

SHINYMAKER DOCUMENTATION

Friday August 26, 2016

This document was written by David Ruvolo.

Introduction

The `shiny__maker.R` script is a standard builder for creating Shiny apps. Generating and setting up initial shiny files (i.e., `server`, `ui`, `global`, `css`, `external`, `assets`, etc.) can be tedious and take a while to setup. In the first instance, this script will generate `navbarPages` with default starter text.

This function generates a shiny `navbarPage` and all necessary files. The app structure looks like this:

```
## directory structure
ShinyApp/
  --- global.R
  --- ui.R
  --- server.R
  --- data/
  --- external/
      | --- main.R
  --- utilities/
      | --- google-analytics.js
  --- www/
      | --- style.css
```

Usage

The `shiny__maker.R` file has 3 functions built in.

1. `%+=%`

The function `%+=%` increments the left argument by a value on the right. This was implemented from this [stackoverflow post](#).

```
`%+=%` = function(e1,e2) eval.parent(substitute(e1 <- e1 + e2))
x = 10
x %+=% 10
x
```

```
## [1] 20
```

2. navbarMaker()

The navbarMaker() is the first shiny app maker in the shinyMaker series.

Parameters

The function navbarMaker() takes the follow parameters:

- directory_name: The name of the shiny app directory.
- dir_path: The path for where you want the app stored. Default is desktop.

Examples

```
# source file
source("shiny_maker.R")

# basic usage
navbarMaker()

# specifying app name
navbarMaker(directory_name = "Project Analysis Tool")

# setting directory path
navbarMaker(dir_path = ".../Documents/")

# both options
navbarMaker(directory_name = "Visualization Tool", dir_path = ".../Documents")
```

3. make_my_text_shiny()

The make_my_text_shiny() writes default text to the shiny app files. Ships with navbarMaker() and reads the dir_path

```
make_my_text_shiny(file_path, file_type)
```

Parameters

The function make_my_text_shiny() takes the follow parameters:

- file_path: path of the file.
- file_type: name the file type you are writing text to. Options include: ui, server, global, main, and css.

Examples

```
# source file
source("shiny_maker.R")

# write ui
make_my_text_shiny(file_path = "/path/to/dir/ui.R", file_type = "ui")

# write server
make_my_text_shiny(file_path = "/path/to/dir/ui.R", file_type = "server")

# write global
make_my_text_shiny(file_path = "/path/to/dir/ui.R", file_type = "global")

# write main tab Panel
make_my_text_shiny(file_path = "/path/to/dir/ui.R", file_type = "main")

# write css
make_my_text_shiny(file_path = "/path/to/dir/ui.R", file_type = "css")
```