

Package ‘tidylab’

November 8, 2021

Type Package

Title Shared Infrastructure of 'Tidylab' Package Collection

Version 4.1.1

Date 2021-10-10

Maintainer Harel Lustiger <tidylab@gmail.com>

Description Selection of packages designed to be loaded in concert. The 'tidylab' package collection revolves around project development and deployment. The purpose of this is to make it easy to install and subsequently load those packages.

License MIT + file LICENSE

URL <https://tidylab.github.io/tidylab/>,
<https://github.com/tidylab/tidylab>

BugReports <https://github.com/tidylab/tidylab/issues>

Depends R (>= 3.3)

Imports cli (>= 2.4.0), crayon (>= 1.4.1), decorators, dplyr (>= 1.0.5), microservices, purrr (>= 0.3.4), rstudioapi (>= 0.13), tibble (>= 3.1.0), usethat

Suggests testthat (>= 3.0.0), withr

Encoding UTF-8

Language en-GB

LazyData true

RoxygenNote 7.1.1

NeedsCompilation no

Author Harel Lustiger [aut, cre] (<<https://orcid.org/0000-0003-2953-9598>>),
Tidylab [cph, fnd]

Repository CRAN

Date/Publication 2021-11-08 12:10:02 UTC

R topics documented:

tidylab_conflicts	2
tidylab_deps	3
tidylab_logo	3
tidylab_packages	4
tidylab_update	4

Index	5
--------------	----------

tidylab_conflicts	<i>Conflicts between the tidylab and other packages</i>
-------------------	---

Description

This function lists all the conflicts between packages in the tidylab and other packages that you have loaded.

This function lists all the conflicts between packages in the tidylab and other packages that you have loaded.

Usage

```
tidylab_conflicts()
```

```
tidylab_conflicts()
```

Details

There are four conflicts that are deliberately ignored: `intersect`, `union`, `setequal`, and `setdiff` from `dplyr`. These functions make the base equivalents generic, so shouldn't negatively affect any existing code.

If `dplyr` is one of the select packages, then the following four conflicts are deliberately ignored: `intersect`, `union`, `setequal`, and `setdiff` from `dplyr`. These functions make the base equivalents generic, so shouldn't negatively affect any existing code.

Value

(tidylab_conflicts) A list with conflicted functions.

Examples

```
tidylab_conflicts()
tidylab_conflicts()
```

tidylab_deps	<i>List all tidylab dependencies</i>
--------------	--------------------------------------

Description

List all tidylab dependencies

Usage

```
tidylab_deps(recursive = FALSE)
```

Arguments

`recursive` If TRUE, will also list all dependencies of tidylab packages.

Value

(tibble) A table with package dependencies.

tidylab_logo	<i>The tidylab logo, using ASCII or Unicode characters</i>
--------------	--

Description

Use `crayon::strip_style()` to get rid of the colors.

Usage

```
tidylab_logo(unicode = 110n_info()$`UTF-8`)
```

Arguments

`unicode` Whether to use Unicode symbols. Default is TRUE on UTF-8 platforms.

Value

(tidylab_logo) ASCII art of the Tidylab logo.

Examples

```
tidylab_logo()
```

tidylab_packages	<i>List all packages in the tidylab</i>
------------------	---

Description

List all packages in the tidylab

Usage

```
tidylab_packages(include_self = TRUE)
```

Arguments

include_self Include tidylab in the list?

Value

(character) The names of the imported packages.

Examples

```
tidylab_packages()
```

tidylab_update	<i>Update tidylab packages</i>
----------------	--------------------------------

Description

This will check to see if all tidylab packages (and optionally, their dependencies) are up-to-date, and will install after an interactive confirmation.

Usage

```
tidylab_update(recursive = FALSE)
```

Arguments

recursive If TRUE, will also check all dependencies of tidylab packages.

Value

No return value, called for side effects.

Examples

```
## Not run:  
tidylab_update()  
  
## End(Not run)
```

Index

`crayon::strip_style()`, 3

`tidylab_conflicts`, 2

`tidylab_deps`, 3

`tidylab_logo`, 3

`tidylab_packages`, 4

`tidylab_update`, 4