

# Package ‘ROhdsiWebApi’

November 7, 2021

**Type** Package

**Title** Interacting With an OHDSI WebApi Instance

**Version** 1.2.2

**Date** 2021-11-07

**Maintainer** Gowtham Rao <rao@ohdsi.org>

**Description** ROhdsiWebApi is an R based interface to OHDSI WebAPI services, and performs GET/PULL/POST/DELETE calls via the WebApi.

All objects starting from R or output to R - are analysis ready R-objects like list and data.frame.

The package handles the intermediary steps by converting R-objects to JSON and vice versa.

**License** Apache License 2.0

**Depends** R (>= 3.1.0),

**Imports** checkmate,  
dplyr,  
httr (>= 1.4.1),  
lifecycle,  
lubridate,  
openxlsx (>= 4.2.1),  
purrr,  
readr,  
RJSONIO,  
scales,  
SqlRender,  
stringr,  
tibble,  
tidyr,  
rlang

**Suggests** testthat,  
knitr,  
rmarkdown,  
httpuv,  
jsonlite,  
withr,  
httptest,  
stringi

**URL** <https://ohdsi.github.io/ROhdsiWebApi/>, <https://github.com/OHDSI/ROhdsiWebApi>

**BugReports** <https://github.com/OHDSI/ROhdsiWebApi/issues>

**NeedsCompilation** no

**RoxygenNote** 7.1.2

**Encoding** UTF-8

**VignetteBuilder** knitr

**Language** en-US

**RdMacros** lifecycle

## R topics documented:

authorizeWebApi . . . . .	4
cancelCharacterizationGeneration . . . . .	5
cancelCohortGeneration . . . . .	5
cancelGeneration . . . . .	6
cancelIncidenceRateGeneration . . . . .	7
cancelPathwayGeneration . . . . .	8
checkInputFileEncoding . . . . .	8
convertConceptSetDefinitionToTable . . . . .	9
createConceptSetWorkbook . . . . .	10
deleteCharacterizationDefinition . . . . .	10
deleteCohortDefinition . . . . .	11
deleteConceptSetDefinition . . . . .	12
deleteDefinition . . . . .	12
deleteEstimationDefinition . . . . .	13
deleteIncidenceRateDefinition . . . . .	14
deletePathwayDefinition . . . . .	14
deletePredictionDefinition . . . . .	15
detectCharacterizationsByName . . . . .	16
detectCohortsByName . . . . .	16
detectConceptSetsByName . . . . .	17
detectEstimationsByName . . . . .	18
detectIncidenceRatesByName . . . . .	19
detectPathwaysByName . . . . .	19
detectPredictionsByName . . . . .	20
existsCharacterizationName . . . . .	21
existsCohortName . . . . .	22
existsConceptSetName . . . . .	22
existsEstimationName . . . . .	23
existsIncidenceRateName . . . . .	24
existsPathwayName . . . . .	24
existsPredictionName . . . . .	25
getCdmSources . . . . .	26
getCharacterizationDefinition . . . . .	26
getCharacterizationDefinitionsMetaData . . . . .	27
getCharacterizationGenerationInformation . . . . .	28
getCharacterizationResults . . . . .	28
getCohortDefinition . . . . .	29
getCohortDefinitionExpression . . . . .	30
getCohortDefinitionName . . . . .	30
getCohortDefinitionsMetaData . . . . .	31
getCohortDefinitionSql . . . . .	32

getCohortGenerationInformation . . . . .	32
getCohortInclusionRulesAndCounts . . . . .	33
getCohortResults . . . . .	34
getCohortSql . . . . .	34
getConcepts . . . . .	35
getConceptSetDefinition . . . . .	36
getConceptSetDefinitionsMetaData . . . . .	37
getDefinition . . . . .	37
getDefinitionsMetadata . . . . .	38
getEstimationDefinition . . . . .	39
getEstimationDefinitionsMetaData . . . . .	40
getGenerationInformation . . . . .	40
getIncidenceRateDefinition . . . . .	41
getIncidenceRateDefinitionsMetaData . . . . .	42
getIncidenceRateGenerationInformation . . . . .	43
getIncidenceRateResults . . . . .	43
getPathwayDefinition . . . . .	44
getPathwayDefinitionsMetaData . . . . .	45
getPathwayGenerationInformation . . . . .	45
getPathwayResults . . . . .	46
getPersonProfile . . . . .	47
getPredictionDefinition . . . . .	48
getPredictionDefinitionsMetaData . . . . .	48
getPriorityVocabularyKey . . . . .	49
getResults . . . . .	50
getSourceConcepts . . . . .	50
getWebApiVersion . . . . .	51
insertCohortDefinitionInPackage . . . . .	52
insertCohortDefinitionSetInPackage . . . . .	53
invokeCharacterizationGeneration . . . . .	54
invokeCohortGeneration . . . . .	55
invokeGeneration . . . . .	55
invokeIncidenceRateGeneration . . . . .	56
invokePathwayGeneration . . . . .	57
isValidCharacterizationId . . . . .	58
isValidCohortId . . . . .	58
isValidConceptSetId . . . . .	59
isValidEstimationId . . . . .	60
isValidId . . . . .	60
isValidIncidenceRateId . . . . .	61
isValidPathwayId . . . . .	62
isValidPredictionId . . . . .	62
isValidSourceKey . . . . .	63
postCharacterizationDefinition . . . . .	64
postCohortDefinition . . . . .	65
postConceptSetDefinition . . . . .	65
postDefinition . . . . .	66
postEstimationDefinition . . . . .	67
postIncidenceRateDefinition . . . . .	68
postPathwayDefinition . . . . .	69
postPredictionDefinition . . . . .	69
resolveConceptSet . . . . .	70

setAuthHeader . . . . .	71
updateCohortDefinition . . . . .	71
updateConceptSetDefinition . . . . .	72
updateDefinition . . . . .	73

<b>Index</b>	<b>74</b>
--------------	-----------

---

authorizeWebApi	<i>Authorize ROhdsiWebApi to access a protected instance of WebAPI. Authorize the ROhdsiWebApi package to access WebAPI on behalf of the user. This can be done with any of the auth methods described below. authorizeWebApi will use attempt to retrieve, cache, and update a token which will grant access to webAPI by all subsequent requests made by the package.</i>
-----------------	---

---

## Description

Authorize ROhdsiWebApi to access a protected instance of WebAPI. Authorize the ROhdsiWebApi package to access WebAPI on behalf of the user. This can be done with any of the auth methods described below. authorizeWebApi will use attempt to retrieve, cache, and update a token which will grant access to webAPI by all subsequent requests made by the package.

## Usage

```
authorizeWebApi(
  baseUrl,
  authMethod,
  webApiUsername = NULL,
  webApiPassword = NULL
)
```

## Arguments

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
authMethod	<p>The method used for authentication to WebAPI. Options are</p> <ul style="list-style-type: none"> <li>• "db" Database authentication using Atlas/WebAPI built in auth</li> <li>• "ad" Active Directory</li> <li>• "windows" Windows NT authentication</li> </ul> <p>The auth method must be enabled in the instance of WebAPI pointed to by baseUrl.</p>
webApiUsername	A character string containing the WebApi username passed on to authentication methods
webApiPassword	An character string containing a WebApi password passed on to authentication methods. By default the user will be prompted for their password when needed.

---

`cancelCharacterizationGeneration`*Cancel generation of Characterization id.*

---

**Description**

Cancel generation of Characterization id.

**Usage**

```
cancelCharacterizationGeneration(characterizationId, baseUrl, sourceKey)
```

**Arguments**

`characterizationId`

An integer id representing the id that uniquely identifies a Characterization definition in a WebApi instance.

`baseUrl`

The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

`sourceKey`

The source key for a CDM instance in WebAPI, as defined in the Configuration page

**Details**

Cancel the generation of Characterization id in the WebApi.

**Value**

A tibble with job status information.

**Examples**

```
## Not run:
cancelCharacterizationGeneration(characterizationId = 13242,
                                baseUrl = "http://server.org:80/WebAPI",
                                sourceKey = "HCUP")

## End(Not run)
```

---

`cancelCohortGeneration`*Cancel generation of Cohort id.*

---

**Description**

Cancel generation of Cohort id.

**Usage**

```
cancelCohortGeneration(cohortId, baseUrl, sourceKey)
```

**Arguments**

cohortId	An integer id representing the id that uniquely identifies a Cohort definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
sourceKey	The source key for a CDM instance in WebAPI, as defined in the Configuration page

**Details**

Cancel the generation of Cohort id in the WebApi.

**Value**

A tibble with job status information.

**Examples**

```
## Not run:
cancelCohortGeneration(cohortId = 13242,
                        baseUrl = "http://server.org:80/WebAPI",
                        sourceKey = "HCUP")

## End(Not run)
```

---

cancelGeneration	<i>Invoke generation. [Stable]</i>
------------------	------------------------------------

---

**Description**

Invoke generation. **[Stable]**

**Usage**

```
cancelGeneration(id, baseUrl, sourceKey, category)
```

**Arguments**

id	An integer id representing the id that uniquely identifies a definition for the category in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
sourceKey	The source key for a CDM instance in WebAPI, as defined in the Configuration page
category	These are the categories in WebApi. The valid string options are 'conceptSet', 'cohort', 'characterization', 'pathway', 'incidenceRate', 'estimation', 'prediction'.

**Details**

Invoke generation (execution) information.

**Value**

Error message if invoke failed.

## Examples

```
## Not run:
cancelGeneration(id = 13242, category = "cohort", baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

cancelIncidenceRateGeneration  
*Cancel generation of IncidenceRate id.*

### Description

Cancel generation of IncidenceRate id.

## Usage

```
cancelIncidenceRateGeneration(incidenceRateId, baseUrl, sourceKey)
```

## Arguments

incidenceRateId	An integer id representing the id that uniquely identifies a IncidenceRate definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
sourceKey	The source key for a CDM instance in WebAPI, as defined in the Configuration page

## Details

Cancel the generation of IncidenceRate id in the WebApi.

## Value

A tibble with job status information.

## Examples

```
## Not run:  
cancelIncidenceRateGeneration(incidenceRateId = 13242,  
                               baseUrl = "http://server.org:80/WebAPI",  
                               sourceKey = "HCUP")  
  
## End(Not run)
```

---

cancelPathwayGeneration

*Cancel generation of Pathway id.*


---

### Description

Cancel generation of Pathway id.

### Usage

```
cancelPathwayGeneration(pathwayId, baseUrl, sourceKey)
```

### Arguments

pathwayId	An integer id representing the id that uniquely identifies a Pathway definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
sourceKey	The source key for a CDM instance in WebAPI, as defined in the Configuration page

### Details

Cancel the generation of Pathway id in the WebApi.

### Value

A tibble with job status information.

### Examples

```
## Not run:
cancelPathwayGeneration(pathwayId = 13242,
                        baseUrl = "http://server.org:80/WebAPI",
                        sourceKey = "HCUP")

## End(Not run)
```

---

checkInputFileEncoding

*Check character encoding of input file*


---

### Description

For its input files, CohortDiagnostics only accepts UTF-8 or ASCII character encoding. This function can be used to check whether a file meets these criteria.

### Usage

```
checkInputFileEncoding(fileName)
```



**Arguments**

fileName            The path to the file to check

**Value**

Throws an error if the input file does not have the correct encoding.

---

convertConceptSetDefinitionToTable

*Convert a concept set definition to a table* **[Maturing]**

---

**Description**

Convert a concept set definition to a table **[Maturing]**

**Usage**

```
convertConceptSetDefinitionToTable(conceptSetDefinition)
```

**Arguments**

conceptSetDefinition

A concept set definition, for example as obtained through the [getConceptSetDefinition](#) function, or taken from a cohort definition.

**Value**

Takes a R (list) representation of the Concept Set expression and returns a table (dataframe) representing the concept set expression. This is useful to create publication friendly output of the concept set expression.

**Examples**

```
## Not run:
conceptSetDefinition <- getConceptSetDefinition(conceptSetId = 282,
                                                baseUrl = "http://server.org:80/WebAPI")
convertConceptSetDefinitionToTable(conceptSetDefinition = conceptSetDefinition)

## End(Not run)
```

---

```
createConceptSetWorkbook
```

*Save a set of concept sets expressions, included concepts, and mapped concepts into a workbook* **[Maturing]**

---

### Description

Save a set of concept sets expressions, included concepts, and mapped concepts into a workbook **[Maturing]**

### Usage

```
createConceptSetWorkbook(
    conceptSetIds,
    fileName,
    baseUrl,
    included = FALSE,
    mapped = FALSE
)
```

### Arguments

conceptSetIds	A vector of concept set IDs.
fileName	The name of the XLSX workbook file.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
included	Should included concepts be included in the workbook?
mapped	Should mapped concepts be included in the workbook?

### Value

A xlsx workbook that includes a list of all concept set IDs and names and a worksheet for the concepts in each set. Options to include an included concepts and mapped concepts worksheet for each concept set are available.

---

```
deleteCharacterizationDefinition
```

*Delete Characterization id definition.* **[Stable]**

---

### Description

Delete Characterization id definition. **[Stable]**

### Usage

```
deleteCharacterizationDefinition(characterizationId, baseUrl)
```

**Arguments**

characterizationId	An integer id representing the id that uniquely identifies a Characterization definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Delete the Characterization definition from WebAPI for a given Characterization id

**Value**

None, unless error.

**Examples**

```
## Not run:
deleteCharacterizationDefinition(characterizationId = 13242,
                                baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

deleteCohortDefinition

*Delete Cohort id definition. [Stable]*

---

**Description**

Delete Cohort id definition. [Stable]

**Usage**

```
deleteCohortDefinition(cohortId, baseUrl)
```

**Arguments**

cohortId	An integer id representing the id that uniquely identifies a Cohort definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Delete the Cohort definition from WebAPI for a given Cohort id

**Value**

None, unless error.

**Examples**

```
## Not run:
deleteCohortDefinition(cohortId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

```
deleteConceptSetDefinition
```

*Delete ConceptSet id definition. [Stable]*

---

### Description

Delete ConceptSet id definition. **[Stable]**

### Usage

```
deleteConceptSetDefinition(conceptSetId, baseUrl)
```

### Arguments

conceptSetId	An integer id representing the id that uniquely identifies a ConceptSet definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Delete the ConceptSet definition from WebAPI for a given ConceptSet id

### Value

None, unless error.

### Examples

```
## Not run:
deleteConceptSetDefinition(conceptSetId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

```
deleteDefinition
```

*Delete a definition id of a chosen category. [Stable]*

---

### Description

Delete a definition id of a chosen category. **[Stable]**

### Usage

```
deleteDefinition(id, baseUrl, category)
```

### Arguments

id	An integer id representing the id that uniquely identifies a definition for the category in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
category	These are the categories in WebApi. The valid string options are 'conceptSet', 'cohort', 'characterization', 'pathway', 'incidenceRate', 'estimation', 'prediction'.

**Details**

Delete the definition for an id of chosen category in WebApi.

**Value**

None, unless error.

**Examples**

```
## Not run:
deleteDefinition(id = 13242, baseUrl = "http://server.org:80/WebAPI", category = "cohort")

## End(Not run)
```

---

```
deleteEstimationDefinition
      Delete Estimation id definition. [Stable]
```

---

**Description**

Delete Estimation id definition. **[Stable]**

**Usage**

```
deleteEstimationDefinition(estimationId, baseUrl)
```

**Arguments**

estimationId	An integer id representing the id that uniquely identifies a Estimation definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Delete the Estimation definition from WebAPI for a given Estimation id

**Value**

None, unless error.

**Examples**

```
## Not run:
deleteEstimationDefinition(estimationId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`deleteIncidenceRateDefinition`*Delete IncidenceRate id definition. [Stable]*

---

**Description**

Delete IncidenceRate id definition. **[Stable]**

**Usage**

```
deleteIncidenceRateDefinition(incidenceRateId, baseUrl)
```

**Arguments**

`incidenceRateId`

An integer id representing the id that uniquely identifies a IncidenceRate definition in a WebApi instance.

`baseUrl`

The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Delete the IncidenceRate definition from WebAPI for a given IncidenceRate id

**Value**

None, unless error.

**Examples**

```
## Not run:
deleteIncidenceRateDefinition(incidenceRateId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`deletePathwayDefinition`*Delete Pathway id definition. [Stable]*

---

**Description**

Delete Pathway id definition. **[Stable]**

**Usage**

```
deletePathwayDefinition(pathwayId, baseUrl)
```

**Arguments**

`pathwayId`

An integer id representing the id that uniquely identifies a Pathway definition in a WebApi instance.

`baseUrl`

The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Delete the Pathway definition from WebAPI for a given Pathway id

**Value**

None, unless error.

**Examples**

```
## Not run:  
deletePathwayDefinition(pathwayId = 13242, baseUrl = "http://server.org:80/WebAPI")  
  
## End(Not run)
```

---

deletePredictionDefinition

*Delete Prediction id definition. [Stable]*

---

**Description**

Delete Prediction id definition. [Stable]

**Usage**

```
deletePredictionDefinition(predictionId, baseUrl)
```

**Arguments**

predictionId	An integer id representing the id that uniquely identifies a Prediction definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Delete the Prediction definition from WebAPI for a given Prediction id

**Value**

None, unless error.

**Examples**

```
## Not run:  
deletePredictionDefinition(predictionId = 13242, baseUrl = "http://server.org:80/WebAPI")  
  
## End(Not run)
```

---

detectCharacterizationsByName

*Detect the presence of string matched Characterization definitions.*  
**[Stable]**

---

### Description

Detect the presence of string matched Characterization definitions. **[Stable]**

### Usage

```
detectCharacterizationsByName(pattern, negate = FALSE, baseUrl)
```

### Arguments

pattern	A pattern to look for. See <a href="#">str_detect</a> for details.
negate	If TRUE, return non-matching elements. See <a href="#">str_detect</a> for details.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Detect string matched Characterization definition names from the WebApi, and retrieve metadata definitions.

### Value

FALSE if no matches. If matched - output from [getCharacterizationDefinitionsMetaData](#)

### Examples

```
## Not run:
detectCharacterizations(pattern = "this text string to search in pattern",
                        baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

detectCohortsByName     *Detect the presence of string matched Cohort definitions.* **[Stable]**

---

### Description

Detect the presence of string matched Cohort definitions. **[Stable]**

### Usage

```
detectCohortsByName(pattern, negate = FALSE, baseUrl)
```



**Arguments**

pattern	A pattern to look for. See <a href="#">str_detect</a> for details.
negate	If TRUE, return non-matching elements. See <a href="#">str_detect</a> for details.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Detect string matched Cohort definition names from the WebApi, and retrieve metadata definitions.

**Value**

FALSE if no matches. If matched - output from [getCohortDefinitionsMetaData](#)

**Examples**

```
## Not run:
detectCohorts(pattern = "this text string to search in pattern",
              baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

detectConceptSetsByName

*Detect the presence of string matched ConceptSet definitions. [Stable]*

---

**Description**

Detect the presence of string matched ConceptSet definitions. **[Stable]**

**Usage**

```
detectConceptSetsByName(pattern, negate = FALSE, baseUrl)
```

**Arguments**

pattern	A pattern to look for. See <a href="#">str_detect</a> for details.
negate	If TRUE, return non-matching elements. See <a href="#">str_detect</a> for details.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Detect string matched ConceptSet definition names from the WebApi, and retrieve metadata definitions.

**Value**

FALSE if no matches. If matched - output from [getConceptSetDefinitionsMetaData](#)

## Examples

```
## Not run:
detectConceptSets(pattern = "this text string to search in pattern",
  baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

detectEstimationsByName

*Detect the presence of string matched Estimation definitions. [Stable]*

---

## Description

Detect the presence of string matched Estimation definitions. [Stable]

## Usage

```
detectEstimationsByName(pattern, negate = FALSE, baseUrl)
```

## Arguments

pattern	A pattern to look for. See <a href="#">str_detect</a> for details.
negate	If TRUE, return non-matching elements. See <a href="#">str_detect</a> for details.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

## Details

Detect string matched Estimation definition names from the WebApi, and retrieve metadata definitions.

## Value

FALSE if no matches. If matched - output from [getEstimationDefinitionsMetaData](#)

## Examples

```
## Not run:
detectEstimations(pattern = "this text string to search in pattern",
  baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`detectIncidenceRatesByName`*Detect the presence of string matched IncidenceRate definitions. [Stable]*

---

### Description

Detect the presence of string matched IncidenceRate definitions. **[Stable]**

### Usage

```
detectIncidenceRatesByName(pattern, negate = FALSE, baseUrl)
```

### Arguments

<code>pattern</code>	A pattern to look for. See <a href="#">str_detect</a> for details.
<code>negate</code>	If TRUE, return non-matching elements. See <a href="#">str_detect</a> for details.
<code>baseUrl</code>	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Detect string matched IncidenceRate definition names from the WebApi, and retrieve metadata definitions.

### Value

FALSE if no matches. If matched - output from [getIncidenceRateDefinitionsMetaData](#)

### Examples

```
## Not run:
detectIncidenceRates(pattern = "this text string to search in pattern",
                     baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`detectPathwaysByName` *Detect the presence of string matched Pathway definitions. [Stable]*

---

### Description

Detect the presence of string matched Pathway definitions. **[Stable]**

### Usage

```
detectPathwaysByName(pattern, negate = FALSE, baseUrl)
```

**Arguments**

pattern	A pattern to look for. See <a href="#">str_detect</a> for details.
negate	If TRUE, return non-matching elements. See <a href="#">str_detect</a> for details.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Detect string matched Pathway definition names from the WebApi, and retrieve metadata definitions.

**Value**

FALSE if no matches. If matched - output from [getPathwayDefinitionsMetaData](#)

**Examples**

```
## Not run:
detectPathways(pattern = "this text string to search in pattern",
               baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

detectPredictionsByName

*Detect the presence of string matched Prediction definitions. [Stable]*

---

**Description**

Detect the presence of string matched Prediction definitions. [Stable]

**Usage**

```
detectPredictionsByName(pattern, negate = FALSE, baseUrl)
```

**Arguments**

pattern	A pattern to look for. See <a href="#">str_detect</a> for details.
negate	If TRUE, return non-matching elements. See <a href="#">str_detect</a> for details.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Detect string matched Prediction definition names from the WebApi, and retrieve metadata definitions.

**Value**

FALSE if no matches. If matched - output from [getPredictionDefinitionsMetaData](#)

## Examples

```
## Not run:
detectPredictions(pattern = "this text string to search in pattern",
  baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

existsCharacterizationName

*Check if Characterization definition name exists. [Stable]*

---

## Description

Check if Characterization definition name exists. **[Stable]**

## Usage

```
existsCharacterizationName(characterizationName, baseUrl)
```

## Arguments

characterizationName

A string name for the Characterization to be checked.

baseUrl

The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

## Details

Check if a string name already exists in the WebApi as a Characterization definition name.

## Value

If found, the function will return a tibble with details of the specification. If not found, FALSE will be returned.

## Examples

```
## Not run:
existsCharacterizationName(characterizationName = "this text string needs to be checked",
  baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

existsCohortName	<i>Check if Cohort definition name exists. [Stable]</i>
------------------	---

---

### Description

Check if Cohort definition name exists. **[Stable]**

### Usage

```
existsCohortName(cohortName, baseUrl)
```

### Arguments

cohortName	A string name for the Cohort to be checked.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Check if a string name already exists in the WebApi as a Cohort definition name.

### Value

If found, the function will return a tibble with details of the specification. If not found, FALSE will be returned.

### Examples

```
## Not run:
existsCohortName(cohortName = "this text string needs to be checked",
                 baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

existsConceptSetName	<i>Check if ConceptSet definition name exists. [Stable]</i>
----------------------	---

---

### Description

Check if ConceptSet definition name exists. **[Stable]**

### Usage

```
existsConceptSetName(conceptSetName, baseUrl)
```

### Arguments

conceptSetName	A string name for the ConceptSet to be checked.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Check if a string name already exists in the WebApi as a ConceptSet definition name.

**Value**

If found, the function will return a tibble with details of the specification. If not found, FALSE will be returned.

**Examples**

```
## Not run:
existsConceptSetName(conceptSetName = "this text string needs to be checked",
                     baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

existsEstimationName	Check if Estimation definition name exists. <b>[Stable]</b>
----------------------	---

---

**Description**

Check if Estimation definition name exists. **[Stable]**

**Usage**

```
existsEstimationName(estimationName, baseUrl)
```

**Arguments**

estimationName A string name for the Estimation to be checked.

baseUrl The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Check if a string name already exists in the WebApi as a Estimation definition name.

**Value**

If found, the function will return a tibble with details of the specification. If not found, FALSE will be returned.

**Examples**

```
## Not run:
existsEstimationName(estimationName = "this text string needs to be checked",
                     baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

existsIncidenceRateName

*Check if IncidenceRate definition name exists. [Stable]*


---

### Description

Check if IncidenceRate definition name exists. **[Stable]**

### Usage

```
existsIncidenceRateName(incidenceRateName, baseUrl)
```

### Arguments

incidenceRateName

A string name for the IncidenceRate to be checked.

baseUrl

The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Check if a string name already exists in the WebApi as a IncidenceRate definition name.

### Value

If found, the function will return a tibble with details of the specification. If not found, FALSE will be returned.

### Examples

```
## Not run:
existsIncidenceRateName(incidenceRateName = "this text string needs to be checked",
                        baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

existsPathwayName

*Check if Pathway definition name exists. [Stable]*


---

### Description

Check if Pathway definition name exists. **[Stable]**

### Usage

```
existsPathwayName(pathwayName, baseUrl)
```

### Arguments

pathwayName

A string name for the Pathway to be checked.

baseUrl

The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".



**Details**

Check if a string name already exists in the WebApi as a Pathway definition name.

**Value**

If found, the function will return a tibble with details of the specification. If not found, FALSE will be returned.

**Examples**

```
## Not run:
existsPathwayName(pathwayName = "this text string needs to be checked",
                  baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

existsPredictionName	Check if Prediction definition name exists. <b>[Stable]</b>
----------------------	---

---

**Description**

Check if Prediction definition name exists. **[Stable]**

**Usage**

```
existsPredictionName(predictionName, baseUrl)
```

**Arguments**

predictionName A string name for the Prediction to be checked.

baseUrl The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Check if a string name already exists in the WebApi as a Prediction definition name.

**Value**

If found, the function will return a tibble with details of the specification. If not found, FALSE will be returned.

**Examples**

```
## Not run:
existsPredictionName(predictionName = "this text string needs to be checked",
                    baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getCdmSources	<i>Get the data sources in the WebAPI instance</i> <b>[Stable]</b>
---------------	--

---

**Description**

Get the data sources in the WebAPI instance **[Stable]**

**Usage**

```
getCdmSources(baseUrl)
```

**Arguments**

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
---------	---

**Details**

Obtains the data sources configured in the WebAPI instance.

**Value**

A data frame.

---

getCharacterizationDefinition	<i>Get Characterization id definition.</i> <b>[Stable]</b>
-------------------------------	--

---

**Description**

Get Characterization id definition. **[Stable]**

**Usage**

```
getCharacterizationDefinition(characterizationId, baseUrl)
```

**Arguments**

characterizationId	An integer id representing the id that uniquely identifies a Characterization definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Obtain the Characterization definition from WebAPI for a given Characterization id

**Value**

An R object representing the Characterization definition

### Examples

```
## Not run:
getCharacterizationDefinition(characterizationId = 13242,
                             baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getCharacterizationDefinitionsMetaData

*Get the meta data for Characterization definitions. [Stable]*

---

### Description

Get the meta data for Characterization definitions. **[Stable]**

### Usage

```
getCharacterizationDefinitionsMetaData(baseUrl)
```

### Arguments

baseUrl            The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Get the meta data of WebApi specifications such as id, name, created/modified details, hash object, etc. from WebApi for Characterization. This function is useful to retrieve the current Characterization specifications.

### Value

A tibble of specification metadata for Characterization. Note: modifiedDate and createdDate are returned as text/character.

### Examples

```
## Not run:
getCharacterizationDefinitionsMetaData(baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`getCharacterizationGenerationInformation`*Get generation information for Characterization id.*

---

**Description**

Get generation information for Characterization id.

**Usage**

```
getCharacterizationGenerationInformation(characterizationId, baseUrl)
```

**Arguments**

`characterizationId`

An integer id representing the id that uniquely identifies a Characterization definition in a WebApi instance.

`baseUrl`

The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Get generation (execution) information about Characterization for a characterizationId.

**Value**

An R object representing the Characterization definition

**Examples**

```
## Not run:
getCharacterizationGenerationInformation(characterizationId = 13242,
                                         baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`getCharacterizationResults`*Get results for a Characterization Id.*

---

**Description**

Get results for a Characterization Id.

**Usage**

```
getCharacterizationResults(characterizationId, baseUrl)
```

**Arguments**

characterizationId	An integer id representing the id that uniquely identifies a characterization analysis definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Get the results for Characterization id.

**Value**

An R object with results.

**Examples**

```
## Not run:
getCharacterizationResults(characterizationId = 342, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getCohortDefinition	<i>Get Cohort id definition.</i> <b>[Stable]</b>
---------------------	--

---

**Description**

Get Cohort id definition. **[Stable]**

**Usage**

```
getCohortDefinition(cohortId, baseUrl)
```

**Arguments**

cohortId	An integer id representing the id that uniquely identifies a Cohort definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Obtain the Cohort definition from WebAPI for a given Cohort id

**Value**

An R object representing the Cohort definition

**Examples**

```
## Not run:
getCohortDefinition(cohortId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

```
getCohortDefinitionExpression
  (Deprecated) Get a cohort definition expression
```

---

### Description

(Deprecated) Get a cohort definition expression

### Usage

```
getCohortDefinitionExpression(cohortId, baseUrl)
```

### Arguments

cohortId	An integer id representing the id that uniquely identifies a cohort definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

(Deprecated) Obtain the JSON expression from WebAPI for a given cohort id

### Value

A JSON list object representing the cohort definition This function has been deprecated. As an alternative please use the following steps as in the example below: 1) cohortDefinition <- getCohortDefinition(baseUrl = baseUrl, cohortId = 15873) 2) validJsonExpression <- RJSONIO::toJSON(cohortDefinition\$expression) 3) save validJsonExpression object as .txt"

### Examples

```
## Not run:
# This will obtain a cohort definition's JSON expression:

getCohortDefinitionExpression(cohortId = 282, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

```
getCohortDefinitionName
  (Deprecated) Get a cohort definition's name from WebAPI
```

---

### Description

(Deprecated) Get a cohort definition's name from WebAPI

### Usage

```
getCohortDefinitionName(baseUrl, cohortId, formatName = FALSE)
```

**Arguments**

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
cohortId	An integer id representing the id that uniquely identifies a cohort definition in a WebApi instance.
formatName	Should the name be formatted to remove prefixes and underscores?

**Details**

(Deprecated) Obtains the name of a cohort. This function has been deprecated. As an alternative please use getCohortDefinition

**Value**

The name of the cohort.

---

`getCohortDefinitionsMetaData`

*Get the meta data for Cohort definitions. [Stable]*

---

**Description**

Get the meta data for Cohort definitions. [Stable]

**Usage**

```
getCohortDefinitionsMetaData(baseUrl)
```

**Arguments**

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
---------	---

**Details**

Get the meta data of WebApi specifications such as id, name, created/modified details, hash object, etc. from WebApi for Cohort. This function is useful to retrieve the current Cohort specifications.

**Value**

A tibble of specification metadata for Cohort. Note: modifiedDate and createdDate are returned as text/character.

**Examples**

```
## Not run:
getCohortDefinitionsMetaData(baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

getCohortDefinitionSql

*Get a cohort definition's SQL from WebAPI*

---

### Description

Get a cohort definition's SQL from WebAPI

### Usage

```
getCohortDefinitionSql(cohortId, baseUrl, generateStats = TRUE)
```

### Arguments

cohortId	An integer id representing the id that uniquely identifies a cohort definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
generateStats	Should the SQL include the code for generating inclusion rule statistics? Note that if TRUE, several additional tables are expected to exist as described in the details. By default this is TRUE.

### Details

Obtains the template SQL of a cohort. When using generateStats = TRUE, the following tables are required to exist when executing the SQL: cohort\_inclusion, cohort\_inclusion\_result, cohort\_inclusion\_stats, and cohort\_summary\_stats. Also note that the cohort\_inclusion table should be populated with the names of the rules prior to executing the cohort definition SQL.

### Value

The templated SQL to generate the cohort

---

getCohortGenerationInformation

*Get generation information for Cohort id.*

---

### Description

Get generation information for Cohort id.

### Usage

```
getCohortGenerationInformation(cohortId, baseUrl)
```

### Arguments

cohortId	An integer id representing the id that uniquely identifies a Cohort definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".



**Details**

Get generation (execution) information about Cohort for a cohortId.

**Value**

An R object representing the Cohort definition

**Examples**

```
## Not run:  
getCohortGenerationInformation(cohortId = 13242, baseUrl = "http://server.org:80/WebAPI")  
  
## End(Not run)
```

---

getCohortInclusionRulesAndCounts

*Get cohort inclusion rules and person counts*

---

**Description**

Get cohort inclusion rules and person counts

**Usage**

```
getCohortInclusionRulesAndCounts(baseUrl, cohortId, sourceKey)
```

**Arguments**

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
cohortId	An integer id representing the id that uniquely identifies a cohort definition in a WebApi instance.
sourceKey	The source key for a CDM instance in WebAPI, as defined in the Configuration page

**Details**

Obtains the inclusion rules from a cohort definition and summarizes the person counts per rule

---

getCohortResults	<i>Get results for a Cohort Id.</i>
------------------	-------------------------------------

---

**Description**

Get results for a Cohort Id.

**Usage**

```
getCohortResults(cohortId, baseUrl)
```

**Arguments**

cohortId	An integer id representing the id that uniquely identifies a cohort definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Get the results for Cohort id.

**Value**

An R object with results.

**Examples**

```
## Not run:
getCohortResults(cohortId = 342, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getCohortSql	<i>Get SQL query for Cohort definition.</i>
--------------	---

---

**Description**

Get SQL query for Cohort definition.

**Usage**

```
getCohortSql(cohortDefinition, baseUrl, generateStats = TRUE)
```

**Arguments**

cohortDefinition	An R list object (not JSON) representing the Cohort definition. It is the output R expression object of list object from CohortDefinition
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
generateStats	Should the SQL include the code for generating inclusion rule statistics? Note that if TRUE, several additional tables are expected to exists as described in the details. By default this is TRUE.

**Details**

Given a valid Cohort definition R-object (not JSON) this function will return the parameterized SQL in OHDSI SQL dialect. This SQL may be used along with OHDSI R-package 'SQLRender' to render/translate to target SQL dialect and parameters rendered.

**Value**

An R object containing the SQL for Cohort definition.

**Examples**

```
## Not run:
getCohortSql(CohortDefinition = (getCohortDefinition(cohortId = 13242, baseUrl = baseUrl)),
             baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getConcepts	<i>Get concepts</i> <b>[Stable]</b>
-------------	-------------------------------------

---

**Description**

Get concepts **[Stable]**

**Usage**

```
getConcepts(
  conceptIds,
  baseUrl,
  vocabularySourceKey = NULL,
  snakeCaseToCamelCase = TRUE
)
```

**Arguments**

conceptIds	A vector of concept IDs.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
vocabularySourceKey	The source key of the Vocabulary. By default, the priority Vocabulary is used.
snakeCaseToCamelCase	Should the column names of the result be converted to camelCase?

**Value**

A tibble containing information on the concepts.

**Examples**

```
## Not run:
conceptSet <- getConceptSet(conceptSetId = 282, baseUrl = "http://server.org:80/WebAPI")
conceptIds <- resolveConceptSet(conceptSet = conceptSet, baseUrl = "http://server.org:80/WebAPI")
concepts <- getConcepts(conceptIds = conceptIds, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getConceptSetDefinition

*Get ConceptSet id definition. [Stable]*


---

**Description**

Get ConceptSet id definition. **[Stable]**

**Usage**

```
getConceptSetDefinition(conceptSetId, baseUrl)
```

**Arguments**

conceptSetId	An integer id representing the id that uniquely identifies a ConceptSet definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Obtain the ConceptSet definition from WebAPI for a given ConceptSet id

**Value**

An R object representing the ConceptSet definition

**Examples**

```
## Not run:
getConceptSetDefinition(conceptSetId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`getConceptSetDefinitionsMetaData`*Get the meta data for ConceptSet definitions. [Stable]*

---

### Description

Get the meta data for ConceptSet definitions. **[Stable]**

### Usage

```
getConceptSetDefinitionsMetaData(baseUrl)
```

### Arguments

`baseUrl`            The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Get the meta data of WebApi specifications such as id, name, created/modified details, hash object, etc. from WebApi for ConceptSet. This function is useful to retrieve the current ConceptSet specifications.

### Value

A tibble of specification metadata for ConceptSet. Note: `modifiedDate` and `createdDate` are returned as text/character.

### Examples

```
## Not run:
getConceptSetDefinitionsMetaData(baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`getDefinition`*Get the definition for an id of chosen category in WebApi. [Stable]*

---

### Description

Get the definition for an id of chosen category in WebApi. **[Stable]**

### Usage

```
getDefinition(id, baseUrl, category)
```

**Arguments**

id	An integer id representing the id that uniquely identifies a definition for the category in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
category	These are the categories in WebApi. The valid string options are 'conceptSet', 'cohort', 'characterization', 'pathway', 'incidenceRate', 'estimation', 'prediction'.

**Details**

Get the definition for an id of chosen category in WebApi. The return object will be a R representation of the definition, that may be reconverted to JSON.

**Value**

An R object representing the definition

**Examples**

```
## Not run:
getDefinition(id = 13242, category = "cohort", baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getDefinitionsMetadata

*Retrieve the meta data for WebApi definitions of a certain category*  
**[Stable]**

---

**Description**

Retrieve the meta data for WebApi definitions of a certain category **[Stable]**

**Usage**

```
getDefinitionsMetadata(baseUrl, category)
```

**Arguments**

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
category	These are the categories in WebApi. The valid string options are 'conceptSet', 'cohort', 'characterization', 'pathway', 'incidenceRate', 'estimation', 'prediction'.

**Details**

Obtains the meta data of WebApi specifications such as id, name, created/modified details, hash object, etc for a certain category. The following function categories are supported. Concept-set, Cohort-definition, Cohort-characterization, Pathway-analysis, Incidence rate (ir), estimation and prediction. This function is useful to retrieve the current specifications.

**Value**

A tibble of specification metadata.

**Examples**

```
## Not run:
getDefinitionsMetadata(baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getEstimationDefinition

*Get Estimation id definition. [Stable]*

---

**Description**

Get Estimation id definition. [Stable]

**Usage**

```
getEstimationDefinition(estimationId, baseUrl)
```

**Arguments**

estimationId	An integer id representing the id that uniquely identifies a Estimation definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Obtain the Estimation definition from WebAPI for a given Estimation id

**Value**

An R object representing the Estimation definition

**Examples**

```
## Not run:
getEstimationDefinition(estimationId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`getEstimationDefinitionsMetaData`*Get the meta data for Estimation definitions. [Stable]*

---

### Description

Get the meta data for Estimation definitions. [Stable]

### Usage

```
getEstimationDefinitionsMetaData(baseUrl)
```

### Arguments

`baseUrl`            The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Get the meta data of WebApi specifications such as id, name, created/modified details, hash object, etc. from WebApi for Estimation. This function is useful to retrieve the current Estimation specifications.

### Value

A tibble of specification metadata for Estimation. Note: `modifiedDate` and `createdDate` are returned as text/character.

### Examples

```
## Not run:
getEstimationDefinitionsMetaData(baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`getGenerationInformation`*Get generation information. [Stable]*

---

### Description

Get generation information. [Stable]

### Usage

```
getGenerationInformation(id, category, baseUrl)
```



**Arguments**

id	An integer id representing the id that uniquely identifies a definition for the category in a WebApi instance.
category	The category of expression in WebApi. Only the following strings are accepted: 'cohort', 'characterization', 'pathway', 'incidenceRate'.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Get generation (execution) information.

**Value**

An R object with the generation information.

**Examples**

```
## Not run:
getGenerationInformation(id = 13242,
                        category = "cohort",
                        baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getIncidenceRateDefinition

*Get IncidenceRate id definition. [Stable]*

---

**Description**

Get IncidenceRate id definition. **[Stable]**

**Usage**

```
getIncidenceRateDefinition(incidenceRateId, baseUrl)
```

**Arguments**

incidenceRateId	An integer id representing the id that uniquely identifies a IncidenceRate definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Obtain the IncidenceRate definition from WebAPI for a given IncidenceRate id

**Value**

An R object representing the IncidenceRate definition

### Examples

```
## Not run:
getIncidenceRateDefinition(incidenceRateId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getIncidenceRateDefinitionsMetaData

*Get the meta data for IncidenceRate definitions. [Stable]*

---

### Description

Get the meta data for IncidenceRate definitions. [Stable]

### Usage

```
getIncidenceRateDefinitionsMetaData(baseUrl)
```

### Arguments

baseUrl            The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Get the meta data of WebApi specifications such as id, name, created/modified details, hash object, etc. from WebApi for IncidenceRate. This function is useful to retrieve the current IncidenceRate specifications.

### Value

A tibble of specification metadata for IncidenceRate. Note: modifiedDate and createdDate are returned as text/character.

### Examples

```
## Not run:
getIncidenceRateDefinitionsMetaData(baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

```
getIncidenceRateGenerationInformation
```

*Get generation information for IncidenceRate id.*

---

**Description**

Get generation information for IncidenceRate id.

**Usage**

```
getIncidenceRateGenerationInformation(incidenceRateId, baseUrl)
```

**Arguments**

incidenceRateId	An integer id representing the id that uniquely identifies a IncidenceRate definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Get generation (execution) information about IncidenceRate for a incidenceRateId.

**Value**

An R object representing the IncidenceRate definition

**Examples**

```
## Not run:
getIncidenceRateGenerationInformation(incidenceRateId = 13242,
                                      baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

```
getIncidenceRateResults
```

*Get results for a IncidenceRate Id.*

---

**Description**

Get results for a IncidenceRate Id.

**Usage**

```
getIncidenceRateResults(incidenceRateId, baseUrl)
```

**Arguments**

incidenceRateId	An integer id representing the id that uniquely identifies a incidence rate analysis definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Get the results for IncidenceRate id.

**Value**

An R object with results.

**Examples**

```
## Not run:
getIncidenceRateResults(incidenceRateId = 342, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getPathwayDefinition    *Get Pathway id definition.* **[Stable]**

---

**Description**

Get Pathway id definition. **[Stable]**

**Usage**

```
getPathwayDefinition(pathwayId, baseUrl)
```

**Arguments**

pathwayId	An integer id representing the id that uniquely identifies a Pathway definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Obtain the Pathway definition from WebAPI for a given Pathway id

**Value**

An R object representing the Pathway definition

**Examples**

```
## Not run:
getPathwayDefinition(pathwayId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`getPathwayDefinitionsMetaData`*Get the meta data for Pathway definitions. [Stable]*

---

### Description

Get the meta data for Pathway definitions. [Stable]

### Usage

```
getPathwayDefinitionsMetaData(baseUrl)
```

### Arguments

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
---------	---

### Details

Get the meta data of WebApi specifications such as id, name, created/modified details, hash object, etc. from WebApi for Pathway. This function is useful to retrieve the current Pathway specifications.

### Value

A tibble of specification metadata for Pathway. Note: modifiedDate and createdDate are returned as text/character.

### Examples

```
## Not run:
getPathwayDefinitionsMetaData(baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`getPathwayGenerationInformation`*Get generation information for Pathway id.*

---

### Description

Get generation information for Pathway id.

### Usage

```
getPathwayGenerationInformation(pathwayId, baseUrl)
```

### Arguments

pathwayId	An integer id representing the id that uniquely identifies a Pathway definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Get generation (execution) information about Pathway for a pathwayId.

**Value**

An R object representing the Pathway definition

**Examples**

```
## Not run:
getPathwayGenerationInformation(pathwayId = 13242, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getPathwayResults	<i>Get results for a Pathway Id.</i>
-------------------	--------------------------------------

---

**Description**

Get results for a Pathway Id.

**Usage**

```
getPathwayResults(pathwayId, baseUrl)
```

**Arguments**

pathwayId	An integer id representing the id that uniquely identifies a pathway analysis definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Get the results for Pathway id.

**Value**

An R object with results.

**Examples**

```
## Not run:
getPathwayResults(pathwayId = 342, baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getPersonProfile	<i>Get person profile data</i> <b>[Maturing]</b>
------------------	--

---

## Description

Get person profile data **[Maturing]**

## Usage

```
getPersonProfile(baseUrl, sourceKey, personId, indexCohortId = NULL)
```

## Arguments

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
sourceKey	The source key for a CDM instance in WebAPI, as defined in the Configuration page
personId	The personId of the person whose profile is being reviewed
indexCohortId	<OPTIONAL> Do you want to use a particular cohortId as the index cohort? If left NULL, the WebApi will identify the earliest cohort for the person by cohort start date and use it as the index cohort. WebApi uses the cohort start date of the index cohort to calculate the person's index age (ageAtIndex). WebApi will also return the relative position, in days, for each event compared to the index cohorts start date. These relative positions are useful to study the relationship of various events with respect to the index cohort start date.

## Details

Get a R object with person profile data. This function may be used for visualizing a patients profile in tables or visualization.

## Value

A list of tibble data frame objects corresponding to cohorts, observationPeriod, records and person.

## Examples

```
## Not run:
getPersonProfile(baseUrl = "http://server.org:80/WebAPI",
                  sourceKey = "sourceKey",
                  personId = 342342)

## End(Not run)
```

---

`getPredictionDefinition`*Get Prediction id definition. [Stable]*

---

**Description**

Get Prediction id definition. **[Stable]**

**Usage**

```
getPredictionDefinition(predictionId, baseUrl)
```

**Arguments**

<code>predictionId</code>	An integer id representing the id that uniquely identifies a Prediction definition in a WebApi instance.
<code>baseUrl</code>	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Obtain the Prediction definition from WebAPI for a given Prediction id

**Value**

An R object representing the Prediction definition

**Examples**

```
## Not run:  
getPredictionDefinition(predictionId = 13242, baseUrl = "http://server.org:80/WebAPI")  
  
## End(Not run)
```

---

`getPredictionDefinitionsMetaData`*Get the meta data for Prediction definitions. [Stable]*

---

**Description**

Get the meta data for Prediction definitions. **[Stable]**

**Usage**

```
getPredictionDefinitionsMetaData(baseUrl)
```

**Arguments**

<code>baseUrl</code>	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
----------------------	---



## Details

Get the meta data of WebApi specifications such as id, name, created/modified details, hash object, etc. from WebApi for Prediction. This function is useful to retrieve the current Prediction specifications.

## Value

A tibble of specification metadata for Prediction. Note: modifiedDate and createdDate are returned as text/character.

## Examples

```
## Not run:
getPredictionDefinitionsMetaData(baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

`getPriorityVocabularyKey`*Get Priority Vocabulary Source Key [Stable]*

---

## Description

Get Priority Vocabulary Source Key **[Stable]**

## Usage

```
getPriorityVocabularyKey(baseUrl)
```

## Arguments

`baseUrl`            The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

## Details

Obtains the source key of the default OMOP Vocabulary in WebApi.

## Value

A string.

---

getResults	<i>Get generation results</i> <b>[Stable]</b>
------------	---

---

### Description

Get generation results **[Stable]**

### Usage

```
getResults(id, baseUrl, category)
```

### Arguments

id	An integer id representing the id that uniquely identifies a definition for the category in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
category	These are the categories in WebApi. The valid string options are 'conceptSet', 'cohort', 'characterization', 'pathway', 'incidenceRate', 'estimation', 'prediction'.

### Details

Get the results objects from WebApi for a definition of a certain category in WebApi.

### Value

Returns the result objects for a given id and category from the WebApi.

### Examples

```
## Not run:
getResults(id = 282, category = "cohort", baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getSourceConcepts	<i>Get source concepts that map to standard concepts</i> <b>[Stable]</b>
-------------------	--

---

### Description

Get source concepts that map to standard concepts **[Stable]**

### Usage

```
getSourceConcepts(
  conceptIds,
  baseUrl,
  vocabularySourceKey = NULL,
  snakeCaseToCamelCase = TRUE
)
```

**Arguments**

conceptIds        A list of concept IDs referring to standard concepts.  
 baseUrl         The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".  
 vocabularySourceKey        The source key of the Vocabulary. By default, the priority Vocabulary is used.  
 snakeCaseToCamelCase       Should the column names of the result be converted to camelCase?

**Value**

A tibble containing information on the source concepts.

**Examples**

```
## Not run:
conceptSet <- getConceptSetDefinition(conceptSetId = 282,
                                     baseUrl = "http://server.org:80/WebAPI")
conceptIds <- resolveConceptSet(conceptSet = conceptSet, baseUrl = "http://server.org:80/WebAPI")
sourceConcepts <- getSourceConcepts(conceptIds = conceptIds,
                                     baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

getWebApiVersion	<i>Get the WebAPI version number</i> <b>[Stable]</b>
------------------	--

---

**Description**

Get the WebAPI version number **[Stable]**

**Usage**

```
getWebApiVersion(baseUrl)
```

**Arguments**

baseUrl         The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Obtains the WebAPI version number. This function is used to check that WebAPI baseUrl can be accessed and is a good first check to make sure you can access a WebAPI endpoint.

**Value**

The WebApi versions as a string.

**Examples**

```
## Not run:
getWebApiVersion("http://server.org:80/WebAPI")

## End(Not run)
```

---

insertCohortDefinitionInPackage

*Load a cohort definition and insert it into this package* **[Maturing]**


---

## Description

Load a cohort definition and insert it into this package **[Maturing]**

## Usage

```
insertCohortDefinitionInPackage(
  cohortId,
  name = NULL,
  jsonFolder = "inst/cohorts",
  sqlFolder = "inst/sql/sql_server",
  baseUrl,
  generateStats = FALSE
)
```

## Arguments

cohortId	An integer id representing the id that uniquely identifies a cohort definition in a WebApi instance.
name	The name that will be used for the JSON and SQL files. If not provided, the name in cohort will be used, but this may not lead to valid file names.
jsonFolder	Path to the folder where the JSON representation will be saved.
sqlFolder	Path to the folder where the SQL representation will be saved.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
generateStats	Should the SQL include the code for generating inclusion rule statistics? Note that if TRUE, several additional tables are expected to exist as described in the details.

## Details

Load a cohort definition from a WebApi instance and insert it into this package. This will fetch the JSON object and store it in a folder (defaults to 'the inst/cohorts' folder), and fetch the template SQL and store it in another folder (defaults to the 'inst/sql/sql\_server' folder). Both folders will be created if they don't exist. When using generateStats = TRUE, the following tables are required to exist when executing the SQL: cohort\_inclusion, cohort\_inclusion\_result, cohort\_inclusion\_stats, and cohort\_summary\_stats. Also note that the cohort\_inclusion table should be populated with the names of the rules prior to executing the cohort definition SQL. Note: generate inclusion statistics are created for all by default.

## Examples

```
## Not run:
# This will create 'inst/cohorts/Angioedema.json' and 'inst/sql/sql_server/Angioedema.sql':

insertCohortDefinitionInPackage(cohortId = 282,
                                name = "Angioedema",
```

```

                                baseUrl = "http://server.org:80/WebAPI")

## End(Not run)

```

---

```

insertCohortDefinitionSetInPackage
    Insert a set of cohort definitions into package

```

---

## Description

Insert a set of cohort definitions into package

## Usage

```

insertCohortDefinitionSetInPackage(
  fileName = "inst/settings/CohortsToCreate.csv",
  baseUrl,
  jsonFolder = "inst/cohorts",
  sqlFolder = "inst/sql/sql_server",
  rFileName = "R/CreateCohorts.R",
  insertTableSql = TRUE,
  insertCohortCreationR = TRUE,
  generateStats = FALSE,
  packageName
)

```

## Arguments

fileName	Name of a CSV file specifying the cohorts to insert. See details for the expected file format.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
jsonFolder	Path to the folder where the JSON representations will be saved.
sqlFolder	Path to the folder where the SQL representations will be saved.
rFileName	Name of R file to generate when insertCohortCreationR = TRUE.
insertTableSql	Should the SQL for creating the cohort table be inserted into the package as well? This file will be called CreateCohortTable.sql.
insertCohortCreationR	Insert R code that will create the cohort table and instantiate the cohorts? This will create a file called R/CreateCohorts.R containing a function called .createCohorts.
generateStats	Should cohort inclusion rule statistics be created?
packageName	The name of the package (only needed when inserting the R code as well).

## Details

The CSV file should have at least the following fields:

**atlasId** The cohort ID in ATLAS.

**name** The name to be used for the cohort. This name will be used to generate file names, so please use letters and numbers only (no spaces).

*Invoke generation of Characterization id.*

Invoke generation of Characterization id.

```
invokeCharacterizationGeneration(characterizationId, baseUrl, sourceKey)
```

An integer id representing the id that uniquely identifies a Characterization definition in a WebApi instance.

sourceKey	The source key for a CDM instance in WebAPI, as defined in the Configuration page
-----------	---

Invoke the generation of Characterization id in the WebApi.

A tibble with job status information.

[illegible]

---

`invokeCohortGeneration`*Invoke generation of Cohort id.*

---

**Description**

Invoke generation of Cohort id.

**Usage**

```
invokeCohortGeneration(cohortId, baseUrl, sourceKey)
```

**Arguments**

<code>cohortId</code>	An integer id representing the id that uniquely identifies a Cohort definition in a WebApi instance.
<code>baseUrl</code>	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
<code>sourceKey</code>	The source key for a CDM instance in WebAPI, as defined in the Configuration page

**Details**

Invoke the generation of Cohort id in the WebApi.

**Value**

A tibble with job status information.

**Examples**

```
## Not run:
invokeCohortGeneration(cohortId = 13242,
                        baseUrl = "http://server.org:80/WebAPI",
                        sourceKey = "HCUP")

## End(Not run)
```

---

`invokeGeneration`*Invoke generation. [Stable]*

---

**Description**

Invoke generation. **[Stable]**

**Usage**

```
invokeGeneration(id, baseUrl, sourceKey, category)
```

**Arguments**

id	An integer id representing the id that uniquely identifies a definition for the category in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
sourceKey	The source key for a CDM instance in WebAPI, as defined in the Configuration page
category	These are the categories in WebApi. The valid string options are 'conceptSet', 'cohort', 'characterization', 'pathway', 'incidenceRate', 'estimation', 'prediction'.

**Details**

Invoke generation (execution) information.

**Value**

A dataframe with generation information such as status, jobName, and time.

**Examples**

```
## Not run:
invokeGeneration(id = 13242, category = "cohort", baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

invokeIncidenceRateGeneration

*Invoke generation of IncidenceRate id.*

---

**Description**

Invoke generation of IncidenceRate id.

**Usage**

```
invokeIncidenceRateGeneration(incidenceRateId, baseUrl, sourceKey)
```

**Arguments**

incidenceRateId	An integer id representing the id that uniquely identifies a IncidenceRate definition in a WebApi instance.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
sourceKey	The source key for a CDM instance in WebAPI, as defined in the Configuration page

**Details**

Invoke the generation of IncidenceRate id in the WebApi.





---

isValidCharacterizationId	<i>is Characterization id a valid definition in the WebApi. [Stable]</i>
---------------------------	--

---

**Description**

is Characterization id a valid definition in the WebApi. **[Stable]**

**Usage**

```
isValidCharacterizationId(characterizationIds, baseUrl)
```

**Arguments**

characterizationIds	A list of integer id(s) of the Characterization to be tested for validity.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Checks if a set of id for a Characterization is valid. The following checks are performed. 1) checks if all the ids exists in the WebApi i.e. valid.

**Value**

A logical vector indicating if an ID is valid.

**Examples**

```
## Not run:
isValidCharacterizationId(characterizationIds = c(13242, 3423, 34),
                          baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

isValidCohortId	<i>is Cohort id a valid definition in the WebApi. [Stable]</i>
-----------------	--

---

**Description**

is Cohort id a valid definition in the WebApi. **[Stable]**

**Usage**

```
isValidCohortId(cohortIds, baseUrl)
```

**Arguments**

cohortIds	A list of integer id(s) of the Cohort to be tested for validity.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Checks if a set of id for a Cohort is valid. The following checks are performed. 1) checks if all the ids exists in the WebApi i.e. valid.

**Value**

A logical vector indicating if an ID is valid.

**Examples**

```
## Not run:
isValidCohortId(cohortIds = c(13242, 3423, 34), baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

isValidConceptSetId	<i>is ConceptSet id a valid definition in the WebApi.</i> <b>[Stable]</b>
---------------------	---

---

**Description**

is ConceptSet id a valid definition in the WebApi. **[Stable]**

**Usage**

```
isValidConceptSetId(conceptSetIds, baseUrl)
```

**Arguments**

conceptSetIds	A list of integer id(s) of the ConceptSet to be tested for validity.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Checks if a set of id for a ConceptSet is valid. The following checks are performed. 1) checks if all the ids exists in the WebApi i.e. valid.

**Value**

A logical vector indicating if an ID is valid.

**Examples**

```
## Not run:
isValidConceptSetId(conceptSetIds = c(13242, 3423, 34), baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

isValidEstimationId	<i>is Estimation id a valid definition in the WebApi. [Stable]</i>
---------------------	--

---

### Description

is Estimation id a valid definition in the WebApi. **[Stable]**

### Usage

```
isValidEstimationId(estimationIds, baseUrl)
```

### Arguments

estimationIds	A list of integer id(s) of the Estimation to be tested for validity.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Checks if a set of id for a Estimation is valid. The following checks are performed. 1) checks if all the ids exists in the WebApi i.e. valid.

### Value

A logical vector indicating if an ID is valid.

### Examples

```
## Not run:
isValidEstimationId(estimationIds = c(13242, 3423, 34), baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

isValidId	<i>Check if an id is valid. [Stable]</i>
-----------	--

---

### Description

Check if an id is valid. **[Stable]**

### Usage

```
isValidId(ids, baseUrl, category)
```

### Arguments

ids	A list of integer id(s) of the category to be tested for validity.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
category	These are the categories in WebApi. The valid string options are 'conceptSet', 'cohort', 'characterization', 'pathway', 'incidenceRate', 'estimation', 'prediction'.

**Details**

Checks if a set of id for a category is valid, i.e. checks if all the ids exists in the WebApi i.e. valid.

**Value**

A logical vector indicating if an ID is valid.

**Examples**

```
## Not run:
isValidId(ids = c(13242, 3423, 34), baseUrl = "http://server.org:80/WebAPI", category = "cohort")

## End(Not run)
```

---

isValidIncidenceRateId

*is IncidenceRate id a valid definition in the WebApi. [Stable]*

---

**Description**

is IncidenceRate id a valid definition in the WebApi. **[Stable]**

**Usage**

```
isValidIncidenceRateId(incidenceRateIds, baseUrl)
```

**Arguments**

incidenceRateIds

A list of integer id(s) of the IncidenceRate to be tested for validity.

baseUrl

The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Checks if a set of id for a IncidenceRate is valid. The following checks are performed. 1) checks if all the ids exists in the WebApi i.e. valid.

**Value**

A logical vector indicating if an ID is valid.

**Examples**

```
## Not run:
isValidIncidenceRateId(incidenceRateIds = c(13242, 3423, 34),
                        baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

isValidPathwayId	<i>is Pathway id a valid definition in the WebApi. [Stable]</i>
------------------	---

---

**Description**

is Pathway id a valid definition in the WebApi. **[Stable]**

**Usage**

```
isValidPathwayId(pathwayIds, baseUrl)
```

**Arguments**

pathwayIds	A list of integer id(s) of the Pathway to be tested for validity.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Checks if a set of id for a Pathway is valid. The following checks are performed. 1) checks if all the ids exists in the WebApi i.e. valid.

**Value**

A logical vector indicating if an ID is valid.

**Examples**

```
## Not run:
isValidPathwayId(pathwayIds = c(13242, 3423, 34), baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

isValidPredictionId	<i>is Prediction id a valid definition in the WebApi. [Stable]</i>
---------------------	--

---

**Description**

is Prediction id a valid definition in the WebApi. **[Stable]**

**Usage**

```
isValidPredictionId(predictionIds, baseUrl)
```

**Arguments**

predictionIds	A list of integer id(s) of the Prediction to be tested for validity.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

## Details

Checks if a set of id for a Prediction is valid. The following checks are performed. 1) checks if all the ids exists in the WebApi i.e. valid.

## Value

A logical vector indicating if an ID is valid.

## Examples

```
## Not run:
isValidPredictionId(predictionIds = c(13242, 3423, 34), baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

isValidSourceKey	Check if source key is valid. <b>[Stable]</b>
------------------	---

### Description

Check if source key is valid. **[Stable]**

## Usage

```
isValidSourceKey(sourceKeys, baseUrl)
```

## Arguments

sourceKeys	The source key(s) for a CDM instance in WebAPI, as defined in the Configuration page.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

## Details

Checks if a set of sourceKey(s) are valid, i.e. checks if all the sourceKey(s) exists in the WebApi i.e. valid.

## Value

A logical vector indicating if an ID is valid.

## Examples

```
## Not run:
isValidSourceKey(sourceKeys = c("HCUP", "CCA"),
  baseUrl = "http://server.org:80/WebAPI",
  category = "cohort")

## End(Not run)
```

---

`postCharacterizationDefinition`*Post Characterization definition. [Maturing]*

---

## Description

Post Characterization definition. **[Maturing]**

## Usage

```
postCharacterizationDefinition(name, characterizationDefinition, baseUrl)
```

## Arguments

<code>name</code>	A valid name for the definition. WebApi will use this name (if valid) as the name of the definition. WebApi checks for validity, such as uniqueness, absence of unacceptable character etc. An error might be thrown.
<code>characterizationDefinition</code>	An R list object containing the expression for the specification. This will be converted to JSON expression by function and posted into the WebApi. Note: only limited checks are performed in R to check the validity of this expression.
<code>baseUrl</code>	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

## Details

Post Characterization definition to WebAPI

## Value

This function will return a dataframe object with one row describing the posted WebApi expression and its details. If unsuccessful a STOP message will be shown.

## Examples

```
## Not run:
postCharacterizationDefinition(name = "new valid name",
                              characterizationDefinition = definition,
                              baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```



---

postCohortDefinition    *Post Cohort definition. [Maturing]*

---

### Description

Post Cohort definition. **[Maturing]**

### Usage

```
postCohortDefinition(name, cohortDefinition, baseUrl)
```

### Arguments

name	A valid name for the definition. WebApi will use this name (if valid) as the name of the definition. WebApi checks for validity, such as uniqueness, absence of unacceptable character etc. An error might be thrown.
cohortDefinition	An R list object containing the expression for the specification. This will be converted to JSON expression by function and posted into the WebApi. Note: only limited checks are performed in R to check the validity of this expression.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Post Cohort definition to WebAPI

### Value

This function will return a dataframe object with one row describing the posted WebApi expression and its details. If unsuccessful a STOP message will be shown.

### Examples

```
## Not run:
postCohortDefinition(name = "new valid name",
                     cohortDefinition = definition,
                     baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

postConceptSetDefinition  
                           *Post ConceptSet definition. [Maturing]*

---

### Description

Post ConceptSet definition. **[Maturing]**

### Usage

```
postConceptSetDefinition(name, conceptSetDefinition, baseUrl)
```

**Arguments**

name	A valid name for the definition. WebApi will use this name (if valid) as the name of the definition. WebApi checks for validity, such as uniqueness, absence of unacceptable character etc. An error might be thrown.
conceptSetDefinition	An R list object containing the expression for the specification. This will be converted to JSON expression by function and posted into the WebApi. Note: only limited checks are performed in R to check the validity of this expression.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Post ConceptSet definition to WebAPI

**Value**

This function will return a dataframe object with one row describing the posted WebApi expression and its details. If unsuccessful a STOP message will be shown.

**Examples**

```
## Not run:
postConceptSetDefinition(name = "new valid name",
                        conceptSetDefinition = definition,
                        baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

postDefinition	<i>Post a definition into WebApi [Maturing]</i>
----------------	---

---

**Description**

Post a definition into WebApi [Maturing]

**Usage**

```
postDefinition(baseUrl, name, category, definition)
```

**Arguments**

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
name	A valid name for the definition. WebApi will use this name (if valid) as the name of the definition. WebApi checks for validity, such as uniqueness, absence of unacceptable character etc. An error might be thrown.
category	These are the categories in WebApi. The valid string options are 'conceptSet', 'cohort', 'characterization', 'pathway', 'incidenceRate', 'estimation', 'prediction'.
definition	An R list object containing the expression for the specification. This will be converted to JSON expression by function and posted into the WebApi. Note: only limited checks are performed in R to check the validity of this expression.

**Details**

Post a definition into WebAPI. Currently only cohort and concept-set are supported.

**Value**

This function will return a dataframe object with one row describing the posted WebApi expression and its details. If unsuccessful a STOP message will be shown.

**Examples**

```
## Not run:
definition <- getCohortDefinition(baseUrl = baseUrl, cohortId = 15873)
postDefinition(name = "new name for expression in sdaddadd",
              baseUrl = "http://server.org:80/WebAPI",
              expression = definition,
              category = "cohort")

## End(Not run)
```

---

postEstimationDefinition

*Post Estimation definition. [Maturing]*

---

**Description**

Post Estimation definition. [Maturing]

**Usage**

```
postEstimationDefinition(name, estimationDefinition, baseUrl)
```

**Arguments**

name	A valid name for the definition. WebApi will use this name (if valid) as the name of the definition. WebApi checks for validity, such as uniqueness, absence of unacceptable character etc. An error might be thrown.
estimationDefinition	An R list object containing the expression for the specification. This will be converted to JSON expression by function and posted into the WebApi. Note: only limited checks are performed in R to check the validity of this expression.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Post Estimation definition to WebAPI

**Value**

This function will return a dataframe object with one row describing the posted WebApi expression and its details. If unsuccessful a STOP message will be shown.

**Examples**

```
## Not run:
postEstimationDefinition(name = "new valid name",
                        estimationDefinition = definition,
                        baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

postIncidenceRateDefinition

*Post IncidenceRate definition. [Maturing]*

---

**Description**

Post IncidenceRate definition. **[Maturing]**

**Usage**

```
postIncidenceRateDefinition(name, incidenceRateDefinition, baseUrl)
```

**Arguments**

name	A valid name for the definition. WebApi will use this name (if valid) as the name of the definition. WebApi checks for validity, such as uniqueness, absence of unacceptable character etc. An error might be thrown.
incidenceRateDefinition	An R list object containing the expression for the specification. This will be converted to JSON expression by function and posted into the WebApi. Note: only limited checks are performed in R to check the validity of this expression.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Post IncidenceRate definition to WebAPI

**Value**

This function will return a dataframe object with one row describing the posted WebApi expression and its details. If unsuccessful a STOP message will be shown.

**Examples**

```
## Not run:
postIncidenceRateDefinition(name = "new valid name",
                          incidenceRateDefinition = definition,
                          baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

postPathwayDefinition *Post Pathway definition. [Maturing]*

---

### Description

Post Pathway definition. **[Maturing]**

### Usage

```
postPathwayDefinition(name, pathwayDefinition, baseUrl)
```

### Arguments

name	A valid name for the definition. WebApi will use this name (if valid) as the name of the definition. WebApi checks for validity, such as uniqueness, absence of unacceptable character etc. An error might be thrown.
pathwayDefinition	An R list object containing the expression for the specification. This will be converted to JSON expression by function and posted into the WebApi. Note: only limited checks are performed in R to check the validity of this expression.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

### Details

Post Pathway definition to WebAPI

### Value

This function will return a dataframe object with one row describing the posted WebApi expression and its details. If unsuccessful a STOP message will be shown.

### Examples

```
## Not run:
postPathwayDefinition(name = "new valid name",
                      pathwayDefinition = definition,
                      baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

postPredictionDefinition  
*Post Prediction definition. [Maturing]*

---

### Description

Post Prediction definition. **[Maturing]**

### Usage

```
postPredictionDefinition(name, predictionDefinition, baseUrl)
```

**Arguments**

name	A valid name for the definition. WebApi will use this name (if valid) as the name of the definition. WebApi checks for validity, such as uniqueness, absence of unacceptable character etc. An error might be thrown.
predictionDefinition	An R list object containing the expression for the specification. This will be converted to JSON expression by function and posted into the WebApi. Note: only limited checks are performed in R to check the validity of this expression.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Post Prediction definition to WebAPI

**Value**

This function will return a dataframe object with one row describing the posted WebApi expression and its details. If unsuccessful a STOP message will be shown.

**Examples**

```
## Not run:
postPredictionDefinition(name = "new valid name",
                        predictionDefinition = definition,
                        baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

resolveConceptSet	<i>Resolve a concept set to the included standard concept IDs</i> <b>[Stable]</b>
-------------------	---

---

**Description**

Resolve a concept set to the included standard concept IDs **[Stable]**

**Usage**

```
resolveConceptSet(conceptSetDefinition, baseUrl, vocabularySourceKey = NULL)
```

**Arguments**

conceptSetDefinition	A concept set definition, for example as obtained through the <a href="#">getConceptSetDefinition</a> function, or taken from a cohort definition.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
vocabularySourceKey	The source key of the Vocabulary. By default, the priority Vocabulary is used.

**Details**

Resolve a concept set to the included standard concept IDs

**Value**

A vector of standard concept ids.

**Examples**

```
## Not run:
conceptSetDefinition <- getConceptSetDefinition(conceptSetId = 282,
                                              baseUrl = "http://server.org:80/WebAPI")
conceptIds <- resolveConceptSet(conceptSetDefinition = conceptSetDefinition,
                               baseUrl = "http://server.org:80/WebAPI")

## End(Not run)
```

---

setAuthHeader	<i>Manually set the authorization http header for a WebAPI baseUrl In some cases the user may want to manually set the authorization header. An authHeader is associated with a particular baseUrl and added to to the header of all http requests sent to that url by ROhdsi-WebApi.</i>
---------------	---

---

**Description**

Manually set the authorization http header for a WebAPI baseUrl In some cases the user may want to manually set the authorization header. An authHeader is associated with a particular baseUrl and added to to the header of all http requests sent to that url by ROhdsiWebApi.

**Usage**

```
setAuthHeader(baseUrl, authHeader)
```

**Arguments**

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
authHeader	A character string containing a Bearer token that will be added to the header of all http requests sent to baseUrl. (e.g. "Bearer lxd9n2nsdsd2329km23mexjop02m23m23mmmsioxiiis")

---

updateCohortDefinition	<i>Update a Cohort definition. [Maturing]</i>
------------------------	---

---

**Description**

Update a Cohort definition. [Maturing]

**Usage**

```
updateCohortDefinition(cohortDefinition, baseUrl)
```

**Arguments**

cohortDefinition	An R list object containing the expression for the specification. Must include id, name and expression. This will be converted to JSON expression by function and posted into the WebApi. The definition will be checked against the WebApi instance for errors
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Update a Cohort definition.

**Examples**

```
## Not run:
definition <- getCohortDefinition(id = 13242,
                                baseUrl = "http://server.org:80/WebAPI",
                                category = cohort)
definition$name <- "My new name for this"
updateCohort(cohortDefinition, baseUrl, category = "cohort")

## End(Not run)
```

---

updateConceptSetDefinition	<i>Update a ConceptSet definition. [Maturing]</i>
----------------------------	---

---

**Description**

Update a ConceptSet definition. **[Maturing]**

**Usage**

```
updateConceptSetDefinition(conceptSetDefinition, baseUrl)
```

**Arguments**

conceptSetDefinition	An R list object containing the expression for the specification. Must include id, name and expression. This will be converted to JSON expression by function and posted into the WebApi. The definition will be checked against the WebApi instance for errors
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".

**Details**

Update a ConceptSet definition.



**Examples**

```
## Not run:
definition <- getConceptSetDefinition(id = 13242,
                                     baseUrl = "http://server.org:80/WebAPI",
                                     category = conceptSet)
definition$name <- "My new name for this"
updateConceptSet(conceptSetDefinition, baseUrl, category = "cohort")

## End(Not run)
```

---

updateDefinition	<i>Update definition</i> <b>[Maturing]</b>
------------------	--

---

**Description**

Update definition **[Maturing]**

**Usage**

```
updateDefinition(definition, baseUrl, category)
```

**Arguments**

definition	An R list object containing the expression for the specification. This will be converted to JSON expression by function and posted into the WebApi.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI".
category	These are the categories in WebApi. The valid string options are 'conceptSet', 'cohort', 'characterization', 'pathway', 'incidenceRate', 'estimation', 'prediction'.

**Details**

Update a definition in WebAPI. Currently only cohorts are supported. Takes the definition as a parameter and converts it to json. This is the full definition (i.e. including name and id fields)

**Examples**

```
## Not run:
definition <- getDefinition(id = 13242, baseUrl = "http://server.org:80/WebAPI", category = "cohort")
definition$name <- "My new name for this"
updateDefinition(definition, baseUrl, category = "cohort")

## End(Not run)
```

# Index

authorizeWebApi, 4

cancelCharacterizationGeneration, 5

cancelCohortGeneration, 5

cancelGeneration, 6

cancelIncidenceRateGeneration, 7

cancelPathwayGeneration, 8

checkInputFileEncoding, 8

convertConceptSetDefinitionToTable, 9

createConceptSetWorkbook, 10

deleteCharacterizationDefinition, 10

deleteCohortDefinition, 11

deleteConceptSetDefinition, 12

deleteDefinition, 12

deleteEstimationDefinition, 13

deleteIncidenceRateDefinition, 14

deletePathwayDefinition, 14

deletePredictionDefinition, 15

detectCharacterizationsByName, 16

detectCohortsByName, 16

detectConceptSetsByName, 17

detectEstimationsByName, 18

detectIncidenceRatesByName, 19

detectPathwaysByName, 19

detectPredictionsByName, 20

existsCharacterizationName, 21

existsCohortName, 22

existsConceptSetName, 22

existsEstimationName, 23

existsIncidenceRateName, 24

existsPathwayName, 24

existsPredictionName, 25

getCdmSources, 26

getCharacterizationDefinition, 26

getCharacterizationDefinitionsMetadata, 16, 27

getCharacterizationGenerationInformation, 28

getCharacterizationResults, 28

getCohortDefinition, 29

getCohortDefinitionExpression, 30

getCohortDefinitionName, 30

getCohortDefinitionsMetadata, 17, 31

getCohortDefinitionSql, 32

getCohortGenerationInformation, 32

getCohortInclusionRulesAndCounts, 33

getCohortResults, 34

getCohortSql, 34

getConcepts, 35

getConceptSetDefinition, 9, 36, 70

getConceptSetDefinitionsMetadata, 17, 37

getDefinition, 37

getDefinitionsMetadata, 38

getEstimationDefinition, 39

getEstimationDefinitionsMetadata, 18, 40

getGenerationInformation, 40

getIncidenceRateDefinition, 41

getIncidenceRateDefinitionsMetadata, 19, 42

getIncidenceRateGenerationInformation, 43

getIncidenceRateResults, 43

getPathwayDefinition, 44

getPathwayDefinitionsMetadata, 20, 45

getPathwayGenerationInformation, 45

getPathwayResults, 46

getPersonProfile, 47

getPredictionDefinition, 48

getPredictionDefinitionsMetadata, 20, 48

getPriorityVocabularyKey, 49

getResults, 50

getSourceConcepts, 50

getWebApiVersion, 51

insertCohortDefinitionInPackage, 52

insertCohortDefinitionSetInPackage, 53

invokeCharacterizationGeneration, 54

invokeCohortGeneration, 55

invokeGeneration, 55

invokeIncidenceRateGeneration, 56

invokePathwayGeneration, 57

isValidCharacterizationId, 58

isValidCohortId, [58](#)  
isValidConceptSetId, [59](#)  
isValidEstimationId, [60](#)  
isValidId, [60](#)  
isValidIncidenceRateId, [61](#)  
isValidPathwayId, [62](#)  
isValidPredictionId, [62](#)  
isValidSourceKey, [63](#)  
  
postCharacterizationDefinition, [64](#)  
postCohortDefinition, [65](#)  
postConceptSetDefinition, [65](#)  
postDefinition, [66](#)  
postEstimationDefinition, [67](#)  
postIncidenceRateDefinition, [68](#)  
postPathwayDefinition, [69](#)  
postPredictionDefinition, [69](#)  
  
resolveConceptSet, [70](#)  
  
setAuthHeader, [71](#)  
str\_detect, [16–20](#)  
  
updateCohortDefinition, [71](#)  
updateConceptSetDefinition, [72](#)  
updateDefinition, [73](#)