Package 'tigreBrowserWriter'

October 14, 2022

Version 0.1.5

Date 2018-06-07

Title 'tigreBrowser' Database Writer

Depends R (>= 2.14.0)

Imports DBI, RSQLite

Description Write modelling results into a database for
'tigreBrowser', a web-based tool for browsing figures and summary
data of independent model fits, such as Gaussian process models
fitted for each gene or other genomic element. The browser is
available at <https://github.com/PROBIC/tigreBrowser>.

License AGPL-3

URL https://github.com/PROBIC/tigreBrowserWriter

BugReports https://github.com/PROBIC/tigreBrowserWriter/issues

RoxygenNote 5.0.1

NeedsCompilation no

Author Antti Honkela [aut, cre],
Miika-Petteri Matikainen [aut]

Maintainer Antti Honkela <antti.honkela@helsinki.fi>

Repository CRAN

Date/Publication 2018-06-07 21:07:56 UTC

R topics documented:

---

closeDb *Finalise and close the database*

Description

Finalise and close the database

Usage

```
closeDb(db, experimentSet = "")
```

Arguments

db   Database object created by [initializeDb](#initializeDb)

experimentSet   Name of the experiment set for all the experiments (optional)

Examples

```
db <- initializeDb("", "My Dataset")
# ...
closeDb(db)
```

---

initializeDb *Create and initialize a database*

Description

Create and initialize a database

Usage

```
initializeDb(dbPath, datasetName, datasetSpecies = "", datasetSource = "",
  datasetPlatform = "", datasetDescription = "", datasetSaveLocation = "",
  datasetFigureFilename = "")
```

Arguments

dbPath   Path to the database file to be created. Empty string

datasetName   Name of the data set

datasetSpecies   Optional data set metadata: species

datasetSource   Optional data set metadata: source

datasetPlatform

                    Optional data set metadata: platform

```
datasetDescription
                Optional data set metadata: description
datasetSaveLocation
                Optional data set metadata: save location
datasetFigureFilename
                Optional data set metadata: figure file name
```

Value

A database object db needed by other tigreBrowserWriter functions

Examples

```
## Not run:
  # Create a real database to a file
  db <- initializeDb("/path/to/the/database.sqlite", "My Dataset")
  closeDb(db)

## End(Not run)

  # Create a temporary database to be deleted at the end
  db <- initializeDb("", "My Dataset")
  closeDb(db)
```

---

insertAliases   *Insert aliases*

Description

Insert aliases

Usage

```
insertAliases(db, aliasType, aliases, aliasSource = "",
  aliasDescription = "")
```

Arguments

db	Database object created by initializeDb
aliasType	Name of the alias
aliases	A vector of aliases with names giving the primary identifiers
aliasSource	Optional alias metadata: source
aliasDescription	
	Optional alias metadata: description

Value

An updated database object db

Examples

```
db <- initializeDb("", "My Dataset")
aliases <- c("aliasA", "aliasB", "aliasC")
names(aliases) <- c("A", "B", "C")
db <- insertAliases(db, "testalias", aliases)
closeDb(db)
```

---

insertFigureData *Insert figure data directly to the database*

Description

Insert figure data directly to the database

Usage

```
insertFigureData(db, experimentName, regulatorName, filenames, name = "",
  description = "", priority = 0)
```

Arguments

db	Database object created by [initializeDb](#)
experimentName	Experiment name
regulatorName	Regulator name (more detailed experiment identifier)
filenames	A list of file names of PNG figures. names of the list must correspond to the names of the entities the figures are for.
name	Optional figure name
description	Optional figure description
priority	Integer priority used for sorting figures (default: 0)

Value

An updated database object db

Examples

```
db <- initializeDb("", "My Dataset")
logl <- c(-4.0, -2.0, 0.0)
names(logl) <- c("A", "B", "C")
baselogl <- c(1.0, -1.0, 4.0)
names(baselogl) <- names(logl)
db <- insertResults(db, "testexperiment", "testregulator", "",
                    logl, baselineloglikelihoods=baselogl)
# Generate a dummy plot (reused for all data elements for simplicity)
examplefile <- tempfile("plot", fileext=".png")
png(examplefile)
```

```
plot(c(0, 1), c(0, 1))
dev.off()
figures <- list(A=examplefile, B=examplefile, C=examplefile)
db <- insertFigureData(db, "testexperiment", "testregulator",
                       figures)
closeDb(db)
```

---

insertFigures   *Insert figure links*

Description

Insert figure links

Usage

```
insertFigures(db, experimentName, regulatorName, filename, name = "",
  description = "", priority = 0)
```

Arguments

db	Database object created by [initializeDb](#)
experimentName	Experiment name
regulatorName	Regulator name (more detailed experiment identifier)
filename	URL path to figures. The path can contain the special form ${probe_name} which will be substituted by the name of the entity (gene, ...) by the browser.
name	Optional figure name
description	Optional figure description
priority	Integer priority used for sorting figures (default: 0)

Value

An updated database object db

Examples

```
db <- initializeDb("", "My Dataset")
logl <- c(-4.0, -2.0, 0.0)
names(logl) <- c("A", "B", "C")
baselogl <- c(1.0, -1.0, 4.0)
names(baselogl) <- names(logl)
db <- insertResults(db, "testexperiment", "testregulator", "",
                    logl, baselineloglikelihoods=baselogl)
db <- insertFigures(db, "testexperiment", "testregulator",
                    "http://foo.invalid/path/${probe_name}_fit.png")
closeDb(db)
```

insertResults   *Insert results*

Description

Insert results

Usage

```
insertResults(db, experimentName, regulatorName, figurePath, loglikelihoods,
  baselineloglikelihoods = NA, experimentDesc = "", loopVariable = 2,
  modelTranslation = FALSE, numberOfParameters = NA, parameterNames = NA,
  experimentProducer = "", experimentTimestamp = "", parameters = NA)
```

Arguments

db	Database object created by [initializeDb](#)
experimentName	Experiment name
regulatorName	Regulator name (more detailed experiment identifier)
figurePath	URL path to figures
loglikelihoods	A vector of log-likelihoods of elements identified by names
baselineloglikelihoods	
	A vector of baseline log-likelihoods of elements identified by names (optional)
experimentDesc	Optional experiment description
loopVariable	Optional: Loop variable (1=regulator, 2=target (default))
modelTranslation	
	Optional: For gene regulation models, is translation modelled
numberOfParameters	
	Optional: Number of parameters
parameterNames	Optional: Parameter names
experimentProducer	
	Optional: Experiment producer (string)
experimentTimestamp	
	Optional: Experiment timestamp (string)
parameters	Optional: A vector of parameter values for the model

Value

An updated database object db

Examples

```
db <- initializeDb("", "My Dataset")
logl <- c(-4.0, -2.0, 0.0)
names(logl) <- c("A", "B", "C")
baselogl <- c(1.0, -1.0, 4.0)
names(baselogl) <- names(logl)
db <- insertResults(db, "testexperiment", "testregulator", "",
                    logl, baselineloglikelihoods=baselogl)
closeDb(db)
```

---

insertSupplementaryData

Insert supplementary data

Description

Insert supplementary data

Usage

```
insertSupplementaryData(db, name, suppData, regulatorName = NA, source = "",
  platform = "", description = "")
```

Arguments

db	Database object created by [initializeDb](#)
name	Name of the supplementary data
suppData	A vector of supplementary data of elements identified by names
regulatorName	Regulator name the data links to (optional)
source	Optional annotation: source
platform	Optional annotation: platform
description	Optional annotation: description

Value

An updated database object db

Examples

```
db <- initializeDb("", "My Dataset")
suppdata <- c(1, 2, 3)
names(suppdata) <- c("A", "B", "C")
db <- insertSupplementaryData(db, "supptest", suppdata)
boolsupp <- c(TRUE, TRUE, FALSE)
names(boolsupp) <- names(suppdata)
db <- insertSupplementaryData(db, "supptest_bool", boolsupp)
closeDb(db)
```

insertZScores   *Insert data z-scores used for filtering*

Description

Insert data z-scores used for filtering

Usage

```
insertZScores(db, zscores)
```

Arguments

db	Database object created by `initializeDb`
zscores	A vector of z-scores of elements identified by names

Value

An updated database object db

Examples

```
db <- initializeDb("", "My Dataset")
zscores <- c(1, 2, 3)
names(zscores) <- c("A", "B", "C")
db <- insertZScores(db, zscores)
closeDb(db)
```

Index