Package ‘rwarrior’

October 14, 2022

Type Package

Title R Warrior - An AI Programming Game

Version 0.4.1

Description A port of Ruby Warrior.
          Teaches R programming in a fun and interactive way.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.1

Imports methods, R6, glue, cli, dplyr, tibble, checkmate, stringr

Suggests covr, knitr, rmarkdown, purrr, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Author Rick M Tankard [cre, aut] (<https://orcid.org/0000-0002-8847-9401>)

Maintainer Rick M Tankard <rickmtankard@gmail.com>

Repository CRAN

Date/Publication 2022-09-07 07:40:02 UTC

R topics documented:

level_readme .......................................................... 2
play_epic ............................................................... 2
play_warrior ........................................................... 3

Index 5
### level_readme

**Level read me**

**Description**

The starting point for R Warrior. Run this before attempting each level with play_warrior().

**Usage**

```r
level_readme(level = 1, tower = c("beginner"))
```

**Arguments**

- **level**: The level number (or custom level).
- **tower**: The tower the level comes from.

**Examples**

```r
level_readme(1)
```

### play_epic

**Play through an epic quest of a tower**

**Description**

Write a single AI function to play through each level of the specified tower. Refine your AI in order to achieve an overall S rank.

**Usage**

```r
play_epic(
  ai,
  tower = c("beginner"),
  warrior_name = "Fisher",
  level_output = TRUE,
  sleep = getOption("rwarrior.sleep", ifelse(interactive(), 0.6, 0))
)
```

**Arguments**

- **ai**: AI function to control your warrior.
- **tower**: Tower to attempt.
- **warrior_name**: Name of your warrior, for flavor.
- **level_output**: A logical denoting whether to give individual level progress.
- **sleep**: Time between text updates. Set to "prompt" to only progress when pressing the return key.
Value

A tibble if successful, or otherwise FALSE.
A tibble giving the scores for each level passed.

Examples

AI <- function(warrior, memory) {
  if(is.null(memory)) {
    # set memory initial values here
  }
  # Modify the following section to be able to complete the tower
  warrior$walk()
  memory
}
play_epic(AI, tower = "beginner", warrior_name = "Euler")

Description

Attempt inbuilt levels of R Warrior.

Usage

play_warrior(
  ai,
  level = 1,
  tower = c("beginner"),
  warrior_name = "Fisher",
  sleep =getOption("rwarrior.sleep", ifelse(interactive(), 0.6, 0)),
  practice = FALSE
)

Arguments

ai AI function to control your warrior.
level Level number.
tower Tower the level comes from.
warrior_name Name of your warrior, for flavor.
sleep Time between text updates in seconds. Set to "prompt" to only progress when pressing the return key.
practice If TRUE, any functions available for that tower may be used.
Value

A tibble if successful, FALSE if unsuccessful, and NA if the AI function caused an error or no action was called.

Examples

```r
AI <- function(warrior, memory) {
  if(is.null(memory)) {
    # set memory initial values here
  }
  # insert AI code here
  memory
}
play_warrior(AI, level = 1)
```
Index

level_readme, 2

play_epic, 2
play_warrior, 3