

Package: ActionSquirrel (via r-universe)

August 17, 2024

Version 0.1.0

Title Play An R6-Class 'Action Squirrel' Game

Description Prints to the console a grid of emoji representing a forest, including a player-controlled squirrel that can be moved by the user to collect nuts and avoid an owl enemy. Collect enough nuts to survive winter.

URL <https://github.com/matt-dray/ActionSquirrel>

BugReports <https://github.com/matt-dray/ActionSquirrel/issues>

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

Imports R6, sonify

Repository <https://matt-dray.r-universe.dev>

RemoteUrl <https://github.com/matt-dray/ActionSquirrel>

RemoteRef HEAD

RemoteSha 19cd3390d7bc34c005a01151ae2087fdb57ee3de

Contents

ActionSquirrel	2
Index	4

`ActionSquirrel`*Generate An R6-Class 'Action Squirrel' Game*

Description

Generate An R6-Class 'Action Squirrel' Game

Generate An R6-Class 'Action Squirrel' Game

Details

Prints to the console a grid of emoji representing an overworld, including a player character that can be controlled by the user. Also provides a live tally of moves taken.

Value

Console output with `cat()`.

Public fields

`active` Logical. Is the game in play?

`overworld` Character vector. The overworld in one dimension.

`s_loc` Numeric. The location of the squirrel.

`n_loc` Numeric. The location of the nut.

`o_loc` Numeric. The location of the owl.

`moves` Numeric. The number of moves made by the player.

`nuts` Numeric. The number of nuts collected by the player.

Methods

Public methods:

- [ActionSquirrel\\$new\(\)](#)
- [ActionSquirrel\\$pause\(\)](#)
- [ActionSquirrel\\$move\(\)](#)
- [ActionSquirrel\\$clone\(\)](#)

Method `new()`: Create a new overworld.

Usage:

```
ActionSquirrel$new()
```

Returns: An R6-class object.

Method `pause()`: Pause menu with instructions.

Usage:

```
ActionSquirrel$pause()
```

Returns: An R6-class object.

Method `move()`: Move the hero around the overworld.

Usage:

```
ActionSquirrel$move(where = c("up", "down", "left", "right"))
```

Arguments:

`where` Character. Which direction to move. One of "up", "down", "left" and "right".

Returns: An R6-class object.

Method `clone()`: The objects of this class are cloneable with this method.

Usage:

```
ActionSquirrel$clone(deep = FALSE)
```

Arguments:

`deep` Whether to make a deep clone.

Examples

```
## Not run:
x <- ActionSquirrel$new() # generate overworld
x$move("up")             # move character up
x$move("right")          # move character right

## End(Not run)
```

Index

ActionSquirrel, [2](#)