

Challenge-5

Sherica Chua

13/09/2023

Questions

```
library(tidyverse)
```

Question-1: Local Variable Shadowing Solutions:

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.

```
x <-5
function_1 <- function() {

  x <- 10
  print(x)
}

print(x)
```

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

```
function_1()
```

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

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 <- 3
function_2 <- function(x) {
  total <- x + total
}

function_2(10)
print(total)
```

```
## [1] 13
```

```
function_2(5)
print(total)
```

```
## [1] 18
```

```
function_2(33)
print(total)
```

```
## [1] 51
```

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<-3

new_total <- function(add) {

  total <- add + total
}

new_total(3)
total
```

```
## [1] 103
```

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, freetype, fftw, ghostscript, heic, lcms, pango, raw, rsvg, webp
## Disabled features: fontconfig, x11
```

```
str(magick::magick_config())
```

```
## List of 24
## $ version      :Class 'numeric_version'  hidden list of 1
## ..$ : int [1:4] 6 9 12 93
## $ modules      : logi FALSE
## $ cairo        : logi TRUE
## $ fontconfig   : logi FALSE
## $ freetype     : logi TRUE
## $ fftw         : logi TRUE
## $ ghostscript  : logi TRUE
## $ heic         : logi TRUE
## $ jpeg         : logi TRUE
## $ lcms         : logi TRUE
## $ libopenjp2   : logi TRUE
## $ lzma         : logi TRUE
## $ pangocairo   : logi TRUE
## $ pango        : logi TRUE
## $ png          : logi TRUE
## $ raw          : logi TRUE
## $ rsvg         : logi TRUE
## $ tiff         : logi TRUE
## $ webp         : logi TRUE
## $ wmf         : logi FALSE
## $ x11         : logi FALSE
## $ xml         : logi TRUE
## $ zero-configuration: logi TRUE
## $ threads      : int 1
```

```
Meme <- function(path) {
  frink<-image_read(path)
  image_annotate(frink, "CONFIDENTIAL", size = 30, color = "red", boxcolor = "pink",
    degrees = 60, location = "+50+100")
}
```

```
Meme("https://jeroen.github.io/images/frink.png")
```



Question-6: Text Analysis Game

Develop a text analysis game in which the user inputs a sentence, and the R function provides statistics.

****Solutions:****

```
'''r
text_analysis_game <- function() {

  cat("Welcome to the Text Analysis Game!\n")
  sentence <- readline(prompt = "Enter a sentence: ")
```

```

words <- strsplit(sentence, " ")[[1]]
num_words <- length(words)
num_chars <- nchar(sentence)
avg_word_length <- num_chars / num_words

cat("\nText Statistics:\n")
cat("Number of words:", num_words, "\n")
cat("Number of characters:", num_chars, "\n")
cat("Average word length:", round(avg_word_length, 2), "\n")

ifelse(avg_word_length < 4, skill_level <- "Novice Communicator", ifelse(avg_word_length < 6, skill_level <- "Intermediate Communicator", "Advanced Communicator"))

cat("\nCommunication Skill Level:", skill_level, "\n")
}

text_analysis_game()

```