# Package: ginmappeR (via r-universe)

October 27, 2026

**Type** Package

**Title** Gene Identifier Mapper

**Version** 1.8.0

**Description** Provides functionalities to translate gene or protein identifiers between state-of-art biological databases: CARD (<https://card.mcmaster.ca/>), NCBI Protein, Nucleotide and Gene (<https://www.ncbi.nlm.nih.gov/>), UniProt (<https://www.uniprot.org/>) and KEGG (<https://www.kegg.jp>). Also offers complementary functionality like NCBI identical proteins or UniProt similar genes clusters retrieval.

**License** GPL-3 + file LICENSE

**Encoding** UTF-8

**VignetteBuilder** knitr

**biocViews** Annotation, KEGG, Genetics, ThirdPartyClient, Software

**Imports** KEGGREST, UniProt.ws, XML, rentrez, httr, utils, memoise, cachem, jsonlite, rvest

**Suggests** RUnit, BiocGenerics, markdown, knitr

**Config/pak/sysreqs** libicu-dev libpng-dev libxml2-dev libssl-dev zlib1g-dev

**Repository** Bioconductor 3.23

**Date/Publication** 2026-04-28 13:02:35 UTC

**RemoteUrl** https://github.com/bioc/ginmappeR

**RemoteRef** RELEASE_3_23

**RemoteSha** 4a78da3d3b03d9193d74a8962ffb30d842fd2ea1

## Contents


changeCARDPath . . . . . 2
getCARD2KEGG . . . . . 3
getCARD2NCBIGene . . . . . 4
getCARD2NCBINucleotide . . . . . 5
getCARD2NCBIProtein . . . . . 5

getCARD2UniProt . . . . . 6
getCARDVersion . . . . . 7
getKEGG2CARD . . . . . 7
getKEGG2NCBIGene . . . . . 8
getKEGG2NCBINucleotide . . . . . 9
getKEGG2NCBIProtein . . . . . 10
getKEGG2UniProt . . . . . 11
getKEGGVersion . . . . . 12
getNCBIGene2CARD . . . . . 13
getNCBIGene2KEGG . . . . . 13
getNCBIGene2NCBINucleotide . . . . . 14
getNCBIGene2NCBIProtein . . . . . 15
getNCBIGene2UniProt . . . . . 16
getNCBIIdenticalProteins . . . . . 17
getNCBINucleotide2CARD . . . . . 18
getNCBINucleotide2KEGG . . . . . 19
getNCBINucleotide2NCBIGene . . . . . 20
getNCBINucleotide2NCBIProtein . . . . . 21
getNCBINucleotide2UniProt . . . . . 21
getNCBIProtein2CARD . . . . . 22
getNCBIProtein2KEGG . . . . . 23
getNCBIProtein2NCBIGene . . . . . 24
getNCBIProtein2NCBINucleotide . . . . . 25
getNCBIProtein2UniProt . . . . . 26
getNCBIVersion . . . . . 27
getUniProt2CARD . . . . . 27
getUniProt2KEGG . . . . . 28
getUniProt2NCBIGene . . . . . 29
getUniProt2NCBINucleotide . . . . . 30
getUniProt2NCBIProtein . . . . . 31
getUniProtSimilarGenes . . . . . 32
getUniProtVersion . . . . . 33
updateCARDDataBase . . . . . 34

**Index** **35**

---

`changeCARDPath` *Change path to download CARD database*

---

## Description

Defines a new path to the folder where to download CARD database. By default, this path is the temporary folder tempdir().

## Usage

```
changeCARDPath(path=tempdir())
```

### Arguments

| | |
|---|---|
| path | valid route to desired folder. Default value: tempdir() |

### Value

Updated path for CARD database to be downloaded.

---

## getCARD2KEGG *Translates a CARD database ID to KEGG database*

---

### Description

Retrieves the ID translation of an specific instance of CARD database ID to the KEGG Genes database.

### Usage

```
getCARD2KEGG(cardId, exhaustiveMapping = FALSE, detailedMapping = FALSE,
byIdenticalProteins = TRUE, bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| cardId | valid CARD database ARO ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of KEGG IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| byIdenticalProteins | logical value (TRUE or FALSE) indicating to translate also through NCBI Identical Proteins. TRUE value could mean higher translation time and FALSE, fewer chances to translate the ID |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as cardId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate CARD ID '3000938' to KEGG
getCARD2KEGG('3000938', byIdenticalProteins = FALSE, bySimilarGenes = FALSE)

# Translate CARD IDs '3000938' and '3002557' to KEGG
getCARD2KEGG(c('3000938', '3002557'))

# Get all possible translations of IDs '3000938', '3002511' with detailed origin
getCARD2KEGG(c('3000938', '3002511'), exhaustiveMapping = TRUE,
detailedMapping = TRUE)
```

---

`getCARD2NCBIGene` *Translates a CARD database ID to NCBI Gene database*

---

## Description

Retrieves the ID translation of an specific instance of CARD database ID to the NCBI Gene database.

## Usage

```
getCARD2NCBIGene(cardId, exhaustiveMapping = FALSE)
```

## Arguments

| | |
|---|---|
| `cardId` | valid CARD database ARO ID or a vector of them |
| `exhaustiveMapping` | logical value (`TRUE` or `FALSE`) indicating to retrieve all possible translations or only the first and fastest one. |

## Value

A character vector of the same length as `cardId`. If parameter `exhaustiveMapping` is `TRUE`, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate CARD ID '3002525' to NCBI Gene
getCARD2NCBIGene('3002525')

# Translate CARD IDs '3002524' and '3002525' to NCBI Gene
getCARD2NCBIGene(c('3002524', '3002525'))

# Get all possible translations of IDs '3005061' and '3002524'
getCARD2NCBIGene(c('3005061', '3002524'), exhaustiveMapping = TRUE)
```

## getCARD2NCBINucleotide *Translates a CARD database ID to NCBI Nucleotide database*

### Description

Retrieves the ID translation of an specific instance of CARD database ID to the NCBI Nucleotide database. Note that, due to the nature of NCBI Nucleotide database, a complete genome is often retrieved.

### Usage

```
getCARD2NCBINucleotide(cardId)
```

### Arguments

| | |
|---|---|
| cardId | valid CARD database ARO ID or a vector of them |

### Value

A character vector of the same length as cardId.

### Examples

```
# Translate CARD ID '3002535' to NCBI Nucleotide
getCARD2NCBINucleotide('3002535')

# Translate CARD ID 'ARO:3002535' to NCBI Nucleotide
getCARD2NCBINucleotide('ARO:3002535')

# Translate CARD IDs '3002535' and '3003988' to NCBI Nucleotide
getCARD2NCBINucleotide(c('3002535', '3003988'))
```

## getCARD2NCBIProtein *Translates a CARD database ID to NCBI Protein database*

### Description

Retrieves the ID translation of an specific instance of CARD database ID to the NCBI Protein database.

### Usage

```
getCARD2NCBIProtein(cardId)
```

### Arguments

| | |
|---|---|
| cardId | valid CARD database ARO ID or a vector of them |

### Value

A character vector of the same length as cardId.

### Examples

```
# Translate CARD ID '3002535' to NCBI Protein
getCARD2NCBIProtein('3002535')

# Translate CARD ID 'ARO:3002535' to NCBI Protein
getCARD2NCBIProtein('ARO:3002535')

# Translate CARD IDs '3002535' and '3003988' to NCBI Protein
getCARD2NCBIProtein(c('3002535', '3003988'))
```

---

## getCARD2UniProt *Translates a CARD database ID to UniProt database*

---

### Description

Retrieves the ID translation of an specific instance of CARD database ID to the UniProt database.

### Usage

```
getCARD2UniProt(cardId, exhaustiveMapping = FALSE, detailedMapping = FALSE)
```

### Arguments

| | |
|---|---|
| cardId | valid CARD database ARO ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of UniProt IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |

### Value

A character vector of the same length as cardId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate CARD ID '3002867' to UniProt
getCARD2UniProt('3002867')

# Translate CARD IDs '3002867' and '3002535' to UniProt
getCARD2UniProt(c('3002867', '3002535'))

# Get all possible translations of IDs '3002867', '3002535' with detailed origin
getCARD2UniProt(c('3002867', '3002535'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

# getCARDVersion *Prints CARD database version being used*

---

## Description

Prints the downloaded CARD version and release date.

## Usage

```
getCARDVersion()
```

## Value

Downloaded CARD database version.

---

# getKEGG2CARD *Translates a KEGG database ID to CARD database*

---

## Description

Retrieves the ID translation of an specific instance of KEGG Genes database ID to the CARD database.

## Usage

```
getKEGG2CARD(keggId, exhaustiveMapping = FALSE, detailedMapping = FALSE,
bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| keggId | valid KEGG Genes database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of CARD IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as keggId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate KEGG ID 'ag:ACC85616' to CARD
getKEGG2CARD('ag:ACC85616')

# Translate KEGG IDs 'ag:ACC85616' and 'ag:CAJ47134' to CARD
getKEGG2CARD(c('ag:ACC85616', 'ag:CAJ47134'))

# Get all possible translations of IDs 'ag:ACC85616', 'ag:CAJ47134'
# with detailed origin
getKEGG2CARD(c('ag:ACC85616', 'ag:CAJ47134'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

## getKEGG2NCBIGene *Translates a KEGG database ID to NCBI Gene database*

---

### Description

Retrieves the ID translation of an specific instance of KEGG Genes database ID to the NCBI Gene database.

### Usage

```
getKEGG2NCBIGene(keggId, exhaustiveMapping = FALSE, detailedMapping = FALSE,
bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| keggId | valid KEGG Genes database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of NCBI Gene IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as keggId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate KEGG ID 'aag:5579347' to NCBI Gene
getKEGG2NCBIGene('aag:5579347', detailedMapping = TRUE)

# Translate KEGG IDs 'aag:5579347' and 'abh:M3Q_2831' to NCBI Gene
getKEGG2NCBIGene(c('aag:5579347', 'abh:M3Q_2831'))

# Get all possible translations of IDs 'aag:5579347', 'abh:M3Q_2831' with
# detailed origin
getKEGG2NCBIGene(c('aag:5579347', 'abh:M3Q_2831'), exhaustiveMapping = TRUE,
detailedMapping = TRUE)
```

---

## getKEGG2NCBINucleotide

*Translates a KEGG database ID to NCBI Nucleotide database*

---

### Description

Retrieves the ID translation of an specific instance of KEGG Genes database ID to the NCBI Nucleotide database. Note that, due to the nature of NCBI Nucleotide database, a complete genome is often retrieved.

### Usage

```
getKEGG2NCBINucleotide(keggId, exhaustiveMapping = FALSE,
detailedMapping = FALSE, bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| keggId | valid KEGG Genes database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of NCBI Nucleotide IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as keggId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate KEGG ID 'llo:LLO_2673' to NCBI Nucleotide
getKEGG2NCBINucleotide('llo:LLO_2673', bySimilarGenes = FALSE)

# Translate KEGG IDs 'llo:LLO_2673' and 'abh:M3Q_2831' to NCBI Nucleotide
getKEGG2NCBINucleotide(c('llo:LLO_2673', 'abh:M3Q_2831'))

# Get all possible translations of IDs 'aag:5579347', 'abh:M3Q_2831' with
# detailed origin
getKEGG2NCBINucleotide(c('aag:5579347', 'abh:M3Q_2831'),
exhaustiveMapping = TRUE, detailedMapping = TRUE)
```

---

## getKEGG2NCBIProtein *Translates a KEGG database ID to NCBI Protein database*

---

### Description

Retrieves the ID translation of an specific instance of KEGG Genes database ID to the NCBI Protein database.

### Usage

```
getKEGG2NCBIProtein(keggId, exhaustiveMapping = FALSE, detailedMapping = FALSE,
bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| keggId | valid KEGG Genes database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of NCBI Protein IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as keggId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate KEGG ID 'llo:LLO_2673' to NCBI Protein
getKEGG2NCBIProtein('llo:LLO_2673')

# Translate KEGG IDs 'llo:LLO_2673' and 'abh:M3Q_2831' to NCBI Protein
getKEGG2NCBIProtein(c('llo:LLO_2673', 'abh:M3Q_2831'))

# Get all possible translations of IDs 'llo:LLO_2673', 'abh:M3Q_2831' with
# detailed origin
getKEGG2NCBIProtein(c('llo:LLO_2673', 'abh:M3Q_2831'),
exhaustiveMapping = TRUE, detailedMapping = TRUE)
```

---

## getKEGG2UniProt *Translates a KEGG Genes database ID to UniProt database*

---

### Description

Retrieves the ID translation of an specific instance of KEGG Genes database ID to the UniProt database.

### Usage

```
getKEGG2UniProt(keggId, exhaustiveMapping = FALSE)
```

### Arguments

| | |
|---|---|
| keggId | valid KEGG Genes database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |

### Value

A character vector of the same length as keggId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate KEGG Genes ID 'llo:LLO_2673' to UniProt
getKEGG2UniProt('llo:LLO_2673')

# Translate KEGG IDs 'llo:LLO_2673' and 'abh:M3Q_2831' to NCBI UniProt
getKEGG2UniProt(c('llo:LLO_2673', 'abh:M3Q_2831'))

# Get all possible translations of IDs 'llo:LLO_2673', 'abh:M3Q_2831'
getKEGG2UniProt(c('llo:LLO_2673', 'abh:M3Q_2831'), exhaustiveMapping = TRUE)
```

---

## getKEGGVersion *Prints KEGG database version being used*

---

### Description

Prints the accessed KEGG version and its corresponding release date. It also prints the API version release date.

### Usage

```
getKEGGVersion()
```

### Value

KEGG database and API version.

## getNCBIGene2CARD *Translates a NCBI Gene database ID to CARD database*

### Description

Retrieves the ID translation of an specific instance of NCBI Gene database ID to the CARD database.

### Usage

```
getNCBIGene2CARD(ncbiId, exhaustiveMapping = FALSE)
```

### Arguments

| | |
|---|---|
| ncbiId | valid NCBI Gene database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |

### Value

A character vector of the same length as ncbiId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate NCBI Gene ID '3510143' to CARD
getNCBIGene2CARD('3510143')

# Translate NCBI Gene IDs '3510143' and '29426913' to CARD
getNCBIGene2CARD(c('3510143', '29426913'))

# Get all possible translations of IDs '3510143' and '29426913'
getNCBIGene2CARD(c('3510143', '29426913'), exhaustiveMapping = TRUE)
```

## getNCBIGene2KEGG *Translates a NCBI Gene database ID to KEGG database*

### Description

Retrieves the ID translation of an specific instance of NCBI Gene database ID to the KEGG Genes database.

### Usage

```
getNCBIGene2KEGG(ncbiId, exhaustiveMapping = FALSE, detailedMapping = FALSE,
byIdenticalProteins = TRUE, bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| ncbiId | valid NCBI Gene database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of KEGG IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| byIdenticalProteins | logical value (TRUE or FALSE) indicating to translate also through NCBI Identical Proteins. TRUE value could mean higher translation time and FALSE, fewer chances to translate the ID |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as ncbiId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate NCBI Gene ID '76524190' to KEGG
getNCBIGene2KEGG('AFU38919', bySimilarGenes = FALSE, byIdenticalProteins = FALSE)

# Translate NCBI Gene IDs '76524190' and 'AFU38919' to KEGG
getNCBIGene2KEGG(c('76524190', 'AFU38919'))

# Get all possible translations of IDs '76524190' and 'AFU38919' with
# detailed origin
getNCBIGene2KEGG(c('76524190', 'AFU38919'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

## getNCBIGene2NCBINucleotide

*Translates a NCBI Gene database ID to NCBI Nucleotide database*

---

### Description

Retrieves the ID translation of an specific instance of NCBI Gene database ID to the NCBI Nucleotide database. Note that, due to the nature of NCBI Nucleotide database, a complete genome is often retrieved.

### Usage

```
getNCBIGene2NCBINucleotide(id, exhaustiveMapping=FALSE)
```

### Arguments

| | |
|---|---|
| id | valid NCBI Gene database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |

### Value

A character vector of the same length as id. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate NCBI Gene ID '76524190' to NCBI Nucleotide
getNCBIGene2NCBINucleotide('76524190')

# Translate NCBI Gene IDs '76524190' and '1272' to NCBI Nucleotide
getNCBIGene2NCBINucleotide(c('76524190', '1272'))

# Get all possible translations of IDs '76524190' and '1272'
getNCBIGene2NCBINucleotide(c('76524190', '1272'), exhaustiveMapping = TRUE)
```

---

## getNCBIGene2NCBIProtein

*Translates a NCBI Gene database ID to NCBI Protein database*

---

### Description

Retrieves the ID translation of an specific instance of NCBI Gene database ID to the NCBI Protein database.

### Usage

```
getNCBIGene2NCBIProtein(id, exhaustiveMapping=FALSE)
```

### Arguments

| | |
|---|---|
| id | valid NCBI Gene database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |

## Value

A character vector of the same length as `id`. If parameter `exhaustiveMapping` is `TRUE`, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate NCBI Gene ID '76524190' to NCBI Protein
getNCBIGene2NCBIProtein('76524190')

# Translate NCBI Gene IDs '76524190' and '1272' to NCBI Protein
getNCBIGene2NCBIProtein(c('76524190', '1272'))

# Get all possible translations of IDs '76524190', '1272'
getNCBIGene2NCBIProtein(c('76524190', '1272'), exhaustiveMapping = TRUE)
```

---

# `getNCBIGene2UniProt` *Translates a NCBI Gene database ID to UniProt database*

---

## Description

Retrieves the ID translation of an specific instance of NCBI Gene database ID to the UniProt database.

## Usage

```
getNCBIGene2UniProt(ncbiId, exhaustiveMapping = FALSE,
detailedMapping = FALSE, byIdenticalProteins = TRUE)
```

## Arguments

| | |
|---|---|
| `ncbiId` | valid NCBI Gene database ID or a vector of them |
| `exhaustiveMapping` | logical value (`TRUE` or `FALSE`) indicating to retrieve all possible translations or only the first and fastest one. |
| `detailedMapping` | logical value (`TRUE` or `FALSE`) indicating to return a vector of UniProt IDs if `FALSE`, or a named vector if `TRUE`. This named vector details the origin of the translated IDs, dividing them in `DT`, if it is a Direct Translation, or `1.0`, `0.9` or `0.5`, if it has been retrieved through any of the identical gene clusters. |
| `byIdenticalProteins` | logical value (`TRUE` or `FALSE`) indicating to translate also through NCBI Identical Proteins. `TRUE` value could mean higher translation time and `FALSE`, fewer chances to translate the ID |

## Value

A character vector of the same length as ncbiId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate NCBI Gene ID '76524190' to UniProt
getNCBIGene2UniProt('76524190')

# Translate NCBI Gene IDs '76524190' and '1272' to UniProt
getNCBIGene2UniProt(c('76524190', '1272'))

# Get all possible translations of IDs '76524190', '1272' with detailed origin
getNCBIGene2UniProt(c('76524190', '1272'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

# getNCBIIdenticalProteins

*Search NCBI Identical Protein database by a given NCBI ID*

---

## Description

Retrieves identical NCBI proteins to that of a given ID.

## Usage

```
getNCBIIdenticalProteins(ncbiId, format='ids')
```

## Arguments

| | |
|---|---|
| ncbiId | valid NCBI ID or a vector of them |
| format | return format of the response. Default value ids returns a vector of the identical proteins IDs; dataframe value returns a dataframe with different information about the found identical proteins |

## Value

Depending on parameter format, it will return a list of same legth as ncbiId with identical NCBI Protein IDs or with dataframes containing different information about the identical proteins found: Id, Source, Nucleotide.Accession, Start, Stop, Strand, Protein, Protein.Name, Organism, Strain and Assembly.

## Examples

```
# Get IDs on identical proteins to the one with ID 'AHA80958'
getNCBIIdenticalProteins('AHA80958', format = 'ids')

# Get IDs on identical proteins to the ones with IDs 'AHA80958' and 'CAA76794'
getNCBIIdenticalProteins(c('AHA80958', 'CAA76794'), format = 'ids')
```

---

# getNCBINucleotide2CARD

*Translates a NCBI Nucleotide database ID to CARD database*

---

## Description

Retrieves the ID translation of an specific instance of NCBI Nucleotide database ID to the CARD database.

## Usage

```
getNCBINucleotide2CARD(ncbiId, exhaustiveMapping = FALSE)
```

## Arguments

| | |
|---|---|
| `ncbiId` | valid NCBI Nucleotide database ID or a vector of them |
| `exhaustiveMapping` | logical value (`TRUE` or `FALSE`) indicating to retrieve all possible translations or only the first and fastest one. |

## Value

A character vector of the same length as `ncbiId`. If parameter `exhaustiveMapping` is `TRUE`, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate NCBI Nucleotide ID 'AY034138' to CARD
getNCBINucleotide2CARD('AY034138')

# Translate NCBI Nucleotide IDs 'AY034138' and 'X54723.1' to CARD
getNCBINucleotide2CARD(c('AY034138', 'X54723.1'))

# Get all possible translations of IDs 'AY034138', 'X54723.1'
getNCBINucleotide2CARD(c('AY034138', 'X54723.1'), exhaustiveMapping = TRUE)
```

getNCBINucleotide2KEGG *Translates a NCBI Nucleotide database ID to KEGG database*

## Description

Retrieves the ID translation of an specific instance of NCBI Nucleotide database ID to the KEGG Genes database.

## Usage

```
getNCBINucleotide2KEGG(ncbiId, exhaustiveMapping = FALSE,
detailedMapping = FALSE, byIdenticalProteins = TRUE, bySimilarGenes = TRUE)
```

## Arguments

| | |
|---|---|
| ncbiId | valid NCBI Nucleotide database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of KEGG IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| byIdenticalProteins | logical value (TRUE or FALSE) indicating to translate also through NCBI Identical Proteins. TRUE value could mean higher translation time and FALSE, fewer chances to translate the ID |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

## Value

A character vector of the same length as ncbiId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate NCBI Nucleotide ID 'NC_013861.1' to KEGG
getNCBINucleotide2KEGG('NC_013861.1', bySimilarGenes = FALSE, byIdenticalProteins = FALSE)

# Translate NCBI Nucleotide IDs 'AY536519' and 'NC_013861.1' to KEGG
getNCBINucleotide2KEGG(c('AY536519', 'NC_013861.1'))

# Get all possible translations of IDs 'AY536519', 'NC_013861.1'
```

```
# with detailed origin
getNCBINucleotide2KEGG(c('AY536519', 'NC_013861.1'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

`getNCBINucleotide2NCBIGene` *Translates a NCBI Nucleotide database ID to NCBI Gene database*

---

## Description

Retrieves the ID translation of an specific instance of NCBI Nucleotide database ID to the NCBI Gene database.

## Usage

```
getNCBINucleotide2NCBIGene(id, exhaustiveMapping=FALSE)
```

## Arguments

| | |
|---|---|
| `id` | valid NCBI Nucleotide database ID or a vector of them |
| `exhaustiveMapping` | logical value (`TRUE` or `FALSE`) indicating to retrieve all possible translations or only the first and fastest one. |

## Value

A character vector of the same length as `id`. If parameter `exhaustiveMapping` is `TRUE`, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate NCBI Nucleotide ID 'Z21488' to NCBI Gene
getNCBINucleotide2NCBIGene('Z21488')

# Get all possible translations of ID 'Z21488'
getNCBINucleotide2NCBIGene(c('Z21488'), exhaustiveMapping = TRUE)
```

## getNCBINucleotide2NCBIProtein

*Translates a NCBI Nucleotide database ID to NCBI Protein database*

### Description

Retrieves the ID translation of an specific instance of NCBI Nucleotide database ID to the NCBI Protein database.

### Usage

```
getNCBINucleotide2NCBIProtein(id, exhaustiveMapping=FALSE)
```

### Arguments

| | |
|---|---|
| id | valid NCBI Nucleotide database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |

### Value

A character vector of the same length as id. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate NCBI Nucleotide ID 'AY536519' to NCBI Protein
getNCBINucleotide2NCBIProtein('AY536519')

# Translate NCBI Nucleotide IDs 'AY536519' and 'X54723.1' to NCBI Protein
getNCBINucleotide2NCBIProtein(c('AY536519', 'X54723.1'))

# Get all possible translations of IDs 'AY536519', 'X54723.1'
getNCBINucleotide2NCBIProtein(c('AY536519', 'X54723.1'),
exhaustiveMapping = TRUE)
```

## getNCBINucleotide2UniProt

*Translates a NCBI Nucleotide database ID to UniProt database*

### Description

Retrieves the ID translation of an specific instance of NCBI Nucleotide database ID to the UniProt database.

### Usage

```
getNCBINucleotide2UniProt(ncbiId, exhaustiveMapping = FALSE,
detailedMapping = FALSE, byIdenticalProteins = TRUE)
```

### Arguments

| | |
|---|---|
| ncbiId | valid NCBI Nucleotide database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of UniProt IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| byIdenticalProteins | logical value (TRUE or FALSE) indicating to translate also through NCBI Identical Proteins. TRUE value could mean higher translation time and FALSE, fewer chances to translate the ID |

### Value

A character vector of the same length as ncbiId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate NCBI Nucleotide ID 'AY536519' to UniProt
getNCBINucleotide2UniProt('AY536519')

# Translate NCBI Nucleotide IDs 'AY536519' and 'X54723.1' to UniProt
getNCBINucleotide2UniProt(c('AY536519', 'X54723.1'))

# Get all possible translations of IDs 'AY536519', 'X54723.1'
# with detailed origin
getNCBINucleotide2UniProt(c('AY536519', 'X54723.1'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

## getNCBIProtein2CARD *Translates a NCBI Protein database ID to CARD database*

---

### Description

Retrieves the ID translation of an specific instance of NCBI Protein database ID to the CARD database.

### Usage

```
getNCBIProtein2CARD(ncbiId, exhaustiveMapping = FALSE)
```

### Arguments

| | |
|---|---|
| ncbiId | valid NCBI Protein database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |

### Value

A character vector of the same length as ncbiId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate NCBI Protein ID 'AAK64581' to CARD
getNCBIProtein2CARD('AAK64581')

# Translate NCBI Protein IDs 'AAK64581' and 'WP_010896559.1' to CARD
getNCBIProtein2CARD(c('AAK64581', 'WP_010896559.1'))

# Get all possible translations of IDs 'AAK64581', 'WP_010896559.1'
getNCBIProtein2CARD(c('AAK64581', 'WP_010896559.1'), exhaustiveMapping = TRUE)
```

---

## getNCBIProtein2KEGG *Translates a NCBI Protein database ID to KEGG database*

---

### Description

Retrieves the ID translation of an specific instance of NCBI Protein database ID to the KEGG Genes database.

### Usage

```
getNCBIProtein2KEGG(ncbiId, exhaustiveMapping = FALSE, detailedMapping = FALSE,
byIdenticalProteins = TRUE, bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| ncbiId | valid NCBI Protein database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |

detailedMapping
: logical value (TRUE or FALSE) indicating to return a vector of KEGG IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters.

byIdenticalProteins
: logical value (TRUE or FALSE) indicating to translate also through NCBI Identical Proteins. TRUE value could mean higher translation time and FALSE, fewer chances to translate the ID

bySimilarGenes
: logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible.

## Value

A character vector of the same length as ncbiId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate NCBI Protein ID 'AFU38919' to KEGG
getNCBIProtein2KEGG('AFU38919', exhaustiveMapping = FALSE,
detailedMapping = FALSE, byIdenticalProteins = FALSE, bySimilarGenes = FALSE)

# Translate NCBI Protein IDs 'AFU38919' and 'WP_010896559.1' to KEGG
getNCBIProtein2KEGG(c('AFU38919', 'WP_010896559.1'),
byIdenticalProteins = FALSE)

# Get all possible translations of ID 'AFU38919', 'WP_010896559.1'
# with detailed origin
getNCBIProtein2KEGG(c('AFU38919', 'WP_010896559.1'), exhaustiveMapping = TRUE,
detailedMapping = TRUE)
```

---

# getNCBIProtein2NCBIGene

*Translates a NCBI Protein database ID to NCBI Gene database*

---

## Description

Retrieves the ID translation of an specific instance of NCBI Protein database ID to the NCBI Gene database.

## Usage

```
getNCBIProtein2NCBIGene(id, exhaustiveMapping=FALSE)
```

### Arguments

| | |
|---|---|
| id | valid NCBI Protein database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |

### Value

A character vector of the same length as id. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate NCBI Protein ID 'CAA79696' to NCBI Gene
getNCBIProtein2NCBIGene('CAA79696')

# Translate NCBI Protein IDs 'CAA79696' and 'WP_001082319' to NCBI Gene
getNCBIProtein2NCBIGene(c('CAA79696', 'WP_001082319'))

# Get all possible translations of IDs 'CAA79696', 'WP_001082319'
getNCBIProtein2NCBIGene(c('CAA79696', 'WP_001082319'), exhaustiveMapping=TRUE)
```

---

## getNCBIProtein2NCBINucleotide

*Translates a NCBI Protein database ID to NCBI Nucleotide database*

---

### Description

Retrieves the ID translation of an specific instance of NCBI Protein database ID to the NCBI Nucleotide database. Note that, due to the nature of NCBI Nucleotide database, a complete genome is often retrieved.

### Usage

```
getNCBIProtein2NCBINucleotide(id, exhaustiveMapping=FALSE)
```

### Arguments

| | |
|---|---|
| id | valid NCBI Protein database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |

### Value

A character vector of the same length as id. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate NCBI Protein ID 'AAS48620' to NCBI Nucleotide
getNCBIProtein2NCBINucleotide('AAS48620')

# Translate NCBI Protein IDs 'AAS48620' and 'CAA38525' to NCBI Nucleotide
getNCBIProtein2NCBINucleotide(c('AAS48620', 'CAA38525'))

# Get all possible translations of IDs 'AAS48620', 'CAA38525'
getNCBIProtein2NCBINucleotide(c('AAS48620', 'CAA38525'), exhaustiveMapping=TRUE)
```

---

# getNCBIProtein2UniProt

*Translates a NCBI Protein database ID to UniProt database*

---

## Description

Retrieves the ID translation of an specific instance of NCBI Protein database ID to the UniProt database.

## Usage

```
getNCBIProtein2UniProt(ncbiId, exhaustiveMapping = FALSE,
detailedMapping = FALSE, byIdenticalProteins = TRUE)
```

## Arguments

| | |
|---|---|
| `ncbiId` | valid NCBI Protein database ID or a vector of them |
| `exhaustiveMapping` | logical value (`TRUE` or `FALSE`) indicating to retrieve all possible translations or only the first and fastest one. |
| `detailedMapping` | logical value (`TRUE` or `FALSE`) indicating to return a vector of UniProt IDs if `FALSE`, or a named vector if `TRUE`. This named vector details the origin of the translated IDs, dividing them in `DT`, if it is a Direct Translation, or `1.0`, `0.9` or `0.5`, if it has been retrieved through any of the identical gene clusters. |
| `byIdenticalProteins` | logical value (`TRUE` or `FALSE`) indicating to translate also through NCBI Identical Proteins. `TRUE` value could mean higher translation time and `FALSE`, fewer chances to translate the ID |

## Value

A character vector of the same length as `ncbiId`. If parameter `exhaustiveMapping` is `TRUE`, it will return a list instead of a character vector to avoid mixing the identifiers.

## Examples

```
# Translate NCBI Protein ID 'WP_010896559.1' to UniProt
getNCBIProtein2UniProt('WP_010896559.1')

# Translate NCBI Protein IDs 'AAS48620' and 'WP_010896559' to UniProt
getNCBIProtein2UniProt(c('AAS48620', 'WP_010896559'))

# Get all possible translations of IDs 'AAS48620', 'WP_010896559'
# with detailed origin
getNCBIProtein2UniProt(c('AAS48620', 'WP_010896559'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

# getNCBIVersion *Prints NCBI databases versions being used*

---

## Description

Prints the build version and release date of the Protein, Gene, Nucleotide and Identical Proteins NCBI databases.

## Usage

```
getNCBIVersion()
```

## Value

NCBI databases versions.

---

# getUniProt2CARD *Translates an UniProt database ID to CARD database*

---

## Description

Retrieves the ID translation of an specific instance of UniProt database ID to the CARD database.

## Usage

```
getUniProt2CARD(upId, exhaustiveMapping = FALSE, detailedMapping = FALSE,
bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| upId | valid UniProt database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of CARD IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as upId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate UniProt ID 'A0A1S7BGS4' to CARD
getUniProt2CARD('A0A1S7BGS4')

# Translate UniProt IDs 'A0A1S7BGS4' and 'A0A6H2TXZ6' to CARD
getUniProt2CARD(c('A0A1S7BGS4', 'A0A6H2TXZ6'))

# Get all possible translations of IDs 'A0A1S7BGS4', 'A0A6H2TXZ6'
# with detailed origin
getUniProt2CARD(c('A0A1S7BGS4', 'A0A6H2TXZ6'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

## getUniProt2KEGG *Translates an UniProt database ID to KEGG database*

---

### Description

Retrieves the ID translation of an specific instance of UniProt database ID to the KEGG Genes database.

### Usage

```
getUniProt2KEGG(upId, exhaustiveMapping = FALSE, bySimilarGenes = TRUE,
detailedMapping = FALSE)
```

### Arguments

| | |
|---|---|
| upId | valid UniProt database ID or a vector of them. |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of KEGG IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |

### Value

A character vector of the same length as upId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate UniProt ID 'G9JVE6' to KEGG
getUniProt2KEGG('G9JVE6')

# Translate UniProt IDs 'A0A1S7BGS4' and 'G9JVE6' to KEGG
getUniProt2KEGG(c('A0A1S7BGS4', 'G9JVE6'))

# Get all possible translations of ID 'A0A2R4PHC7' with detailed origin
getUniProt2KEGG(c('test_id', 'A0A2R4PHC7'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

## getUniProt2NCBIGene *Translates an UniProt database ID to NCBI Gene database*

---

### Description

Retrieves the ID translation of an specific instance of UniProt database ID to the NCBI Gene database.

### Usage

```
getUniProt2NCBIGene(upId, exhaustiveMapping = FALSE, detailedMapping = FALSE,
bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| upId | valid UniProt database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of NCBI Gene IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as upId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate UniProt ID 'G9JVE6' to NCBI Gene
getUniProt2NCBIGene('G9JVE6')

# Translate UniProt IDs 'A0SNL9' and 'G9JVE6' to NCBI Gene
getUniProt2NCBIGene(c('A0SNL9', 'G9JVE6'))

# Get all possible translations of IDs 'A0SNL9', 'G9JVE6' with detailed origin
getUniProt2NCBIGene(c('A0SNL9', 'G9JVE6'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

## getUniProt2NCBINucleotide

*Translates an UniProt database ID to NCBI Nucleotide database*

---

### Description

Retrieves the ID translation of an specific instance of UniProt database ID to the NCBI Nucleotide database. Note that, due to the nature of NCBI Nucleotide database, a complete genome is often retrieved.

### Usage

```
getUniProt2NCBINucleotide(upId, exhaustiveMapping = FALSE,
detailedMapping = FALSE, bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| upId | valid UniProt database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of NCBI Nucleotide IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as upId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate UniProt ID 'G9JVE6' to NCBI Nucleotide
getUniProt2NCBINucleotide('G9JVE6')

# Translate UniProt IDs 'A0SNL9' and 'G9JVE6' to NCBI Nucleotide
getUniProt2NCBINucleotide(c('A0SNL9', 'G9JVE6'))

# Get all possible translations of IDs 'A0SNL9', 'G9JVE6' with detailed origin
getUniProt2NCBINucleotide(c('A0SNL9', 'G9JVE6'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

## getUniProt2NCBIProtein

*Translates an UniProt database ID to NCBI Protein database*

---

### Description

Retrieves the ID translation of an specific instance of UniProt database ID to the NCBI Protein database.

### Usage

```
getUniProt2NCBIProtein(upId, exhaustiveMapping = FALSE, detailedMapping = FALSE,
bySimilarGenes = TRUE)
```

### Arguments

| | |
|---|---|
| upId | valid UniProt database ID or a vector of them |
| exhaustiveMapping | logical value (TRUE or FALSE) indicating to retrieve all possible translations or only the first and fastest one. |
| detailedMapping | logical value (TRUE or FALSE) indicating to return a vector of NCBI Protein IDs if FALSE, or a named vector if TRUE. This named vector details the origin of the translated IDs, dividing them in DT, if it is a Direct Translation, or 1.0, 0.9 or 0.5, if it has been retrieved through any of the identical gene clusters. |
| bySimilarGenes | logical value (TRUE or FALSE) indicating to translate also through similar UniProt genes (100%, 90% or 50% of identity). Using TRUE value may take a long time, only advised if there is not a direct translation possible. |

### Value

A character vector of the same length as upId. If parameter exhaustiveMapping is TRUE, it will return a list instead of a character vector to avoid mixing the identifiers.

### Examples

```
# Translate UniProt ID 'G9JVE6' to NCBI Protein
getUniProt2NCBIProtein('G9JVE6')

# Translate UniProt IDs 'A0SNL9' and 'G9JVE6' to NCBI Protein
getUniProt2NCBIProtein(c('A0SNL9', 'G9JVE6'))

# Get all possible translations of IDs 'A0SNL9', 'G9JVE6' with detailed origin
getUniProt2NCBIProtein(c('A0SNL9', 'G9JVE6'), detailedMapping = TRUE,
exhaustiveMapping = TRUE)
```

---

## getUniProtSimilarGenes

*Get similar UniProt genes by a given identity percentage*

---

### Description

Retrieves the cluster of genes with a certain percentage of identity where a given UniProt gene belongs.

### Usage

```
getUniProtSimilarGenes(upId, clusterIdentity = '1.0', clusterNames = FALSE)
```

### Arguments

| | |
|---|---|
| upId | valid UniProt ID or a vector of them |
| clusterIdentity | identity of the cluster to be retrieved. Default value 1.0 returns the 100% identical genes cluster, values 0.9 and 0.5, the 90% and 50% identical genes clusters, respectively |
| clusterNames | logical value (TRUE or FALSE) indicating if cluster names are retrieved |

### Value

A list of the same length as upId. If parameter clusterNames is TRUE it will return a named list.

### Examples

```
# Get UniProt IDs of 50% identical genes of UniProt gene 'G0L217'
getUniProtSimilarGenes('G0L217', clusterIdentity = '0.5')

# Get UniProt IDs of 50% identical genes of UniProt genes 'G0L217' and 'A0SNL9'
getUniProtSimilarGenes(c('G0L217', 'A0SNL9'), clusterIdentity = '0.5')
```

---

## getUniProtVersion *Prints UniProt database version being used*

---

### Description

Prints the accessed UniProt version and its corresponding release date.

### Usage

```
getUniProtVersion()
```

### Value

UniProt database version.

`updateCARDDataBase` *Downloads the most up-to-date version of CARD database*

## Description

Cleans previous CARD database versions if there were, and downloads its latest version.

## Usage

```
updateCARDDataBase()
```

## Value

Downloaded CARD database.

# Index

changeCARDPath, 2

getCARD2KEGG, 3
getCARD2NCBIGene, 4
getCARD2NCBINucleotide, 5
getCARD2NCBIProtein, 5
getCARD2UniProt, 6
getCARDVersion, 7
getKEGG2CARD, 7
getKEGG2NCBIGene, 8
getKEGG2NCBINucleotide, 9
getKEGG2NCBIProtein, 10
getKEGG2UniProt, 11
getKEGGVersion, 12
getNCBIGene2CARD, 13
getNCBIGene2KEGG, 13
getNCBIGene2NCBINucleotide, 14
getNCBIGene2NCBIProtein, 15
getNCBIGene2UniProt, 16
getNCBIIdenticalProteins, 17
getNCBINucleotide2CARD, 18
getNCBINucleotide2KEGG, 19
getNCBINucleotide2NCBIGene, 20
getNCBINucleotide2NCBIProtein, 21
getNCBINucleotide2UniProt, 21
getNCBIProtein2CARD, 22
getNCBIProtein2KEGG, 23
getNCBIProtein2NCBIGene, 24
getNCBIProtein2NCBINucleotide, 25
getNCBIProtein2UniProt, 26
getNCBIVersion, 27
getUniProt2CARD, 27
getUniProt2KEGG, 28
getUniProt2NCBIGene, 29
getUniProt2NCBINucleotide, 30
getUniProt2NCBIProtein, 31
getUniProtSimilarGenes, 32
getUniProtVersion, 33

updateCARDDataBase, 34