Package 'biodbChebi'

November 21, 2024

Title biodbChebi, a library for connecting to the ChEBI Database

Version 1.13.0

Description The biodbChebi library provides access to the ChEBI Database, using biodb package framework. It allows to retrieve entries by their accession number. Web services can be accessed for searching the database by name, mass or other fields.

URL https://github.com/pkrog/biodbChebi

BugReports https://github.com/pkrog/biodbChebi/issues

biocViews Software, Infrastructure, DataImport

Depends R (>= 4.1)

License AGPL-3

Encoding UTF-8

VignetteBuilder knitr

Suggests BiocStyle, roxygen2, devtools, testthat (>= 2.0.0), knitr, rmarkdown, lgr

Imports R6, biodb (>= 1.1.5)

RoxygenNote 7.1.2

Collate 'ChebiConn.R' 'ChebiEntry.R' 'package.R'

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

git_branch devel

git_last_commit 686a27c

git_last_commit_date 2024-10-29

Repository Bioconductor 3.21

Date/Publication 2024-11-21

Author Pierrick Roger [aut, cre] (ORCID: <https://orcid.org/0000-0001-8177-4873>)

Maintainer Pierrick Roger <pierrick.roger@cea.fr>

Contents

`ChebiConn` *ChEBI connector class.*

Description

ChEBI connector class.

ChEBI connector class.

Details

This is the connector class for connecting to the ChEBI database through its web services.

Super classes

`biodb::BiodbConnBase` -> `biodb::BiodbConn` -> `ChebiConn`

Methods

Public methods:

- `ChebiConn$new()`
- `ChebiConn$wsWsdl()`
- `ChebiConn$wsGetLiteEntity()`
- `ChebiConn$convIdsToChebiIds()`
- `ChebiConn$convInchiToChebi()`
- `ChebiConn$convCasToChebi()`
- `ChebiConn$getWsdl()`
- `ChebiConn$getWsdlEnumeration()`
- `ChebiConn$getStarsCategories()`
- `ChebiConn$getSearchCategories()`
- `ChebiConn$clone()`

Method `new()`**:** New instance initializer. Connector classes must not be instantiated directly. Instead, you must use the createConn() method of the factory class.

Usage:

```
ChebiConn$new(...)
```

Arguments:

`...` All parameters are passed to the super class initializer.

Returns: Nothing.

Method `wsWsdl()`**:** Retrieves the complete WSDL from the web server.

Usage:

```
ChebiConn$wsWsdl(retfmt = c("plain", "parsed", "request"))
```

Arguments:

`retfmt` The return format to use. 'plain' will return the value as it is returned by the server. 'parsed' will return an XML object. 'request' will return a BiodbRequest object representing the request that would have been sent.

Returns: Depending on 'retfmt' value.

Method `wsGetLiteEntity()`**:** Calls getLiteEntity web service and returns the XML result. Be careful when searching by mass (search.category='MASS' or 'MONOISOTOPIC MASS'), since the search is made in text mode, thus the number must be exactly written as it is stored in database, eventually padded with 0 in order to have exactly 5 digits after the decimal. An easy solution is to use wildcards to search a mass '410;.718*'. See http //www.ebi.ac.uk/chebi/webServices.do for more details.

Usage:

```
ChebiConn$wsGetLiteEntity(
  search = NULL,
  search.category = "ALL",
  stars = "ALL",
  max.results = 10,
  retfmt = c("plain", "parsed", "request", "ids")
)
```

Arguments:

`search` The text or pattern to search.

`search.category` The search category. Call 'getSearchCategories()' to get a full list of search categories.

`stars` How many starts the returned entities should have. Call 'getStarsCategories() to get a full list of starts categories.'

`max.results` The maximum of results to return.

`retfmt` The return format to use. 'plain' will return the results as given by the server, in a string. 'parsed' will return an XML object. 'request' will return a BiodbRequest object representing the request as would have been sent. 'ids' will return a list of matched entity IDs.

Returns: Depending on 'retfmt' value.

Method `convIdsToChebiIds()`**:** Converts a list of IDs (InChI, InChI Keys, CAS, ...) into a list of ChEBI IDs. Several ChEBI IDs may be returned for a single ID.

Usage:

```
ChebiConn$convIdsToChebiIds(ids, search.category, simplify = TRUE)
```

Arguments:

`ids` The identifiers to convert.

`search.category` The search category. Call 'getSearchCategories()' to get a full list of search categories.

`simplify` If set to TRUE and only one ChEBI ID has been found for each ID, then a character vector is returned. Otherwise a list of character vectors is returned.

Returns: Depending on the value of simplify.

Method `convInchiToChebi()`**:** Converts a list of InChI or InChI KEYs into a list of ChEBI IDs. Several ChEBI IDs may be returned for a single InChI or InChI KEY.

Usage:

```
ChebiConn$convInchiToChebi(inchi, simplify = TRUE)
```

Arguments:

`inchi` The InChI values to convert.

`simplify` If set to TRUE and only one ChEBI ID has been found for each ID, then a character vector is returned. Otherwise a list of character vectors is returned.

Returns: Depending on the value of simplify.

Method `convCasToChebi()`**:** Converts a list of CAS IDs into a list of ChEBI IDs. Several ChEBI IDs may be returned for a single InChI or InChI KEY.

Usage:

```
ChebiConn$convCasToChebi(cas, simplify = TRUE)
```

Arguments:

`cas` The CAS IDs to convert.

`simplify` If set to TRUE and only one ChEBI ID has been found for each ID, then a character vector is returned. Otherwise a list of character vectors is returned.

Returns: Depending on the value of simplify.

Method `getWsdl()`**:** Gets the WSDL as an XML object.

Usage:

```
ChebiConn$getWsdl()
```

Returns: The ChEBI WSDL as an XML object.

Method `getWsdlEnumeration()`**:** Extracts a list of values from an enumeration in the WSDL.

Usage:

```
ChebiConn$getWsdlEnumeration(name)
```

Arguments:

`name` The name of the enumeration for which to retrieve the values.

Returns: A character vector listing the enumerated values.

Method `getStarsCategories()`**:** Gets the list of allowed stars categories for the getLiteEntity web service.

Usage:

```
ChebiConn$getStarsCategories()
```

Returns: Returns all the possible stars categories as a character vector.

Method `getSearchCategories()`**:** Gets the list of allowed search categories for the getLiteEntity web service.

Usage:

```
ChebiConn$getSearchCategories()
```

Returns: Returns all the possible search categories as a character vector.

Method `clone()`**:** The objects of this class are cloneable with this method.

Usage:

```
ChebiConn$clone(deep = FALSE)
```

Arguments:

`deep` Whether to make a deep clone.

Examples

```
# Create an instance with default settings:
mybiodb <- biodb::newInst()

# Create a connector
conn <- mybiodb$getFactory()$createConn('chebi')

# Get an entry
e <- conn$getEntry('15440')

# Convert an InChI KEY to a ChEBI identifier
conn$convInchiToChebi('YYGNTYWPHWGJRM-AAJYLUCBSA-N')

# Terminate instance.
mybiodb$terminate()
```

`ChebiEntry` *ChEBI entry class.*

Description

This is the entry class for ChEBI database.

Super classes

`biodb::BiodbEntry` -> `biodb::BiodbXmlEntry` -> `ChebiEntry`

Methods

Public methods:

- `ChebiEntry$clone()`

Method `clone()`**:** The objects of this class are cloneable with this method.

Usage:

```
ChebiEntry$clone(deep = FALSE)
```

Arguments:

`deep` Whether to make a deep clone.

Examples

```
# Create an instance with default settings:
mybiodb <- biodb::newInst()

# Create a connector to ChEBI
conn <- mybiodb$getFactory()$createConn('chebi')

# Get an entry
e <- conn$getEntry('15440')

# Terminate instance.
mybiodb$terminate()
```

Index