**org.Sco.eg.db**

December 9, 2016

### org.Sco.egACCNUM

**Map Entrez Gene identifiers to GenBank Accession Numbers**

#### Description

`org.Sco.egACCNUM` is an R object that contains mappings between Entrez Gene identifiers and GenBank accession numbers.

#### Details


#### Examples

```r
x <- org.Sco.egACCNUM
# Get the entrez gene identifiers that are mapped to an ACCNUM
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the ACCNUM for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
# For the reverse map ACCNUM2EG:
# Convert to a list
xx <- as.list(org.Sco.egACCNUM2EG)
if(length(xx) > 0) {
  # Gets the entrez gene identifiers for the first five Entrez Gene ID's
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```
org.Sco.eg.db

Description

org.Sco.egALIAS is an R object that provides mappings between common gene symbol identifiers and entrez gene identifiers.

Details

Each gene symbol maps to a named vector containing the corresponding entrez gene identifier. The name of the vector corresponds to the gene symbol. Since gene symbols are sometimes redundantly assigned in the literature, users are cautioned that this map may produce multiple matching results for a single gene symbol. Users should map back from the entrez gene IDs produced to determine which result is the one they want when this happens.

Because of this problem with redundant assignment of gene symbols, it is never advisable to use gene symbols as primary identifiers.

This mapping includes ALL gene symbols including those which are already listed in the SYMBOL map. The SYMBOL map is meant to only list official gene symbols, while the ALIAS maps are meant to store all used symbols.

References


Examples

# Convert the object to a list
xx <- as.list(org.Sco.egALIAS2EG)
# Remove pathway identifiers that do not map to any entrez gene id
xx <- xx[!is.na(xx)]
if(length(xx) > 0){
  # The entrez gene identifiers for the first two elements of XX
  xx[1:2]
  # Get the first one
  xx[[1]]
}

org.Sco.eg.db

Bioconductor annotation data package

Description

Welcome to the org.Sco.eg.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.Sco.eg.db. "Hs" is for Homo sapiens. This package is updated biannually.

You can learn what objects this package supports with the following command:
ls("package:org.Sco.eg.db")
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

```r
ls("package:org.Sco.eg.db")
```

---

**org.Sco.egCHR**

Map Entrez Gene IDs to Chromosomes

**Description**

`org.Sco.egCHR` is an R object that provides mappings between entrez gene identifiers and the chromosome that contains the gene of interest.

**Details**

Each entrez gene identifier maps to a vector of a chromosome.


**Examples**

```r
x <- org.Sco.egCHR
# Get the entrez gene 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.Sco.egCHRLENGTHS**

A named vector for the length of each of the chromosomes

**Description**

`org.Sco.egCHRLENGTHS` 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.
Examples

```r
tt <- org.Sco.egCHRLENGTHS
# Length of chromosome 1
tt["1"]
```

**Description**

`org.Sco.egCHRLOC` is an R object that maps entrez gene 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 entrez gene 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: Entrez Gene http://www.ncbi.nlm.nih.gov/nuccore/ With a date stamp from the source of: 2010-Feb3

**Examples**

```r
x <- org.Sco.egCHRLOC
# Get the entrez gene 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]]
}
```
Map between Entrez Gene IDs and Enzyme Commission (EC) Numbers

**Description**

`org.Sco.egENZYME` is an R object that provides mappings between entrez gene identifiers and EC numbers.

**Details**

Each entrez gene identifier maps to a named vector containing the EC number that corresponds to the enzyme produced by that gene. The name corresponds to the entrez gene 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/](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.Sco.egENZYME2EG`, 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](http://www.chem.qmul.ac.uk/iupac/jcbn/index.html#6)


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

**References**


**Examples**

```r
x <- org.Sco.egENZYME
# Get the entrez gene 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 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.Sco.egENZYME2EG)
if(length(xx) > 0){
    # Gets the entrez gene identifiers for the first five enzyme
    # commission numbers
    xx[1:5]
    # Get the first one
    xx[[1]]
}

---

**org.Sco.egGENENAME**  
*Map between Entrez Gene IDs and Genes*

**Description**

`org.Sco.egGENENAME` is an R object that maps entrez gene identifiers to the corresponding gene name.

**Details**

Each entrez gene identifier maps to a named vector containing the gene name. The vector name corresponds to the entrez gene 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.

With a date stamp from the source of: 2010-Feb3

**Examples**

```r
x <- org.Sco.egGENENAME
# Get the gene names that are mapped to an entrez gene 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]]
}
```
**Map between Entrez Gene IDs and Gene Ontology (GO)**

**Description**

org.Sco.egGO is an R object that provides mappings between entrez gene 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 Entrez Gene 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. 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 between entrez gene identifiers and GO information were obtained through their mappings to Entrez Gene identifiers. NAs are assigned to entrez gene identifiers that cannot be mapped to any Gene Ontology information. 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: Gene Ontology http://www.geneontology.org/ontology/obo\_format\_1\_2/geo

With a date stamp from the source of: 2010-Feb9

For the reverse map GO2EG, each GO term maps to a named vector of entrez gene identifiers. A GO identifier may be mapped to the same entrez gene 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**

org.Sco.egGO2ALLEGS

Map Between Gene Ontology (GO) Identifiers and all Entrez Gene Identifiers in the subtree

Description

org.Sco.egGO2ALLEGS is an R object that provides mappings between a given GO identifier and all Entrez Gene 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 Entrez Gene id corresponds to the evidence code for that GO identifier. This object org.Sco.egGO2ALLEGS maps between a given GO identifier and all Entrez Gene 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

Examples

```r
x <- org.Sco.egGO
# Get the entrez gene 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.Sco.egGO2EG)
if(length(xx) > 0)
  # Gets the entrez gene ids for the top 2nd and 3rd GO identifiers
  goids <- xx[2:3]
  # Gets the entrez gene ids for the first element of goids
  goids[[1]]
  # Evidence code for the mappings
  names(goids[[1]])
```
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 Entrez Gene 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**


**Examples**

```r
# Convert to a list
xx <- as.list(org.Sco.egGO2ALLEGS)
if(length(xx) > 0){
  # Gets the Entrez Gene identifiers for the top 2nd and 3nd GO identifiers
goids <- xx[2:3]
  # Gets all the Entrez Gene identifiers for the first element of goids
goids[[1]]
  # Evidence code for the mappings
  names(goids[[1]])
}
```

---

**Description**

`org.Sco.egLOCUSTAG` is an R object that provides mappings between entrez gene identifiers and the locus tag identifier associated to the gene of interest.

**Details**

Each entrez gene identifier maps to a locus tag identifier.

Mappings were based on data provided by: Entrez Gene http://www.ncbi.nlm.nih.gov/nuccore/
With a date stamp from the source of: 2010-Feb3
Examples

```r
x <- org.Sco.egLOCUSTAG
# Get the entrez gene identifiers that are mapped to a locus tag
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the LOCUSTAG for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

Description

`org.Sco.egMAPCOUNTS` provides the "map count" (i.e. the count of mapped keys) for each map in package `org.Sco.eg.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

```r
mapnames <- names(org.Sco.egMAPCOUNTS)
org.Sco.egMAPCOUNTS[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.Sco.eg.db
checkMAPCOUNTS("org.Sco.eg.db")
```
**org.Sco.egORGANISM**

**The Organism for org.Sco.eg**

**Description**

`org.Sco.egORGANISM` is an R object that contains a single item: a character string that names the organism for which `org.Sco.eg` was built.

**Details**

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

**Examples**

```r
org.Sco.egORGANISM
```

---

**org.Sco.egPATH**

**Mappings between Entrez Gene identifiers and KEGG pathway identifiers**

**Description**

KEGG (Kyoto Encyclopedia of Genes and Genomes) maintains pathway data for various organisms. `org.Sco.egPATH` maps entrez gene identifiers to the identifiers used by KEGG for pathways.

**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](http://www.bioconductor.org).


Mappings were based on data provided by: KEGG PATHWAY ftp://ftp.genome.jp/pub/kegg/pathway/organisms/sco

With a date stamp from the source of: 2010-Jan30

**References**


**Examples**

```r
x <- org.Sco.egPATH
# Get the entrez gene 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]]

# For the reverse map:
# Convert the reverse map to a list
xx <- as.list(org.Sco.egPATH2EG)
# Remove pathway identifiers that do not map to any entrez gene id
xx <- xx[!is.na(xx)]
if(length(xx) > 0){
    # The entrez gene identifiers for the first two elements of XX
    xx[1:2]
    # Get the first one
    xx[[1]]
}

---

**org.Sco.egPMID**  
*Map between Entrez Gene Identifiers and PubMed Identifiers*

**Description**

org.Sco.egPMID is an R object that provides mappings between entrez gene identifiers and PubMed identifiers.

**Details**

Each entrez gene identifier is mapped to a named vector of PubMed identifiers. The name associated with each vector corresponds to the entrez gene identifier. The length of the vector may be one or greater, depending on how many PubMed identifiers a given entrez gene identifier is mapped to. An NA is reported for any entrez gene 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: Entrez Gene http://www.ncbi.nlm.nih.gov/nuccore/  
With a date stamp from the source of: 2010-Feb3

**References**


**Examples**

```r
x <- org.Sco.egPMID
# Get the entrez gene 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 entrez gene 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")

} # For the reverse map:
# Convert the object to a list
xx <- as.list(org.Sco.egPMID2EG)
if(length(xx) > 0){
  # The entrez gene 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.Sco.egPROTEINGI    Map Entrez Gene IDs to protein GenInfo (GI) numbers

Description

org.Sco.egPROTEINGI is an R object that provides mappings between entrez gene identifiers and
the protein GI number associated to the gene of interest.

Details

Each entrez gene identifier maps to a protein gi number.

Mappings were based on data provided by: Entrez Gene http://www.ncbi.nlm.nih.gov/nuccore/
With a date stamp from the source of: 2010-Feb3

Examples

x <- org.Sco.egPROTEINGI
# Get the entrez gene identifiers that are mapped to a protein GI number
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the PROTEINGI for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
org.Sco.egREFSEQ is an R object that provides mappings between entrez gene identifiers and RefSeq identifiers.

Details

Each entrez gene identifier is mapped to a named vector of RefSeq identifiers. The name represents the entrez gene 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 entrez gene identifier can be mapped to. An NA is reported for any entrez 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.


References


Examples

```r
x <- org.Sco.egREFSEQ
# Get the entrez gene 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]]
}
# For the reverse map:
x <- org.Sco.egREFSEQ2EG
# Get the RefSeq identifier that are mapped to an entrez gene ID
mapped_segs <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_segs])
if(length(xx) > 0) {
  # Get the entrez gene for the first five Refseqs
  xx[1:5]
  # Get the first one
  xx[[1]]
}

---

### org.Sco.egSYMBOL

#### Map between Entrez Gene Identifiers and Gene Symbols

**Description**

`org.Sco.egSYMBOL` is an R object that provides mappings between entrez gene identifiers and gene abbreviations.

**Details**

Each entrez gene identifier is mapped to the a common abbreviation for the corresponding gene. The locus tag is assigned as symbol if there is no known abbreviation for a given gene.

Symbols typically consist of 3 letters that define either a single gene (ABC) or multiple genes (ABC1, ABC2, ABC3). Gene symbols can be used as key words to query public databases such as Entrez Gene.

Mappings were based on data provided by: Entrez Gene http://www.ncbi.nlm.nih.gov/nuccore/ With a date stamp from the source of: 2010-Feb3

**References**


**Examples**

```r
x <- org.Sco.egSYMBOL
# Get the gene symbol that are mapped to an entrez gene identifiers
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the SYMBOL for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
# For the reverse map:
x <- org.Sco.egSYMBOL2EG
# Get the entrez gene identifiers that are mapped to a gene symbol
mapped_genes <- mappedkeys(x)
# Convert to a list
```
org.Sco.eg_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.Sco.eg_dbconn()
org.Sco.eg_dbfile()
org.Sco.eg_dbschema(file='', show.indices=FALSE)
org.Sco.eg_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.Sco.eg_dbconn returns a connection object to the package annotation DB. IMPORTANT: Don’t call dbDisconnect on the connection object returned by org.Sco.eg_dbconn or you will break all the AnnDbObj objects defined in this package!
org.Sco.eg_dbfile returns the path (character string) to the package annotation DB (this is an SQLite file).
org.Sco.eg_dbschema prints the schema definition of the package annotation DB.
org.Sco.eg_dbInfo prints other information about the package annotation DB.

Value

org.Sco.eg_dbconn: a DBIConnection object representing an open connection to the package annotation DB.
org.Sco.eg_dbfile: a character string with the path to the package annotation DB.
org.Sco.eg_dbschema: none (invisible NULL).
org.Sco.eg_dbInfo: none (invisible NULL).

See Also

dbGetQuery, dbConnect, dbi_conn, dbfile, dbschema, dbInfo
Examples

## Count the number of rows in the "genes" table:
\[\text{dbGetQuery}(\text{org.Sco.eg_dbconn}(), \text{"SELECT COUNT(*) FROM genes"})\]

## The connection object returned by \text{org.Sco.eg_dbconn}() was
## created with:
\[\text{dbConnect}(\text{SQLite}(), \text{dbname}=\text{org.Sco.eg_dbfile}(), \text{cache_size}=64000, \text{synchronous}=0)\]

\text{org.Sco.eg_dbschema}()

\text{org.Sco.eg_dbInfo}()
Index

*Topic datasets
  org.Sco.eg.db, 2
  org.Sco.eg.dbconn, 16
  org.Sco.egACCNUM, 1
  org.Sco.egALIAS2EG, 2
  org.Sco.egCHR, 3
  org.Sco.egCHRLENGTHS, 3
  org.Sco.egCHRLOC, 4
  org.Sco.egENZYME, 5
  org.Sco.egGENENAME, 6
  org.Sco.egGO, 7
  org.Sco.egGO2ALLEGS, 8
  org.Sco.egLOCUSTAG, 9
  org.Sco.egMAPCOUNTS, 10
  org.Sco.egORGANISM, 11
  org.Sco.egPATH, 11
  org.Sco.egPMID, 12
  org.Sco.egPROTEINGI, 13
  org.Sco.egREFSEQ, 14
  org.Sco.egSYMBOL, 15

*Topic utilities
  org.Sco.eg.dbconn, 16

AnnDbObj, 16

cat, 16
checkMAPCOUNTS, 10
count.mappedkeys, 10

dbconn, 16
dbConnect, 16
dbDisconnect, 16
dbfile, 16
dbGetQuery, 16
dbInfo, 16
dbschema, 16

mappedkeys, 10

org.Sco.eg (org.Sco.eg.db), 2
org.Sco.eg.db, 2
org.Sco.eg.dbconn, 16
org.Sco.eg_dbfile (org.Sco.eg_dbconn), 16

org.Sco.eg_dbInfo (org.Sco.eg_dbconn), 16
org.Sco.eg_dbschema (org.Sco.eg_dbconn), 16
org.Sco.egACCNUM, 1
org.Sco.egACCNUM2EG (org.Sco.egACCNUM), 1
org.Sco.egALIAS2EG, 2
org.Sco.egCHR, 3
org.Sco.egCHRLENGTHS, 3
org.Sco.egCHRLOC, 4
org.Sco.egCHRLOCEND (org.Sco.egCHRLOC), 4
org.Sco.egENZYME, 5
org.Sco.egENZYME2EG (org.Sco.egENZYME), 5
org.Sco.egGENENAME, 6
org.Sco.egGO, 7
org.Sco.egGO2ALLEGS, 8
org.Sco.egGO2EG (org.Sco.egGO), 7
org.Sco.egLOCUSTAG, 9
org.Sco.egMAPCOUNTS, 10
org.Sco.egORGANISM, 11
org.Sco.egPATH, 11
org.Sco.egPATH2EG (org.Sco.egPATH), 11
org.Sco.egPMID, 12
org.Sco.egPMID2EG (org.Sco.egPMID), 12
org.Sco.egPROTEINGI, 13
org.Sco.egREFSEQ, 14
org.Sco.egREFSEQ2EG (org.Sco.egREFSEQ), 14
org.Sco.egSYMBOL, 15
org.Sco.egSYMBOL2EG (org.Sco.egSYMBOL), 15