Challenge-5

Ho Wei Ni

2023-09-11

Questions

Question-1: Local Variable Shadowing Create an R function that defines a global variable called `x` with a value of 5. Inside the function, declare a local variable also named `x` with a value of 10. Print the value of `x` both inside and outside the function to demonstrate shadowing.

Solutions:

```
x <- 5
print(x)
```

```
## [1] 5
```

```
new_x <- function(x) {
  x <- 10
  return(x)
}
new_x()
```

```
## [1] 10
```

```
x
```

```
## [1] 5
```

Question-2: Modify Global Variable Create an R function that takes an argument and adds it to a global variable called `total`. Call the function multiple times with different arguments to accumulate the values in `total`.

Solutions:

```
total <- 0

accumulated_total <- function(value) {
  total + value
}

total <- accumulated_total(5)
total <- accumulated_total(10)
total <- accumulated_total(90)
total
```

```
## [1] 105
```

Question-3: Global and Local Interaction Write an R program that includes a global variable `total` with an initial value of 100. Create a function that takes an argument, adds it to `total`, and returns the updated `total`. Demonstrate how this function interacts with the global variable.

Solutions:

```
total <- 100
add_to_total <- function(x) {
  total <- total + x
}

total <- add_to_total(200)
total <- add_to_total(20)
total
```

```
## [1] 320
```

Question-4: Nested Functions Define a function `outer_function` that declares a local variable `x` with a value of 5. Inside `outer_function`, define another function `inner_function` that prints the value of `x`. Call both functions to show how the inner function accesses the variable from the outer function's scope.

Solutions:

```
outer_function <- function() {
  x <- 5

  inner_function <- function() {
    print(x)
  }

  inner_function()
}

outer_function()
```

```
## [1] 5
```

Question-5: Meme Generator Function Create a function that takes a text input and generates a humorous meme with the text overlaid on an image of your choice. You can use the `magick` package for image manipulation. You can find more details about the commands offered by the package, with some examples of annotating images here: https://cran.r-project.org/web/packages/magick/vignettes/intro.html

Solutions:

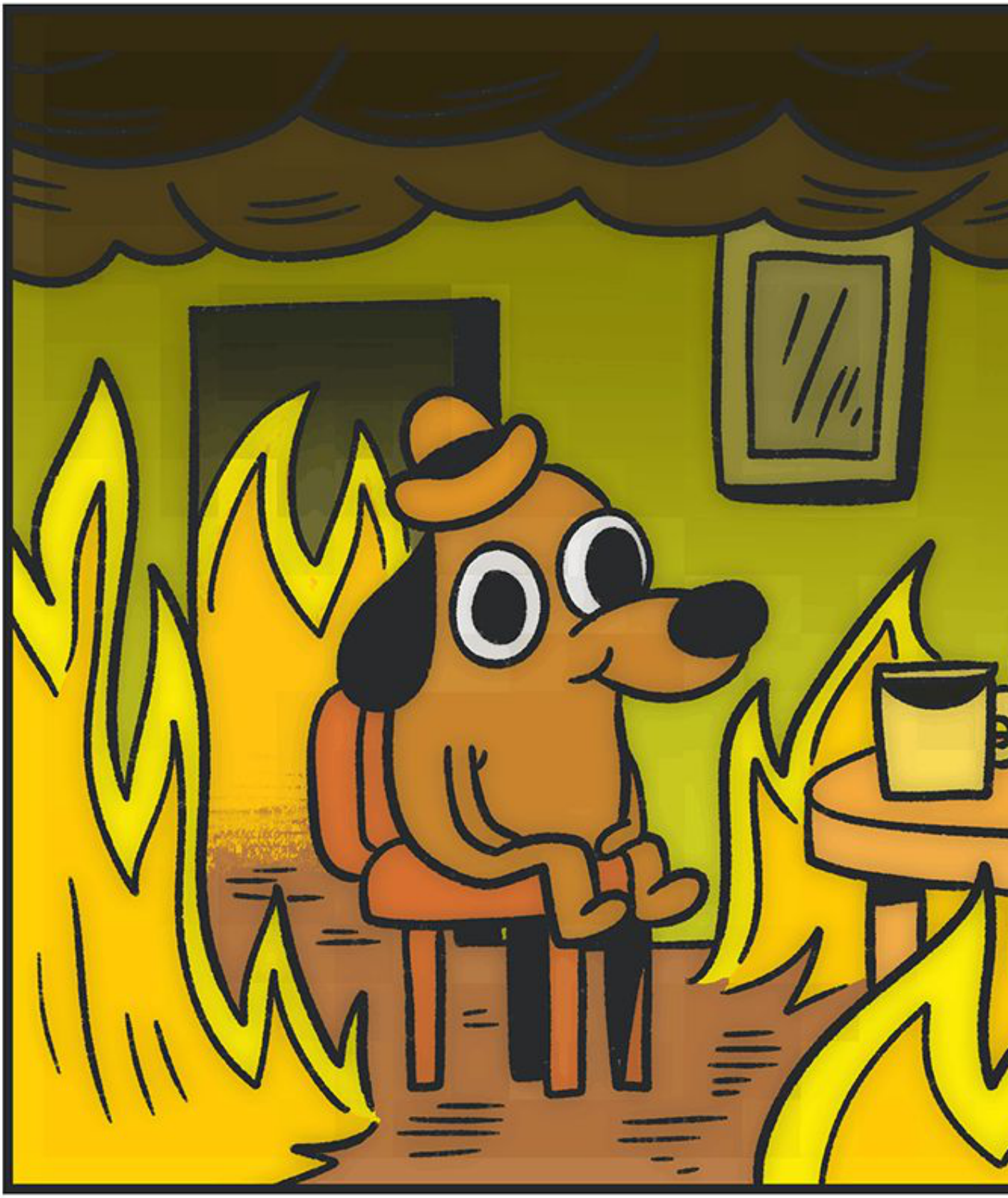
```
library(magick)
```

```
## Linking to ImageMagick 6.9.12.93
## Enabled features: cairo, fontconfig, freetype, heic, lcms, pango, raw, rsvg, webp
## Disabled features: fftw, ghostscript, x11
```

```r
meme_generator <- function(text_input) {
  img <- image_read('thisisfinetemplate.jpeg')
  #img<- image_annotate(img, text_input, size = 70, gravity = "southwest", color = "green")
  #image_annotate is not working :()
  print(img)
}

meme_generator("Hello")
```

```
##   format width height colorspace matte filesize density
## 1   JPEG   800    815       sRGB FALSE   108138   72x72
```



Question-6: Text Analysis Game Develop a text analysis game in which the user inputs a sentence, and the R function provides statistics like the number of words, characters, and average word length. Reward the user with a "communication skill level" based on their input.

Solutions:

```
text_analysis_game <- function(sentence) {
  num_of_words <- length(strsplit(sentence, " ")[[1]])
  num_of_chars <- nchar(sentence)
  avg_word_length <- num_of_chars / num_of_words

  comm_skill_level <- ifelse(avg_word_length < 4, "Beginner",
                        ifelse(avg_word_length < 6, "Intermediate", "Advanced"))

  print(paste0("Number of words: ", num_of_words))
  print(paste0("Number of characters: ", num_of_chars))
  print(paste0("Average word length: ", avg_word_length))
  print(paste0("Your communication level is: ", comm_skill_level))
}

text_analysis_game("The quick brown fox jumps over the lazy dog in the backyard")
```

```
## [1] "Number of words: 12"
## [1] "Number of characters: 59"
## [1] "Average word length: 4.91666666666667"
## [1] "Your communication level is: Intermediate"
```