Challenge-5

Jing Ying

2023-09-11

```
knitr::opts_chunk$set(echo = TRUE)
```

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

shadowing_example <- function() {
  x <- 10

  cat("Value of x inside the function:", x, "\n")
}

x <- 5

cat("Value of x outside the function:", x, "\n")
```

```
## Value of x outside the function: 5
```

```
shadowing_example()
```

```
## Value of x inside the function: 10
```

```
cat("Value of x outside the function after shadowing:", x, "\n")
```

```
## Value of x outside the function after shadowing: 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:

```r
# Enter code here

total <- 1

accumulate_total <- function(value) {
  total <<- total + value}
accumulate_total(5)
accumulate_total(10)
accumulate_total(3)

print(total)
```

```
## [1] 19
```

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:

```r
# Enter code here

total <- 100

add_to_total <- function(num) {
  total <<- total + num
  return(total)
}

cat("Initial value of total:", total, "\n")
```

```
## Initial value of total: 100
```

```r
result1 <- add_to_total(50)
cat("After adding 50, total becomes:", result1, "\n")
```

```
## After adding 50, total becomes: 150
```

```r
result2 <- add_to_total(25)
cat("After adding 25, total becomes:", result2, "\n")
```

```
## After adding 25, total becomes: 175
```

```r
cat("Final value of total:", total, "\n")
```

```
## Final value of total: 175
```

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() {
    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:

```
# Enter code here

install.packages("magick",repos = "http://cran.us.r-project.org")
```

```
## Installing package into 'C:/Users/yeojy/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## package 'magick' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'magick'

## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying
## C:\Users\yeojy\AppData\Local\R\win-library\4.3\00LOCK\magick\libs\x64\magick.dll
## to C:\Users\yeojy\AppData\Local\R\win-library\4.3\magick\libs\x64\magick.dll:
## Permission denied

## Warning: restored 'magick'

##
## The downloaded binary packages are in
##  C:\Users\yeojy\AppData\Local\Temp\Rtmpysbt77\downloaded_packages
```

```
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
```

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

```
## $version
## [1] '6.9.12.93'
##
## $modules
## [1] FALSE
##
## $cairo
## [1] TRUE
##
## $fontconfig
## [1] FALSE
##
## $freetype
## [1] TRUE
##
## $fftw
## [1] TRUE
##
```

```
## $ghostscript
## [1] TRUE
##
## $heic
## [1] TRUE
##
## $jpeg
## [1] TRUE
##
## $lcms
## [1] TRUE
##
## $libopenjp2
## [1] TRUE
##
## $lzma
## [1] TRUE
##
## $pangocairo
## [1] TRUE
##
## $pango
## [1] TRUE
##
## $png
## [1] TRUE
##
## $raw
## [1] TRUE
##
## $rsvg
## [1] TRUE
##
## $tiff
## [1] TRUE
##
## $webp
## [1] TRUE
##
## $wmf
## [1] FALSE
##
## $x11
## [1] FALSE
##
## $xml
## [1] TRUE
##
## $`zero-configuration`
## [1] TRUE
##
## $threads
## [1] 1
```

```r
# Enter code here

generate_meme <- function(text, image_path) {
  image <- image_read(image_path)
  image <- image_scale(image, "500")
  meme <- image_annotate(image, text, gravity = "center", color = "white", size = 20, boxcolor = "black
  image_browse(meme)
  output_path <- paste0("meme_", format(Sys.time(), "%Y%m%d%H%M%S"), ".png")
  image_write(meme, output_path)
  return(output_path)
}

text_input <- "when the deck cai fan becomes a scam"
image_path <- "C:/Users/yeojy/Pictures/jiakcao.jpg"
gimmemycaipngback <- generate_meme(text_input, image_path)

print(gimmemycaipngback)
```

```
## [1] "meme_20230911201328.png"
```

```r
knitr::include_graphics("C:/Users/yeojy/Documents/Y2S1/NM2207/Week 5/meme_20230911194057.png")
```

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:

```r
# Enter code here

text_analysis_game <- function() {
  cat("Enter a sentence: ")
  sentence <- readLines(n = 1)

  if (nchar(trimws(sentence)) == 0) {
    cat("Error: Empty sentence. Please enter a valid sentence.\n")
    return()
  }
```

```
  sentence <- sentence[1]

  words <- strsplit(sentence, "\\s+")
  num_words <- length(words[[1]])

  num_chars <- nchar(sentence)

  word_lengths <- sapply(words[[1]], nchar)
  avg_word_length <- mean(word_lengths)

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

  skill_level <- ifelse(avg_word_length < 4, "NOOB",
                        ifelse(avg_word_length < 6, "UR OK", "PRETENTIOUS"))

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

text_analysis_game()
```

```
## Enter a sentence: Error: Empty sentence. Please enter a valid sentence.
```

```
## NULL
```

```
knitr::include_graphics("C:/Users/yeojy/Documents/Y2S1/NM2207/Week 5/YAYYAYYAYYYYAYYYAYYYYYY.png")
```