

# Package ‘gendoo.Hs.db’

March 25, 2013

**Title** Gendoo Human Database  
**Description** Entrez Gene ID to MeSH ID table from Gendoo  
**Version** 0.99.0  
**Author** Itoshi NIKAIDO, Koki Tsuyuzaki, Gota Morota, Takeru Nakazato  
**Maintainer** Itoshi NIKAIDO <dritoshi@gmail.com>  
**Depends** R (>= 2.7.0), methods, AnnotationDbi, AnnotationForge, DBI, RSQLite  
**Imports** methods, AnnotationDbi, AnnotationForge, DBI, RSQLite  
**Suggests** RUnit, BiocGenerics  
**License** Artistic-2.0  
**biocViews** AnnotationData

## R topics documented:

gendoo.Hs.db . . . . .	1
gendoo.Hs_dbconn . . . . .	2
GendooGene . . . . .	3
GendooMeSHA . . . . .	3
<b>Index</b>	<b>5</b>

---

gendoo.Hs.db	<i>Gendoo Human Database package</i>
--------------	--------------------------------------

---

## Description

Welcome to the gendoo.Hs.db annotation package. The purpose of this package is to provide mapping table for Entrez Gene ID and MeSH ID. The original data are provided at <http://gendoo.dbcls.jp/>. You can learn what objects this package supports with the following command:

```
ls ("package:gendoo.Hs.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:gendoo.Hs.db")
```

---

gendoo.Hs_dbconn	<i>Collect information about the package annotation DB</i>
------------------	--

---

## Description

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

## Usage

```
gendoo.Hs ()
gendoo.Hs_dbconn ()
gendoo.Hs_dbfile ()
gendoo.Hs_dbschema ()
gendoo.Hs_dbInfo ()
```

## Details

`gendoo.Hs` returns the meta data (character string) of the package annotation DB.

`gendoo.Hs_dbconn` returns a connection object to the package annotation DB. **IMPORTANT:** Don't call `dbDisconnect` on the connection object returned by `gendoo.Hs_dbconn`.

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

`gendoo.Hs_dbschema` prints the schema definition of the package annotation DB.

`gendoo.Hs_dbInfo` prints other information about the package annotation DB.

## Value

`gendoo.Hs`: none (invisible NULL).

`gendoo.Hs_dbconn`: a `DBIConnection` object representing an open connection to the package annotation DB.

`gendoo.Hs_dbfile`: a character string with the path to the package annotation DB.

`gendoo.Hs_dbschema`: none (invisible NULL).

`gendoo.Hs_dbInfo`: none (invisible NULL).

## See Also

[dbGetQuery](#), [dbConnect](#), [dbshow](#), [dbconn](#), [dbfile](#), [dbschema](#), [dbInfo](#)

**Examples**

```
## Count the number of rows in the "metadata" table:
dbGetQuery(gendoo.Hs_dbconn(), "SELECT COUNT(*) FROM metadata")

## The connection object returned by gendoo.Hs_dbconn() was
## created with:
dbConnect(SQLite(), dbname=gendoo.Hs_dbfile(), cache_size=64000,
synchronous=0)

gendoo.Hs()

gendoo.Hs_dbschema()

gendoo.Hs_dbInfo()

gendoo.Hs_dbfile()
```

---

GendooGene

*The GendooGene for gendoo.Hs*


---

**Description**

GendooGene is an R object that contains Entrez Gene ID and its annotations.

**Details**

GendooGene provides Entrez Gene IDs with its annotations, Gene ID, symbol, description, synonyms and gene type. You can retrieve these data from the object using by select method. See examples.

**Examples**

```
keytypes(GendooGene)
cols(GendooGene)

my.keytype <- c("gene_id")
gene_ids <- keys(GendooGene, keytype = my.keytype)

## search
my.keys <- c(1)
my.cols <- c("gene_id", "gene_symbol", "gene_description",
"gene_synonyms", "gene_type")
my.genes <- select(GendooGene, keys = my.keys, cols = my.cols, keytype
= my.keytype)
head(my.genes)
```

## Description

GendooMeSH is an R object that contains mapping table of Entrez Gene ID to MeSH ID.

## Details

GendooMeSH provides mapping table for Entrez Gene ID to MeSH ID. You can retrieve these data from the object using by select method. See examples.

A: Anatomy B: Organisms C: Diseases D: Drugs and chemicals G: Biological phenomena S: Substance Names

## Examples

```
keytypes(GendooMeSHA)
cols(GendooMeSHA)

my.keytype <- c("gene_id")
gene_ids <- keys(GendooMeSHA, keytype = my.keytype)

## search
my.keys <- c(1, 2, 3)
my.cols <- c("gene_id", "mesh")
my.mesh <- select(GendooMeSHA, keys = my.keys, cols = my.cols, keytype
= my.keytype)
head(my.mesh)
```

# Index

## \*Topic **datasets**

- `gendoo.Hs.db`, [1](#)
- `gendoo.Hs_dbconn`, [2](#)
- `GendooGene`, [3](#)
- `GendooMeSHA`, [3](#)

## \*Topic **utilities**

- `gendoo.Hs_dbconn`, [2](#)

- `dbconn`, [2](#)
- `dbConnect`, [2](#)
- `dbDisconnect`, [2](#)
- `dbfile`, [2](#)
- `dbGetQuery`, [2](#)
- `dbInfo`, [2](#)
- `dbschema`, [2](#)
- `dbshow`, [2](#)

- `gendoo.Hs` (`gendoo.Hs.db`), [1](#)
- `gendoo.Hs` (`gendoo.Hs_dbconn`), [2](#)
- `gendoo.Hs.db`, [1](#)
- `gendoo.Hs_dbconn`, [2](#)
- `gendoo.Hs_dbfile`
  - (`gendoo.Hs_dbconn`), [2](#)
- `gendoo.Hs_dbInfo`
  - (`gendoo.Hs_dbconn`), [2](#)
- `gendoo.Hs_dbschema`
  - (`gendoo.Hs_dbconn`), [2](#)
- `GendooGene`, [3](#)
- `GendooMeSHA`, [3](#)
- `GendooMeSHB` (`GendooMeSHA`), [3](#)
- `GendooMeSHC` (`GendooMeSHA`), [3](#)
- `GendooMeSHD` (`GendooMeSHA`), [3](#)
- `GendooMeSHG` (`GendooMeSHA`), [3](#)
- `GendooMeSHS` (`GendooMeSHA`), [3](#)