

# Package ‘OmopViewer’

October 21, 2024

**Type** Package

**Title** Visualise OMOP Results using 'shiny' Applications

**Version** 0.1.0

**Maintainer** Martí Català <martí.catalasabate@ndorms.ox.ac.uk>

**Description** Visualise results obtained from analysing data mapped to the Observational Medical Outcomes Partnership (OMOP) common data model using 'shiny' applications.

**License** Apache License (>= 2)

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**URL** <https://ohdsi.github.io/OmopViewer/>

**Depends** R (>= 2.10)

**Imports** bslib, cli, dplyr, DT, glue, markdown, omopgenerics (>= 0.3.1), purrr, rlang, shiny, snakecase, stringr, styler, tidyr, usethis, vctrs, visOmopResults (>= 0.4.0), yaml

**Suggests** CDMConnector, CodelistGenerator (>= 3.2.0), CohortCharacteristics (>= 0.3.0), CohortConstructor, DiagrammeR, DrugUtilisation (>= 0.7.1), duckdb, ggplot2, gt, here, htmlwidgets, IncidencePrevalence (>= 0.8.0), knitr, omock (>= 0.3.0), OmopSketch, PatientProfiles, plotly, rmarkdown, RPostgres, shinycssloaders, shinyWidgets, sortable, testthat (>= 3.1.5), utils, webshot2, zip

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Config/testthat/parallel** true

**NeedsCompilation** no

**Author** Martí Català [aut, cre] (<<https://orcid.org/0000-0003-3308-9905>>),  
Yuchen Guo [aut] (<<https://orcid.org/0000-0002-0847-4855>>),  
Edward Burn [ctb] (<<https://orcid.org/0000-0002-9286-1128>>),  
Nuria Mercade-Besora [ctb] (<<https://orcid.org/0009-0006-7948-3747>>),  
Elin Rowlands [ctb] (<<https://orcid.org/0009-0005-5166-0417>>),  
Cecilia Campanile [ctb] (<<https://orcid.org/0009-0007-6629-4661>>)

**Repository** CRAN

**Date/Publication** 2024-10-21 12:00:52 UTC

## Contents

cardFromMd . . . . .	2
cardSummary . . . . .	3
correctSettings . . . . .	3
exportStaticApp . . . . .	4
filterData . . . . .	5
getChoices . . . . .	5
launchDynamicApp . . . . .	6
omopViewerTable . . . . .	6
tidyData . . . . .	7
<b>Index</b>	<b>8</b>

---

cardFromMd	<i>Create a 'bslib::card()' object from a '.md' file.</i>
------------	---

---

## Description

Create a 'bslib::card()' object from a '.md' file.

## Usage

```
cardFromMd(fileName)
```

## Arguments

fileName      Name of the .md file.

## Value

Ui 'bslib::card' with the background details.

---

cardSummary	Create a <code>'bslib::card()'</code> object from a <code>'&lt;summarised_result&gt;'</code> object.
-------------	--

---

**Description**

Create a `'bslib::card()'` object from a `'<summarised_result>'` object.

**Usage**

```
cardSummary(result)
```

**Arguments**

result            A `'<summarised_result>'` object.

**Value**

A `'bslib::card()'` object.

---

correctSettings	Add the strata, group and additional columns to settings with the tidyColumns ( <code>'visOmopResults::splitGroup()'</code> , <code>'visOmopResults::splitStrata()'</code> , <code>'visOmopResults::splitAdditional()'</code> ) for each <code>'result_id'</code> .
-----------------	---

---

**Description**

Add the strata, group and additional columns to settings with the tidyColumns (`'visOmopResults::splitGroup()'`, `'visOmopResults::splitStrata()'`, `'visOmopResults::splitAdditional()'`) for each `'result_id'`.

**Usage**

```
correctSettings(result)
```

**Arguments**

result            A `'<summarised_result>'` object.

**Value**

The summarised\_result object with the settings corrected

---

exportStaticApp      *Export and launch a static shiny specific to the provided results.*

---

## Description

Export and launch a static shiny specific to the provided results.

## Usage

```
exportStaticApp(  
  result,  
  directory,  
  logo = "ohdsi",  
  title = "",  
  background = TRUE,  
  summary = TRUE,  
  panels = list(),  
  theme = NULL,  
  open = rlang::is_interactive()  
)
```

## Arguments

result	A summarised_result object.
directory	Directory to create the shiny.
logo	Name of a logo or path to a logo. If NULL no logo is included. Only svg format allowed for the moment.
title	title of the shiny
background	Whether to include a background panel. Background panel content will be controlled from the generated background.md file.
summary	Whether to include a panel with a summary of content in the 'result'.
panels	List specifying order of results. Each panel is determined by the available result types in the result object. Panels for any available results not specified will be included after the specified result tabs.
theme	Assign a theme to the shiny app using bslib::bs_theme()
open	Whether to open the shiny app project.

## Value

The shiny app will be created in directory.

**Examples**

```
exportStaticApp(
  result = emptySummarisedResult(),
  directory = tempdir(),
  theme = "bslib::bs_theme(bg = '#bb0a1e', fg = '#0000ff')")
)
```

---

filterData	<i>This function is used to filter data in shinys that inputs are defined following.</i>
------------	--

---

**Description**

This function is used to filter data in shinys that inputs are defined following.

**Usage**

```
filterData(result, resultType, input)
```

**Arguments**

result	A summarised_result object.
resultType	A resultType of interest to filter by.
input	Input of the shiny to filter by.

**Value**

The filtered result.

---

getChoices	<i>Get the different options that a summarised_result have.</i>
------------	---

---

**Description**

Get the different options that a summarised\_result have.

**Usage**

```
getChoices(result, flatten = FALSE)
```

**Arguments**

result	A '<summarised_result>' object.
flatten	Whether to flatten to a single list or not.

**Value**

A named list with the options

**Examples**

```
library(CohortCharacteristics)

cdm <- mockCohortCharacteristics()

result <- cdm$cohort1 |>
  summariseCharacteristics()

getChoices(result)
```

---

launchDynamicApp	<i>Launch a dynamic shiny app where you can upload results.</i>
------------------	---

---

**Description**

Launch a dynamic shiny app where you can upload results.

**Usage**

```
launchDynamicApp()
```

**Value**

Launches the shiny app.

---

omopViewerTable	<i>Create a formatted gt table.</i>
-----------------	-------------------------------------

---

**Description**

Create a formatted gt table.

**Usage**

```
omopViewerTable(
  result,
  header = character(),
  group = character(),
  hide = character()
)
```

**Arguments**

result	A tibble it must contain at least two columns: estimate_value and estimate_type.
header	Column names to move to header, must be columns in result.
group	Column names to group by, must be columns in result.
hide	Column names to hide, must be columns in result.

**Value**

A gt table.

---

tidyData	<i>Get a tidy tibble from a 'summarised_result' object.</i>
----------	---

---

**Description**

Get a tidy tibble from a 'summarised\_result' object.

**Usage**

```
tidyData(result)
```

**Arguments**

result	A 'summarised_result' object.
--------	-------------------------------

**Value**

A tibble.

# Index

cardFromMd, [2](#)  
cardSummary, [3](#)  
correctSettings, [3](#)  
  
exportStaticApp, [4](#)  
  
filterData, [5](#)  
  
getChoices, [5](#)  
  
launchDynamicApp, [6](#)  
  
omopViewerTable, [6](#)  
  
tidyData, [7](#)