

Package: owdata (via r-universe)

October 9, 2026

Title Index of openwashdata R Data Packages

Version 0.1.0

Description Index metapackage for the openwashdata R data packages.

Ships a harvested catalog of every package in the openwashdata
GitHub organisation, with its datasets and variables, as
internal data behind accessor and search functions, and
installs any listed package from the openwashdata R-universe.
The catalog is refreshed by a weekly harvest.

License CC BY 4.0

URL <https://openwashdata.github.io/owdata/>,

<https://github.com/openwashdata/owdata>

Encoding UTF-8

Roxygen list(markdown = TRUE)

Suggests jsonlite, testthat (>= 3.0.0)

Config/testthat/edition 3

Imports tibble, utils

Language en-GB

LazyData true

Config/Needs/website rmarkdown, brand.yml

Date 2026-09-09

BugReports <https://github.com/openwashdata/owdata/issues>

Config/Needs/harvest desc, cffr, gh, jsonlite

Depends R (>= 3.5)

Config/roxygen2/version 8.1.0

Repository <https://openwashdata.r-universe.dev>

Date/Publication 2026-09-09 14:07:05 UTC

RemoteUrl <https://github.com/openwashdata/owdata>

RemoteRef HEAD

RemoteSha 54b1e4dc74293dcae7a0c993ac83fc9c655f7335

Contents

owd_datasets	2
owd_install	3
owd_packages	3
owd_search	4
owd_variables	5
Index	6

owd_datasets	<i>List the datasets shipped by published packages</i>
--------------	--

Description

Dataset detail is harvested for published packages only; an unpublished package appears in [owd_packages\(\)](#) but not here.

Usage

```
owd_datasets(pkg = NULL)
```

Arguments

pkg	Optional character vector of package names to keep. NULL (the default) returns all rows.
-----	--

Value

A tibble with one row per dataset, with columns pkg_name, dataset_name, title, description, n_rows, n_vars, n_vars_dictionary, csv_url and xls_url.

Examples

```
owd_datasets()
owd_datasets(pkg = "washmalawi")
```

owd_install	<i>Install openwashdata packages from the openwashdata R-universe</i>
-------------	---

Description

A thin wrapper around `utils::install.packages()` that points at <https://openwashdata.r-universe.dev> (with CRAN as fallback for dependencies).

Usage

```
owd_install(pkg = NULL, ...)
```

Arguments

<code>pkg</code>	Character vector of package names to install. NULL (the default) installs every published package after an interactive confirmation showing the count; in non-interactive sessions NULL is an error, so scripts must name packages explicitly.
<code>...</code>	Passed on to <code>utils::install.packages()</code> .

Value

Invisibly, the character vector of package names requested.

Examples

```
## Not run:
owd_install("washmalawi")
owd_install()

## End(Not run)
```

owd_packages	<i>List the openwashdata packages in the catalog</i>
--------------	--

Description

Returns the harvested package table. Every detected package is listed, whether or not it has been published; use `published` to restrict the result.

Usage

```
owd_packages(published = NULL)
```

Arguments

<code>published</code>	Optional logical. TRUE keeps published packages, FALSE keeps the rest, NULL (the default) returns all rows.
------------------------	---

Details

A package counts as published when its CITATION.cff carries a Zenodo DOI. There are no version tiers, so a version number says nothing about whether a package is published.

Value

A tibble with one row per package and these columns: pkg_name, title, description, version (leading v stripped), published, reviewed (NA until the pkgreview check is wired), maintainer, maintainer_orcid, authors, license, date, keywords, spatial_coverage, temporal_coverage, doi, cff_version, url_github, url_docs, n_datasets, latest_release, default_branch, created_at, last_commit, topics, has_dictionary, has_extdata, and the tooling provenance columns washr_version, brand_ref, pkgreview_standard, has_r_cmd_check, has_pkgdown_workflow and has_jsonld.

Examples

```
owd_packages()
owd_packages(published = TRUE)
```

owd_search	<i>Search the openwashdata catalog</i>
------------	--

Description

Runs a case-insensitive regular expression over package, dataset and variable metadata, so `owd_search("groundwater")` answers "which package has data about groundwater". The search runs against a prebuilt flat index, so it is a single vectorized match regardless of catalog size.

Usage

```
owd_search(
  query,
  fields = c("title", "description", "variable_name", "dataset_name")
)
```

Arguments

query	A single regular expression (case-insensitive). Plain words work as-is.
fields	Character vector of fields to search. Any of "title", "description", "variable_name", "dataset_name", "pkg_name". The default covers titles, descriptions, variable names and dataset names.

Value

A tibble of hits with columns pkg_name, match_type ("package", "dataset" or "variable"), dataset_name, field and text (the matched text). Zero rows when nothing matches.

Examples

```
owd_search("sanitation")
owd_search("water", fields = "variable_name")
```

`owd_variables`*List the documented variables across the catalog*

Description

The cross-package variable dictionary, harvested from each package's `data-raw/dictionary.csv`. Published packages only, as in `owd_datasets()`.

Usage

```
owd_variables(pkg = NULL, dataset = NULL)
```

Arguments

<code>pkg</code>	Optional character vector of package names to keep.
<code>dataset</code>	Optional character vector of dataset names to keep.

Value

A tibble with one row per documented variable, with columns `pkg_name`, `dataset_name`, `variable_name`, `variable_type` (first class only) and `description`.

Examples

```
owd_variables(pkg = "washmalawi")
head(owd_variables())
```

Index

`owd_datasets`, [2](#)
`owd_datasets()`, [5](#)
`owd_install`, [3](#)
`owd_packages`, [3](#)
`owd_packages()`, [2](#)
`owd_search`, [4](#)
`owd_variables`, [5](#)

`utils::install.packages()`, [3](#)