

# Package ‘mygene’

February 14, 2019

**Type** Package

**Title** Access MyGene.Info\_ services

**Version** 1.18.0

**Date** 2014-04-18

**Author** Adam Mark, Ryan Thompson, Cyrus Afrasiabi, Chunlei Wu

**Maintainer** Adam Mark, Cyrus Afrasiabi, Chunlei Wu <cwu@scripps.edu>

**Description** MyGene.Info\_ provides simple-to-use REST web services to query/retrieve gene annotation data. It's designed with simplicity and performance emphasized. \*mygene\*, is an easy-to-use R wrapper to access MyGene.Info\_ services.

**License** Artistic-2.0

**Depends** R (>= 3.2.1), GenomicFeatures,

**Imports** httr (>= 0.3), jsonlite (>= 0.9.7), S4Vectors, Hmisc, sqldf, plyr

**Suggests** BiocStyle

**biocViews** Annotation

**git\_url** <https://git.bioconductor.org/packages/mygene>

**git\_branch** RELEASE\_3\_8

**git\_last\_commit** 247454a

**git\_last\_commit\_date** 2018-10-30

**Date/Publication** 2019-02-13

## R topics documented:

getGene . . . . .	2
getGenes . . . . .	3
makeTxDbFromMyGene . . . . .	4
metadata . . . . .	5
MyGene . . . . .	5
mygene . . . . .	6
MyGene-class . . . . .	6
query . . . . .	7
queryMany . . . . .	8

<b>Index</b>	<b>10</b>
--------------	-----------

---

getGene	<i>Return the gene object for the given geneid.</i>
---------	---

---

### Description

This is a wrapper for GET query of "/gene/<geneid>" service.

### Usage

```
getGene(geneid, fields = c("symbol", "name", "taxid", "entrezgene"),
        ..., return.as=c("records", "text"), mygene)
```

### Arguments

geneid	Entrez/ensembl gene id
fields	Fields to return, a list of a comma-sep string. If fields=="all", all available fields are returned.
...	Includes species as well as several other fields. View available fields by calling ?metadata. Also, see <a href="http://docs.mygene.info/en/latest/doc/annotation_service.html">http://docs.mygene.info/en/latest/doc/annotation_service.html</a> for complete argument details and syntax.
return.as	"records" (list), "text" (JSON).
mygene	A MyGene object that describes how to connect to data resources. See <a href="#">MyGene-class</a> . If missing, default object will be used that accesses the main MyGene.info portal. Default is recommended.

### Value

returns a gene object containing the queried annotations

### References

Ref: [http://docs.mygene.info/en/latest/doc/annotation\\_service.html](http://docs.mygene.info/en/latest/doc/annotation_service.html)

### See Also

[getGenes query queryMany](#)

### Examples

```
## return the gene object for the given gene id
getGene(1017)

## customize fields
getGene(1017, fields=c("name", "symbol", "refseq"), return.as="text")

## all fields
getGene(1017, fields="all")
```

---

getGenes	<i>Return the list of gene objects for the given list of geneids.</i>
----------	---

---

### Description

This is a wrapper for POST query of "/gene" service.

### Usage

```
getGenes(geneids, fields = c("symbol", "name", "taxid", "entrezgene"), ...,
         return.as=c("DataFrame", "records", "text"), mygene)
```

### Arguments

geneids	A vector, list, or comm-sep string entrez/ensembl gene ids
fields	A vector of fields to return. If fields=="all", all available fields are returned.
...	Includes species as well as several other fields. View available fields by calling ?metadata. Also, see <a href="http://docs.mygene.info/en/latest/doc/annotation_service.html">http://docs.mygene.info/en/latest/doc/annotation_service.html</a> for complete argument details and syntax.
return.as	"DataFrame" (default), "records" (list), "text" (JSON).
mygene	A MyGene object that describes how to connect to data resources. See <a href="#">MyGene-class</a> . If missing, default object will be used that accesses the main MyGene.info portal. Default is recommended.

### Value

returns a gene object (DataFrame, list, or JSON text) containing the queried annotations

### References

Ref: [http://docs.mygene.info/en/latest/doc/annotation\\_service.html](http://docs.mygene.info/en/latest/doc/annotation_service.html)

### See Also

[getGene query queryMany](#)

### Examples

```
## Return the list of gene object for the given list of gene ids.
getGenes(c(1017,1018))

## mix types of gene ids
getGenes(c(1017,1018,"ENSG00000148795"))
```

---

makeTxDbFromMyGene     *makeTxDbFromMyGene*

---

### Description

makeTxDbFromMyGene allows the user to make a TxDb object in the Genomic Features package from a mygene "exons" query using a default MyGene object.

### Usage

```
makeTxDbFromMyGene(gene.list, scopes, species, returnall=FALSE)
```

### Arguments

gene.list	A list, vector, or comma-separated string of query terms.
scopes	Type of types of identifiers, either a list, vector, or a comma-separated fields to specify type of input qterms, e.g. c("reporter", "ensembl.gene", "symbol") refer to "http://docs.mygene.info/en/latest/doc/data.html#available-fields" for full list of fields.
species	Names or taxonomy ids
returnall	Logical, if TRUE, return list of genes without exons annotations. False by Default.

### Details

makeTxDbFromMyGene invokes either the query or queryMany method and passes the response to construct a TxDb object. See ?TxDb for utilizing transcript annotations.

### Value

returns TxDb object

### See Also

[makeTxDb](#)

### Examples

```
xli <- c('DDX26B', 'CCDC83', 'MAST3', 'RPL11')
txdb <- makeTxDbFromMyGene(xli, scopes="symbol", species="human")
```

---

metadata	<i>metadata</i>
----------	-----------------

---

**Description**

Get metadata for MyGene.info services.

**Usage**

```
metadata(x, ...)
```

**Arguments**

x	MyGene object
...	MyGene object slot parameters

**Value**

returns the metadata including available fields, genome assemblies, sources, statistics, taxonomy, and timestamp

**References**

<http://mygene.info/v2/metadata>

**Examples**

```
## Get metadata
mygene<-MyGene()
metadata(mygene)

## get available fields to return
metadata(mygene)$available_fields
```

---

MyGene	<i>MyGene</i>
--------	---------------

---

**Description**

Construct a MyGene object.

**Usage**

```
MyGene(...)
```

**Arguments**

...	See help page for MyGene-class
-----	--------------------------------

**Value**

MyGene object

**Examples**

```
MyGene()
```

---

mygene	<i>Access MyGene.info annotation services</i>
--------	---

---

**Description**

MyGene.Info\_ provides simple-to-use REST web services to query/retrieve gene annotation data. It's designed with simplicity and performance emphasized. \*mygene\* is an easy-to-use R wrapper to access MyGene.info services.

**Details**

Package:	mygene
Type:	Package
Version:	0.99.0
Date:	2014-04-18
License:	BSD
Depends:	httr jsonlite Hmisc

**Author(s)**

Adam Mark, Chunlei Wu

Maintainer: Chunlei Wu <help@mygene.info>

**References**

Wu C, MacLeod I, Su AI (2013) BioGPS and MyGene.info: organizing online, gene-centric information. Nucl. Acids Res. 41(D1): D561-D565. [http://mygene.info/doc/annotation\\_service.html](http://mygene.info/doc/annotation_service.html)  
[http://mygene.info/doc/query\\_service.html](http://mygene.info/doc/query_service.html)

---

MyGene-class	<i>Class "MyGene"</i>
--------------	-----------------------

---

**Description**

R Client to access MyGene.Info annotation services

**Objects from the Class**

Objects can be created by calls of the form `MyGene(base.url="http://mygene.info/v2", delay=1, step=1000`

**Slots**

**base.url:** "http://mygene.info/v2". Object of class "character"  
**delay:** Sleep time between batch retrieval. Object of class "numeric"  
**step:** Batch limit. Object of class "numeric"  
**version:** httr package version. Object of class "character"  
**verbose:** Object of class "logical"  
**debug:** Object of class "logical"

**Methods**

**getGene(geneid, fields = c("symbol", "name", "taxid", "entrezgene"), ..., return.as=c("records", ...))**  
 Return the gene object for the given geneid  
**getGenes(geneids, fields = c("symbol", "name", "taxid", "entrezgene"), ..., return.as=c("DataFrame", ...))**  
 Return the list of gene object for the given list of geneids.  
**query(q, fields=c("name", "symbol", "taxid", "entrezgene"), ..., return.as=c("DataFrame", "records", ...))**  
 Return the query result.  
**queryMany(qterms, scopes=NULL, fields=c("name", "symbol", "taxid", "entrezgene"), ..., return.as=c("records", ...))**  
 Return the batch query result.  
**metadata(x, ...):** Get metadata for MyGene.info services.  
**makeTxDbFromMyGene(gene.list, scopes, species, returnall=FALSE):** Make a TxDb object from transcript annotations

**Author(s)**

Adam Mark, Chunlei Wu, Ryan Thompson

**References**

Wu C, MacLeod I, Su AI (2013) BioGPS and MyGene.info: organizing online, gene-centric information. *Nucl. Acids Res.* 41(D1): D561-D565.

**Examples**

```
showClass("MyGene")
```

---

query

*Return the query result.*

---

**Description**

This is a wrapper for GET query of "/query?q=<query>" service.

**Usage**

```
query(q, ..., return.as=c("DataFrame", "records", "text"), mygene)
```

**Arguments**

q	query term(s), see query syntax at <a href="http://mygene.info/doc/query_service.html#query-syntax">http://mygene.info/doc/query_service.html#query-syntax</a>
...	Commonly queried fields include <code>species</code> , <code>fields</code> , <code>size</code> as well as several other fields. View available fields by calling <code>?metadata</code> . Also, see <a href="http://docs.mygene.info/en/latest/">http://docs.mygene.info/en/latest/</a> for complete argument details and syntax.
return.as	"DataFrame" (default), "records" (list), or "text" (JSON).
mygene	A MyGene object that describes how to connect to data resources. See <a href="#">MyGene-class</a> . If missing, default object will be used that accesses the main MyGene.info portal. Default is recommended.

**Value**

returns a gene object (DataFrame, list, or JSON text) containing the queried annotations

**References**

Ref: [http://docs.mygene.info/en/latest/doc/query\\_service.html](http://docs.mygene.info/en/latest/doc/query_service.html)

**See Also**

[queryMany](#) [getGene](#) [getGenes](#)

**Examples**

```
## return the query result
query("cdk2", size=5)

query("reporter:1000_at")

query("symbol:cdk2", return.as="text")

query(q="cyclin-dependent kinase", fields="uniprot")
```

---

queryMany

*Return the batch query result.*

---

**Description**

This is a wrapper for POST query of `/query` service.

**Usage**

```
queryMany(qterms, scopes=NULL, ..., return.as=c("DataFrame", "records", "text"), mygene)
```



**Arguments**

qterms	A vector or list, or string of comma-separated query terms
scopes	Type of types of identifiers, either a list or a comma-separated fields to specify type of input qterms, e.g. c("reporter", "ensembl.gene", "symbol") refer to "http://mygene.info/doc/query_service.html#available_fields" for full list of fields.
...	Commonly queried fields include species, fields, size as well as several other fields. returnall returns a list of all related data including duplicated and missing qterms. False by default. View available fields by calling ?metadata. Also, see <a href="http://docs.mygene.info/en/latest/doc/query_service.html">http://docs.mygene.info/en/latest/doc/query_service.html</a> for complete argument details and syntax.
return.as	"DataFrame" (default), "records" (list), "text" (JSON).
mygene	A MyGene object that describes how to connect to data resources. See <a href="#">MyGene-class</a> . If missing, default object will be used that accesses the main MyGene.info portal. Default is recommended.

**Value**

returns a gene object (DataFrame, list, or JSON text) containing the queried annotations

**References**

Ref: [http://docs.mygene.info/en/latest/doc/query\\_service.html](http://docs.mygene.info/en/latest/doc/query_service.html)

**See Also**

[query](#) [getGene](#) [getGenes](#)

**Examples**

```
## return the batch query result
queryMany(c('1053_at', '117_at', '121_at'), scopes="reporter", fields="ensembl.gene",
          species="human", return.as="records")

queryMany(c('1053_at', '117_at', '121_at'), scopes="reporter", species=9606)

queryMany(c('DDX26B', 'CCDC83', 'MAST3', 'FLOT1'), scopes="symbol", fields="entrezgene", species="human")
```

# Index

## \*Topic **classes**

MyGene-class, 6

## \*Topic **package**

mygene, 6

getGene, 2, 3, 8, 9

getGene,missing-method (getGene), 2

getGene,MyGene-method (getGene), 2

getGenes, 2, 3, 8, 9

getGenes,missing-method (getGenes), 3

getGenes,MyGene-method (getGenes), 3

makeTxDb, 4

makeTxDbFromMyGene, 4

metadata, 5

metadata,MyGene-method (metadata), 5

MyGene, 5

mygene, 6

MyGene-class, 6

query, 2, 3, 7, 9

query,missing-method (query), 7

query,MyGene-method (query), 7

queryMany, 2, 3, 8, 8

queryMany,missing-method (queryMany), 8

queryMany,MyGene-method (queryMany), 8