Package 'FSsetup'

May 23, 2024

Title Package for Producing R Prjects and Packages using the Defra Farming Stats Template

Version 0.0.1

Author Josh Moatt

Maintainer Josh Moatt <joshua.moatt@defra.gov.uk>

Description This package contains the functions needed to create an R package
or project using the Defra Farming Stats Team template. It adds all the
necessary files and liceneces. It can be run in local repositories or in
repositories cloned from GitHub.

License MIT + file LICENSE

Encoding UTF-8

Imports cli (>= 3.6.2),
devtools (>= 2.4.5),
glue (>= 1.7.0),
here (>= 1.0.1),
pkgbuild (>= 1.4.4),
quarto (>= 1.4),
stringr (>= 1.5.1),
usethis (>= 2.2.3)

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

R topics documented:

---

add_fs_script   *Create a new script with farming stats header.*

Description

This function will create a new script with the farming stats header added. There a multiple options for cutomisation (see details).

Usage

```
add_fs_script(
  file_name = NULL,
  folder = NULL,
  author = NULL,
  email = NULL,
  date = format(Sys.Date(), "%d/%m/%Y")
)
```

Arguments

file_name	string containing desired name for script.
folder	string containing folder name to save script. This is built on the here() function in R, so will follow your root directory. If you want to save in a sub-folder, enter the full folder sequence, e.g. "folder/sub-folder".
author	string containing author's name.
email	string containing author's email.
date	string containing a date. By default, this will be set as today's date.

Details

This is a simple function to create a new script with the farming stats header template added.

By just calling the function a script will be created with the template entered. All fields of the template will be blank except for the date, which is set to today's date.

There are options for customising and entering some details automatically. The details that can be pre-entered are: #'

- script name
- location to save
- author
- author's email
- date created

By default these are set to NULL (except for the date). But can be set to any string.

Value

An R script will be saved in the root directory or in the specified folder.

create_fs_pkg	*Create an R package based on the Farming Stats template*

Description

This function will create an R package using the Farming Stats template.

Usage

```
create_fs_pkg(
  pkg_name,
  author,
  readme_format = c("github", "html", "markdown")
)
```

Arguments

pkg_name	string containing package name.
author	string containing author name and email.
readme_format	string indicating readme format. can be "github", "html" or "markdown".

Details

Use this function to create an R package based on the farming stats template. The function should be run within an already established R project.

For easy git integration, first set up a repo on GitHub, then clone the repo to you local directory, then run this function. Though this is not necessary if your package is not being pushed to GitHub.

This function will add:

- DESCRIPTION
- NAMESPACE
- MIT License
- Defra licence
- R folder
- gitignore
- buildignore
- README Qmd and MD

Value

The output will be all the necessary package files within your repo.

---

create_fs_proj   *Create R project based on the Farming Stats template*

Description

This function will create an R project using the Farming Stats template.

Usage

```
create_fs_proj(
  readme_format = c("markdown", "github", "html"),
  file_path = NULL,
  proj_struc = NULL
)
```

Arguments

readme_format   controls the output format of the README. Default is "markdown", but can be "html" or "github".

file_path   optional string containing file path to project folder. Default is the current working directory.

proj_struc   optional vector containing the project structure you want. Default is to add the accepted Farming Stats skeleton, which is described in the details.

Details

Use this function to create an R project based on the farming stats template. The function should be run within an already established R project.

For easy git integration, first set up a repo on GitHub, then clone the repo to you local directory, then run this function. Though this is not necessary if your package is not being pushed to GitHub.

This function will add folders by default (see below). However, if you would like, you can choose an alternative project structure that suits your needs.

By default this function will add:

- inputs folder - this should contain all the inputs used in the project (if appropriate). This could be data, templates, txt files etc. This can be customised with sub-folders as required.

- src folder - this should contain all the scripts sourced into your pipeline. E.g. this may contain functions, or scripts for reading in the data that can be sourced into the pipeline. This can be customised with sub-folders as required.

- output folder - this should be where your outputs are saved (where appropriate). This can be customised with sub-folders as required.

- main pipeline script - this should be where the main code is that executes your project. This may need to expand to multiple scripts, but should be the active pipeline code (rather than scripts containing functions etc).

- README (Qmd and MD) - with all R projects you should provide a full README that gives users all the information they need to run your project. You should edit this as you develop you project.

Value

Output is all the files needed for a basic R project based on the Farming Stats template.

---

create_fs_readme *Create a Farming Stats README*

Description

Use this function to create a README for a Farming Stats project or package.

Usage

```
create_fs_readme(
  type = c("project", "package"),
  format = c("markdown", "github", "html"),
  file_path = NULL,
  author = NULL,
  readme_title = NULL
)
```

Arguments

type	controls the README template that is produced. Can be "project" or "package".
format	controls the output format of the README. Default is "markdown", but can be "html" or "github".
file_path	sting containing file path where README will be saved.
author	string containing authors name. If no string provided author will be set to "add author".
readme_title	string containing README title. If no string provided will be set to "README (edit title)".

Details

This function will create a README for a project or package using a pre-defined Farming Stats template. The function will create a Quarto (.qmd) file and do a first render producing a desired output.

The output type can be controlled using the format option (default is markdown). Can also be html or github (gfm).

The template for projects and packages is slightly different. You can control the type of README by using the "type" argument (project or package).

Value

Output is a .qmd file containing the desired README template and an initial render of the README in the desired output.

```
create_script_template
```
Create an default R script template.

Description

This function can be used to create a default R script template. All scripts opened from this point will have this template applied as a header. More information provided in the details section.

Usage

```
create_script_template(
  format = c("farming_stats", "custom", "manual_edit", "blank"),
  template = NULL
)
```

Arguments

format	what format the template will take. There are four options: "farming_stats" (default farming stats template), "custom" will apply a custom template (provided as a string), "manual_edit" will open the template so you can manually edit the template, and "blank" can be used to remove all pre-existing templates. Note: manual edit can also be used to edit existing templates.
template	default is NULL, only used if format = "custom". Used to provide custom template design. Must be provided as a string. Must be provided if using format = "custom" or the function will return an error.

Details

This function is used to create a new template for R scripts. This will be the default that all subsequent R scripts opened will contain. By doing this is it should be easier to follow best practice and properly comment all scripts you create.

The default is stored in the appdata folder on your c drive: "~/AppData/Roaming/RStudio". It will create a "templates" folder where the default will be stored.

The function has various ways it can work which will give you the ability to create whatever header template you wish. It has four ways formatting options:

- farming_stats
- custom
- manual_edit
- blank

"farming_stats" will pre-load the default script with the farming stats template.

"custom" will allow you to provide your own custom template as a string, which will then be added to the default.

"manual" will allow you to manually edit the default template. It will open it in your R studio window and manual edits can be saved. Note, this can also be used to tweak a pre-existing template (e.g. to add your name and email to all scripts).

"blank" will delete he default R script and template. This returns R back to normal, and any script opened subsequently will be blank.

Value

New .R file created at "~/AppData/Roaming/RStudio/templates/default.R" containing the script template

Index