Package ‘CirceR’

April 18, 2024

<table>
<thead>
<tr>
<th>Type</th>
<th>Package</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Construct Cohort Inclusion and Restriction Criteria Expressions</td>
</tr>
<tr>
<td>Version</td>
<td>1.3.3</td>
</tr>
<tr>
<td>Date</td>
<td>2024-03-20</td>
</tr>
<tr>
<td>Maintainer</td>
<td>Chris Knoll <a href="mailto:cknoll@ohdsi.org">cknoll@ohdsi.org</a></td>
</tr>
<tr>
<td>Description</td>
<td>Wraps the 'CIRCE' (<a href="https://github.com/ohdsi/circe-be">https://github.com/ohdsi/circe-be</a>) 'Java' library allowing cohort definition expressions to be edited and converted to 'Markdown' or 'SQL'.</td>
</tr>
<tr>
<td>License</td>
<td>Apache License 2.0</td>
</tr>
<tr>
<td>BugReports</td>
<td><a href="https://github.com/OHDSI/CirceR/issues/">https://github.com/OHDSI/CirceR/issues/</a></td>
</tr>
<tr>
<td>SystemRequirements</td>
<td>Java (&gt;= 8)</td>
</tr>
<tr>
<td>Depends</td>
<td>R (&gt;= 3.5.0)</td>
</tr>
<tr>
<td>Imports</td>
<td>rJava, RJSONIO</td>
</tr>
<tr>
<td>Suggests</td>
<td>testthat, knitr, rmarkdown</td>
</tr>
<tr>
<td>NeedsCompilation</td>
<td>no</td>
</tr>
<tr>
<td>RoxygenNote</td>
<td>7.3.1</td>
</tr>
<tr>
<td>Language</td>
<td>en-US</td>
</tr>
<tr>
<td>Encoding</td>
<td>UTF-8</td>
</tr>
<tr>
<td>Author</td>
<td>Chris Knoll [aut, cre], Martijn Schuemie [aut]</td>
</tr>
<tr>
<td>Repository</td>
<td>CRAN</td>
</tr>
<tr>
<td>Date/Publication</td>
<td>2024-04-18 19:42:49 UTC</td>
</tr>
</tbody>
</table>

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Documented Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>buildCohortQuery</td>
<td>2</td>
</tr>
<tr>
<td>buildConceptSetQuery</td>
<td>2</td>
</tr>
<tr>
<td>cohortDefinitionJson</td>
<td>3</td>
</tr>
</tbody>
</table>
buildCohortQuery

**Build cohort SQL**

**Description**
Generates the OMOP CDM Sql to generate the cohort expression

**Usage**
buildCohortQuery(expression, options)

**Arguments**
- **expression**: result of cohortExpressionFromJson containing the cohort expression.
- **options**: The options built using createGenerateOptions()

**Value**
A character vector containing the OHDSI Sql for the cohort expression

buildConceptSetQuery

**Build conceptset SQL**

**Description**
Generates the OMOP CDM Sql to resolve the concept set expression

**Usage**
buildConceptSetQuery(conceptSetJson)

**Arguments**
- **conceptSetJson**: a string containing the JSON for the conceptset expression.

**Value**
A character vector containing the OHDSI Sql for the conceptset expression
cohortDefinitionJson

---

**cohortDefinitionJson**  *An example cohort definition*

---

**Description**

*An example cohort definition*

**Usage**

data(cohortDefinitionJson)

**Format**

*A character vector containing a JSON string*

**Details**

*An example cohort expression in JSON format*

---

cohortExpressionFromJson

---

*Rend read JSON into a CohortExpression instance*

---

**Description**

*Reads a String (json) and deserializes it into a org.ohdsi.circe.cohortdefinition.CohortExpression*

**Usage**

cohortExpressionFromJson(expressionJson)

**Arguments**

expressionJson  A character vector containing the cohort expression.

**Value**

*A java instance of org.ohdsi.circe.cohortdefinition.CohortExpression.*
**cohortPrintFriendly**  
*Render cohort print-friendly*

**Description**
Generates a print-friendly (human-readable) representation of the cohort definition. This can for example be used in a study protocol.

**Usage**
```r
cohortPrintFriendly(expression)
```

**Arguments**
- `expression`  
  A character vector or result of `cohortExpressionFromJson` containing the cohort expression.

**Value**
A character vector containing the markdown.

---

**conceptSetExpressionFromJson**  
*Render read JSON into a ConceptSetExpression instance*

**Description**
Reads a String (json) and deserializes it into a `org.ohdsi.circe.vocabulary.CohortExpression`

**Usage**
```r
conceptSetExpressionFromJson(expressionJson)
```

**Arguments**
- `expressionJson`  
  A character vector containing the cohort expression.

**Value**
A java instance of `org.ohdsi.circe.cohortdefinition.CohortExpression`.
### conceptSetJson

<table>
<thead>
<tr>
<th>conceptSetJson</th>
<th>An example concept set</th>
</tr>
</thead>
</table>

**Description**

An example concept set

**Usage**

```r
data(conceptSetJson)
```

**Format**

A character vector containing a JSON string

**Details**

An example concept set in JSON format

### conceptSetListJson

<table>
<thead>
<tr>
<th>conceptSetListJson</th>
<th>An example concept set list</th>
</tr>
</thead>
</table>

**Description**

An example concept set list

**Usage**

```r
data(conceptSetListJson)
```

**Format**

A character vector containing a JSON string

**Details**

An example concept set list in JSON format
**conceptSetListPrintFriendly**

*Render conceptSet array for print-friendly*

**Description**

Generates a print-friendly (human-readable) representation of an array of concept sets. This can for example be used in a study protocol.

**Usage**

```r
conceptSetListPrintFriendly(conceptSetList)
```

**Arguments**

- `conceptSetList`: A ConceptSet[] (from cohortExpression.conceptSets), a JSON string, or a list from RJSONIO::fromJson

**Value**

A character vector containing the markdown.

---

**conceptSetPrintFriendly**

*Render conceptSet array for print-friendly*

**Description**

Generates a print-friendly (human-readable) representation of a single concept set. This can for example be used in a study protocol.

**Usage**

```r
conceptSetPrintFriendly(conceptSet)
```

**Arguments**

- `conceptSet`: A ConceptSet (from cohortExpression.conceptSets[i]), a JSON string, or a list from RJSONIO::fromJson

**Value**

A character vector containing the markdown.
**createGenerateOptions**  

**Create Generation Options**

**Description**

Creates the generation options object for use in generateSql().

**Usage**

```javascript
createGenerateOptions(
  cohortIdFieldName,
  cohortId,
  cdmSchema,
  targetTable,
  resultSchema,
  vocabularySchema,
  generateStats
)
```

**Arguments**

- **cohortIdFieldName**
  The field that contains the cohortId in the cohort table.
- **cohortId**
  The generated cohort ID.
- **cdmSchema**
  The value of the CDM schema.
- **targetTable**
  The cohort table name.
- **resultSchema**
  The schema the cohort table belongs to.
- **vocabularySchema**
  The schema of the vocabulary tables (defaults to cdmSchema)
- **generateStats**
  A boolean representing if the query should include inclusion rule statistics calculation

**Value**

A jObj that can be passed to generateSql().
## Index

* **datasets**
  - cohortDefinitionJson, 3
  - conceptSetJson, 5
  - conceptSetListJson, 5

buildCohortQuery, 2
buildConceptSetQuery, 2

cohortDefinitionJson, 3
cohortExpressionFromJson, 3
cohortPrintFriendly, 4
conceptSetExpressionFromJson, 4
conceptSetJson, 5
conceptSetListJson, 5
conceptSetListPrintFriendly, 6
conceptSetPrintFriendly, 6
createGenerateOptions, 7