Package 'mypackage'

January 29, 2024

Title Demo Package as an Example

Version 0.0.1.0000

Description This package is used as a demo for a simple package for the course MATH 3190 at Southern Utah University. It contains functions on adding, subtracting, multiplying, and dividing, as well as graphing a simple scatterplot.

License MIT + file LICENSE

URL https://github.com/rbrown53/mypackage

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Depends R (>= 2.10),
ggplot2,
shiny

Imports magrittr,
tidyverse,
plotly

LazyData true

Suggests knitr,
rmarkdown

Author Rick Brown [aut, cre]

Maintainer Rick Brown <richardbrown1@suu.edu>

VignetteBuilder knitr

R topics documented:

add 2
brainbody 2
cranes 3
divide 3
ggraph 4
hello 4
multiply 5
runCor 5
subtract 6

Index 7

add *This is my addition function*

Description

This is my addition function

Usage

```
add(x, y)
```

Arguments

x	this is the first value to add
y	this is the second value to add

Value

This function returns the sum of x and y

Examples

```
## Start with something simple
add(1,1)

## Now something more difficult
add(49,60)
```

brainbody *Read in brainbody.txt and create a brainbody object*

Description

This function reads in the brainbody.txt file and creates a brainbody object.

This data set contains information on brain weight, body weight, gestation length, and two additional variables for a sample of subjects.

Usage

```
brainbody()

brainbody
```

Format

A data frame with 5 variables: `species`, `brain` (grams), `body` (kilograms), `gestation` (days), and `litter`.

Examples

```
brainbody <- read_brainbody()
# Returns: A data frame containing brain and body weights
```

---

cranes *Cranes Data Set*

Description

This data set contains information on the number of cranes at Aransas National Wildlife Refuge in Austwell, Texas by year from 1938 to 2016.

Usage

```
cranes
```

Format

A data frame with 2 variables: `cranes` and `year`.

---

divide *Divide two numbers with a check for division by zero*

Description

This function takes two numbers and divides the first number by the second number. If the second number is 0, it outputs an error message.

Usage

```
divide(x, y)
```

Arguments

`x`	The numerator
`y`	The denominator

Value

The result of x divided by y

Examples

```
divide(6, 2)
# Returns: 3

divide(4, 0)
# Outputs an error message: "You cannot divide by 0."
```

ggraph *Create a quick scatter plot in ggplot.*

Description

This will graph two given vectors in a ggplot-style scatter plot with the x-axis labeled "x" and the y-axis labeled "y".

Usage

```
ggraph(x, y, point_color = "black", point_size = 1.5, point_shape = 19)
```

Arguments

x	This is the first vector to be plotted.
y	This is the first vector to be plotted.
point_color	This is the color of the points that will be plotted.
point_size	This is the size of the points that will be plotted. The default is size 1.5.
point_shape	This is the shape of the points that will be plotted. The default is 19: a filled circle.

Value

This function returns a ggplot scatter plot object.

Examples

```
## Create a scatter plot of y vs x.
x <- rnorm(100)
y <- x + rnorm(100, 0, 0.3)
ggraph(x, y)
```

hello *This is my hello function. There are no parameters.*

Description

This is my hello function. There are no parameters.

Usage

```
hello()
```

Value

This function returns the message "hello world".

Examples

```
## This is the only thing this function does.
hello()
```

`multiply` *Multiply two numbers*

Description

This function takes two numbers and returns their product.

Usage

```
multiply(x, y)
```

Arguments

x	The first number
y	The second number

Value

The product of x and y

Examples

```
multiply(2, 3)
# Returns: 6
```

`runCor` *Correlation App*

Description

This function allows the correlation shiny app to run. The app is a little game where you are presented with a graph and you guess the correlation between the two variables. The true correlation will then be shown and the difference between your guess and the true correlation will be given

Usage

```
runCor()
```

subtract *This is my subtract function*

Description

This is my subtract function

Usage

```
subtract(x, y)
```

Arguments

x	this is the first value
y	this is the second value to subtract

Value

This function returns the difference of x and y

Examples

```
## Start with something simple
subtract(1, 1)

## Now something more difficult
subtract(49, 60)
```

Index

∗ **datasets**
brainbody, 2
cranes, 3

add, 2

brainbody, 2

cranes, 3

divide, 3

ggraph, 4

hello, 4

multiply, 5

runCor, 5

subtract, 6