

# Package ‘OmopViewer’

December 10, 2025

**Type** Package

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

**Version** 0.6.0

**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

**LazyData** true

**RoxygenNote** 7.3.3

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

**BugReports** <https://github.com/ohdsi/OmopViewer/issues>

**Depends** R (>= 4.1)

**Imports** bslib, cli, dplyr, DT, glue, gt, lifecycle, markdown, omopgenerics (>= 0.4.0), purrr, rlang, shiny, snakecase, stringr, styler, tidyr, usethis, visOmopResults (>= 1.4.0), yaml

**Suggests** CDMConnector, CodelistGenerator (>= 3.2.0), CohortCharacteristics (>= 1.0.0), CohortConstructor, CohortSurvival, cpp11, devtools, DiagrammeR, DiagrammeRsvg, DrugUtilisation (>= 0.7.1), duckdb, ggplot2, here, htmlwidgets, IncidencePrevalence (>= 0.8.0), knitr, MeasurementDiagnostics, omock (>= 0.3.0), OmopSketch, PatientProfiles, plotly, progress, reactable, renv, rmarkdown, RPostgres, rsconnect, rsvg, shiny.fluent, shinycssloaders, shinyTree, shinyWidgets (>= 0.9.0), 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] (ORCID:

<<https://orcid.org/0000-0003-3308-9905>>),

Yuchen Guo [aut] (ORCID: <<https://orcid.org/0000-0002-0847-4855>>),

Edward Burn [ctb] (ORCID: <<https://orcid.org/0000-0002-9286-1128>>),

Nuria Mercade-Besora [ctb] (ORCID:

<<https://orcid.org/0009-0006-7948-3747>>),

Elin Rowlands [ctb] (ORCID: <<https://orcid.org/0009-0005-5166-0417>>),

Cecilia Campanile [ctb] (ORCID:

<<https://orcid.org/0009-0007-6629-4661>>)

**Repository** CRAN

**Date/Publication** 2025-12-10 10:00:18 UTC

## Contents

defaultPanels . . . . .	2
exportStaticApp . . . . .	3
getPanel . . . . .	4
launchDynamicApp . . . . .	5
omopViewerPanels . . . . .	5
omopViewerResults . . . . .	6
omopViewerThemes . . . . .	6
panelDetailsFromResult . . . . .	7
<b>Index</b>	<b>8</b>

---

defaultPanels	<i>Default panels defined in the package.</i>
---------------	---

---

## Description

Default panels defined in the package.

## Usage

```
defaultPanels()
```

## Value

Names of the default panels defined in the package.

## Examples

```
defaultPanels()
```

---

exportStaticApp	<i>Export and launch a static shiny specific to the provided results.</i>
-----------------	---

---

### Description

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

### Usage

```
exportStaticApp(
  result,
  directory,
  logo = "ohdsi",
  title = "",
  background = TRUE,
  summary = TRUE,
  report = FALSE,
  panelDetails = panelDetailsFromResult(result),
  panelStructure = NULL,
  theme = NULL,
  template = NULL,
  updateButtons = TRUE,
  includeOneChoiceFilters = TRUE,
  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.
report	Whether to include a quarto report.
panelDetails	A named list to provide details for each one of the panels, such as: result_id, result_type, title, icon, filters and content. By default it is created using the panelDetailsFromResult() function.
panelStructure	A named list of panel identifiers to organise them in drop-down menus. Identifiers names are the ones used in panelDetails. By default one panel per each panelDetails element is created.

theme	Specify the theme for the Shiny application. You can either select a predefined theme provided by the package (e.g., "theme1"), or define a custom theme using <code>bslib::bs_theme()</code> . If using a custom theme, it must be provided as a character string (e.g., " <code>bslib::bs_theme(bg = 'white', fg = 'black')</code> ").
template	Path to a template .docx document to be used for the report.
updateButtons	Whether to include update buttons for visualisations.
includeOneChoiceFilters	Whether to include filter buttons for filters with just one choice.
open	Whether to open the shiny app project.

**Value**

The shiny app will be created in directory.

**Examples**

```
exportStaticApp(
  result = omopgenerics::emptySummarisedResult(),
  directory = tempdir()
)
```

---

getPanel

*Get one of the default pre-built panels.*

---

**Description**

Get one of the default pre-built panels.

**Usage**

```
getPanel(panelId)
```

**Arguments**

panelId      Name of the one of the default panels. Use `defaultPanels()` to see the available default panels.

**Value**

A panel definition.

**Examples**

```
getPanel("incidence")
```

---

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

---

### Description

**[Experimental]**

### Usage

```
launchDynamicApp()
```

### Value

Launches the shiny app.

---

omopViewerPanels	<i>Panels defined in the package.</i>
------------------	---------------------------------------

---

### Description

The panels are generated using the following file: <https://github.com/OHDSI/OmopViewer/blob/main/data-raw/panels.R>.

### Usage

```
omopViewerPanels
```

### Format

omopViewerPanels:  
A list of 26 omopviewer\_panel objects

omopViewerResults      *Mock results obtained from GiBleed dataset.*

---

**Description**

The results are generated using the following file: <https://github.com/OHDSI/OmopViewer/blob/main/data-raw/results.R>.

**Usage**

```
omopViewerResults
```

**Format**

```
omopViewerResults:  
A summarised_result object with 12,017 rows
```

---

omopViewerThemes      *List the available configured OmopViewer themes*

---

**Description**

List the available configured OmopViewer themes

**Usage**

```
omopViewerThemes()
```

**Value**

A character vector with the available OmopViewer themes.

**Examples**

```
library(OmopViewer)  
omopViewerThemes()
```

---

panelDetailsFromResult

*Obtain default panel details from a <summarised\_result> object.*

---

### **Description**

Obtain default panel details from a <summarised\_result> object.

### **Usage**

```
panelDetailsFromResult(result, includeOneChoiceFilters = TRUE)
```

### **Arguments**

result            A summarised\_result object.  
includeOneChoiceFilters  
                  Whether to include filters that contain only one choice.

### **Value**

A list of omop\_viewer\_panel objects.

### **Examples**

```
panelDetailsFromResult(omopViewerResults)
```

# Index

## \* **datasets**

- omopViewerPanels, [5](#)
- omopViewerResults, [6](#)

defaultPanels, [2](#)

exportStaticApp, [3](#)

getPanel, [4](#)

launchDynamicApp, [5](#)

omopViewerPanels, [5](#)

omopViewerResults, [6](#)

omopViewerThemes, [6](#)

panelDetailsFromResult, [7](#)