org.Hs.ipi.db

October 2, 2015

`org.Hs.ipi.db` *annotation data package*

Description

Welcome to the org.Hs.ipi.db annotation Package. The annotation package was built using a downloadable R package - PAnnBuilder (download and build your own). The purpose is to provide detailed information about the proteins in IPI database: ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz Homo sapiens 3.75, 19 Aug 2010

Each of these objects has their own manual page detailing where relevant data was obtained along with examples of how to use it. Many of these objects also have a reverse map available. When this is true, expect to usually find relevant information on the same manual page as the forward map.

Examples

```
# You can learn what objects this package supports with the following command:
ls("package:org.Hs.ipi.db")
```

`org.Hs.ipiDE` *Maps protein identifier to textural descriptions*

Description

org.Hs.ipiDE maps protein identifiers to their descriptive information.

Details

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiDE
  # Get the protein identifiers that are mapped to textural descriptions.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
# Get the values for a few keys
if(length(xx) >= 3){
xx[1:3]
}
}
```

`org.Hs.ipiGENEID` *Map protein identifier to Entrez gene identifier*

Description

org.Hs.ipiGENEID maps protein identifiers to Entrez Gene identifiers.

Details

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiGENEID
  # Get the protein identifiers that are mapped to Entrez gene identifier.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
# Get the values for a few keys
if(length(xx) >= 3){
xx[1:3]
}
}
```

org.Hs.ipiGI *Map protein identifier to the NCBI Protein GI*

Description

org.Hs.ipiGI maps protein identifiers to the NCBI Protein GI identifiers.

Details

Each protein identifier maps to the NCBI Protein GI identifiers.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
  x <- org.Hs.ipiGI
  # Get the protein identifiers that are mapped to the NCBI Protein GI.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

org.Hs.ipiGO *Map protein identifier to GO*

Description

org.Hs.ipiGO maps protein identifiers to Gene Ontology identifiers .

Details

Each Protein identifier is mapped to a list. Each component contain : GO ID, Evidence and Ontology (C,F,P). NAs are assigned to probe identifiers that can not be mapped to any Gene Ontology information.

The Evidence element contains a code indicating what kind of evidence supports the association of the GO id to the protein id. The evidence codes in use include: IMP: inferred from mutant phenotype IGI: inferred from genetic interaction IPI: inferred from physical interaction ISS: inferred from sequence similarity IDA: inferred from direct assay IEP: inferred from expression pattern IEA: inferred from electronic annotation TAS: traceable author statement NAS: non-traceable author statement ND: no biological data available IC: inferred by curator

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
  x <- org.Hs.ipiGO
  # Get the protein identifiers that are mapped to Gene Ontology annotation.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}

# For the reverse map:
  xx <- as.list(org.Hs.ipiGO2IPIID)
  if(length(xx) > 0){
      goids <- xx[2:3]
  }
```

`org.Hs.ipiINTERPRO` *Map protein identifier to Interpro doamins*

Description

org.Hs.ipiINTERPRO maps protein identifiers to Interpro identifiers .

Details

Each protein identifier maps to a vector of InterPro identifiers.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiINTERPRO
  # Get the protein identifiers that are mapped to Interpro doamins.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

org.Hs.ipiIPIAC *Map protein identifier to IPI Primary accession number*

Description

org.Hs.ipiIPIAC maps protein identifiers to IPI Primary Accession Number.

Details

Each protein identifier maps to a IPI primary accession number (eg: IPI00000045).

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiIPIAC
  # Get the protein identifiers that are mapped to IPI Primary accession number.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}

  # For the reverse map:
  xx <- as.list(org.Hs.ipiIPIAC2IPIID)
  if(length(xx) > 0){
      goids <- xx[2:3]
  }
```

org.Hs.ipiIPIACs *Map protein identifier to IPI accession numbers*

Description

org.Hs.ipiIPIACs maps protein identifiers to IPI accession numbers.

Details

Each protein identifier maps to a vector of IPI accession numbers (eg: IPI00000045, IPI00004985).

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiIPIACs
  # Get the protein identifiers that are mapped to IPI accession numbers.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

`org.Hs.ipiKEGG` *Map protein identifier to KEGG gene identifiers*

Description

org.Hs.ipiKEGG maps protein identifiers to KEGG gene identifiers.

Details

Each protein identifier maps to KEGG gene identifiers.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiKEGG
  # Get the protein identifiers that are mapped to KEGG gene identifiers.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

`org.Hs.ipiLEN` *Map protein identifier to the length of protein sequence*

Description

org.Hs.ipiLEN maps protein identifiers to the length of protein sequence.

Details

Each protein identifier maps to the length of protein sequence.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
  x <- org.Hs.ipiLEN
  # Get the protein identifiers that are mapped to the length of protein sequence.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

---

org.Hs.ipiMAPCOUNTS *Number of mapped keys for the maps in package org.Hs.ipi.db*

Description

org.Hs.ipiMAPCOUNTS provides the "map count" (i.e. the count of mapped keys) for each map in package org.Hs.ipi.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

```
org.Hs.ipiMAPCOUNTS
mapnames <- names(org.Hs.ipiMAPCOUNTS)
org.Hs.ipiMAPCOUNTS[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 org.Hs.ipi.db
checkMAPCOUNTS("org.Hs.ipi.db")
```

org.Hs.ipiMW *Map protein identifier to its molecular weight*

Description

org.Hs.ipiMW maps protein identifiers to its molecular weight.

Details

Each protein identifier maps to its molecular weight.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiMW
  # Get the protein identifiers that are mapped to its molecular weight.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

org.Hs.ipiORGANISM *The Organism for org.Hs.ipi.db*

Description

org.Hs.ipiORGANISM is an R object that contains a single item: a character string that names the organism for which org.Hs.ipi.db was built.

Details

Although the package name is suggestive of the organism for which it was built, org.Hs.ipiORGANISM provides a simple way to programmatically extract the organism name.

Examples

```
org.Hs.ipiORGANISM
```

org.Hs.ipiPATH *Map protein identifier to KEGG pathway*

Description

org.Hs.ipiPATH maps protein identifiers to KEGG pathway identifiers.

Details

Each protein identifier maps to KEGG pathway identifiers.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiPATH
  # Get the protein identifiers that are mapped to KEGG pathway.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}

# For the reverse map:
  xx <- as.list(org.Hs.ipiPATH2IPIID)
  if(length(xx) > 0){
      goids <- xx[2:3]
  }
```

org.Hs.ipiPFAM *Map protein identifier to Pfam domain*

Description

org.Hs.ipiPFAM maps protein identifiers to Pfam domain identifiers.

Details

Each protein identifier maps to a vector of Pfam domain identifiers.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiPFAM
  # Get the protein identifiers that are mapped to Pfam domain.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

`org.Hs.ipiPROSITE` *Map protein identifier to PROSITE domain*

Description

org.Hs.ipiPROSITE maps protein identifiers to PROSITE domain identifiers.

Details

Each protein identifier maps to a vector of PROSITE domain identifiers.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiPROSITE
  # Get the protein identifiers that are mapped to PROSITE domain.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

`org.Hs.ipiREFSEQ` *Map protein identifier to RefSeq identifiers*

Description

org.Hs.ipiREFSEQ maps protein identifiers to RefSeq identifiers.

Details

Each protein identifier maps to RefSeq identifiers.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiREFSEQ
  # Get the protein identifiers that are mapped to the RefSeq identifiers.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}

# For the reverse map:
  xx <- as.list(org.Hs.ipiREFSEQ2IPIID)
  if(length(xx) > 0){
      goids <- xx[2:3]
  }
```

---

org.Hs.ipiSEQ *Map protein identifier to the protein sequence*

Description

org.Hs.ipiSEQ provides mappings between a protein identifier and the protein Sequence.

Details

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiSEQ
# Get the protein identifiers that are mapped to protein sequence.
mapped_proteins <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_proteins])
# randomly display 10 proteins
sample(xx, 10)
```

org.Hs.ipiSPAC *Map protein identifier to SwissProt primary accession number*

Description

org.Hs.ipiSPAC maps protein identifiers to SwissProt primary accession number.

Details

Each protein identifier maps to SwissProt primary accession number.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiSPAC
  # Get the protein identifiers that are mapped to SwissProt primary accession number.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}

# For the reverse map:
  xx <- as.list(org.Hs.ipiSPAC2IPIID)
  if(length(xx) > 0){
      goids <- xx[2:3]
  }
```

org.Hs.ipiSPID *Map protein identifier to SwissProt identifiers*

Description

org.Hs.ipiSPID maps protein identifiers to SwissProt identifiers.

Details

Each protein identifier maps to SwissProt identifiers.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiSPID
  # Get the protein identifiers that are mapped to SwissProt identifiers.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}

# For the reverse map:
  xx <- as.list(org.Hs.ipiSPID2IPIID)
  if(length(xx) > 0){
      goids <- xx[2:3]
  }
```

---

org.Hs.ipiSYMBOL *Map protein identifier to gene symbols*

Description

org.Hs.ipiSYMBOL maps protein identifiers to gene symbols.

Details

Each protein identifier maps to an abbreviation for the corresponding gene. .

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiSYMBOL
  # Get the protein identifiers that are mapped to gene symbols.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

org.Hs.ipiUNIGENE *Map protein identifier to UniGene cluster identifiers*

Description

org.Hs.ipiUNIGENE maps protein identifiers to UniGene cluster identifiers.

Details

Each protein identifier maps to UniGene cluster identifiers. A UniGene identifier represents a cluster of sequences of a gene. Using UniGene identifiers one can query the UniGene database for information about the sequences or the Entrez Gene database for information about the genes.

Mappings were based on data provided by: IPI (ftp://ftp.ebi.ac.uk/pub/databases/IPI/current/ipi.HUMAN.dat.gz) on Homo sapiens 3.75, 19 Aug 2010

Examples

```
x <- org.Hs.ipiUNIGENE
  # Get the protein identifiers that are mapped to UniGene cluster identifiers.
  mapped_proteins <- mappedkeys(x)
  # Convert to a list
  xx <- as.list(x[mapped_proteins])
if(length(xx) > 0){
# Get the value of the first key
xx[[1]]
}
```

org.Hs.ipi_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

```
org.Hs.ipi_dbconn()
org.Hs.ipi_dbfile()
org.Hs.ipi_dbschema(file="", show.indices=FALSE)
org.Hs.ipi_dbInfo()
```

Arguments

file	A connection, or a character string naming the file to print to (see the file argument of the cat function for the details).
show.indices	The CREATE INDEX statements are not shown by default. Use show.indices=TRUE to get them.

Details

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

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

org.Hs.ipi_dbschema prints the schema definition of the package annotation DB.

org.Hs.ipi_dbInfo prints other information about the package annotation DB.

Examples

```
## Show the first three rows.
dbGetQuery(org.Hs.ipi_dbconn(), "select * from basic limit 3")

## The connection object returned by org.Hs.ipi_dbconn() was created with:
dbConnect(SQLite(), dbname=org.Hs.ipi_dbfile(), cache_size=64000, synchronous=0)

org.Hs.ipi_dbschema()

org.Hs.ipi_dbInfo()
```

Index