org.Sc.sgd.db

November 11, 2026

`org.Sc.sgdALIAS` *Map Open Reading Frame (ORF) Identifiers to Alias Gene Names*

Description

A set of gene names may have been used to report yeast genes represented by ORF identifiers. One of these names is chosen to be the primary gene name, and the others are considered aliases. This R object provides mappings between the primary name and aliases.

Details

Each primary name maps to a vector of alias names. If there are no aliases, the vector will contain `NA`.

Annotation based on data provided by: Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23

References

http://www.yeastgenome.org/DownloadContents.shtml

See Also

- `AnnotationDb-class` for use of the `select()` interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdALIAS
# Get the probe identifiers that are mapped to alias names
mapped_probes <- mappedkeys(x)
# Convert to a list
```

```
xx <- as.list(x[mapped_probes])
if(length(xx) > 0) {
    # Get the alias names for the first five probes
    xx[1:5]
    # For the first probe
    xx[[1]]
}
```

`org.Sc.sgd.db` *Bioconductor annotation data package*

Description

Welcome to the org.Sc.sgd.db annotation Package. This is an organism specific package. The purpose is to provide detailed information about the species abbreviated in the second part of the package name org.Sc.sgd.db. This package is updated biannually.

Objects in this package are accessed using the `select()` interface. See ?select in the AnnotationDbi package for details.

See Also

- `AnnotationDb-class` for use of `keys()`, `columns()` and `select()`.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.
columns(org.Sc.sgd.db)

## Bimap interface:
## The 'old style' of interacting with these objects is manipulation as
## bimaps. While this approach is still available we strongly encourage the
## use of select().
ls("package:org.Sc.sgd.db")
```

`org.Sc.sgdCHR` *Map ORF IDs to Chromosomes*

Description

org.Sc.sgdCHR is an R object that provides mappings between ORF identifiers and the chromosome that contains the gene of interest.

Details

Each ORF identifier maps to a vector of a chromosome.

Mappings were based on data provided by: Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdCHR
# Get the ORF identifiers that are mapped to a chromosome
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the CHR for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

---

org.Sc.sgdCHRLENGTHS *A named vector for the length of each of the chromosomes*

Description

org.Sc.sgdCHRLENGTHS provides the length measured in base pairs for each of the chromosomes.

Details

This is a named vector with chromosome numbers as the names and the corresponding lengths for chromosomes as the values.

Total lengths of chromosomes were derived by calculating the number of base pairs on the sequence string for each chromosome.

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
    tt <- org.Sc.sgdCHRLENGTHS
    # Length of chromosome 1
    tt["1"]
```

`org.Sc.sgdCHRLOC` *ORF IDs to Chromosomal Location*

Description

org.Sc.sgdCHRLOC is an R object that maps ORF identifiers to the starting position of the gene. The position of a gene is measured as the number of base pairs.

The CHRLOCEND mapping is the same as the CHRLOC mapping except that it specifies the ending base of a gene instead of the start.

Details

Each ORF identifier maps to a named vector of chromosomal locations, where the name indicates the chromosome.

Chromosomal locations on both the sense and antisense strands are measured as the number of base pairs from the p (5' end of the sense strand) to q (3' end of the sense strand) arms. Chromosomal locations on the antisense strand have a leading "-" sign (e. g. -1234567).

Since some genes have multiple start sites, this field can map to multiple locations.

Mappings were based on data provided by: Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23

See Also

- AnnotationDb-class for use of the `select()` interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdCHRLOC
# Get the ORF identifiers that are mapped to chromosome locations
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
```

```
if(length(xx) > 0) {
  # Get the CHRLOC for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

`org.Sc.sgdCOMMON2ORF` *Map Between Yeast Common Names and ORF Identifiers*

Description

org.Sc.sgdCOMMON2ORF is an R object that maps the yeast common names (gene names and aliases) to the corresponding yeast ORF identifiers.

Details

Each yeast common name, either gene name or alias, maps to a vector of ORF identifiers. This mapping is the reverse mappings of org.Sc.sgdGENENAME.

Mappings were based on data provided by:

Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23

See Also

- AnnotationDb-class for use of the `select()` interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
# Convert to a list
xx <- as.list(org.Sc.sgdCOMMON2ORF)
# Remove probes that do not map in COMMON2ORF
xx <- xx[!is.na(xx)]
if(length(xx) > 0){
    # Gets the ORF identifiers for the first five gene names/alias
    xx[1:5]
    # Get the first one
    xx[[1]]
}
```

org.Sc.sgdDESCRIPTION *An annotation data file that maps Open Reading Frame (ORF) identifiers to textural descriptions of the corresponding genes*

Description

org.Sc.sgdDESCRIPTION maps yeast ORF identifiers to descriptive information about genes corresponding to the ORF identifiers

Details

This is an R object containing key and value pairs. Keys are ORF identifiers and values are the corresponding descriptions of genes. Values are vectors of length 1. Probe identifiers that can not be mapped to descriptive information are assigned a value NA.

Annotation based on data provided by: Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23

References

http://www.yeastgenome.org/DownloadContents.shtml

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdDESCRIPTION
# Get the probe identifiers that are mapped to gene descriptions
mapped_probes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_probes])
if(length(xx) > 0) {
  # Get the gene descriptions for the first five probes
  xx[1:5]
  # For the first probe
  xx[[1]]
}
```

org.Sc.sgdENSEMBL *Map Ensembl gene accession numbers with SGD Gene identifiers*

Description

org.Sc.sgdENSEMBL is an R object that contains mappings between SGD Gene identifiers and Ensembl gene accession numbers.

Details

This object is a simple mapping of SGD Gene identifiers ftp://genome-ftp.stanford.edu/pub/yeast/data_download to Ensembl gene Accession Numbers.

Mappings were based on data provided by ALL of these sources: ftp://ftp.ensembl.org/pub/current_fasta ftp://genome-ftp.stanford.edu/pub/yeast/data_download ftp://ftp.ncbi.nlm.nih.gov/gene/DATA

This mapping is a combination of NCBI to ensembl IDs from BOTH NCBI and ensembl. These mappings are based upon the ensembl table which is contains data from BOTH of these sources in an effort to maximize the chances that you will find a match.

Mappings were based on data provided by: Ensembl ftp://ftp.ensembl.org/pub/current_fasta With a date stamp from the source of: 2025-Sep03

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdENSEMBL
# Get the SGD gene IDs that are mapped to an Ensembl ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the Ensembl gene IDs for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
#For the reverse map ENSEMBL2ORF:
# Convert to a list
xx <- as.list(org.Sc.sgdENSEMBL2ORF)
if(length(xx) > 0){
   # Gets the SGD gene IDs for the first five Ensembl IDs
```

```
   xx[1:5]
   # Get the first one
   xx[[1]]
}
```

---

org.Sc.sgdENSEMBLPROT *Map Ensembl protein acession numbers with SGD Gene identifiers*

Description

org.Sc.sgdENSEMBL is an R object that contains mappings between SGD Gene identifiers and Ensembl protein accession numbers.

Details

This object is a simple mapping of SGD Gene identifiers ftp://genome-ftp.stanford.edu/pub/yeast/data_download to Ensembl protein accession numbers.

Mappings were based on data provided by: ftp://ftp.ensembl.org/pub/current_fasta ftp://genome-ftp.stanford.edu/pub/yeast/data_download ftp://ftp.ncbi.nlm.nih.gov/gene/DATA

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdENSEMBLPROT
# Get the SGD gene IDs that are mapped to an Ensembl ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the Ensembl gene IDs for the first five proteins
  xx[1:5]
  # Get the first one
  xx[[1]]
}
#For the reverse map ENSEMBLPROT2ORF:
# Convert to a list
xx <- as.list(org.Sc.sgdENSEMBLPROT2ORF)
if(length(xx) > 0){
   # Gets the SGD gene IDs for the first five Ensembl IDs
   xx[1:5]
```

```
   # Get the first one
   xx[[1]]
}
```

---

org.Sc.sgdENSEMBLTRANS *Map Ensembl transcript acession numbers with SGD Gene identifiers*

Description

org.Sc.sgdENSEMBL is an R object that contains mappings between SGD Gene identifiers and Ensembl transcript accession numbers.

Details

This object is a simple mapping of SGD Gene identifiers ftp://genome-ftp.stanford.edu/pub/yeast/data_download to Ensembl transcript accession numbers.

Mappings were based on data provided by: ftp://ftp.ensembl.org/pub/current_fasta ftp://genome-ftp.stanford.edu/pub/yeast/data_download ftp://ftp.ncbi.nlm.nih.gov/gene/DATA

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdENSEMBLTRANS
# Get the SGD gene IDs that are mapped to an Ensembl ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the Ensembl gene IDs for the first five proteins
  xx[1:5]
  # Get the first one
  xx[[1]]
}
#For the reverse map ENSEMBLTRANS2ORF:
# Convert to a list
xx <- as.list(org.Sc.sgdENSEMBLTRANS2ORF)
if(length(xx) > 0){
   # Gets the SGD gene IDs for the first five Ensembl IDs
   xx[1:5]
```

```
  # Get the first one
  xx[[1]]
}
```

`org.Sc.sgdENTREZID` *Map Systematic ORF identifiers with Entrez Gene identifiers*

Description

org.Sc.sgdENTREZID is an R object that contains mappings between Systematic ORF accession numbers and NCBI Entrez Gene identifiers.

Details

This object is a simple mapping of Entrez Gene identifiers https://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene to systematic ORF locus Accession Numbers.

Mappings were based on data provided by: Entrez Gene ftp://ftp.ncbi.nlm.nih.gov/gene/DATA With a date stamp from the source of: 2025-Sep24

See Also

- AnnotationDb-class for use of the `select()` interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdENTREZID
# Get the ORF IDs that are mapped to an Entrez Gene ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the Entrez gene IDs for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

org.Sc.sgdENZYME *Map between ORF IDs and Enzyme Commission (EC) Numbers*

Description

org.Sc.sgdENZYME is an R object that provides mappings between ORF identifiers and EC numbers.

Details

Each ORF identifier maps to a named vector containing the EC number that corresponds to the enzyme produced by that gene. The name corresponds to the ORF identifier. If this information is unknown, the vector will contain an NA.

Enzyme Commission numbers are assigned by the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology http://www.chem.qmw.ac.uk/iubmb/enzyme/ to allow enzymes to be identified.

An Enzyme Commission number is of the format EC x.y.z.w, where x, y, z, and w are numeric numbers. In org.Sc.sgdENZYME2ORF, EC is dropped from the Enzyme Commission numbers.

Enzyme Commission numbers have corresponding names that describe the functions of enzymes in such a way that EC x is a more general description than EC x.y that in turn is a more general description than EC x.y.z. The top level EC numbers and names are listed below:

EC 1 oxidoreductases

EC 2 transferases

EC 3 hydrolases

EC 4 lyases

EC 5 isomerases

EC 6 ligases

The EC name for a given EC number can be viewed at http://www.chem.qmul.ac.uk/iupac/jcbn/index.html#6

Mappings between ORF identifiers and enzyme identifiers were obtained using files provided by: KEGG GENOME ftp://ftp.genome.jp/pub/kegg/genomes With a date stamp from the source of: 2011-Mar15

For the reverse map, each EC number maps to a named vector containing the ORF identifier that corresponds to the gene that produces that enzyme. The name of the vector corresponds to the EC number.

References

ftp://ftp.genome.ad.jp/pub/kegg/pathways

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdENZYME
# Get the ORF identifiers that are mapped to an EC number
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the ENZYME id for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
# For the reverse map:
# Convert to a list
xx <- as.list(org.Sc.sgdENZYME2ORF)
if(length(xx) > 0){
   # Gets the ORF identifiers for the first five enzyme
   #commission numbers
   xx[1:5]
   # Get the first one
   xx[[1]]
}
```

---

org.Sc.sgdGENENAME *Map between ORF IDs and Genes*

Description

org.Sc.sgdGENENAME is an R object that maps ORF identifiers to the corresponding gene name.

Details

Each ORF identifier maps to a named vector containing the gene name. The vector name corresponds to the ORF identifier. If the gene name is unknown, the vector will contain an NA.

Gene names currently include both the official (validated by a nomenclature committee) and preferred names (interim selected for display) for genes. Efforts are being made to differentiate the two by adding a name to the vector.

Mappings were based on data provided by: Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdGENENAME
# Get the gene names that are mapped to an ORF identifier
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the GENE NAME for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

`org.Sc.sgdGO` *Map between ORF IDs and Gene Ontology (GO)*

Description

org.Sc.sgdGO is an R object that provides mappings between ORF identifiers and the GO identifiers that they are directly associated with. This mapping and its reverse mapping do NOT associate the child terms from the GO ontology with the gene. Only the directly evidenced terms are represented here.

Details

Each ORF identifier is mapped to a list of lists. The names on the outer list are GO identifiers. Each inner list consists of three named elements: GOID, Ontology, and Evidence.

The GOID element matches the GO identifier named in the outer list and is included for convenience when processing the data using 'lapply'.

The Ontology element indicates which of the three Gene Ontology categories this identifier belongs to. The categories are biological process (BP), cellular component (CC), and molecular function (MF).

The Evidence element contains a code indicating what kind of evidence supports the association of the GO identifier to the Entrez Gene id. Some of 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

A more complete listing of evidence codes can be found at:

http://www.geneontology.org/GO.evidence.shtml

Mappings between ORF identifiers and GO information were obtained through their mappings to ORF identifiers. NAs are assigned to ORF identifiers that can not be mapped to any Gene Ontology information. Mappings between Gene Ontology identifiers an Gene Ontology terms and other information are available in a separate data package named GO.

Mappings were based on data provided by: Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23

For the reverse map org.Sc.sgdGO2ORF, each GO term maps to a named vector of ORF identifiers. A GO identifier may be mapped to the same ORF identifier more than once but the evidence code can be different. Mappings between Gene Ontology identifiers and Gene Ontology terms and other information are available in a separate data package named GO.

References

ftp://ftp.ncbi.nlm.nih.gov/gene/DATA/

See Also

- org.Sc.sgdGO2ALLORFS
- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdGO
# Get the ORF identifiers that are mapped to a GO ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
    # Try the first one
    got <- xx[[1]]
    got[[1]][["GOID"]]
    got[[1]][["Ontology"]]
    got[[1]][["Evidence"]]
}
# For the reverse map:
# Convert to a list
```

```
xx <- as.list(org.Sc.sgdGO2ORF)
if(length(xx) > 0){
    # Gets the ORF ids for the top 2nd and 3nd GO identifiers
    goids <- xx[2:3]
    # Gets the ORF ids for the first element of goids
    goids[[1]]
    # Evidence code for the mappings
    names(goids[[1]])
}
```

`org.Sc.sgdGO2ALLORFS` *Map Between Gene Ontology (GO) Identifiers and all ORF Identifiers in the subtree*

Description

org.Sc.sgdGO2ALLORFS is an R object that provides mappings between a given GO identifier and all ORF identifiers annotated at that GO term or one of its children in the GO ontology.

Details

GO consists of three ontologies—molecular function (MF), biological process (BP), and cellular component (CC). All ontologies are structured as directed acyclic graphs (DAGs). Each node in each DAG (tree) is a GO term (id) associated with a named vector of manufacturer identifiers. The name associated with each ORF id corresponds to the evidence code for that GO identifier. This object org.Sc.sgdGO2ALLORFS maps between a given GO identifier and all ORF identifiers annotated at that GO term or one of its children in the GO ontology.

The evidence code indicates what kind of evidence supports the association between the GO and Entrez Gene identifiers. Evidence codes currently 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

A GO identifier may be mapped to the same ORF identifier more than once but the evidence code can be different. Mappings between Gene Ontology identifiers and Gene Ontology terms and other information are available in a separate data package named GO.

Mappings were based on data provided by: Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23 and Gene Ontology http://current.geneontology.org/ontology/go-basic.obo With a date stamp from the source of: 2025-07-22

For GO2ALL style mappings, the intention is to return all the genes that are the same kind of term as the parent term (based on the idea that they are more specific descriptions of the general term). However because of this intent, not all relationship types will be counted as offspring for this mapping. Only "is a" and "has a" style relationships indicate that the genes from the child terms would be the same kind of thing.

References

ftp://ftp.ncbi.nlm.nih.gov/gene/DATA/

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
# Convert to a list
xx <- as.list(org.Sc.sgdGO2ALLORFS)
if(length(xx) > 0){
    # Gets the ORF identifiers for the top 2nd and 3nd GO identifiers
    goids <- xx[2:3]
    # Gets all the ORF identifiers for the first element of goids
    goids[[1]]
    # Evidence code for the mappings
    names(goids[[1]])
}
```

---

org.Sc.sgdINTERPRO *Map Yeast Systematic Names to InterPro IDs*

Description

org.Sc.sgdINTERPRO is an R object that provides mappings between yeast ORF identifiers and the associated InterPro identifiers.

Details

Each yeast ORF identifier maps to a vector of InterPro identifiers.

Mappings were based on data provided by:

Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23

References

InterPro website: http://www.ebi.ac.uk/interpro/

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
# Convert to a list
xxx <- as.list(org.Sc.sgdINTERPRO)
# randomly display 10 probes
sample(xxx, 10)
```

---

org.Sc.sgdMAPCOUNTS *Number of mapped keys for the maps in package org.Sc.sgd.db*

Description

org.Sc.sgdMAPCOUNTS provides the "map count" (i.e. the count of mapped keys) for each map in package org.Sc.sgd.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
- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
org.Sc.sgdMAPCOUNTS
mapnames <- names(org.Sc.sgdMAPCOUNTS)
org.Sc.sgdMAPCOUNTS[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.Sc.sgd.db
checkMAPCOUNTS("org.Sc.sgd.db")
```

`org.Sc.sgdORGANISM` *The Organism for org.Sc.sgd*

Description

org.Sc.sgdORGANISM is an R object that contains a single item: a character string that names the organism for which org.Sc.sgd was built.

Details

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

See Also

- AnnotationDb-class for use of the `select()` interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
org.Sc.sgdORGANISM
```

org.Sc.sgdPATH *Mappings between ORF identifiers and KEGG pathway identifiers*

Description

KEGG (Kyoto Encyclopedia of Genes and Genomes) maintains pathway data for various organisms. org.Sc.sgdPATH maps ORF identifiers to the identifiers used by KEGG for pathways. The reverse map org.Sc.sgdPATH2ORF maps back from KEGG pathway IDs to the ORF identifiers.

Details

Each KEGG pathway has a name and identifier. Pathway name for a given pathway identifier can be obtained using the KEGG data package that can either be built using AnnBuilder or downloaded from Bioconductor http://www.bioconductor.org.

Graphic presentations of pathways are searchable at url http://www.genome.ad.jp/kegg/pathway.html by using pathway identifiers as keys.

Mappings were based on data provided by: KEGG GENOME ftp://ftp.genome.jp/pub/kegg/genomes With a date stamp from the source of: 2011-Mar15

References

http://www.genome.ad.jp/kegg/

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdPATH
# Get the ORF identifiers that are mapped to a KEGG pathway ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the PATH for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
# Convert the object to a list
xx <- as.list(org.Sc.sgdPATH2ORF)
# Remove pathway identifiers that do not map to any ORF id
```

```
xx <- xx[!is.na(xx)]
if(length(xx) > 0){
    # The ORF identifiers for the first two elements of XX
    xx[1:2]
    # Get the first one
    xx[[1]]
}
```

---

org.Sc.sgdPMID *Map between ORF Identifiers and PubMed Identifiers*

Description

org.Sc.sgdPMID is an R object that provides mappings between ORF identifiers and PubMed identifiers.

Details

Each ORF identifier is mapped to a named vector of PubMed identifiers. The name associated with each vector corresponds to the ORF identifier. The length of the vector may be one or greater, depending on how many PubMed identifiers a given ORF identifier is mapped to. An NA is reported for any ORF identifier that cannot be mapped to a PubMed identifier.

Titles, abstracts, and possibly full texts of articles can be obtained from PubMed by providing a valid PubMed identifier. The pubmed function of annotate can also be used for the same purpose.

Mappings were based on data provided by: Yeast Genome http://sgd-archive.yeastgenome.org With a date stamp from the source of: 2025-Sep23

References

https://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdPMID
# Get the ORF identifiers that are mapped to any PubMed ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0){
```

```
    # The ORF identifiers for the first two elements of XX
    xx[1:2]
    # Get the first one
    xx[[1]]
    if(interactive() && !is.null(xx[[1]]) && !is.na(xx[[1]])
       && require(annotate)){
        # Gets article information as XML files
        xmls <- pubmed(xx[[1]], disp = "data")
        # Views article information using a browser
        pubmed(xx[[1]], disp = "browser")
    }
}
# Convert the object to a list
xx <- as.list(org.Sc.sgdPMID2ORF)
if(length(xx) > 0){
    # The ORF identifiers for the first two elements of XX
    xx[1:2]
    # Get the first one
    xx[[1]]
    if(interactive() && require(annotate)){
        # Gets article information as XML files for a PubMed id
        xmls <- pubmed(names(xx)[1], disp = "data")
        # Views article information using a browser
        pubmed(names(xx)[1], disp = "browser")
    }
}
```

`org.Sc.sgdREFSEQ` *Map between systematic ORF identifiers and RefSeq Identifiers*

Description

org.Sc.sgdREFSEQ is an R object that provides mappings between systematic ORF identifiers and RefSeq identifiers.

Details

Each systematic ORF identifier is mapped to a named vector of RefSeq identifiers. The name represents the systematic ORF identifier and the vector contains all RefSeq identifiers that can be mapped to that entrez gene identifier. The length of the vector may be one or greater, depending on how many RefSeq identifiers a given systematic ORF identifier can be mapped to. An `NA` is reported for any entrex gene identifier that cannot be mapped to a RefSeq identifier at this time.

RefSeq identifiers differ in format according to the type of record the identifiers are for as shown below:

NG_XXXXX: RefSeq accessions for genomic region (nucleotide) records

NM_XXXXX: RefSeq accessions for mRNA records

NC_XXXXX: RefSeq accessions for chromosome records

NP_XXXXX: RefSeq accessions for protein records

XR_XXXXX: RefSeq accessions for model RNAs that are not associated with protein products

XM_XXXXX: RefSeq accessions for model mRNA records

XP_XXXXX: RefSeq accessions for model protein records

Where XXXXX is a sequence of integers.

NCBI https://www.ncbi.nlm.nih.gov/RefSeq/ allows users to query the RefSeq database using RefSeq identifiers.

Mappings were based on data provided by: Entrez Gene ftp://ftp.ncbi.nlm.nih.gov/gene/DATA With a date stamp from the source of: 2025-Sep24

References

https://www.ncbi.nlm.nih.gov https://www.ncbi.nlm.nih.gov/RefSeq/

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdREFSEQ
# Get the systematic ORF identifiers that are mapped to any RefSeq ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the REFSEQ for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

---

org.Sc.sgdREJECTORF *Rejected Yeast Genes (ORF)*

Description

This is based upon Real and rejected yeast ORFs from Kellis et al. (2003) to list the rejected genes by this criteria.

Details

A character vector which contains yeast ORFs which are rejected in the reading frame conseration (RFC) test in Kellis et al. (2003).

References

Manolis Kellis, Nick Patterson, Matthew Endrizzi, Bruce Birren and Eric S. Lander, Sequencing and comparison of yeast species to identify genes and regulatory elements. Nature 423, 241-254 (15 May 2003)

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
head(org.Sc.sgdREJECTORF)
```

---

org.Sc.sgdSGD *Map Systematic ORF identifiers with SGD accession numbers*

Description

org.Sc.sgdSGD is an R object that contains mappings between Systematic ORF accessions and SGD accession numbers.

Details

This object is a simple mapping of Systematic ORF Accession Numbers to SGD identifiers.

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdSGD
# Get the Systematic ORF Accessions that are mapped to a SGD ID
mapped_genes <- mappedkeys(x)
```

```
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the SGD gene IDs for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

---

org.Sc.sgdSMART *Map Yeast ORF Identifiers to SMART IDs*

Description

org.Sc.sgdSMART is an R object that provides mappings between yeast ORF Identifiers and the associated SMART identifiers.

Details

Each yeast systematic name maps to a vector of SMART identifiers.

Mappings were based on data provided by:

Saccharomyces Genome Database

Package built on Thu Mar 15 18:04:19 2007

References

SMART website: http://smart.embl-heidelberg.de/

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
# Convert to a list
xxx <- as.list(org.Sc.sgdSMART)
# randomly display 10 probes
sample(xxx, 10)
```

`org.Sc.sgdUNIPROT` *Map Uniprot accession numbers with Systematic ORF identifiers*

Description

org.Sc.sgdUNIPROT is an R object that contains mappings between Systematic ORF identifiers and Uniprot accession numbers.

Details

This object is a simple mapping of Systematic ORF identifiers https://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene to Uniprot Accession Numbers.

Mappings were based on data provided by: Entrez Gene ftp://ftp.ncbi.nlm.nih.gov/gene/DATA With a date stamp from the source of: 2025-Sep24

See Also

- AnnotationDb-class for use of the `select()` interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Sc.sgdUNIPROT
# Get the Systematic ORF IDs that are mapped to a Uniprot ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the Uniprot gene IDs for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

`org.Sc.sgd_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.Sc.sgd_dbconn()
org.Sc.sgd_dbfile()
org.Sc.sgd_dbschema(file="", show.indices=FALSE)
org.Sc.sgd_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.Sc.sgd_dbconn returns a connection object to the package annotation DB. IMPORTANT: Don't call dbDisconnect on the connection object returned by org.Sc.sgd_dbconn or you will break all the AnnDbObj objects defined in this package!

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

org.Sc.sgd_dbschema prints the schema definition of the package annotation DB.

org.Sc.sgd_dbInfo prints other information about the package annotation DB.

Value

org.Sc.sgd_dbconn: a DBIConnection object representing an open connection to the package annotation DB.

org.Sc.sgd_dbfile: a character string with the path to the package annotation DB.

org.Sc.sgd_dbschema: none (invisible NULL).

org.Sc.sgd_dbInfo: none (invisible NULL).

See Also

dbGetQuery, dbConnect, dbconn, dbfile, dbschema, dbInfo

Examples

```
library(DBI)
## Count the number of rows in the "sgd" table:
dbGetQuery(org.Sc.sgd_dbconn(), "SELECT COUNT(*) FROM sgd")

org.Sc.sgd_dbschema()

org.Sc.sgd_dbInfo()
```

Index

∗ **datasets**
org.Sc.sgd.db, 2
org.Sc.sgd_dbconn, 25
org.Sc.sgdALIAS, 1
org.Sc.sgdCHR, 2
org.Sc.sgdCHRLENGTHS, 3
org.Sc.sgdCHRLOC, 4
org.Sc.sgdCOMMON2ORF, 5
org.Sc.sgdDESCRIPTION, 6
org.Sc.sgdENSEMBL, 7
org.Sc.sgdENSEMBLPROT, 8
org.Sc.sgdENSEMBLTRANS, 9
org.Sc.sgdENTREZID, 10
org.Sc.sgdENZYME, 11
org.Sc.sgdGENENAME, 12
org.Sc.sgdGO, 13
org.Sc.sgdGO2ALLORFS, 15
org.Sc.sgdINTERPRO, 16
org.Sc.sgdMAPCOUNTS, 17
org.Sc.sgdORGANISM, 18
org.Sc.sgdPATH, 19
org.Sc.sgdPMID, 20
org.Sc.sgdREFSEQ, 21
org.Sc.sgdREJECTORF, 22
org.Sc.sgdSGD, 23
org.Sc.sgdSMART, 24
org.Sc.sgdUNIPROT, 25
∗ **utilities**
org.Sc.sgd_dbconn, 25

AnnDbObj, *26*

cat, *26*
checkMAPCOUNTS, *17*
count.mappedkeys, *17*

dbconn, *26*
dbConnect, *26*
dbDisconnect, *26*
dbfile, *26*
dbGetQuery, *26*
dbInfo, *26*
dbschema, *26*

mappedkeys, *17*

org.Sc.sgd *(*org.Sc.sgd.db*)*, 2
org.Sc.sgd.db, 2
org.Sc.sgd_dbconn, 25
org.Sc.sgd_dbfile *(*org.Sc.sgd_dbconn*)*, 25
org.Sc.sgd_dbInfo *(*org.Sc.sgd_dbconn*)*, 25
org.Sc.sgd_dbschema *(*org.Sc.sgd_dbconn*)*, 25
org.Sc.sgdALIAS, 1
org.Sc.sgdALIAS2ORF *(*org.Sc.sgdALIAS*)*, 1
org.Sc.sgdCHR, 2
org.Sc.sgdCHRLENGTHS, 3
org.Sc.sgdCHRLOC, 4
org.Sc.sgdCHRLOCEND *(*org.Sc.sgdCHRLOC*)*, 4
org.Sc.sgdCOMMON2ORF, 5
org.Sc.sgdDESCRIPTION, 6
org.Sc.sgdENSEMBL, 7
org.Sc.sgdENSEMBL2ORF *(*org.Sc.sgdENSEMBL*)*, 7
org.Sc.sgdENSEMBLPROT, 8
org.Sc.sgdENSEMBLPROT2ORF *(*org.Sc.sgdENSEMBLPROT*)*, 8
org.Sc.sgdENSEMBLTRANS, 9
org.Sc.sgdENSEMBLTRANS2ORF *(*org.Sc.sgdENSEMBLTRANS*)*, 9
org.Sc.sgdENTREZID, 10
org.Sc.sgdENZYME, 11
org.Sc.sgdENZYME2ORF *(*org.Sc.sgdENZYME*)*, 11
org.Sc.sgdGENENAME, 12
org.Sc.sgdGO, 13
org.Sc.sgdGO2ALLORFS, *14*, 15

org.Sc.sgdGO2ORF *(org.Sc.sgdGO)*, 13
org.Sc.sgdINTERPRO, 16
org.Sc.sgdMAPCOUNTS, 17
org.Sc.sgdORGANISM, 18
org.Sc.sgdPATH, 19
org.Sc.sgdPATH2ORF *(org.Sc.sgdPATH)*, 19
org.Sc.sgdPMID, 20
org.Sc.sgdPMID2ORF *(org.Sc.sgdPMID)*, 20
org.Sc.sgdREFSEQ, 21
org.Sc.sgdREJECTORF, 22
org.Sc.sgdSGD, 23
org.Sc.sgdSMART, 24
org.Sc.sgdUNIPROT, 25