Package 'Eunomia'

October 6, 2021

Type Package

Title A Standard Dataset in the OMOP Common Data Model

Version 1.0.1

Date 2020-10-21

Maintainer Frank DeFalco <fdefalco@ohdsi.org>

Description A sample dataset in the OMOP (Observational Medical Outcomes Partnership) Common Data Model (CDM) format. The CDM enables uniform storage of observational health care data, and is widely used for health care analytics. 'Eunomia' contains simulated data as well as a subset of the OMOP Vocabulary, and enables testing of additional packages and is used for educational and demonstration purposes.

License Apache License 2.0

URL https://github.com/OHDSI/Eunomia

BugReports https://github.com/OHDSI/Eunomia/issues

Depends DatabaseConnector (>= 2.2.0)

Imports SqlRender,
RSQLite (> 2.1.1),
rJava

Suggests testthat

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

R topics documented:

createCohorts *Construct cohorts*

Description

Creates a set of predefined cohorts in a cohort table. WARNING: this will delete all existing cohorts in the table!

Usage

```
createCohorts(
  connectionDetails,
  cdmDatabaseSchema = "main",
  cohortDatabaseSchema = "main",
  cohortTable = "cohort"
)
```

Arguments

connectionDetails	The connection details to connect to the (Eunomia) database.
cdmDatabaseSchema	The name of the database schema holding the CDM data.
cohortDatabaseSchema	The name of the database schema where the cohorts will be written.
cohortTable	The name of the table in the cohortDatabaseSchema where the cohorts will be written.

Value

A data frame listing all created cohorts.

Examples

```
connectionDetails <- getEunomiaConnectionDetails()
createCohorts(connectionDetails)

connection <- connect(connectionDetails)

sql <- "SELECT COUNT(*)
FROM main.cohort
WHERE cohort_definition_id = 1;"

renderTranslateQuerySql(connection, sql)

disconnect(connection)
```

exportToCsv *Extract the Eunomia database to CSV files*

Description

Extract the Eunomia database to CSV files

Usage

```
exportToCsv(
  outputFolder = file.path(getwd(), "csv"),
  connectionDetails = getEunomiaConnectionDetails()
)
```

Arguments

outputFolder	A folder where the CSV files will be written.
connectionDetails	Connection details for the Eunomia database. Defaults to a fresh Eunomia database.

Examples

```
# For this example we'll create a temp folder:
folder <- tempfile()
dir.create(folder)

exportToCsv(folder)

list.files(folder)

 #  [1] "CARE_SITE.csv"             "CDM_SOURCE.csv"            "COHORT.csv"
 #  [4] "COHORT_ATTRIBUTE.csv"      "CONCEPT.csv"               "CONCEPT_ANCESTOR.csv"
 #  [7] "CONCEPT_CLASS.csv"         "CONCEPT_RELATIONSHIP.csv"  "CONCEPT_SYNONYM.csv"
 # [10] "CONDITION_ERA.csv"         "CONDITION_OCCURRENCE.csv"  "COST.csv"
 # [13] "DEATH.csv"                 "DEVICE_EXPOSURE.csv"       "DOMAIN.csv"
 # [16] "DOSE_ERA.csv"              "DRUG_ERA.csv"              "DRUG_EXPOSURE.csv"
 # [19] "DRUG_STRENGTH.csv"         "FACT_RELATIONSHIP.csv"     "LOCATION.csv"
 # [22] "MEASUREMENT.csv"           "METADATA.csv"              "NOTE.csv"
 # [25] "NOTE_NLP.csv"              "OBSERVATION.csv"           "OBSERVATION_PERIOD.csv"
 # [28] "PAYER_PLAN_PERIOD.csv"     "PERSON.csv"                "PROCEDURE_OCCURRENCE.csv"
 # [31] "PROVIDER.csv"              "RELATIONSHIP.csv"          "SOURCE_TO_CONCEPT_MAP.csv"
 # [34] "SPECIMEN.csv"              "VISIT_DETAIL.csv"          "VISIT_OCCURRENCE.csv"
 # [37] "VOCABULARY.csv"

 # Cleaning up the temp folder used in this example:
 unlink(folder, recursive = TRUE)
```

getEunomiaConnectionDetails *Get Eunomia Connection Details*

Description

Creates a copy of the Eunomia database, and provides details for connecting to that copy.

Usage

```
getEunomiaConnectionDetails(databaseFile = tempfile(fileext = ".sqlite"))
```

Arguments

databaseFile	The path where the database file will be copied to. By default, the database will be copied to a temporary folder, and will be deleted at the end of the R session.

Value

A ConnectionDetails object, to be used with the DatabaseConnector package.

Examples

```
connectionDetails <- getEunomiaConnectionDetails()
connection <- connect(connectionDetails)
querySql(connection, "SELECT COUNT(*) FROM person;")
disconnect(connection)
```

supportsJava8 *Determine if Java virtual machine supports Java*

Description

Tests Java virtual machine (JVM) java.version system property to check if version >= 8.

Usage

```
supportsJava8()
```

Value

Returns TRUE if JVM supports Java >= 8.

Examples

```
supportsJava8()
```

Index