Package 'assignmentr'

November 16, 2019

Title What the Package Does (One Line, Title Case)

Version 0.0.0.9000

Description This packages satisfies the requirement of creating an R package for the Coursera course ``Building R Packages", one of the courses in the ``Mastering Software Development in R" specialization.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.0.0

R topics documented:

`fars_map_state` *Creates a geographical plot showing the accidents for a given year in a given state. Each accident is plotted as a single dot.*

Description

Creates a geographical plot showing the accidents for a given year in a given state. Each accident is plotted as a single dot.

Usage

```
fars_map_state(state.num, year)
```

Arguments

`state.num`	The state number which you would like displayed.
`year`	The year that you would like in the filename.

Value

Displays a plot (no object returned).

Note

If the state.num argument doesn't correspond to a state within the specified data frame's STATE column, and error will be thrown.

If there are no accidents to plot for the specified state.num/year combination a message will display saying so.

Examples

```
fars_map_state(42, 2015)
# plot of state 42 pops up with accidents plotted as dots
```

---

fars_read *Read a CSV file into a tibble (tidyverse dataframe)*

Description

This function reads a CSV file into R as a tibble.

Usage

```
fars_read(filename)
```

Arguments

`filename`	The aboslute or relative path to the filename to read into R (should be a CSV file) .

Value

The CSV data in tibble (tidyverse version of data frames) format.

Note

If the file doesn't exist, a message will be displayed saying so.

Examples

```
df <- fars_read("~/data/jason_bournes_data.csv")
df <- fars_read("C:\\Documents\\jack_bauers_data.csv")
```

`fars_read_years` *Collects the MONTH and year associated with a fatal car incident for each year provided.*

Description

Collects the MONTH and year associated with a fatal car incident for each year provided.

Usage

```
fars_read_years(years)
```

Arguments

`years`	A vector of years which have an associated file in the current working directory.

Value

A list of data frames, each with two columns; MONTH and year. Each row in a data frame represent a fatal car accident in that month and year.

Note

If one of the years specified doesn't have an associated file, and error will be thrown.

Examples

```
years <- fars_read_years(c(2013, 2014, 2015))
print(class(years))
[1] "list"

print(years)
[[1]]
# A tibble: 30,202 x 2
   MONTH  year
   <dbl> <dbl>
 1     1  2013
 2     1  2013
 3     1  2013
 4     1  2013
 5     1  2013
 6     1  2013
 7     1  2013
 8     1  2013
 9     1  2013
10     1  2013
# . . .  with 30,192 more rows

[[2]]
# A tibble: 30,056 x 2
   MONTH  year
   <dbl> <dbl>
 1     1  2014
 2     1  2014
```

```
 3     1  2014
 4     1  2014
 5     1  2014
 6     1  2014
 7     1  2014
 8     1  2014
 9     1  2014
10     1  2014
# . . .  with 30,046 more rows

[[3]]
# A tibble: 32,166 x 2
   MONTH  year
   <dbl> <dbl>
 1     1  2015
 2     1  2015
 3     1  2015
 4     1  2015
 5     1  2015
 6     1  2015
 7     1  2015
 8     1  2015
 9     1  2015
10     1  2015
# . . .  with 32,156 more rows
```

`fars_summarize_years` *Creates a table summarizing the count of fatal car incidents in a given month and year combination.*

Description

Creates a table summarizing the count of fatal car incidents in a given month and year combination.

Usage

```
fars_summarize_years(years)
```

Arguments

`years`	a vector of years for which the number of fatal car incidents should be displayed, grouped by Month.

Value

a data frame.

Note

If one of the years specified doesn't have an associated file, and error will be thrown.

Examples

```
summary_df <- fars_summarize_years(c(2013, 2014, 2015))
# A tibble: 12 x 4
   MONTH `2013` `2014` `2015`
   <dbl>  <int>  <int>  <int>
 1     1   2230   2168   2368
 2     2   1952   1893   1968
 3     3   2356   2245   2385
 4     4   2300   2308   2430
 5     5   2532   2596   2847
 6     6   2692   2583   2765
 7     7   2660   2696   2998
 8     8   2899   2800   3016
 9     9   2741   2618   2865
10    10   2768   2831   3019
11    11   2615   2714   2724
12    12   2457   2604   2781
```

---

make_filename *Creates a descriptive filename based on the input year.*

Description

Creates a descriptive filename based on the input year.

Usage

```
make_filename(year)
```

Arguments

year	The year that you would like in the filename.

Value

A character object that represents the filename for car fatality data.

Examples

```
filename_2018 <- make_filename(2018)
print(filename_2018)
[1] "accident_2018.csv.bz2"
```

Index