synaptome.db

May 12, 2026

`findGeneByCompartmentPaperCnt` *Get gene table of frequently found genes within compartments*

Description

Get gene table and paper count for genes mentioned `cnt` or more times in different compartment-paper pairs.

Usage

```
findGeneByCompartmentPaperCnt(cnt = 1)
```

Arguments

`cnt`	minimal number of times mentioned gene

Value

`data.frame` with 9 columns: 8 specified in `getGenesByID` and `Npmid` column for the paper count.

See Also

getGenesByID

Other Gene functions: `findGeneByPaperCnt()`, `findGeneByPapers()`, `findGenesByEntrez()`, `findGenesByName()`, `getAllGenes4Compartment()`, `getGenes4Compartment()`, `getGenesByID()`

Examples

```
cntT <- findGeneByPaperCnt(47)
head(cntT)
```

`findGeneByPaperCnt` *Get gene table of frequently found genes*

Description

Get gene table and paper count for genes mentioned `cnt` or more times in different papers.

Usage

```
findGeneByPaperCnt(cnt = 1)
```

Arguments

`cnt`	minimal number of papers that mentioned gene

Value

`data.frame` with 9 columns: 8 specified in `getGenesByID` and `Npmid` column for the paper count.

See Also

getGenesByID

Other Gene functions: `findGeneByCompartmentPaperCnt()`, `findGeneByPapers()`, `findGenesByEntrez()`, `findGenesByName()`, `getAllGenes4Compartment()`, `getGenes4Compartment()`, `getGenesByID()`

Examples

```
cntT <- findGeneByPaperCnt(47)
head(cntT)
```

`findGeneByPapers` *Get gene table of frequently found genes*

Description

Get gene table and paper count for genes mentioned `cnt` or more times in different papers.

Usage

```
findGeneByPapers(pmids, cnt = 1)
```

Arguments

`pmids`	vector of PMIDs to search for genes
`cnt`	minimal number of papers that mentioned gene

Value

`data.frame` with 9 columns: 8 specified in `getGenesByID` and `Npmid` column for the paper count.

See Also

getGenesByID

Other Gene functions: `findGeneByCompartmentPaperCnt()`, `findGeneByPaperCnt()`, `findGenesByEntrez()`, `findGenesByName()`, `getAllGenes4Compartment()`, `getGenes4Compartment()`, `getGenesByID()`

Examples

```
res<- getAllGenes4Compartment(compartmentID = 1)
gnt<-getGeneInfoByIDs(res$GeneID)
pmids<-names(sort(table(gnt$PaperPMID))[1:5])
cntT <- findGeneByPapers(pmids,cnt=3)
head(cntT)
```

`findGenesByEntrez` *Internal Gene representation for given list of gene Entrez IDs*

Description

Get internal gene representation for set of gene Entrez IDs. Function lookups for provided values in Human Entrez ID, Mouse Entrez ID and Rat Entrez ID columns and return following features for all found genes: GeneID (internal database ID), MGI ID, Human Entrez ID, Mouse Entrez ID, Rat Entrez ID, Human gene name, Mouse gene name and Rat gene name.

Usage

```
findGenesByEntrez(entrez)
```

Arguments

`entrez`	`vector` of Entres IDs. Function accepts both integers and characters.

Details

Could be used as an intermediate step for building Protein-Protein interaction map from the list of Gene IDs returned in the first column. Also, this function provides a useful sanity check, e.g. how many Gene IDs correspond to the specific gene name or Entrez ID, which could be specie-specific.

Value

`data.frame` with columns specified above.

See Also

Other Lookup functions: `findGenesByName()`

Other Gene functions: `findGeneByCompartmentPaperCnt()`, `findGeneByPaperCnt()`, `findGeneByPapers()`, `findGenesByName()`, `getAllGenes4Compartment()`, `getGenes4Compartment()`, `getGenesByID()`

Examples

```
# get information for specific gene
t <- findGenesByEntrez(c(1742, 1741, 1739, 1740))
```

`findGenesByName` *Find GeneIDs for names*

Description

Get internal gene representation for set of gene names. Function lookups for provided values in Human Name, Mouse Name and Rat Name columns and return following features for all found genes: GeneID (internal database ID), MGI ID, Human Entrez ID, Mouse Entrez ID, Rat Entrez ID, Human gene name, Mouse gene name and Rat gene name.

Usage

```
findGenesByName(name)
```

Arguments

`name`	`vector` of gene names.

Details

Could be used as an intermediate step for building Protein-Protein interaction map from the list of Gene IDs returned in the first column. Also, this function provides a useful sanity check, e.g. how many Gene IDs correspond to the specific gene name or Entrez ID, which could be specie-specific.

Value

`data.frame` with columns specified above.

See Also

Other Lookup functions: `findGenesByEntrez()`

Other Gene functions: `findGeneByCompartmentPaperCnt()`, `findGeneByPaperCnt()`, `findGeneByPapers()`, `findGenesByEntrez()`, `getAllGenes4Compartment()`, `getGenes4Compartment()`, `getGenesByID()`

Examples

```
# Find GeneIDs for names
t <- findGenesByName(c("Src", "Srcin1", "Fyn"))
```

`getAllGenes4BrainRegion` *Get all genes for brain region for particular specie*

Description

Get all genes annotated for specific brain region for specific specie. Function lookups Brain Region ID and Specie Tax ID columns and returns main information like internal Gene IDs, MGI ID, Human Entrez ID, Human Gene Name, Mouse Entrez ID, Mouse Gene Name, Localisation (presynaptic, postsynaptic, synaptosomal), PaperPMID and BrainRegion.

Usage

```
getAllGenes4BrainRegion(brainRegion, taxID)
```

Arguments

`brainRegion`	ID for specific brain region
`taxID`	specie ID

Value

data.frame with the following columns:

- GeneID
- Localisation
- MGI
- HumanEntrez
- MouseEntrez
- HumanName
- MouseName
- PMID
- Paper
- Year
- SpeciesTaxID
- BrainRegion

See Also

Other BrainRegion functions: `getBrainRegions()`, `getGenes4BrainRegion()`, `getPPIbyIDs4BrainRegion()`

Other BrainRegion Gene functions: `getGenes4BrainRegion()`

Examples

```
gns <- getAllGenes4BrainRegion(brainRegion = "Striatum", taxID = 10090)
head(gns)
```

`getAllGenes4Compartment`

Extract all genes found in compartment

Description

Get all genes annotated for specific compartment. Function returns main information like internal Gene IDs, MGI ID, Human Entrez ID, Human Gene Name, Mouse Entrez ID, Mouse Gene Name, Rat Entrez ID, Rat Gene Name.

Usage

```
getAllGenes4Compartment(compartmentID)
```

Arguments

`compartmentID`	ID for specific compartment

Value

data.frame described in `getGenesByID()`

See Also

Other Gene functions: `findGeneByCompartmentPaperCnt()`, `findGeneByPaperCnt()`, `findGeneByPapers()`, `findGenesByEntrez()`, `findGenesByName()`, `getGenes4Compartment()`, `getGenesByID()`

Other Compartment functions: `getCompartments()`, `getGenes4Compartment()`, `getPPIbyIDs4Compartment()`

Examples

```
G <- getAllGenes4Compartment(compartmentID = 1) # 5560 rows
```

`getBrainRegions` *Get list of all Brain regions in the database*

Description

Get full list of brain regions considered in the database. Returns table with regions and their respective IDs.

Usage

```
getBrainRegions()
```

Value

data.frame with following columns:

- ID: Brain region internal ID
- Name: name of the region
- Description: extended description of the region
- InterlexID
- ParentID: ID of the containing brain region

See Also

Other BrainRegion functions: `getAllGenes4BrainRegion()`, `getGenes4BrainRegion()`, `getPPIbyIDs4BrainRegion`

Examples

```
t <- getBrainRegions()
```

`getCompartments` *Get full list of compartments*

Description

Get list of synaptic compartments considered in the database. Returns table with compartments: “Presynaptic”, “Postsynaptic”, “Synaptosome”, and their internal IDs.

Usage

```
getCompartments()
```

Value

data.frame described above

See Also

Other Compartment functions: `getAllGenes4Compartment()`, `getGenes4Compartment()`, `getPPIbyIDs4Compartmen`

Examples

```
c <- getCompartments()
```

`getGeneDiseaseByEntres` *Disease information for Human Entrez IDs*

Description

Get Human disease information (HDO provided) for the set of Human Entrez IDs. Function lookups for diseases associated with Human Entrez IDs and returns list of available diseases.

Usage

```
getGeneDiseaseByEntres(entrez)
```

Arguments

`entrez`	`vector` of Human Entrez gene IDs

Value

data.frame with following columns:

- HumanEntrez
- HumanName
- HDOID
- Description

See Also

Other Disease functions: `getGeneDiseaseByIDs()`, `getGeneDiseaseByName()`

Examples

```
t <- getGeneDiseaseByEntres(c(8573, 1742, 1739)) # (95 rows)
```

`getGeneDiseaseByIDs` *Disease information for internal gene IDs.*

Description

Get Human disease information (HDO provided) for the set of internal gene IDs. Function lookups for diseases associated with internal GeneIDs and returns list of available diseases.

Usage

```
getGeneDiseaseByIDs(ids)
```

Arguments

`ids`	vector of gene IDs.

Value

data.frame with following columns:

- HumanEntrez
- HumanName
- HDOID
- Description

See Also

Other Disease functions: `getGeneDiseaseByEntres()`, `getGeneDiseaseByName()`

Examples

```
t <- getGeneDiseaseByIDs(c(48, 585, 710))
```

`getGeneDiseaseByName` *Disease information for Human Gene Names*

Description

Get Human disease information (HDO provided) for the set of Human Gene Names. Function lookups for diseases associated with internal Human gene names and returns list of available diseases.

Usage

```
getGeneDiseaseByName(names)
```

Arguments

`names`	`vector` of Human gene names

Value

data.frame

See Also

Other Disease functions: `getGeneDiseaseByEntres()`, `getGeneDiseaseByIDs()`

Examples

```
t <- getGeneDiseaseByName(c("CASK", "DLG2", "DLG1")) # (115 rows)
```

`getGeneInfoByEntrez` *Gene information for given list of gene Entrez IDs*

Description

Get gene information for set of gene Entrez IDs. Function lookup for name in Human Entrez ID and Mouse Entrez Id data and return following features for all found genes: GeneID (internal database ID), Localisation (presynaptic, postsynaptic, synaptosome), MGI (MGI ID), HumanEntrez (Human Entrez ID), MouseEntrez (Mouse Entrez ID), HumanName (Human gene name), MouseName (Mouse gene name), PaperPMID (PMID IDs for the publications where the genes were reported), Paper (papers where specific genes were reported in a format FIRSTAUTHOR_YEAR), Year, SpeciesTaxID (specie the original experiment was performed on), BrainRegion (Brain region where the specific genes were identified, according to the paper)

Usage

```
getGeneInfoByEntrez(entrez)
```

Arguments

`entrez`	`vector` of Entres IDs. Function accepts both integers and characters.

Value

`data.frame` with fields specified above.

See Also

Other GeneInfo functions: `getGeneInfoByIDs()`, `getGeneInfoByName()`, `getGeneInfoByPapers()`

Examples

```
# get information for specific gene
t <- getGeneInfoByEntrez(1742)
# get information for specific character string Entres representation
t <- getGeneInfoByEntrez("1742")

# get information for the list of genes
t <- getGeneInfoByName(c(1741, 1742, 1739, 1740))
```

`getGeneInfoByIDs` *Get GeneInfo table for set of GeneIDs*

Description

Function lookup for internal GeneID values and return following features for all found genes: GeneID (internal database ID), Localisation (one of the following: presynaptic, postsynaptic, synaptosome), MGI (MGI ID), HumanEntrez (Human Entrez ID), MouseEntrez (Mouse Entrez ID), HumanName (Human gene name), MouseName (Mouse gene name), PaperPMID (PMID IDs for the publications where the genes were reported), Paper (papers where specific genes were reported in a format FIRSTAUTHOR_YEAR), Year, SpeciesTaxID (specie the original experiment was performed on), BrainRegion (Brain region where the specific genes were identified, according to the paper)

Usage

```
getGeneInfoByIDs(ids)
```

Arguments

`ids`	`vector` of Gene IDs.

Details

This function then returns following features for all found genes:

- GeneID,
- Localisation,
- MGI,
- HumanEntrez,
- MouseEntrez,
- HumanName,
- MouseName,

- PaperPMID,
- Paper,
- Year,
- SpeciesTaxID,
- BrainRegion

Value

`data.frame` with column specified above.

See Also

Other GeneInfo functions: `getGeneInfoByEntrez()`, `getGeneInfoByName()`, `getGeneInfoByPapers()`

Examples

```
gdf <- getGeneInfoByIDs(c(46, 6, 15, 1))
```

`getGeneInfoByName` *Get gene information for set of gene names.*

Description

Function lookup for name in Human Gene name, Rat Gene name and Mouse Gene name data and return following features for all found genes: GeneID (internal database ID), Localisation (one of the following: presynaptic, postsynaptic, synaptosome), MGI (MGI ID), HumanEntrez (Human Entrez ID), MouseEntrez (Mouse Entrez ID), HumanName (Human gene name), MouseName (Mouse gene name), PaperPMID (PMID IDs for the publications where the genes were reported), Paper (papers where specific genes were reported in a format FIRSTAUTHOR_YEAR), Year, SpeciesTaxID (specie the original experiment was performed on), BrainRegion (Brain region where the specific genes were identified, according to the paper)

Usage

```
getGeneInfoByName(name)
```

Arguments

`name`	`vector` of gene names

Details

This function then returns following features for all found genes:

- GeneID,
- Localisation,
- MGI,
- HumanEntrez,
- MouseEntrez,
- HumanName,

- MouseName,
- PaperPMID,
- Paper,
- Year,
- SpeciesTaxID,
- BrainRegion

Value

`data.frame` with fields specified above.

See Also

Other GeneInfo functions: `getGeneInfoByEntrez()`, `getGeneInfoByIDs()`, `getGeneInfoByPapers()`

Examples

```
# get information for specific gene
t <- getGeneInfoByName("CASK")

# get information for the list of genes
t <- getGeneInfoByName(c("CASK", "DLG2"))
```

`getGeneInfoByPapers` *Get gene information for set of genes mentioned by certain papers.*

Description

Function lookup for specified PubMedIDs in the gene reference data and return following features for genes referenced by requested papers at least `cnt` times: GeneID (internal database ID), Localisation (one of the following: presynaptic, postsynaptic, synaptosome), MGI (MGI ID), HumanEntrez (Human Entrez ID), MouseEntrez (Mouse Entrez ID), HumanName (Human gene name), MouseName (Mouse gene name), PaperPMID (PMID IDs for the publications where the genes were reported if it is within `pmids` list), Paper (papers where specific genes were reported in a format FIRSTAUTHOR_YEAR), Year, SpeciesTaxID (specie the original experiment was performed on), BrainRegion (Brain region where the specific genes were identified, according to the paper)

Usage

```
getGeneInfoByPapers(pmids, cnt = 1)
```

Arguments

`pmids`	vector of PMIDs to search for genes
`cnt`	minimal number of papers that mentioned gene

Details

This function then returns following features for all found genes:

- GeneID,
- Localisation,
- MGI,
- HumanEntrez,
- MouseEntrez,
- HumanName,
- MouseName,
- PaperPMID,
- Paper,
- Year,
- SpeciesTaxID,
- BrainRegion

Value

`data.frame` with fields specified above.

See Also

Other GeneInfo functions: `getGeneInfoByEntrez()`, `getGeneInfoByIDs()`, `getGeneInfoByName()`

Examples

```
res<- getAllGenes4Compartment(compartmentID = 1)
gnt<-getGeneInfoByIDs(res$GeneID)
pmids<-names(sort(table(gnt$PaperPMID))[1:5])
cntT <- getGeneInfoByPapers(pmids,cnt=3)
head(cntT)
```

`getGenes4BrainRegion` *Select genes from the list that found in brain region of particular specie*

Description

Select genes from your list annotated for specific brain region. Should be used with `findGenesByEntrez()` or `findGenesByName()` functions to obtain list of internal IDs for your list of genes. Function lookups the Gene table for specified localisation and returns main gene information like internal Gene IDs, MGI ID, Human Entrez ID, Human Gene Name, Mouse Entrez ID, Mouse Gene Name, PaperPMID, Localisation and BrainRegion.

Usage

```
getGenes4BrainRegion(ids, brainRegion, taxID)
```

Arguments

`ids`	internal IDs for list of genes
`brainRegion`	ID for the brain region of interest
`taxID`	taxon ID specie tax ID

Value

data.frame with the following columns:

- GeneID
- Localisation
- MGI
- HumanEntrez
- MouseEntrez
- HumanName
- MouseName
- PMID
- Paper
- Year
- SpeciesTaxID
- BrainRegion

See Also

Other BrainRegion functions: `getAllGenes4BrainRegion()`, `getBrainRegions()`, `getPPIbyIDs4BrainRegion()`

Other BrainRegion Gene functions: `getAllGenes4BrainRegion()`

Examples

```
Genes <- getGenes4BrainRegion(c(1, 15, 156, 1500, 3000, 7000),
    brainRegion = "Striatum", taxID = 10090
) # 5 rows
```

`getGenes4Compartment` *Select genes from the list that found in compartment*

Description

Select all genes from your list annotated for specific compartment. Should be used with `findGenesByEntrez()` or `findGenesByName()` functions to obtain list of internal IDs for your list of genes. Function lookups the Gene table for specified localisation and returns main gene information like internal Gene IDs, MGI ID, Human Entrez ID, Human Gene Name, Mouse Entrez ID, Mouse Gene Name, Rat Entrez ID, Rat Gene Name.

Usage

```
getGenes4Compartment(ids, compartmentID)
```

Arguments

`ids`	Gene IDs
`compartmentID`	compartment ID

Value

data.frame described in `getGenesByID()`

See Also

Other Gene functions: `findGeneByCompartmentPaperCnt()`, `findGeneByPaperCnt()`, `findGeneByPapers()`, `findGenesByEntrez()`, `findGenesByName()`, `getAllGenes4Compartment()`, `getGenesByID()`

Other Compartment functions: `getAllGenes4Compartment()`, `getCompartments()`, `getPPIbyIDs4Compartment()`

Examples

```
Genes <- getGenes4Compartment(c(1, 15, 156, 1500, 3000, 7000),
    compartmentID = 1
)
```

`getGenesByID` *Get gene table from list of GeneIDs.*

Description

Takes internal gene IDs as input and return the following features for all found genes:

GeneID internal database ID

MGI MGI ID

HumanEntrez Human Entrez ID

MouseEntrez Mouse Entrez ID

HumanName Human gene name

MouseName Mouse gene name

RatEntrez Rat Entrez ID

Rat Name Rat gene name

Usage

```
getGenesByID(ids)
```

Arguments

`ids`	`vector` of GeneID values.

Value

`data.frame` with 8 columns specified above.

See Also

Other Gene functions: `findGeneByCompartmentPaperCnt()`, `findGeneByPaperCnt()`, `findGeneByPapers()`, `findGenesByEntrez()`, `findGenesByName()`, `getAllGenes4Compartment()`, `getGenes4Compartment()`

Examples

```
gdf <- getGenesByID(c(46, 6, 15, 1))
```

`getIGraphFromPPI` *Get Igraph representation of PPI*

Description

Combine information from PPI `data.frame` obtained with functions like `getPPIbyName` or `getPPIbyEntrez` with information about genes obtained from `getGenesByID` to make interpretable undirected PPI graph in `igraph` format. In this format network could be further analysed and visualized by algorithms in `igraph` package.

Usage

```
getIGraphFromPPI(ppi)
```

Arguments

`ppi`	PPI `data.frame` with columns A and B, obtaioed from functions like `getPPIbyName`

Value

`igraph` object with specified PPI network.

See Also

`getPPIbyName()`, `getPPIbyEntrez()` and `getPPIbyIDs()` to get PPI `data.frame`, `getTableFromPPI()` to get interpretable table representation of the PPI `data.frame`.

Other PPI functions: `getPPIbyEntrez()`, `getPPIbyIDs()`, `getPPIbyIDs4BrainRegion()`, `getPPIbyIDs4Compartmen`
`getPPIbyName()`, `getTableFromPPI()`

Examples

```
library(igraph)
g <- getIGraphFromPPI(
    getPPIbyIDs(c(48, 129, 975, 4422, 5715, 5835), type = "lim")
)
plot(g, vertex.label = V(g)$RatName, vertex.size = 35)
```

---

getMutations4DiseaseByEntres

Get mutational information for Gene and Disease

Description

Function looks up the information for particular set of Entrez IDs and disease HDOID.

Usage

```
getMutations4DiseaseByEntres(entrez, hdoid)
```

Arguments

`entrez`	list of Entrez IDs for genes to select
`hdoid`	disease HDOID to get mutational association.

Value

`data.frame` as described in `getMutations4DiseaseByIDs`.

See Also

findGenesByEntrez

Other Mutation functions: `getMutations4DiseaseByIDs()`, `getMutations4DiseaseByName()`

Examples

```
hdoid<-'DOID:0060041'
entrez<-c("23859", "17754", "18673", "268566", "12293", "320840", "24012")
mdf<-getMutations4DiseaseByEntres(entrez, hdoid)
is.null(mdf)
```

---

getMutations4DiseaseByIDs

Get mutational information for Gene and Disease

Description

Function looks up the information for particular set of GeneIDs and disease HDOID.

Usage

```
getMutations4DiseaseByIDs(ids, hdoid)
```

Arguments

`ids`	internal GeneIDs to filter mutation data
`hdoid`	disease HDOID to get mutational association.

Details

This function then returns following features for all found genes:

- GeneID,
- MGI,
- MouseEntrez,
- MouseName,
- HumanName,
- HumanEntrez,
- HDOID,
- Disease,
- Chromosome,
- Position,
- Variant,
- FunctionClass,
- cDNAvariant,
- ProteinVariant,
- DENOVO,
- SFARI,
- EpilepsyGene,
- ClinVar,
- PMID,
- Paper

Value

`data.frame` with fields specified above.

See Also

Other Mutation functions: `getMutations4DiseaseByEntres()`, `getMutations4DiseaseByName()`

Examples

```
hdoid<-'DOID:0060041'
ids<-c(6,32,127,181,240,267,558)
mdf<-getMutations4DiseaseByIDs(ids, hdoid)
is.null(mdf)
```

`getMutations4DiseaseByName` *Get mutational information for Gene and Disease*

Description

Function lookups for provided values in Human Name, Mouse Name and Rat Name columns, extract known mutations, filter requested HDOID and return data.frame in the format described in `getMutations4DiseaseByIDs`.

Usage

```
getMutations4DiseaseByName(name, hdoid)
```

Arguments

`name`	gene names to look for
`hdoid`	disease HDOID to get mutational association.

Value

`data.frame` as described in `getMutations4DiseaseByIDs`.

See Also

findGenesByName

Other Mutation functions: `getMutations4DiseaseByEntres()`, `getMutations4DiseaseByIDs()`

Examples

```
hdoid<-'DOID:0060041'
name<-c("Dlg2", "Map1a", "Phb", "Gphn", "Cacna2d1", "Negr1", "Rgs7")
mdf<-getMutations4DiseaseByName(name, hdoid)
is.null(mdf)
```

`getPPIbyEntrez` *Extract the PPIs for my list of genes defined by Entrez IDs*

Description

Get Protein-Protein interactions (PPIs) for the set of ID. Function lookups for PPIs for the list of Entrez IDs and returns either “induced” (all available interactions for these genes) or “limited” (only interactions between genes specified in the query) table of interacting genes A and B, where A and B are respective Internal IDs.

Usage

```
getPPIbyEntrez(entrez, type = c("limited", "induced"))
```

Arguments

`entrez`	Entrez IDs
`type`	type of the PPI network should be either `induced` (for all the PPIs for specific genes, including external genes) or `limited` (for PPIs between the genes specified in the query). Type could be shortened to recognizable minimum like 'ind' or 'lim'.

Value

data.frame with interactors internal GeneID in columns A and B

See Also

`getPPIbyName()` and `getPPIbyIDs()` to get PPI `data.frame`, `getIGraphFromPPI()` to get igraph representation of the PPI `data.frame` and `getTableFromPPI()` to get interpretable table representation of the PPI `data.frame`.

Other PPI functions: `getIGraphFromPPI()`, `getPPIbyIDs()`, `getPPIbyIDs4BrainRegion()`, `getPPIbyIDs4Compartme`
`getPPIbyName()`, `getTableFromPPI()`

Examples

```
t <- getPPIbyEntrez(c(1739, 1740, 1742, 1741), type = "ind")
```

`getPPIbyIDs` *Extract the PPIs for my list of genes defined by GeneID*

Description

Get Protein-Protein interactions (PPIs) for the set of internal GeneIDs. Function lookups for PPIs for specific GeneID and returns either “induced” (all interaction for this GeneID) or “limited” (only interactions between GeneIDs specified in the query) table of A and B interacting genes, where A and B are respective GeneIDs.

Usage

```
getPPIbyIDs(ids, type = c("limited", "induced"))
```

Arguments

`ids`	Gene IDs
`type`	type of the PPI network should be either `induced` (for all the PPIs for specific genes, including external genes) or `limited` (for PPIs between the genes specified in the query). Type could be shortened to recognizable minimum like 'ind' or 'lim'.

Value

data.frame with interactors internal GeneID in columns A and B

See Also

`getPPIbyName()` and `getPPIbyEntrez()` to get PPI `data.frame`, `getIGraphFromPPI()` to get igraph representation of the PPI `data.frame` and `getTableFromPPI()` to get interpretable table representation of the PPI `data.frame`.

Other PPI functions: `getIGraphFromPPI()`, `getPPIbyEntrez()`, `getPPIbyIDs4BrainRegion()`, `getPPIbyIDs4Compartment()`, `getPPIbyName()`, `getTableFromPPI()`

Examples

```
t <- getPPIbyIDs(c(48, 585, 710), type = "limited") # (16 rows)
t <- getPPIbyIDs(c(48, 585, 710), type = "induced") # 306 rows
```

`getPPIbyIDs4BrainRegion`

Prepare induced or limited network for brain region

Description

Prepare PPI network for genes from your list annotated for specific brain region. Should be used with `findGenesByEntrez()`) or `findGenesByName()` functions to obtain list of internal IDs for your list of genes. Could be used with getAllGenes4BrainRegion functions to obtain all genes belonging to respective brain region. Function lookups the PPI table for gene IDs from the list and returns “limited” or “induced” interactors GeneIDs for the specified region.

Usage

```
getPPIbyIDs4BrainRegion(
  ids,
  brainRegion,
  taxID,
  type = c("limited", "induced")
)
```

Arguments

`ids`	gene IDs
`brainRegion`	region ID
`taxID`	taxon ID
`type`	type of the PPI network should be either `induced` (for all the PPIs for specific genes, including external genes) or `limited` (for PPIs between the genes specified in the query). Type could be shortened to recognizable minimum like ’ind’ or ’lim’.

Value

data.frame with interactors internal GeneID in columns A and B

See Also

Other PPI functions: `getIGraphFromPPI()`, `getPPIbyEntrez()`, `getPPIbyIDs()`, `getPPIbyIDs4Compartment()`, `getPPIbyName()`, `getTableFromPPI()`

Other BrainRegion functions: `getAllGenes4BrainRegion()`, `getBrainRegions()`, `getGenes4BrainRegion()`

Examples

```
# getting all genes for mouse Striatum
gns <- getAllGenes4BrainRegion(brainRegion = "Striatum", taxID = 10090)
head(gns)

# getting full PPI network for postsynaptic compartment
ppi <- getPPIbyIDs4BrainRegion(
    gns$GeneID,
    brainRegion = "Striatum",
    taxID = 10090,
    type = "limited"
)
head(ppi)
```

`getPPIbyIDs4Compartment` *Prepare induced or limited network for compartment*

Description

Prepare PPI network for genes from your list annotated for specific compartment. Should be used with findGenesByEntrez or findGenesByName functions to obtain list of internal IDs for your list of genes. Could be used with getAllGenes4Compartment functions to obtain all genes belonging to respective compartment. Function lookups the PPI table for gene IDs from the list and returns “limited” or “induced” interactors GeneIDs from the same compartment.

Usage

```
getPPIbyIDs4Compartment(ids, compartmentID, type = c("limited", "induced"))
```

Arguments

`ids`	internal gene IDs
`compartmentID`	ID for specific compartment
`type`	type of the PPI network should be either ‘induced‘ (for all the PPIs for specific genes, including external genes) or ‘limited‘ (for PPIs between the genes specified in the query). Type could be shortened to recognizable minimum like ’ind’ or ’lim’.

Value

data.frame with interactors internal GeneID in columns A and B

See Also

Other PPI functions: `getIGraphFromPPI()`, `getPPIbyEntrez()`, `getPPIbyIDs()`, `getPPIbyIDs4BrainRegion()`, `getPPIbyName()`, `getTableFromPPI()`

Other Compartment functions: `getAllGenes4Compartment()`, `getCompartments()`, `getGenes4Compartment()`

Examples

```
ppi <- getPPIbyIDs4Compartment(c(1, 15, 156, 1500, 3000, 7000),
    compartmentID = 1, type = "induced"
) # 201 rows
```

`getPPIbyName` *Extract the PPIs for my list of genes defined by Gene name*

Description

Get Protein-Protein interactions (PPIs) for the set of gene names. Function lookups for PPIs for the list of GeneIDs and returns either "induced" (all interaction for this GeneID) or "limited" (only interactions between GeneIDs specified in the query) table of interacting genes A and B, where A and B are respective gene names.

Usage

```
getPPIbyName(name, type = c("limited", "induced"))
```

Arguments

`name`	Gene names
`type`	type of the PPI network should be either `induced` (for all the PPIs for specific genes, including external genes) or `limited` (for PPIs between the genes specified in the query). Type could be shortened to recognizable minimum like 'ind' or 'lim'.

Value

data.frame with interactors internal GeneID in columns A and B

See Also

`getPPIbyEntrez()` and `getPPIbyIDs()` to get PPI `data.frame`, `getIGraphFromPPI()` to get igraph representation of the PPI `data.frame` and `getTableFromPPI()` to get interpretable table representation of the PPI `data.frame`.

Other PPI functions: `getIGraphFromPPI()`, `getPPIbyEntrez()`, `getPPIbyIDs()`, `getPPIbyIDs4BrainRegion()`, `getPPIbyIDs4Compartment()`, `getTableFromPPI()`

Examples

```
t <- getPPIbyName(
    c("CASK", "DLG4", "GRIN2A", "GRIN2B", "GRIN1"),
    type = "lim"
)
```

`getPapers` *Get synaptome papers overview*

Description

Get synaptome papers overview

Usage

```
getPapers()
```

Value

data.frame with following columns:

- PaperPMID
- SpeciesTaxID
- Year
- Name
- Localisation
- BrainRegion
- Method
- Ngenes

Examples

```
p <- getPapers()
head(p)
```

`getTableFromPPI` *Get table representation of the PPI.*

Description

Combine information from PPI `data.frame` obtained with functions like `getPPIbyName` or `getPPIbyEntrez` with information about genes obtained from `getGenesByID` to make interpretable table representation.

Usage

```
getTableFromPPI(ppi)
```

Arguments

`ppi`	PPI `data.frame` with columns A and B, obtained from functions like `getPPIbyName`

Value

table with the following columns added for both interactors (A and B):

- GeneID internal database ID
- MGI MGI ID
- HumanEntrez Human Entrez ID
- MouseEntrez Mouse Entrez ID
- HumanName Human gene name
- MouseName Mouse gene name
- RatEntrez Rat Entrez ID
- Rat Name Rat gene name

See Also

`getPPIbyName()`, `getPPIbyEntrez()` and `getPPIbyIDs()` to get PPI `data.frame`, `getIGraphFromPPI()` to get igraph representation of the PPI `data.frame`.

Other PPI functions: `getIGraphFromPPI()`, `getPPIbyEntrez()`, `getPPIbyIDs()`, `getPPIbyIDs4BrainRegion()`, `getPPIbyIDs4Compartment()`, `getPPIbyName()`

Examples

```
tbl <- getTableFromPPI(getPPIbyIDs(c(48, 585, 710), type = "limited"))
```

`graphFromSynaptomeByEntrez` *Utility function to create network from* `synaptome.db` *data*

Description

Utility function to create network from `synaptome.db` data

Usage

```
graphFromSynaptomeByEntrez(
  entrez,
  LCC = TRUE,
  simplify = TRUE,
  type = c("limited", "induced")
)
```

Arguments

`entrez`	vector of EntrezIDs for network vertices
`LCC`	if TRUE only largest connected component is returned
`simplify`	if TRUE loops and multiple edges will be removed
`type`	type of the PPI network should be either ‘induced‘ (for all the PPIs for specific genes, including external genes) or ‘limited‘ (for PPIs between the genes specified in the query). Type could be shortened to recognizable minimum like ’ind’ or ’lim’.

Value

network defined by the gene table

Examples

```
library(synaptome.db)
cid<-match('Presynaptic', getCompartments()$Name)
geneTable<-getAllGenes4Compartment(cid)
gg<-graphFromSynaptomeByEntrez(geneTable$HumanEntrez)
```

---

graphFromSynaptomeGeneTable

Utility function to create network from synaptome.db data

Description

Utility function to create network from `synaptome.db` data

Usage

```
graphFromSynaptomeGeneTable(
  geneTable,
  LCC = FALSE,
  simplify = FALSE,
  type = c("limited", "induced")
)
```

Arguments

`geneTable`	data.frame described in `getGenesByID`
`LCC`	if TRUE only largest connected component is returned
`simplify`	if TRUE loops and multiple edges will be removed
`type`	type of the PPI network should be either ‘induced‘ (for all the PPIs for specific genes, including external genes) or ‘limited‘ (for PPIs between the genes specified in the query). Type could be shortened to recognizable minimum like ’ind’ or ’lim’.

Value

network defined by the gene table

See Also

[getPPIbyIDs()]

Examples

```
library(synaptome.db)
cid<-match('Presynaptic', getCompartments()$Name)
geneTable<-getAllGenes4Compartment(cid)
gg<-graphFromSynaptomeGeneTable(geneTable)
```

`synaptome.db` *synaptome.db: programmatic access to the Synaptic proteome database*

Description

The package obtains a local copy of the Synaptic proteome database from AnnotationHub package `synaptome.data` and provides a set of utility R functions to query and analyse its content.

Author(s)

Maintainer: Oksana Sorokina <`oksana.sorokina@ed.ac.uk`>

Authors:

- Anatoly Sorokin <`lptolik@gmail.com`>
- J. Douglas Armstrong

References

\insertRefSorokina:2021hlsynaptome.db

See Also

Useful links:

- Report bugs at `https://github.com/lptolik/synaptome.db/issues/`

Index