guildifyR

January 26, 2018

---

check.parameters	*Check whether provided species, tissue and scoring parameter values are valid*

Description

Check whether provided species, tissue and scoring parameter values are valid

Usage

```
check.parameters(species, tissue, scoring.options = NULL)
```

---

get.species.info	*Get info on available species and tissues*

Description

Get info on available species and tissues

Usage

```
get.species.info()
```

Value

result List containing available species tax ids and names for tissue-specific networks

Examples

```
result = get.species.info()
```

---

get.url   *Return the base url*

Description

Return the base url

Usage

```
get.url()
```

Value

html address

---

query   *Get protein/gene info associated with the query keywords*

Description

Get protein/gene info associated with the query keywords

Usage

```
query(keywords, species = "9606", tissue = "All")
```

Arguments

keywords	Text containing description of a phenotype or list of genes (seperated by ;)
species	Species tax identifier (9606: human, 10090: mouse, etc., see get.species.info method)
tissues	Tissue identifier (All, brain, liver, etc., see get.species.info method)

Value

result.table Data frame containing list of matching proteins/genes and their description

Examples

```
result.table = query("alzheimer", species="10090", tissue="All")
```

---

retrieve.job *Retrieve results using job id*

Description

Retrieve results using job id

Usage

```
retrieve.job(job.id, n.top = NULL, fetch.files = F, output.dir = "./")
```

Arguments

job.id	Job id
n.top	Number of top proteins to retrieve. If NULL top enriched proteins are provided (upto 500 proteins)
fetch.files	Flag to fetch result files from server and save them locally in output.dir
output.dir	Directory to save the ranking, function, subnetwork and drug info files fetched from the server

Value

result List containing scores of top-ranking proteins, functions enriched among top-ranking proteins, drugs targeting top-ranking proteins, top-ranking subnetwork cutoff

Examples

```
result = retrieve.job(job.id)
names(result)
head(result$score.table)
```

---

retrieve.overlap *Retrieve overlap between two results using two job ids*

Description

Retrieve overlap between two results using two job ids

Usage

```
retrieve.overlap(job.id1, job.id2, fetch.files = F, output.dir = "./")
```

Arguments

job.id1	Job id 1
job.id2	Job id 2
fetch.files	Flag to fetch result files from server and save them locally in output.dir
output.dir	Directory to save the ranking, function, subnetwork and drug info files fetched from the server

Value

result List containing scores of common top-ranking proteins, common functions enriched among top-ranking proteins, drugs targeting common top-ranking proteins (Note that the number of top-ranking proteins and common functions are limited to 500)

Examples

```
result = retrieve.overlap(job.id1, job.id2)
names(result)
head(result$score.table)
```

---

submit.job   *Submit job using the guildifyR query result table*

Description

Submit job using the guildifyR query result table

Usage

```
submit.job(result.table, species, tissue, scoring.options = list(netcombo =
    T))
```

Arguments

scoring.options

Prioritization method options (default NetCombo)

query.table   Data frame contaning information from guildifyR query (user entitiy info, network info, source etc...)

Value

job id

Examples

```
species="10090"
tissue="All"
result.table = query("alzheimer", species)
NetScore
scoring.options = list(netscore=T, repetitionSelector=3, iterationSelector=2)
NetZcore
scoring.options = list(netzcore=T, repetitionZelector=3)
NetShort
scoring.options = list(netshort=T)
NetCombo
scoring.options = list(netcombo=T)
job.id = submit.job(result.table, species, tissue, scoring.options)
```

Index