DO.db

April 2, 2013

DOANCESTOR *Annotation of DO Identifiers to their Ancestors*

Description

This data set describes associations between DO terms and their ancestor terms, based on the directed acyclic graph (DAG) defined by the Disease Ontology Consortium. The format is an R object mapping the DO terms to all ancestor terms, where an ancestor term is a more general DO term that precedes the given DO term in the DAG (in other words, the parents, and all their parents, etc.).

Details

Each DO term is mapped to a vector of ancestor DO terms.

Mappings were based on data provided by: Disease Ontology With a date stamp from the source of: (sub_version)

References

http://do-wiki.nubic.northwestern.edu/index.php/Main_Page

Examples

```
# Convert the object to a list
xx <- as.list(DOANCESTOR)
# Remove DO IDs that do not have any ancestor
xx <- xx[!is.na(xx)]
if(length(xx) > 0){
  # Get the ancestor DO IDs for the first two elents of xx
  doids <- xx[1:2]
}
```

DO.db *Bioconductor annotation data package*

Description

Welcome to the DO.db annotation Package. The purpose of this package is to provide detailed information about the latest version of the Disease Ontology. This package is updated biannually.

You can learn what objects this package supports with the following command:

`ls("package:DO.db")`

Each of these objects has their own manual page detailing where relevant data was obtained along with some examples of how to use it.

Examples

```
ls("package:DO.db")
```

DOCHILDREN *Annotation of DO Identifiers to their Children*

Description

This data set describes associations between DO terms and their direct children terms, based on the directed acyclic graph (DAG) defined by the Disease Ontology Consortium. The format is an R object mapping the DO terms to all direct children terms, where a direct child term is a more specific DO term that is immediately preceded by the given DO term in the DAG.

Details

Each DO term is mapped to a vector of children DO terms.

Mappings were based on data provided by: Disease Ontology With a date stamp from the source of: (sub_version)

References

http://do-wiki.nubic.northwestern.edu/index.php/Main_Page

Examples

```
# Convert the object to a list
xx <- as.list(DOCHILDREN)
# Remove DO IDs that do not have any children
xx <- xx[!is.na(xx)]

if(length(xx) > 0){
  # Get the parent DO IDs for the first elents of xx
    doids <- xx[[1]]
    # Find out the DO terms for the first parent doid
    DOID(DOTERM[[doids[1]]])
    Term(DOTERM[[doids[1]]])
```

```
    Synonym(DOTERM[[doids[1]]])
    Secondary(DOTERM[[doids[1]]])
}
```

DO_dbconn *Collect information about the package annotation DB*

Description

Some convenience functions for getting a connection object to (or collecting information about) the package annotation DB.

Usage

```
DO_dbconn()
DO_dbfile()

DO_dbschema()
DO_dbInfo()
```

Details

DO_dbconn returns a connection object to the package annotation DB. IMPORTANT: Don't call dbDisconnect on the connection object returned by DO_dbconn or you will break all the AnnDbObj objects defined in this package!

DO_dbfile returns the path (character string) to the package annotation DB (this is an SQLite file).

DO_dbschema prints the schema definition of the package annotation DB.

DO_dbInfo prints other information about the package annotation DB.

Value

DO_dbconn: a DBIConnection object representing an open connection to the package annotation DB.

DO_dbfile: a character string with the path to the package annotation DB.

DO_dbschema: none (invisible NULL).

DO_dbInfo: none (invisible NULL).

See Also

dbGetQuery, dbConnect, dbconn, dbfile, dbschema, dbInfo

Examples

```
## Count the number of rows in the "do_term" table:
dbGetQuery(DO_dbconn(), "SELECT COUNT(*) FROM do_term")

## The connection object returned by DO_dbconn() was
## created with:
dbConnect(SQLite(), dbname=DO_dbfile(), cache_size=64000,
synchronous=0)
```

```
 #DO_dbschema()

# DO_dbInfo()
```

DOMAPCOUNTS *Number of mapped keys for the maps in package DO.db*

Description

DOMAPCOUNTS provides the "map count" (i.e. the count of mapped keys) for each map in package DO.db.

Details

This "map count" information is precalculated and stored in the package annotation DB. This allows some quality control and is used by the checkMAPCOUNTS function defined in AnnotationDbi to compare and validate different methods (like `count.mappedkeys(x)` or `sum(!is.na(as.list(x)))`) for getting the "map count" of a given map.

See Also

mappedkeys, count.mappedkeys, checkMAPCOUNTS

Examples

```
DOMAPCOUNTS
mapnames <- names(DOMAPCOUNTS)
DOMAPCOUNTS[mapnames[1]]
x <- get(mapnames[1])
sum(!is.na(as.list(x)))
count.mappedkeys(x)   # much faster!

## Check the "map count" of all the maps in package DO.db
#checkMAPCOUNTS("DO.db")
```

DOOBSOLETE *Annotation of DO identifiers by terms defined by Disease Ontology Consortium and their status are obsolete*

Description

This is an R object mapping DO identifiers to the specific terms in defined by Disease Ontology Consortium and their definition are obsolete

Details

All the obsolete DO terms that are collected in this index will no longer exist in other mapping objects.

Mappings were based on data provided by: Disease Ontology With a date stamp from the source of: (sub_version)

References

http://do-wiki.nubic.northwestern.edu/index.php/Main_Page

Examples

```
    # Convert the object to a list
    xx <- as.list(DOOBSOLETE)
    if(length(xx) > 0){
        # Get the TERMS for the first elent of xx
        DOID(xx[[1]])
        Term(xx[[1]])
    }
```

DOOFFSPRING *Annotation of DO Identifiers to their Offspring*

Description

This data set describes associations between DO terms and their offspring terms, based on the directed acyclic graph (DAG) defined by the Disease Ontology Consortium. The format is an R object mapping the DO terms to all offspring terms, where an ancestor term is a more specific DO term that is preceded by the given DO term in the DAG (in other words, the children and all their children, etc.).

Details

Each DO term is mapped to a vector of offspring DO terms.

Mappings were based on data provided by: Disease Ontology With a date stamp from the source of: (sub_version)

References

http://do-wiki.nubic.northwestern.edu/index.php/Main_Page

Examples

```
  # Convert the object to a list
  xx <- as.list(DOOFFSPRING)
  # Remove DO IDs that do not have any offspring
  xx <- xx[!is.na(xx)]
   if(length(xx) > 0){
    # Get the offspring DO identifiers for the first two elents of xx
    doidentifiers <- xx[1:2]
  }
```

DOPARENTS *Annotation of DO Identifiers to their Parents*

Description

This data set describes associations between DO terms and their direct parent terms, based on the directed acyclic graph (DAG) defined by the Disease Ontology Consortium. The format is an R object mapping the DO terms to all direct parent terms, where a direct parent term is a more general DO term that immediately precedes the given DO term in the DAG.

Details

Each DO term is mapped to a named vector of DO terms. The name associated with the parent term will be either *isa*, *partof*, where *isa* indicates that the child term is a more specific version of the parent, and *partof* indicate that the child term is a part of the parent.

Mappings were based on data provided by: Disease Ontology With a date stamp from the source of: (sub_version)

References

http://do-wiki.nubic.northwestern.edu/index.php/Main_Page

Examples

```
# Convert the object to a list
xx <- as.list(DOPARENTS)
# Remove DO IDs that do not have any parent
xx <- xx[!is.na(xx)]
if(length(xx) > 0){
  doids <- xx[[1]]
  # Find out the DO terms for the first parent do ID
  DOID(DOTERM[[doids[1]]])
  Term(DOTERM[[doids[1]]])
  Synonym(DOTERM[[doids[1]]])
  Secondary(DOTERM[[doids[1]]])
}
```

DOSYNONYM *Map from DO synonyms to DO terms*

Description

DOSYNONYM is an R object that provides mapping from DO synonyms to DO terms

Details

Mappings were based on data provided by: Disease Ontology With a date stamp from the source of: (sub_version)

References

http://do-wiki.nubic.northwestern.edu/index.php/Main_Page

Examples

```
x <- DOSYNONYM
sample(x, 3)
# DO ID "DOID:8757" has a lot of synonyms
DOTERM[["DOID:8757"]]
# DO ID "DOID:9134" is a synonym of DO ID "DOID:8757"
DOID(DOSYNONYM[["DOID:9134"]])
```

DOTERM *Annotation of DO Identifiers to DO Terms*

Description

This data set gives mappings between DO identifiers and their respective terms.

Details

Each DO identifier is mapped to a `DOTerms` object that has 4 slots: DOID: DO Identifier; Term: The term for that DO id; Secondary: Secondary terms that have been merged into this term; Synonym: other ontology terms that are considered to be synonymous to the primary term attached to the DO id

Mappings were based on data provided by: Disease Ontology With a date stamp from the source of: (sub_version)

References

http://do-wiki.nubic.northwestern.edu/index.php/Main_Page

Examples

```
# Convert the object to a list
FirstTenDOBimap <- DOTERM[1:10] ##grab the 1st ten
xx <- as.list(FirstTenDOBimap)
 if(length(xx) > 0){
    # Get the TERMS for the 2nd element of xx
    DOID(xx[[2]])
    Term(xx[[2]])
    Synonym(xx[[2]])
    Secondary(xx[[2]])
}
```

DOTermsAnnDbBimap *Class "DOTermsAnnDbBimap"*

Description

A sub-class of Bimap, specific for DO.db. Please see Bimap for details.

See Also

Bimap, DOTerms

Examples

```
class(DOTERM)
```

DOTerms-class *Class "DOTerms"*

Description

A class to represent Disease Ontology nodes

Objects from the Class

Objects can be created by calls of the form `DOTerms(DOId, term, synonym, secondary)`. DOId, term are required.

Slots

`DOID`**:** Object of class `"character"` A character string for the DO id of a primary node.

`Term`**:** Object of class `"character"` A character string that defines the role of gene product corresponding to the primary DO id.

`Synonym`**:** Object of class `"character"` other ontology terms that are considered to be synonymous to the primary term attached to the DO id. Synonymous here can mean that the synonym is an exact synonym of the primary term, is related to the primary term, is broader than the primary term, is more precise than the primary term, or name is related to the term, but is not exact, broader or narrower.

`Secondary`**:** Object of class `"character"` DO ids that are secondary to the primary DO id as results of merging DO terms so that One DO id becomes the primary DO id and the rest become the secondary.

Methods

DOID `signature(object = "DOTerms")`: The get method for slot DOID.

Term `signature(object = "DOTerms")`: The get method for slot Term.

Synonym `signature(object = "DOTerms")`: The get method for slot Synonym.

Secondary `signature(object = "DOTerms")`: The get method for slot Secondary.

show `signature(x = "DOTerms")`: The method for pretty print.

Note

DOTerms objects are used to represent primary DO nodes in the SQLite-based annotation data package DO.db

References

http://do-wiki.nubic.northwestern.edu/index.php/Main_Page

Examples

```
  DOnode <- new("DOTerms", DOID="DOID:1234567", Term="Test")
  DOID(DOnode)
  Term(DOnode)

  ##Or you can just use these methods on a DOTermsAnnDbBimap
## Not run: ##I want to show an ex., but don't want to require DO.db
  require(DO.db)
  FirstTenDOBimap <- DOTERM[1:10] ##grab the 1st ten
  Term(FirstTenDOBimap)

  ##Or you can just use DO IDs directly
  ids = keys(FirstTenDOBimap)
  Term(ids)

## End(Not run)
```

Index