

Package ‘KEGGREST’

October 18, 2017

Version 1.16.1

Title Client-side REST access to KEGG

Author Dan Tenenbaum

Imports methods, httr, png, Biostrings

Suggests RUnit, BiocGenerics, knitr

Maintainer Bioconductor Package Maintainer

<maintainer@bioconductor.org>

Description A package that provides a client interface to the KEGG REST server. Based on KEGGSOAP by J. Zhang, R. Gentleman, and Marc Carlson, and KEGG (python package) by Aurelien Mazurie.

License Artistic-2.0

VignetteBuilder knitr

biocViews Annotation, Pathways, ThirdPartyClient, KEGG

NeedsCompilation no

R topics documented:

keggConv	2
keggFind	3
keggGet	4
keggInfo	5
keggLink	6
keggList	7
listDatabases	8
mark.pathway.by.objects	8

Index	10
--------------	-----------

keggConv

Convert KEGG identifiers to/from outside identifiers

Description

Convert KEGG identifiers to/from outside identifiers.

Usage

```
keggConv(target, source)
```

Arguments

target	A KEGG organism code (), T number, or one of the external databases ncbi-gi, ncbi-geneid, ncbi-proteinid, uniprot, or (for chemical substance identifiers) drug, compound, or glycan, pubchem, or chebi.
source	Same as target, but may also be a list of KEGG identifiers representing internal or external names.

Value

A named character vector.

Author(s)

Dan Tenenbaum

References

<http://www.kegg.jp/kegg/docs/keggapi.html>

Examples

```
head(keggConv("eco", "ncbi-geneid")) ## conversion from NCBI GeneID to
## KEGG ID for E. coli genes
head(keggConv("ncbi-geneid", "eco")) ## opposite direction
head(keggConv("ncbi-proteinid", c("hsa:10458", "ece:Z5100"))) ## conversion from KEGG ID
## to NCBI GI
## conversion from NCBI GI to KEGG ID when the organism code is not known:
head(keggConv("genes", "ncbi-geneid:3113320"))
```

keggFind	<i>Finds entries with matching query keywords or other query data in a given database</i>
----------	---

Description

Finds entries with matching query keywords or other query data in a given database.

Usage

```
keggFind(database, query, option = c("formula", "exact_mass",  
  "mol_weight"))
```

Arguments

database	Either the name of a single KEGG database (list available via <code>listDatabases()</code>), a "T number" genome identifier, or a KEGG organism code (lists of both available via <code>keggList("organism")</code>).
query	One or more keywords, or a range of integers representing molecular weights. If query includes identifiers not known to KEGG, the results will not contain any information about those identifiers.
option	Optional. If database is compound or drug, option can be formula, exact_mass, or weight. Chemical formula search is a partial match irrespective of the order of atoms given. The exact mass (or molecular weight) is checked by rounding off to the same decimal place as the query data.

Value

A named character vector.

Author(s)

Dan Tenenbaum

References

<http://www.kegg.jp/kegg/docs/keggapi.html>

Examples

```
keggFind("genes", c("shiga", "toxin")) ## for keywords "shiga" and "toxin"  
keggFind("genes", "shiga toxin") ## for keywords "shiga toxin"  
keggFind("compound", "C7H1005", "formula") ## for chemical formula "C7H1005"  
keggFind("compound", "05C7", "formula") ## for chemical formula containing  
  ## "05" and "C7"  
keggFind("compound", 174.05, "exact_mass") ## for 174.045  
  ## =< exact mass < 174.055  
keggFind("compound", 300:310, "mol_weight") ## for 300 =<  
  ## molecular weight =< 310
```



```
library(png)
writePNG(png, t)
keggGet("hsa05130", "kgml")
```

keggInfo	<i>Displays the current statistics of a given database</i>
----------	--

Description

Displays statistics of a given database, such as number of entries, version, release date, and source.

Usage

```
keggInfo(database)
```

Arguments

database	Either a KEGG database (list available via <code>listDatabases()</code>), a KEGG organism code (list available by calling <code>keggList()</code> with the organism argument), or a T number (list available by calling <code>keggList()</code> with the genome argument.)
----------	---

Value

A character vector containing statistics about database.

Author(s)

Dan Tenenbaum

References

<http://www.kegg.jp/kegg/docs/keggapi.html>

Examples

```
keggInfo("kegg") ## displays the current statistics of the KEGG database
keggInfo("pathway") ## displays the number pathway entries including both
                    ## the reference and organism-specific pathways
keggInfo("hsa") ## displays the number of gene entries for the KEGG organism
                ## Homo sapiens
```

keggLink

Find related entries by using database cross-references.

Description

Find related entries by using database cross-references.

Usage

```
keggLink(target, source)
```

Arguments

target	Either the name of a single KEGG database (list available via <code>listDatabases()</code> , a "T number" genome identifier, or a KEGG organism code (lists of both available via <code>keggList("organism")</code>).
source	The same as target, but may also be one or more KEGG identifiers.

Details

Many of the old KEGGSOAP functions whose names started with 'get', such as `get.pathways.by.genes` and `get.pathways.by.reactions`, are replaced by using `keggLink` (see examples).

Value

A named character vector.

Author(s)

Dan Tenenbaum

References

<http://www.kegg.jp/kegg/docs/keggapi.html>

Examples

```
keggLink("pathway", "hsa") ## KEGG pathways linked from each of the human genes
                          ## equivalent to 'get.genes.by.pathway' in KEGGSOAP
keggLink("hsa", "pathway") ## human genes linked from each of the KEGG pathways
                          ## equivalent to 'get.pathways.by.genes' in KEGGSOAP
keggLink("pathway", c("hsa:10458", "ece:Z5100")) ## KEGG pathways linked from a
                                                  ## human gene and an E. coli
                                                  ## 0157 gene
keggLink("hsa:126") ## LinkDB search shows all KEGG resources related to hsa:126
```

keggList	<i>Returns a list of entry identifiers and associated definition for a given database or a given set of database entries.</i>
----------	---

Description

Returns a list of entry identifiers and associated definition for a given database or a given set of database entries.

Usage

```
keggList(database, organism)
```

Arguments

database	Either a KEGG database (list available via <code>listDatabases()</code>), a KEGG organism code (list available via <code>keggList()</code> with the organism argument), a T number (list available via <code>keggList()</code> with the genome argument), or a character vector of KEGG identifiers.
organism	Optional. A KEGG organism identifier (list available via <code>keggList()</code> with the organism argument).

Value

A named character vector containing entry identifiers and associated definition.

Author(s)

Dan Tenenbaum

References

<http://www.kegg.jp/kegg/docs/keggapi.html>

Examples

```
keggList("pathway") ## returns the list of reference pathways
keggList("pathway", "hsa") ## returns the list of human pathways
keggList("organism") ## returns the list of KEGG organisms with
  ## taxonomic classification
keggList("hsa") ## returns the entire list of human genes
keggList("T01001") ## same as above
keggList(c("hsa:10458", "ece:Z5100")) ## returns the list of a human gene
  ## and an E.coli 0157 gene
keggList(c("cpd:C01290", "gl:G00092")) ## returns the list of a compound entry
  ## and a glycan entry
keggList(c("C01290+G00092")) ## same as above (prefixes are not necessary)
```


Arguments

- `pathway.id` `pathway.id` a character string for a KEGG pathway id. KEGG pathway ids consist of the string path followed by a colon, a three-letter code for the organism of concern, and then a number (e. g. "path:eco00020"). The three-letter organism code consists of the first letter of the genus name and the first two letters of the species name of the scientific name of the organism of concern
- `object.id.list` `object.id.list` a vector of character strings for KEGG gene ids. KEGG gene ids normally consist of three letters followed by a column and then several numeric numbers. The three letters are from the first letter of the genus name and the first two letters of the species name of the scientific name of the organism of concern (e. g. hsa:111 for Homo Sapiens)
- `fg.color.list` `fg.color.list` a vector of two character strings to indicate the color for the text and border, respectively, of the objects in a pathway diagram. The strings can either be a color code like `\#ff0000` or letter like yellow
- `bg.color.list` `bg.color.list` a vector of character strings of the same length of `object.id.list` to indicate the background color of the objects in a pathway diagram. The strings can either be a color code like `\#ff0000` or letter like yellow

Details

This function only returns the URL of the KEGG pathway diagram. Use the function [browseURL](#) to view the diagram.

These functions are not part of the KEGG REST API; they are provided because they existed in KEGGSOAP and an alternative implementation was possible.

Value

This function returns a character string for the url

Author(s)

Jianhua Zhang

References

<http://www.kegg.jp/kegg/docs/keggapi.html>

See Also

[browseURL](#)

Examples

```
url <- mark.pathway.by.objects("path:eco00260",
                              c("eco:b0002", "eco:c00263"))
if(interactive()){
  browseURL(url)
}
url <- color.pathway.by.objects("path:eco00260",
                               c("eco:b0002", "eco:c00263"),
                               c("#ff0000", "#00ff00"), c("#ffff00", "yellow"))
```

Index

- *Topic **conv**
 - keggConv, 2
- *Topic **databases**
 - listDatabases, 8
- *Topic **database**
 - listDatabases, 8
- *Topic **datasets**
 - mark.pathway.by.objects, 8
- *Topic **find**
 - keggFind, 3
- *Topic **get**
 - keggGet, 4
- *Topic **info**
 - keggInfo, 5
- *Topic **link**
 - keggLink, 6
- *Topic **list**
 - keggList, 7
- *Topic **metadata**
 - keggInfo, 5

bconv (keggConv), 2
browseURL, 9

color.pathway.by.objects
 (mark.pathway.by.objects), 8
conv (keggConv), 2

info (keggInfo), 5

keggConv, 2
keggFind, 3
keggGet, 4
keggInfo, 5
keggLink, 6
keggList, 5, 7, 7, 8

link (keggLink), 6
listDatabases, 3, 5–7, 8

mark.pathway.by.objects, 8