

# Challenge-5

Marzuki Nooranas

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:

```
# Enter code here
x <- 5

shadowing <- function() {
  x <- 10
  print(paste("Inside the function, x is:", x))
}
shadowing()

## [1] "Inside the function, x is: 10"
print(paste("Outside the function, x is:", x))

## [1] "Outside the function, x is: 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:

```
# Enter code here

#this is to initiate the variable
total <- 0

add_total <- function(number) {
```

```

total <- total + number
print(paste("total is now ", total))
}

add_total(5)

## [1] "total is now 5"

add_total(5)

## [1] "total is now 10"

add_total(5)

## [1] "total is now 15"

```

### *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:**

```

# Enter code here

total <- 100

add_total <- function(number) {
  total <- total + number
  return(total)
}

new_total <- add_total(10)
sprintf ("The new total is %d", new_total)

## [1] "The new total is 110"

new_total <- add_total(20)
sprintf ("The new total is %d", new_total)

## [1] "The new total is 130"

new_total <- add_total(30)
sprintf ("The new total is %d", new_total)

## [1] "The new total is 160"

```

#### *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:**

```
# Enter code here

outer_function <- function(){
  x <- 5
  inner_function <- function() {
    sprintf ("The value of x is %d", x)
  }
  inner_function()
}
outer_function()

## [1] "The value of x is 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:**

```
# Enter code here
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

create_meme <- function(image_path, meme_text) {
  img <- image_read(image_path)

  # add text
  meme <- img %>%
    image_annotate(text = meme_text,
                  gravity = 'south',
                  location = "+0+100",
                  size = 200,
```

```

        font = "Impact",
        strokecolor = "black",
        color = "white")

return(meme)
}

create_meme("/Users/marzuki/Desktop/NM2207/NM2207/Week-5/meme.png", "Me after
NM2207")

```



#### *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:**

```

# Enter code here

text_analysis_game <- function(sentence) {

```

```

cat("Hello everyone, time for the Text Analysis Game!\n")
cat("Analyzing the sentence:\n")

# number of words
word_count <- strsplit(sentence, " ")[[1]]
n_words <- length(word_count)

# number of char
n_chars <- nchar(sentence)

# average word length
word_lengths <- sapply(word_count, nchar)
avg_length <- mean(word_lengths)

# statistics
cat("\nHere are the results:\n")
cat("Number of words:", n_words, "\n")
cat("Number of characters:", n_chars, "\n")
cat("Average word length:", avg_length, "\n")

# Calculate a score
score <- (n_words * 2) + (n_chars / 10) - (avg_length * 3)

# calculate score i think
if (score < 20) {
  skill_level <- "Room for improvement. Keep practicing!"
} else if (score >= 20 & score <= 50) {
  skill_level <- "Not bad at all! You're doing great."
} else {
  skill_level <- "Excellent! Your communication skills are top-notch."
}

# print
cat("Your score is:", score, "\n")
cat("Your communication skill level is:", skill_level, "\n")
}

text_analysis_game("This is an example of the sentence maybe I could try to
calculate the score")

## Hello everyone, time for the Text Analysis Game!
## Analyzing the sentence:
##
## Here are the results:
## Number of words: 15
## Number of characters: 75
## Average word length: 4.066667

```

```
## Your score is: 25.3
## Your communication skill level is: Not bad at all! You're doing great.
```