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

setEdgeTargetArrowColorBypass()

---

setEdgeTargetArrowColorDefault

Set Edge Target Arrow Color Default

Description

Set the default edge target arrow color.
**setEdgeTargetArrowColorDefault**

Usage

```r
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

```r
setEdgeTargetArrowColorDefault('#FD5903')
```

**setEdgeTargetArrowColorMapping**

*Set Edge Target Arrow Color Mapping*

Description

Map table column values to colors to set the target arrow color.

Usage

```r
setEdgeTargetArrowColorMapping(
    table.column,
    table.column.values = NULL,
    colors = NULL,
    mapping.type = "c",
    default.color = NULL,
    style.name = NULL,
    network = NULL,
    base.url = .defaultBaseUrl
)
```

**setEdgeTargetArrowColorMapping**

*Set Edge Target Arrow Color Mapping*
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
setEdgeTargetArrowColorMapping('score', c(0,5), c('#FFFFFF', '#FF7755'))
setEdgeTargetArrowColorMapping('score', colors=paletteColorBrewerRdBu)
setEdgeTargetArrowColorMapping('score', colors=paletteColorBrewerSet3, mapping.type='d')
```

---

**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>
<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>table.column.values</code></td>
<td>List of values from Cytoscape table to be used in mapping. Leave NULL to perform an automatic mapping to all column values.</td>
</tr>
<tr>
<td><code>shapes</code></td>
<td>List of shapes to map to <code>table.column.values</code>. Leave NULL to perform an automatic mapping to available shapes. See <code>getArrowShapes</code>.</td>
</tr>
<tr>
<td><code>default.shape</code></td>
<td>Shape to set as default. See <code>getArrowShapes</code>.</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
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

Arguments

- **edge.names**: List of edge names or SUIDs
- **new.shapes**: List of values to set, 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
setEdgeTargetArrowShapeBypass()
```

---

**setEdgeTargetArrowShapeDefault**

*Set Edge Target Arrow Shape Default*

Description

Set the default edge target arrow shape.

Usage

```r
setEdgeTargetArrowShapeDefault(
  new.shape,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```
setEdgeTargetArrowShapeMapping

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

```
setEdgeTargetArrowShapeDefault('ARROW')
```

Description

Map table column values to shapes to set the target arrow shape.

Usage

```
setEdgeTargetArrowShapeMapping(
  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`
**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.

**Examples**

```r
setEdgeTargetArrowShapeMapping('type')
setEdgeTargetArrowShapeMapping('type', c('activation', 'inhibition'), c('ARROW', 'T'))
```
**setEdgeTooltipDefault**  

*Set Edge Tooltip Default*

**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()
```

---

**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
setEdgeTooltipDefault('unknown')
```

---

**setEdgeTooltipMapping  Set Edge Tooltip Mapping**

---

**Description**

Pass the values from a table column to display as edge tooltips.

**Usage**

```r
setEdgeTooltipMapping(
  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
setEdgeTooltipMapping('description')
```
**setLayoutProperties**

### Description
Sets the specified properties for the specified layout. Unmentioned properties are left unchanged.

### Usage
```
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
```
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

```r
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

```r
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`
**setNetworkPropertyBypass**

*Set Network Property Bypass*

**Examples**

```r
setNetworkCenterBypass()
```

**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**

```r
setNetworkPropertyBypass()
```

---

**setNetworkZoomBypass**  *Set Network Zoom Bypass*

**Description**

Set the bypass value for scale factor for the network.

**Usage**

```r
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**

```r
setNetworkZoomBypass()
```
**Description**

Override the border color for particular nodes.

**Usage**

```r
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 to 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
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
setNodeBorderColorDefault(
  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
setNodeBorderColorDefault('#FD5903')
```

**setNodeBorderColorMapping**

*Set Node Border Color Mapping*

**Description**

Map table column values to colors to set the node border color.
Usage

```
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

```
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

```r
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

```r
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.
Usage

```r
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

```r
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')
```
setNodeBorderWidthDefault

*Set Node Border Width Default*

**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)
```

---

setNodeBorderWidthMapping

*Set Node Border Width Mapping*

**Description**

Map table column values to widths to set the node border width.
Usage

```r
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

```r
setNodeBorderWidthMapping('score')
setNodeBorderWidthMapping('score', widths=c(1,10))
setNodeBorderWidthMapping('score', c(0,30), c(1,5))
```
**setNodeColorBypass**

**Set Node Color Bypass**

**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
)
```
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 = 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

setNodeComboOpacityMapping('score')
setNodeComboOpacityMapping('score', opacities=c(0,100))
setNodeComboOpacityMapping('score', c(-5,5), c(50,255))
**setNodeCustomBarChart**

```r
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 is 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**

```r
setNodeCustomBarChart(c("data1","data2","data3"))
```
**setNodeCustomBoxChart**  
*Set Node Custom Box Chart*

**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

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

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
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

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

setNodeCustomLineChart(
columns,
colors = NULL,
range = NULL,
lineWidth = 1,
rangeAxis = FALSE,
zeroLine = FALSE,
axisWidth = 0.25,
axisColor = "#000000",
axisFontSize = 1,
slot = 1,
style.name = NULL,
baseUrl = .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
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.
setNodeCustomRadialGradient

Usage

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

setNodeCustomPosition()

Description

Makes a gradient fill per node by setting a default custom graphic style.
Usage

```
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

```
setNodeCustomRadialGradient()
```

Description

Makes a ring chart per node using specified node table columns by setting a default custom graphic style.
setNodeFillOpacityBypass

Set Node Fill Opacity Bypass

Description

Override the fill opacity for particular nodes.

Usage

```r
setNodeCustomRingChart(
  columns,  # List of node column names to be displayed.
  colors = NULL,  # (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 = 0,  # (optional) Angle to start filling ring. Default is 0.0.
  holeSize = 0.5,  # (optional) Size of hole in ring. Ranges 0-1. Default is 0.5.
  slot = 1,  # (optional) Which custom graphics slot to modify. Slots 1-9 are available for independent charts, gradients and images. Default is 1.
  style.name = NULL,  # (optional) Name of style; default is "default" style
  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

- `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

```r
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**
```
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**
```
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

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

setNodeFontFaceBypass()
**setNodeFontFaceDefault**

*Set Node Font Face Default*

**Description**

Set the default node font.

**Usage**

```r
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
setNodeFontFaceDefault("Dialog,plain,10")
```

---

**setNodeFontFaceMapping**

*Set Node Font Face Mapping*

**Description**

Sets font face for node labels.
setNodeFontFaceMapping

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  

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**

```
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**

```
setNodeFontSizeDefault(12)
```

setNodeFontSizeMapping

*Set Node Font Size Mapping*

**Description**

Map table column values to sizes to set the node size.
setNodeFontSizeMapping

Usage

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

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
)
```
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))
```

Description

Override the label for particular nodes.

Usage

```r
setNodeLabelBypass(
  node.names,  # Node names to change labels for.
  new.labels,  # New labels for the nodes.
  network = NULL,  # Network name or SUID (optional).
  base.url = .defaultBaseUrl  # Base URL for CyREST API connection.
)
```
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

```r
setNodeLabelBypass('Node 1', 'Custom Label')
setNodeLabelBypass(c('Node 1', 'Node 2'), 'Custom Label')
clearNodePropertyBypass(c('Node 1', 'Node 2'), 'NODE_LABEL')
```
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
setNodeLabelColorBypass('Node 1', '#FF55AA')
setNodeLabelColorBypass(c('Node 1', 'Node 2'), '#FF55AA')
clearNodePropertyBypass(c('Node 1', 'Node 2'), 'NODE_LABEL_COLOR')
```

Description

Set the default node label color.

Usage

```r
setNodeLabelColorDefault(
  new.color,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```
setNodeLabelColorMapping

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
setNodeLabelDefault(new.label, style.name = NULL, base.url = .defaultBaseUrl)
```

**Arguments**

- **new.label**: String label 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
setNodeLabelDefault('unknown')
```
setNodeLabelMapping  

Set Node Label Mapping

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  

Set Node Label Opacity Bypass

Description

Override the label opacity for particular nodes.
setNodeLabelOpacityDefault

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()
```

Description

Set default opacity value for all unmapped node labels.
setNodeLabelOpacityMapping

Usage

```
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

```
setNodeLabelOpacityDefault(50)
```

---

setNodeLabelOpacityMapping

*Set Node Label Opacity Mapping*

Description

Sets opacity for node label only.

Usage

```
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

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

setNodeLabelPositionBypass(
  node.names,
  new.positions,
  network = NULL,
  base.url = .defaultBaseUrl
)
setNodeLabelPositionDefault

Description

Set the default node label position

Usage

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,  # List of node names or SUIDs
  new.values,  # List of values to set, 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.
)
```
setNodePositionBypass

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,  # List of node names or SUIDs
  new.x.locations = NULL,  # List of x position values, or single value, default is current x position
  new.y.locations = NULL,  # List of y position values, or single value, default is current y position
  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

**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)
```

---

**setNodePropertyBypass  Set Node Property Bypass**

**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,  # Names or SUIDs of the nodes
  new.values,  # New values for the properties
  visual.property,  # Visual property to set
  bypass = TRUE,  # Set bypass for the property
  network = NULL,  # Network to use
  base.url = .defaultBaseUrl
)
```
**setNodeSelectionColorDefault**

*Set Node Selection Color Default*

**Description**

Set the default selection node color.

**Usage**

```r
setNodeSelectionColorDefault(
  new.color,
  style.name = NULL,
  base.url = .defaultBaseUrl
)
```
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
graphnelSelectionColorDefault('#FD5903')
```

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\')

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**
```
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**
```
setNodeShapeMapping('type')
setNodeShapeMapping('type',c('protein','dna'),c('ELLIPSE','RECTANGLE'))
```
setNodeSizeBypass  Set Node Size Bypass

Description

Sets the bypass value of node size for one or more nodes. Only applicable if node dimensions are locked. See lockNodeDimensions.

Usage

setNodeSizeBypass(
  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

setNodeSizeBypass('Node 1', 35)
setNodeSizeBypass(c('Node 1','Node 2'), 35)
clearNodePropertyBypass(c('Node 1','Node 2'), 'NODE_SIZE')
setNodeSizeDefault  

Set Node Size Default

Description
Set the default node size.

Usage
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
setNodeSizeDefault(35)

setNodeSizeMapping  

Set Node Size Mapping

Description
Map table column values to node font sizes.

Usage
setNodeSizeMapping(
  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.

**Details**

Using this function will lock node width and height to use a singular "size" value.

**Value**

None

**Examples**

```r
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**

```r
setNodeTooltipBypass(
  node.names,  # Names of nodes to set tooltips for
  new.tooltip, # New tooltip text
  network = NULL,  # Name or SUID of the network
  base.url = .defaultBaseUrl  # Base URL for CyREST API connection
)
```
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

setNodeTooltipBypass('Node 1', 'This is an important node.')

---

setNodeTooltipDefault  Set Node Tooltip Default

Description

Set the default node tooltip

Usage

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

```
setNodeTooltipDefault('unknown')
```

---

**Description**

Pass the values from a table column to display as node tooltips.

**Usage**

```
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
setNodeWidthBypass

Examples

setNodeTooltipMapping('description')

Description

Override the width for particular nodes.

Usage

setNodeWidthBypass(
  node.names,
  new.widths,
  network = NULL,
  base.url = .defaultBaseUrl
)

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
**setNodeWidthDefault**  
*Set Node Width Default*

**Description**
Set the default node width.

**Usage**
```r
setNodeWidthDefault(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
setNodeWidthDefault(35)
```

---

**setNodeWidthMapping**  
*Set Node Width Mapping*

**Description**
Map table column values to the node widths.
Usage

```r
setNodeWidthMapping(
  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 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 

@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
setNotebookIsRunning(newState = NULL)

Arguments
newState new state of running remote

Value
oldState

Examples
setNotebookIsRunning()

setSandboxReinitialize

Description
Set and return flag indicating that next command should reinitialize the sandbox according to the default_sandbox.

Usage
setSandboxReinitialize(doReinitialize = TRUE)

Arguments
doReinitialize default is TRUE

Value
sandbox reinitialize

Examples
setCurrentSandbox()
**setDescription**  

*setDescription*

**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
default.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
default.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
- **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, # (Boolean) Whether to sync node custom graphics size
  style.name = NULL,  # Name of style; default is "default" style
  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.
)
```
Value

None

Examples

syncNodeCustomGraphicsSize(TRUE)

toggleGraphicsDetails  Toggle Graphics Details

Description

Regardless of the current zoom level and network size, show (or hide) graphics details, e.g., node labels.

Usage

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

showGraphicsDetails(TRUE)
ungroupAnnotation  

**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")
ungroupAnnotation(c("316869a4-39fc-4731-8f45-199dec9af10d","c3621eb4-4687-490f-9396-b829dd8767d5"))
ungroupAnnotation("Group 1")
ungroupAnnotation(c("Group1","Group2", "Group3"))
```
unhideAll  

Unhide All

Description

Unhide all previously hidden nodes and edges, by clearing the Visible property bypass value.

Usage

unhideAll(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>

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

Unhide Nodes

Description

Unhide specified nodes that were previously hidden, by clearing the Node Visible property bypass value.

Usage

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

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
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
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)

updateAnnotationImage  Update Image Annotation

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

Usage
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**

```r
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
updateAnnotationShape

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>borderColor</td>
<td>(optional) Hexadecimal color; default is #000000 (black)</td>
</tr>
<tr>
<td>borderOpacity</td>
<td>(optional) Integer between 0 and 100; default is 100.</td>
</tr>
<tr>
<td>height</td>
<td>(optional) Height of shape; default is based on text height.</td>
</tr>
<tr>
<td>width</td>
<td>(optional) Width of shape; default is based on text length.</td>
</tr>
<tr>
<td>name</td>
<td>(optional) Name of annotation object; default is &quot;Shape&quot;</td>
</tr>
<tr>
<td>canvas</td>
<td>(optional) Canvas to display annotation, i.e., foreground (default) or background</td>
</tr>
<tr>
<td>z.order</td>
<td>(optional) Arrangement order specified by number (larger values are in front of smaller values); default is 0</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

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,
  base.url = NULL,
  network = NULL,
  z.order = NULL,
  canvas = NULL,
  height = NULL,
  width = NULL,
  borderColor = NULL,
  borderOpacity = NULL,
  name = "Shape")

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**  
*Update App*

**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**  
*Update Group Annotation*

**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 UUID

**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

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
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to 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 the updated submission

Examples

```
updateNetworkInNDEx("user", "pass", TRUE)
```

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

- **style.name**: (char) name for style
- **defaults**: (list) a list of visual property default settings
- **base.url**: (optional) Ignore unless you need to specify a custom domain, port or version to 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
updateStyleMapping

See Also

mapVisualProperty

Examples

updateStyleDefaults('myStyle', list('node fill color' = '#0000FF', 'node size' = 50))

updateStyleMapping(Usage)

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

.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
INDEX

getNodeLabelPositionDefault, 136
getNodePosition, 136
getNodeProperty, 137
getNodeSelectionColorDefault, 138
getNodeShapes, 139, 322–324
getNodeSize, 139
getNodeWidth, 140
getNotebookIsRunning, 141
getSandboxReinitialize, 141
getSelectedEdgeCount, 142
getSelectedEdges, 142
getSelectedNodeCount, 143
getSelectedNodes, 144
getStyleDependencies, 145
getStyleMapping, 145
getTableColumnNames, 146
getTableColumns, 147
getTableColumnTypes, 148
getTableValue, 149
getUninstalledApps, 150
getVisualStyleJSON, 152
getVisualStyleNames, 152

hideAllPanels, 154
hideEdges, 154, 157
hideNodes, 155, 158
hidePanel, 156
hideSelectedEdges, 155, 157
hideSelectedNodes, 156, 158

importFileFromUrl, 159
importFilters, 160
importNetworkFromFile, 160
importNetworkFromNDEx, 161
importVisualStyle, 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
paletteColorBrewerPiYG, 187
paletteColorBrewerPRGn, 188
paletteColorBrewerPuBu, 189
paletteColorBrewerPuBuGn, 189
paletteColorBrewerPuOr, 190
paletteColorBrewerPuRd, 190
paletteColorBrewerPurples, 191
paletteColorBrewerRdBu, 191
paletteColorBrewerRdPb, 192
paletteColorBrewerRdYlBu, 193
paletteColorBrewerReds, 193
paletteColorBrewerSet1, 194
paletteColorBrewerSet2, 194
paletteColorBrewerSet3, 195
paletteColorBrewerSpectral, 195
paletteColorBrewerYlGnBu, 196
paletteColorBrewerYlGnBu, 196
paletteColorBrewerYlOrBr, 197
paletteColorBrewerYlOrRd, 197
paletteColorBrewerPaletteBrewerAccent, 199

RCy3, 198
RemoveFromGroup, 199
removeNodeCustomGraphics, 200
INDEX

setNodeBorderWidthDefault, 281
setNodeBorderWidthMapping, 281
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
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
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 (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
updateNetworkInNDEx, 350
updateStyleDefaults, 351
updateStyleMapping, 352