Package 'ready4'

April 16, 2024

Title Implement Modular and Open-Source Health Economic Models

Version 0.1.9

Description Programming syntax, a template model module and tools to help maintain a modular and open-source health economic model's project documentation website. These elements are the foundation for a prototype software framework to support replicable and transferable health economic models. The software framework is extended by other R libraries. For detailed documentation about the framework and how to use it visit <https://www.ready4-dev.com/>. For a background to the methodological issues that the framework is attempting to help solve, see Hamilton et al. (2024) <doi:10.48550/arXiv.2310.14138>.

License GPL-3

URL https://ready4-dev.github.io/ready4/,
https://github.com/ready4-dev/ready4,
https://www.ready4-dev.com/

Encoding UTF-8

RoxygenNote 7.3.1

Collate 'C4_Ready4Module.R' 'C4_Ready4Private.R' 'C4_Ready4Public.R' 'fn_add.R' 'fn_bind.R' 'fn_get.R' 'fn_make.R' 'fn_print.R' 'fn_remove.R' 'fn_rowbind.R' 'fn_transform.R' 'fn_update.R' 'fn_write.R' 'grp_generics.R' 'imp_fns.R' 'imp_mthds.R' 'mthd_authorSlot.R' 'mthd_characterizeSlot.R' 'mthd_depictSlot.R' 'mthd_enhanceSlot.R' 'mthd_exhibitSlot.R' 'mthd_ingestSlot.R' 'mthd_investigateSlot.R' 'mthd_manufactureSlot.R' 'mthd_metamorphoseSlot.R' 'mthd_procureSlot.R' 'mthd_prognosticateSlot.R' 'mthd_ratifySlot.R' 'mthd_reckonSlot.R' 'mthd_renewSlot.R' 'mthd_shareSlot.R' 'pkg_ready4.R' 'ready4-package.R'

Suggests devtools, Hmisc, knitr, pkgload, readr, readxl, rmarkdown, testthat, usethis, zen4R

VignetteBuilder knitr

Imports dataverse, dplyr, gh, kableExtra, lifecycle, magrittr, methods, piggyback, purrr, rlang, rvest, stats, stringi, stringr, tibble, tidyRSS, tidyselect, tools, utils

NeedsCompilation no

Author Matthew Hamilton [aut, cre, cph] (<https://orcid.org/0000-0001-7407-9194>), Orygen [cph, fnd], Australian Government Research Training Program [fnd], VicHealth [fnd], Victoria University [fnd]

Maintainer Matthew Hamilton <matthew.hamilton1@monash.edu>

Repository CRAN

Date/Publication 2024-04-16 09:10:02 UTC

R topics documented:

`author` *Author and save files*

Description

author() is a method that authors and saves files.

Usage

```
author(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method or no return value (when called for side-effects only).

---

authorClasses *Author and document classes*

Description

authorClasses() is a method that authors and saves R package files for creating and documenting classes to describe the data structures of model modules.

Usage

```
authorClasses(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method or no return value (when called for side-effects only).

---

authorData *Author and document datasets*

Description

authorData() is a method that authors, documents and saves model module datasets.

Usage

```
authorData(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method or no return value (when called for side-effects only).

---

authorFunctions *Author and document functions*

Description

authorFunctions() is a method that authors and saves R package files files necessary for creating and documenting functions that implement model module algorithms.

Usage

```
authorFunctions(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method or no return value (when called for side-effects only).

---

authorReport *Author and save a report*

Description

authorReport() is a method that authors and saves a report.

Usage

```
authorReport(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method or no return value (when called for side-effects only).

`authorSlot` *Apply the author method to a model module slot*

Description

authorSlot() is a convenience method that applies the author method to a specified slot of a model module.

authorSlot method applied to Ready4Module

Usage

```
authorSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
authorSlot(x, slot_nm_1L_chr, ...)
```

Arguments

`x`	An object of class Ready4Module
`slot_nm_1L_chr`	Slot name (a length one character vector)
`...`	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or no value (when called for side effects only).

Either a ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or no value (when called for side effects only).

characterize *Characterize model module data by generating (tabular) descriptive statistics*

Description

characterize() is a method that generates descriptive tabular summaries about data contained in a model module.

Usage

```
characterize(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A data.frame, tibble or other table based class.

characterizeSlot *Apply the characterize method to a model module slot*

Description

characterizeSlot() is a convenience method that applies the characterize method to a specified slot of a model module.

characterizeSlot method applied to Ready4Module

Usage

```
characterizeSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
characterizeSlot(x, slot_nm_1L_chr, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or a data.frame, tibble or other table class.

Either a ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or a data.frame, tibble or other table class.

---

depict *Depict (plot) features of model module data*

Description

depict() is a method that plots features of data contained in a model module (or sub-module).

Usage

```
depict(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A ggplot, gg or other plot type class.

---

depictSlot *Apply the depict method to a model module slot*

Description

depictSlot() is a convenience method that applies the depict method to a specified slot of a model module.

depictSlot method applied to Ready4Module

Usage

```
depictSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
depictSlot(x, slot_nm_1L_chr, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or no value (when called for side effects only).

Either a ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or no value (when called for side effects only).

---

enhance *Enhance a model module by adding new elements*

Description

enhance() is a method that adds new data fields (columns for tabular data, elements for arrays) and values to a model module by transforming it into a module of an inheriting class.

Usage

```
enhance(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method.

enhanceSlot *Apply the enhance method to a model module slot*

Description

enhanceSlot() is a convenience method that applies the enhance method to a specified slot a model module.

enhanceSlot method applied to Ready4Module

Usage

```
enhanceSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
enhanceSlot(x, slot_nm_1L_chr, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method.

A ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method.

exhibit *Exhibit features of model module data by printing them to the R console*

Description

exhibit() is a method that prints to console selected features of data contained in a model module.

Usage

```
exhibit(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method or no return value (when called for side-effects only).

`exhibitSlot` *Apply the exhibit method to a model module slot*

Description

exhibitSlot() is a convenience method that applies the exhibit method to a specified slot a model module.

exhibitSlot method applied to Ready4Module

Usage

```
exhibitSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
exhibitSlot(x, slot_nm_1L_chr, ...)
```

Arguments

`x`	An object of class Ready4Module
`slot_nm_1L_chr`	Slot name (a length one character vector)
`...`	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or no return value (when called purely for side effects).

Either a ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or no return value (when called purely for side effects).

get_from_lup_obj *Get a value from a lookup table*

Description

get_from_lup_obj() retrieves from a lookup table (a data.frame) the values in a target column for cases where values in a second column match a specified value.

Usage

```
get_from_lup_obj(
  data_lookup_tb,
  match_value_xx,
  match_var_nm_1L_chr,
  target_var_nm_1L_chr,
  evaluate_1L_lgl = FALSE
)
```

Arguments

data_lookup_tb	Data lookup (a tibble)
match_value_xx	Match value (an output object of multiple potential types)
match_var_nm_1L_chr	Match variable name (a character vector of length one)
target_var_nm_1L_chr	Target variable name (a character vector of length one)
evaluate_1L_lgl	Evaluate (a logical vector of length one), Default: FALSE

Value

Cell value (an output object of multiple potential types)

Examples

```
lookup_tb <- tibble::tibble(Name = c("Sajid","Siobhan"),
                            Treat = c("Cake", "Chocolate"))
get_from_lup_obj(lookup_tb, match_value_xx = "Siobhan",
                 match_var_nm_1L_chr = "Name", target_var_nm_1L_chr = "Treat")
get_from_lup_obj(lookup_tb, match_value_xx = "Cake",
                 match_var_nm_1L_chr = "Treat", target_var_nm_1L_chr = "Name")
```

get_libraries_tb *Get a table of ready4 libraries*

Description

get_libraries_tb() retrieves a tabular summary of ready4 libraries that have been developed within a specified GitHub organisation.

Usage

```
get_libraries_tb(
  gh_repo_1L_chr = "ready4-dev/ready4",
  gh_tag_1L_chr = "Documentation_0.0"
)
```

Arguments

gh_repo_1L_chr	Github repository (a character vector of length one), Default: 'ready4-dev/ready4'
gh_tag_1L_chr	Github tag (a character vector of length one), Default: 'Documentation_0.0'

Value

Libraries (a tibble)

Examples

```
get_libraries_tb("ready4-dev/ready4")
```

get_methods *Get the methods associated with a ready4 model module*

Description

get_methods() retrieves the ready4 methods that are available for a specified ready4 model module.

Usage

```
get_methods(pkg_nm_1L_chr = "ready4", cls_nm_1L_chr = "Ready4Module")
```

Arguments

pkg_nm_1L_chr	Package name (a character vector of length one), Default: 'ready4'
cls_nm_1L_chr	Class name (a character vector of length one), Default: 'Ready4Module'

Value

Methods (a character vector)

Examples

```
get_methods()
```

`get_methods_tb` *Get a table of methods associated with ready4 model modules*

Description

get_methods_tb() ingests 'methods_tb.RDS' (a table of methods associated with ready4 model modules) from a specified GitHub repository release.

Usage

```
get_methods_tb(
  gh_repo_1L_chr = "ready4-dev/ready4",
  gh_tag_1L_chr = "Documentation_0.0"
)
```

Arguments

`gh_repo_1L_chr`	Github repository (a character vector of length one), Default: 'ready4-dev/ready4'
`gh_tag_1L_chr`	Github tag (a character vector of length one), Default: 'Documentation_0.0'

Value

Methods (a tibble)

Examples

```
get_methods_tb("ready4-dev/ready4")
```

get_modules_tb *Get a table of ready4 model modules*

Description

get_modules_tb() ingests 'modules_tb.RDS' (a table of ready4 model modules) from a specified GitHub repository release.

Usage

```
get_modules_tb(
  gh_repo_1L_chr = "ready4-dev/ready4",
  gh_tag_1L_chr = "Documentation_0.0"
)
```

Arguments

gh_repo_1L_chr	Github repository (a character vector of length one), Default: 'ready4-dev/ready4'
gh_tag_1L_chr	Github tag (a character vector of length one), Default: 'Documentation_0.0'

Value

Modules (a tibble)

Examples

```
get_modules_tb("ready4-dev/ready4")
```

ingest *Ingest data*

Description

ingest() is a method that ingests data saved in external files into a model module or submodule.

Usage

```
ingest(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance).

`ingestSlot` *Apply the ingest method to a model module slot*

Description

ingestSlot() is a convenience method that applies the ingest method to a specified slot of a model module.

ingestSlot method applied to Ready4Module

Usage

```
ingestSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
ingestSlot(x, slot_nm_1L_chr, ...)
```

Arguments

`x`	An object of class Ready4Module
`slot_nm_1L_chr`	Slot name (a length one character vector)
`...`	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method.

A ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method.

`investigate` *Investigate solutions to an inverse problem*

Description

investigate() is a method that applies an algorithm to data contained in a model module in order to solve an inverse problem (ie, identify a statistical model that can generate approximations of that data).

Usage

```
investigate(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance).

---

investigateSlot *Apply the investigate method to a model module slot*

Description

investigateSlot() is a convenience method that applies the investigate method to a specified slot of a model module.

investigateSlot method applied to Ready4Module

Usage

```
investigateSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
investigateSlot(x, slot_nm_1L_chr, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module).

A ready4 model module (an instance of a class that inherits from Ready4Module).

make_code_releases_tbl *Make a tabular summary of release history of ready4 code libraries and executables*

Description

make_code_releases_tbl() scrapes the details of a specified GitHub repository to generate a release history of ready libraries and executables. To work all repositories without any release need to be supplied using the 'exclude_chr' argument.

Usage

```
make_code_releases_tbl(
  repo_type_1L_chr = c("Framework", "Module", "Package", "Program", "Subroutine",
    "Program_and_Subroutine"),
  as_kbl_1L_lgl = TRUE,
  brochure_repos_chr = character(0),
  exclude_chr = character(0),
  format_1L_chr = "%d-%b-%Y",
  framework_repos_chr = character(0),
  gh_repo_1L_chr = "ready4-dev/ready4",
  gh_tag_1L_chr = "Documentation_0.0",
  model_repos_chr = character(0),
  program_repos_chr = character(0),
  org_1L_chr = "ready4-dev",
  repos_chr = character(0),
  subroutine_repos_chr = character(0),
  tidy_desc_1L_lgl = TRUE,
  url_stub_1L_chr = "https://ready4-dev.github.io/",
  ...
)
```

Arguments

repo_type_1L_chr
: Repository type (a character vector of length one), Default: c("Framework", "Module", "Package", "Program", "Subroutine", "Program_and_Subroutine")

as_kbl_1L_lgl
: As kable (a logical vector of length one), Default: TRUE

brochure_repos_chr
: Brochure repositories (a character vector), Default: character(0)

exclude_chr
: Exclude (a character vector), Default: character(0)

format_1L_chr
: Format (a character vector of length one), Default: '%d-%b-%Y'

framework_repos_chr
: Framework repositories (a character vector), Default: character(0)

gh_repo_1L_chr
: Github repository (a character vector of length one), Default: 'ready4-dev/ready4'

gh_tag_1L_chr	Github tag (a character vector of length one), Default: 'Documentation_0.0'
model_repos_chr	Model repositories (a character vector), Default: character(0)
program_repos_chr	Program repositories (a character vector), Default: character(0)
org_1L_chr	Organisation (a character vector of length one), Default: 'ready4-dev'
repos_chr	Repositories (a character vector), Default: character(0)
subroutine_repos_chr	Subroutine repositories (a character vector), Default: character(0)
tidy_desc_1L_lgl	Tidy description (a logical vector of length one), Default: TRUE
url_stub_1L_chr	Url stub (a character vector of length one), Default: 'https://ready4-dev.github.io/'
...	Additional arguments

Value

Releases (an output object of multiple potential types)

Examples

```
# Likely to take more than one minute to execute.
  if(requireNamespace("tidyRSS", quietly = TRUE)) {
    make_code_releases_tbl("Framework",
                           gh_repo_1L_chr = "ready4-dev/ready4")
    make_code_releases_tbl("Module",
                           gh_repo_1L_chr = "ready4-dev/ready4")
    make_code_releases_tbl("Program",
                           gh_repo_1L_chr = "ready4-dev/ready4")
    make_code_releases_tbl("Subroutine",
                           gh_repo_1L_chr = "ready4-dev/ready4")
  }
```

---

make_datasets_tb *Make a tabular summary of ready4 model data collections*

Description

make_datasts_tb() function searches the contents of a specified Dataverse collection and returns a summary of the the data collections it contains.

Usage

```
make_datasets_tb(
  dv_nm_1L_chr = "ready4",
  dvs_tb = NULL,
  filter_cdns_ls = NULL,
  key_1L_chr = NULL,
  server_1L_chr = "dataverse.harvard.edu",
  toy_data_dv_1L_chr = "fakes",
  type_1L_chr = c("collections", "datasets"),
  what_1L_chr = "all"
)
```

Arguments

dv_nm_1L_chr	Dataverse name (a character vector of length one), Default: 'ready4'
dvs_tb	Dataverses (a tibble), Default: NULL
filter_cdns_ls	Filter conditions (a list), Default: NULL
key_1L_chr	Key (a character vector of length one), Default: NULL
server_1L_chr	Server (a character vector of length one), Default: 'dataverse.harvard.edu'
toy_data_dv_1L_chr	Toy data dataverse (a character vector of length one), Default: 'fakes'
type_1L_chr	Type (a character vector of length one), Default: c("collections", "datasets")
what_1L_chr	What (a character vector of length one), Default: 'all'

Value

Datasets (a tibble)

Examples

```
# Likely to take more than one minute to execute.
make_datasets_tb("ready4")
dvs_tb <- get_datasets_tb("ready4-dev/ready4")
make_datasets_tb("ready4", dvs_tb = dvs_tb)
make_datasets_tb("ready4", dvs_tb = dvs_tb, what_1L_chr = "real")
make_datasets_tb("ready4", dvs_tb = dvs_tb, what_1L_chr = "fakes")
make_datasets_tb("ready4", dvs_tb = dvs_tb, type_1L_chr = "datasets")
make_datasets_tb("ready4", dvs_tb = dvs_tb, type_1L_chr = "datasets", what_1L_chr = "real")
make_datasets_tb("ready4", dvs_tb = dvs_tb, type_1L_chr = "datasets", what_1L_chr = "fakes")
```

`make_ds_releases_tbl` *Make a tabular summary of release history of ready4 model data collections*

Description

make_datasts_tb() scrapes metadata from a specified Dataverse collection to create a summary table of its contents. The contents table can detail either subsidiary data collections or individual datasets from those subsidiary data collections.

Usage

```
make_ds_releases_tbl(
  ds_dois_chr,
  format_1L_chr = "%d-%b-%Y",
  key_1L_chr = NULL,
  server_1L_chr = "dataverse.harvard.edu",
  as_kbl_1L_lgl = TRUE,
  ...
)
```

Arguments

`ds_dois_chr`	Dataset digital object identifiers (a character vector)
`format_1L_chr`	Format (a character vector of length one), Default: '%d-%b-%Y'
`key_1L_chr`	Key (a character vector of length one), Default: NULL
`server_1L_chr`	Server (a character vector of length one), Default: 'dataverse.harvard.edu'
`as_kbl_1L_lgl`	As kable (a logical vector of length one), Default: TRUE
`...`	Additional arguments

Value

Dataset releases (an output object of multiple potential types)

Examples

```
make_ds_releases_tbl("10.7910/DVN/RIQTKK", as_kbl_1L_lgl = FALSE)
```

`make_methods_tb` *Make a tabular summary of methods associated with ready model modules*

Description

make_ds_releases_tbl() scrapes metadata from Dataverse datasets for which a valid Digital Object Identifier (DOI) has been supplied to create a table summarising the entire release history of these datasets.

Usage

```
make_methods_tb(
  packages_tb = NULL,
  exclude_mthds_for_chr = NA_character_,
  framework_only_1L_lgl = TRUE,
  gh_repo_1L_chr = "ready4-dev/ready4",
  gh_tag_1L_chr = "Documentation_0.0",
  module_pkgs_chr = character(0),
  ns_var_nm_1L_chr = "pt_ns_chr",
  path_1L_chr = character(0),
  return_1L_chr = "all"
)
```

Arguments

`packages_tb`	Packages (a tibble), Default: NULL
`exclude_mthds_for_chr`	Exclude methods for (a character vector), Default: 'NA'
`framework_only_1L_lgl`	Framework only (a logical vector of length one), Default: TRUE
`gh_repo_1L_chr`	Github repository (a character vector of length one), Default: 'ready4-dev/ready4'
`gh_tag_1L_chr`	Github tag (a character vector of length one), Default: 'Documentation_0.0'
`module_pkgs_chr`	Module packages (a character vector), Default: character(0)
`ns_var_nm_1L_chr`	Namespace variable name (a character vector of length one), Default: 'pt_ns_chr'
`path_1L_chr`	Path (a character vector of length one), Default: character(0)
`return_1L_chr`	Return (a character vector of length one), Default: 'all'

Value

Methods (a tibble)

Examples

```
# Likely to take more than one minute to execute.
make_methods_tb(gh_repo_1L_chr = "ready4-dev/ready4")
```

`make_modules_tb` *Make a tabular summary of ready4 model modules and sub-modules*

Description

make_methods_tb() scrapes the documentation websites of all libraries of ready4 modules in a specified GitHub organisation and then creates a tabular summary of vignette examples of ready4 module methods.

Usage

```
make_modules_tb(
  pkg_extensions_tb = NULL,
  cls_extensions_tb = NULL,
  gh_repo_1L_chr = "ready4-dev/ready4",
  gh_tag_1L_chr = "Documentation_0.0",
  module_pkgs_chr = character(0),
  include_1L_chr = "modules",
  ns_var_nm_1L_chr = "pt_ns_chr",
  url_stub_1L_chr = "https://ready4-dev.github.io/",
  what_chr = "all"
)
```

Arguments

`pkg_extensions_tb`	Package extensions (a tibble), Default: NULL
`cls_extensions_tb`	Class extensions (a tibble), Default: NULL
`gh_repo_1L_chr`	Github repository (a character vector of length one), Default: 'ready4-dev/ready4'
`gh_tag_1L_chr`	Github tag (a character vector of length one), Default: 'Documentation_0.0'
`module_pkgs_chr`	Module packages (a character vector), Default: character(0)
`include_1L_chr`	Include (a character vector of length one), Default: 'modules'
`ns_var_nm_1L_chr`	Namespace variable name (a character vector of length one), Default: 'pt_ns_chr'
`url_stub_1L_chr`	Url stub (a character vector of length one), Default: 'https://ready4-dev.github.io/'
`what_chr`	What (a character vector), Default: 'all'

Value

Modules (a tibble)

Examples

```
# Likely to take more than one minute to execute.
make_modules_tb(gh_repo_1L_chr = "ready4-dev/ready4")
```

`make_programs_tbl` *Make a tabular summary of programs using ready4 model modules*

Description

make_modules_tb() scrapes the documentation websites of all libraries of ready4 modules in a specified GitHub organisation and then creates a tabular summary of the modules included in those libraries and vignette examples of their use.

Usage

```
make_programs_tbl(
  what_1L_chr = c("Program", "Subroutine", "Program_and_Subroutine"),
  as_kbl_1L_lgl = FALSE,
  exclude_chr = character(0),
  format_1L_chr = "%d-%b-%Y",
  gh_repo_1L_chr = "ready4-dev/ready4",
  gh_tag_1L_chr = "Documentation_0.0",
  tidy_desc_1L_lgl = TRUE,
  url_stub_1L_chr = "https://ready4-dev.github.io/",
  zenodo_1L_chr = "ready4",
  ...
)
```

Arguments

`what_1L_chr`	What (a character vector of length one), Default: c("Program", "Subroutine", "Program_and_Subroutine")
`as_kbl_1L_lgl`	As kable (a logical vector of length one), Default: FALSE
`exclude_chr`	Exclude (a character vector), Default: character(0)
`format_1L_chr`	Format (a character vector of length one), Default: '%d-%b-%Y'
`gh_repo_1L_chr`	Github repository (a character vector of length one), Default: 'ready4-dev/ready4'
`gh_tag_1L_chr`	Github tag (a character vector of length one), Default: 'Documentation_0.0'
`tidy_desc_1L_lgl`	Tidy description (a logical vector of length one), Default: TRUE

url_stub_1L_chr
: Url stub (a character vector of length one), Default: 'https://ready4-dev.github.io/'

zenodo_1L_chr
: Zenodo (a character vector of length one), Default: 'ready4'

...
: Additional arguments

Value

Programs (an output object of multiple potential types)

See Also

zen4R::ZenodoManager()

Examples

```
# Likely to take more than one minute to execute.
if(requireNamespace("zen4R", quietly = TRUE)) {
  make_programs_tbl("Program",
                    gh_repo_1L_chr = "ready4-dev/ready4")
  make_programs_tbl("Subroutine",
                    gh_repo_1L_chr = "ready4-dev/ready4")
}
```

---

manufacture *Manufacture a new object*

Description

manufacture() is a method that used data contained in a model module or submodule to create a new object (other than a model module).

Usage

```
manufacture(x, ...)
```

Arguments

x
: A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)

...
: Additional arguments

Value

An object other than a model module (an instance of a class that inherits from Ready4Module).

manufactureSlot *Apply the manufacture method to a model module slot*

Description

manufactureSlot() is a convenience method that applies the manufacture method to a specified slot of a model module.

manufactureSlot method applied to Ready4Module

Usage

```
manufactureSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
manufactureSlot(x, slot_nm_1L_chr, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments

Value

An object that is not the the same class as that supplied to the method.

An object that is not the the same class as that supplied to the method.

metamorphose *Metamorphose a model module to a model module of a different (non-inheriting) class*

Description

metamorphose() is a method that transforms a model module into a model module of a different (non-inheriting) class.

Usage

```
metamorphose(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) of a different class to that supplied to the method.

`metamorphoseSlot` *Apply the metamorphose method to a model module slot*

Description

metamorphoseSlot() is a convenience method that applies the metamorphose method to a specified slot of a model module.

metamorphoseSlot method applied to Ready4Module

Usage

```
metamorphoseSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
metamorphoseSlot(x, slot_nm_1L_chr, ...)
```

Arguments

`x`	An object of class Ready4Module
`slot_nm_1L_chr`	Slot name (a length one character vector)
`...`	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module).

A ready4 model module (an instance of a class that inherits from Ready4Module).

`print_data` *Print a table of ready4 model data collections*

Description

make_programs_tbl() scrapes the GitHub organisation and Zenodo community associated specified for a ready4 model implementation to create a tabular summary of programs and sub-routines associated with that implementation.

Usage

```
print_data(
  datasets_tb,
  by_dv_1L_lgl = FALSE,
  filter_cdns_ls = NULL,
  root_1L_chr = "https://dataverse.harvard.edu/dataverse/",
  scroll_height_1L_chr = character(0),
  scroll_width_1L_chr = character(0),
  toy_data_dv_1L_chr = "fakes",
  what_1L_chr = "all",
  ...
)
```

Arguments

datasets_tb	Datasets (a tibble)
by_dv_1L_lgl	By dataverse (a logical vector of length one), Default: FALSE
filter_cdns_ls	Filter conditions (a list), Default: NULL
root_1L_chr	Root (a character vector of length one), Default: 'https://dataverse.harvard.edu/dataverse/'
scroll_height_1L_chr	Scroll height (a character vector of length one), Default: character(0)
scroll_width_1L_chr	Scroll width (a character vector of length one), Default: character(0)
toy_data_dv_1L_chr	Toy data dataverse (a character vector of length one), Default: 'fakes'
what_1L_chr	What (a character vector of length one), Default: 'all'
...	Additional arguments

Value

Datasets (a kable)

Examples

```
datasets_tb <- get_datasets_tb("ready4-dev/ready4")
print_data(datasets_tb, by_dv_1L_lgl = TRUE)
print_data(datasets_tb, what_1L_chr = "real")
print_data(datasets_tb, what_1L_chr = "fakes")
```

print_methods *Print a table of methods associated with ready4 model modules*

Description

print_data() formats the output of either get_datasts_tb() or make_datasts_tb() as HTML. The type of output can be customised to display Dataverse data collections or Dataverse datasets. Similarly output can be restricted to real or toy datasets.

Usage

```
print_methods(
  methods_tb = NULL,
  exclude_mthds_for_chr = NA_character_,
  gh_repo_1L_chr = "ready4-dev/ready4",
  gh_tag_1L_chr = "Documentation_0.0",
  methods_chr = NULL,
  module_pkgs_chr = character(0),
  ns_var_nm_1L_chr = "pt_ns_chr",
  path_1L_chr = character(0),
  packages_tb = NULL,
  return_1L_chr = "all",
  scroll_height_1L_chr = character(0),
  scroll_width_1L_chr = character(0),
  ...
)
```

Arguments

methods_tb	Methods (a tibble), Default: NULL
exclude_mthds_for_chr	Exclude methods for (a character vector), Default: 'NA'
gh_repo_1L_chr	Github repository (a character vector of length one), Default: 'ready4-dev/ready4'
gh_tag_1L_chr	Github tag (a character vector of length one), Default: 'Documentation_0.0'
methods_chr	Methods (a character vector), Default: NULL
module_pkgs_chr	Module packages (a character vector), Default: character(0)
ns_var_nm_1L_chr	Namespace variable name (a character vector of length one), Default: 'pt_ns_chr'
path_1L_chr	Path (a character vector of length one), Default: character(0)
packages_tb	Packages (a tibble), Default: NULL
return_1L_chr	Return (a character vector of length one), Default: 'all'
scroll_height_1L_chr	Scroll height (a character vector of length one), Default: character(0)

scroll_width_1L_chr
: Scroll width (a character vector of length one), Default: character(0)

...
: Additional arguments

Value

Methods (a kable)

Examples

```
methods_tb <- get_methods_tb("ready4-dev/ready4")
print_methods(methods_tb)
print_methods(methods_tb, return_1L_chr = "core")
print_methods(methods_tb, return_1L_chr = "slot")
print_methods(methods_tb, return_1L_chr = "extended")
```

---

print_modules *Print a table of ready4 model modules*

Description

print_methods() formats the output of either get_methods_tb() or make_methods_tb() as HTML.

Usage

```
print_modules(
  modules_tb,
  scroll_height_1L_chr = character(0),
  scroll_width_1L_chr = character(0),
  what_1L_chr = "All",
  ...
)
```

Arguments

modules_tb
: Modules (a tibble)

scroll_height_1L_chr
: Scroll height (a character vector of length one), Default: character(0)

scroll_width_1L_chr
: Scroll width (a character vector of length one), Default: character(0)

what_1L_chr
: What (a character vector of length one), Default: 'All'

...
: Additional arguments

Value

Modules (a kable)

Examples

```
modules_tb <- get_modules_tb("ready4-dev/ready4")
# Print sub-modules
print_modules(modules_tb, what_1L_chr = "S3")
# Print full-modules
print_modules(modules_tb, what_1L_chr = "S4")
```

`print_packages` *Print a table of ready4 libraries*

Description

print_modules() formats the output of either get_modules_tb() or make_modules_tb() as HTML.

Usage

```
print_packages(
  pkg_extensions_tb = NULL,
  gh_repo_1L_chr = "ready4-dev/ready4",
  gh_tag_1L_chr = "Documentation_0.0",
  include_1L_chr = "modules",
  module_pkgs_chr = character(0),
  ns_var_nm_1L_chr = "pt_ns_chr",
  project_badges_url_1L_chr = "https://img.shields.io/badge/ready4",
  reference_var_nm_1L_chr = "Reference",
  scroll_height_1L_chr = character(0),
  scroll_width_1L_chr = character(0),
  url_stub_1L_chr = "https://ready4-dev.github.io/",
  vignette_var_nm_1L_chr = "Vignettes",
  vignette_url_var_nm_1L_chr = "Vignettes_URLs",
  what_chr = "all",
  ...
)
```

Arguments

`pkg_extensions_tb`	Package extensions (a tibble), Default: NULL
`gh_repo_1L_chr`	Github repository (a character vector of length one), Default: 'ready4-dev/ready4'
`gh_tag_1L_chr`	Github tag (a character vector of length one), Default: 'Documentation_0.0'
`include_1L_chr`	Include (a character vector of length one), Default: 'modules'
`module_pkgs_chr`	Module packages (a character vector), Default: character(0)
`ns_var_nm_1L_chr`	Namespace variable name (a character vector of length one), Default: 'pt_ns_chr'

project_badges_url_1L_chr
: Project badges url (a character vector of length one), Default: 'https://img.shields.io/badge/ready4'

reference_var_nm_1L_chr
: Reference variable name (a character vector of length one), Default: 'Reference'

scroll_height_1L_chr
: Scroll height (a character vector of length one), Default: character(0)

scroll_width_1L_chr
: Scroll width (a character vector of length one), Default: character(0)

url_stub_1L_chr
: Url stub (a character vector of length one), Default: 'https://ready4-dev.github.io/'

vignette_var_nm_1L_chr
: Vignette variable name (a character vector of length one), Default: 'Vignettes'

vignette_url_var_nm_1L_chr
: Vignette url variable name (a character vector of length one), Default: 'Vignettes_URLs'

what_chr
: What (a character vector), Default: 'all'

...
: Additional arguments

Value

Package extensions (a kable)

Examples

```
# Method 1
libraries_tb <- get_libraries_tb(gh_repo_1L_chr = "ready4-dev/ready4")
## Print framework libraries
update_libraries_tb(libraries_tb,
                    url_stub_1L_chr = "https://ready4-dev.github.io/",
                    include_1L_chr = "framework") %>%
  print_packages()
## Print module libraries
update_libraries_tb(libraries_tb,
                    url_stub_1L_chr = "https://ready4-dev.github.io/",
                    include_1L_chr = "modules") %>%
  print_packages()
# Method 2
## Print framework libraries
print_packages(gh_repo_1L_chr = "ready4-dev/ready4",
               include_1L_chr = "framework")
## Print module libraries
print_packages(gh_repo_1L_chr = "ready4-dev/ready4",
               include_1L_chr = "modules")
```

procure *Procure items from a dataset*

Description

procure() is a "getter" method that retrieves data contained within a model module or sub-module.

Usage

```
procure(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

An object of the same class as that supplied to the method or of one of the same classes that constitute the input object's slots or elements.

procureSlot *Procure (get) data from a slot*

Description

procureSlot() is a "getter" method that, depending on input arguments, retrieves either data contained in a specified model module slot (the default behaviour) or the value returned by applying the procure method to the specified slot.

procureSlot method applied to Ready4Module

Usage

```
procureSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
procureSlot(x, slot_nm_1L_chr, use_procure_mthd_1L_lgl = FALSE, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments
use_procure_mthd_1L_lgl	Use procure method (a length one logical vector)

Value

A model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or an instance of a class contained in that Ready4Module's slots.

A ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or an instance of a class contained in that Ready4Module's slots.

Examples

```
X <- Ready4Module()
procureSlot(X, "dissemination_1L_chr")
```

---

prognosticate *Prognosticate (make predictions) by solving a forward problem*

Description

prognosticate() is a method that applies an algorithm to data contained in a model module to solve a forward problem (i.e., use simulation and statistical methods to make predictions).

Usage

```
prognosticate(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module).

---

prognosticateSlot *Apply the prognosticate method to a model module slot*

Description

prognosticateSlot() is a convenience method that applies the prognosticate method to a specified slot of a model module.

prognosticateSlot method applied to Ready4Module

Usage

```
prognosticateSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
prognosticateSlot(x, slot_nm_1L_chr, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module).

A ready4 model module (an instance of a class that inherits from Ready4Module).

---

ratify *Ratify that input or output data meet validity criteria*

Description

ratify() is a method that validates that a model module or submodule conforms to specified internal consistency criteria, potentially updating the invalid values in the model module so that these criteria are met.

Usage

```
ratify(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method.

ratifySlot *Apply the ratify method to a model module slot*

Description

ratifySlot() is a convenience method that applies the ratify method to a specified slot of a model module.

ratifySlot method applied to Ready4Module

Usage

```
ratifySlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
ratifySlot(x, slot_nm_1L_chr, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method.

A ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method.

Ready4Module-class *Ready4Module*

Description

A module of the ready4 representational system.

Slots

dissemination_1L_chr	Dissemination (a character vector of length one)

Ready4Private-class *Ready4Private*

Description

A module of the ready4 representational system that contains data not intended for public dissemination.

Slots

`dissemination_1L_chr` Dissemination (a character vector of length one)

Ready4Public-class *Ready4Public*

Description

A virtual class denoting a module of the ready4 representational system hat is suitable for public dissemination in its current form.

Slots

`dissemination_1L_chr` Dissemination (a character vector of length one)

reckon *Reckon (calculate) a value*

Description

reckon() is a method that performs a calculation using data contained in a model module (or submodule).

Usage

```
reckon(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A numeric class.

reckonSlot *Apply the reckon method to a model module slot*

Description

reckonSlot() is a convenience method that applies the reckon method to a specified slot of a model module.

reckonSlot method applied to Ready4Module

Usage

```
reckonSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
reckonSlot(x, slot_nm_1L_chr, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments

Value

A numeric class.

A numeric class.

renew *Renew (update) values*

Description

renew() is a "setter" method that updates values of selected data contained in a model module or sub-module.

Usage

```
renew(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method.

`renewSlot` *Apply the renew method to a model module slot*

Description

renewSlot() is a "setter" method that renews (sets) the value of a specified model module slot with either the value returned by applying the renew method to that slot (the default behaviour) or a supplied new value.

renewSlot method applied to Ready4Module

Usage

```
renewSlot(x, slot_nm_1L_chr, new_val_xx = "use_renew_mthd", ...)

## S4 method for signature 'Ready4Module'
renewSlot(x, slot_nm_1L_chr, new_val_xx = "use_renew_mthd", ...)
```

Arguments

`x`	An object of class Ready4Module
`slot_nm_1L_chr`	Slot name (a length one character vector)
`new_val_xx`	New value (slot dependent object type), Default 'use_renew_mthd'
`...`	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method.

A ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method.

Examples

```
X <- Ready4Module()
X <- renewSlot(X, "dissemination_1L_chr", new_val_xx = "Some new text.")
```

share *Share data via an online repository*

Description

share() is a method that uploads data contained in a model module to an online repository. If requested, the method will also publish the updated repository.

Usage

```
share(x, ...)
```

Arguments

x	A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance)
...	Additional arguments

Value

A model module (an instance of a class that inherits from Ready4Module) or submodule (any S3 class instance) of the same class as that supplied to the method or no return value (when called for side-effects only).

shareSlot *Apply the share method to a model module slot*

Description

shareSlot() is a convenience method that applies the share method to a specified slot of a model module.

shareSlot method applied to Ready4Module

Usage

```
shareSlot(x, slot_nm_1L_chr, ...)

## S4 method for signature 'Ready4Module'
shareSlot(x, slot_nm_1L_chr, ...)
```

Arguments

x	An object of class Ready4Module
slot_nm_1L_chr	Slot name (a length one character vector)
...	Additional arguments

Value

Either a model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or no value (when called purely for side effects).

Either a ready4 model module (an instance of a class that inherits from Ready4Module) of the same class as that supplied to the method or no value (when called purely for side effects).

---

write_to_copy_rmds *Write a local copy of RMD or Rmarkdown files*

Description

print_packages() formats the output of get_libraries_tb() as HTML.

Usage

```
write_to_copy_rmds(
  dir_path_1L_chr,
  fl_nm_1L_chr,
  consent_1L_chr = "",
  rmds_dir_1L_chr = "R/RMD Templates",
  consent_indcs_int = 1L,
  options_chr = c("Y", "N"),
  return_1L_lgl = FALSE
)
```

Arguments

dir_path_1L_chr	Directory path (a character vector of length one)
fl_nm_1L_chr	File name (a character vector of length one)
consent_1L_chr	Consent (a character vector of length one), Default: ”
rmds_dir_1L_chr	R Markdowns directory (a character vector of length one), Default: 'R/RMD Templates'
consent_indcs_int	Consent indices (an integer vector), Default: 1
options_chr	Options (a character vector), Default: c("Y", "N")
return_1L_lgl	Return (a logical vector of length one), Default: FALSE

Value

No return value, called for side effects.

Examples

```
write_to_copy_rmds(dir_path_1L_chr = tempdir(),
                   fl_nm_1L_chr = "RMDs",
                   rmds_dir_1L_chr = system.file("MD_RMDs",
                                                 package = "ready4"))
```

`write_to_render_post` *Write ready4 model documentation website page from an RMD or Rmarkdown file*

Description

write_to_copy_rmds() is used to copy template RMD or Rmarkdown files to specified sub-directories of a model documentation website. These template copies can then be manually edited before being rendered with write_to_render_post().

Usage

```
write_to_render_post(
  included_dirs_chr,
  path_to_main_dir_1L_chr,
  consent_1L_chr = "",
  consent_indcs_int = 1L,
  is_rmd_1L_lgl = TRUE,
  options_chr = c("Y", "N")
)
```

Arguments

`included_dirs_chr`	Included directories (a character vector)
`path_to_main_dir_1L_chr`	Path to main directory (a character vector of length one)
`consent_1L_chr`	Consent (a character vector of length one), Default: ”
`consent_indcs_int`	Consent indices (an integer vector), Default: 1
`is_rmd_1L_lgl`	Is Markdown (a logical vector of length one), Default: TRUE
`options_chr`	Options (a character vector), Default: c("Y", "N")

Value

No return value, called for side effects.

See Also

rmarkdown::render()

Examples

```
# Note, In addition to rmarkdown, the non CRAN package "hugodown" is also required.
if(requireNamespace("rmarkdown", quietly = TRUE)) {
# Example 1 - RMD files
#
# Copy template RMD files
write_to_copy_rmds(dir_path_1L_chr = tempdir(),
                   fl_nm_1L_chr = "RMDs",
                   rmds_dir_1L_chr = system.file("MD_RMDs",
                                                 package = "ready4"))
# Typically you would now edit these templates before proceeding.
# Render post from RMD files.
write_to_render_post("RMDs", path_to_main_dir_1L_chr = tempdir())
#
# Example 2 - Rmarkdown file
#
# Copy template Rmarkdown file
write_to_copy_rmds(dir_path_1L_chr = tempdir(),
                   fl_nm_1L_chr = "Rmarkdown",
                   rmds_dir_1L_chr = system.file("MD_Rmarkdown",
                                                 package = "ready4"))
# Typically you would now edit these templates before proceeding.
# Render post from RMD files.
write_to_render_post("Rmarkdown",
                     path_to_main_dir_1L_chr = tempdir(),
                     is_rmd_1L_lgl = F)
}
```

`write_ws` *Write ready4 software develoment local directories*

Description

write_to_render_post() is designed for help overcome practical challenges of rendering RMD or Rmarkdown files to Markdown output in a modelling project's Hugo Docsy documentation website. You must have 'hugodown' installed for this function to work.

Usage

```
write_ws(
  path_1L_chr,
  consent_1L_chr = "",
  consent_indcs_int = 1L,
```

```
  options_chr = c("Y", "N")
)
```

Arguments

path_1L_chr	Path (a character vector of length one)
consent_1L_chr	Consent (a character vector of length one), Default: ”
consent_indcs_int	Consent indices (an integer vector), Default: 1
options_chr	Options (a character vector), Default: c("Y", "N")

Value

No return value, called for side effects.

Examples

```
write_ws(tempdir())
```

Index