Challenge-5

Tan Ee Xuan

2023-09-13

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

print(paste0("Outside the function, x = ", x))
```

```
## [1] "Outside the function, x = 5"
```

```
one <- function() {
  x <- 10
  paste0("Inside the function, x = ", x)
}

one()
```

```
## [1] "Inside the function, x = 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:

```
# Enter code here
total <- 5

questiontwo <- function(x) {
  total <<- x + total
}

questiontwo(5)
print(total)
```

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

```
questiontwo(12)
print(total)
```

```
## [1] 22
```

```
questiontwo(73)
print(total)
```

```
## [1] 95
```

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
questionthree <- function(x) {
  total <<- total + x
}

questionthree(67)
total
```

```
## [1] 167
```

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)
  }
  return(inner_function())
}

print(outer_function())
```

```
## [1] 5
## [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 (https://cran.r-project.org/web/packages/magick/vignettes/intro.html)

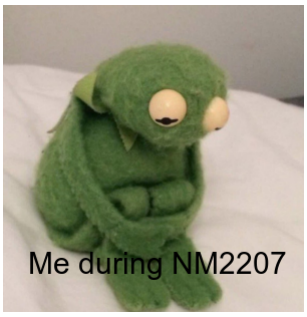
Solutions:

```
# Enter code here


library(magick)
```

```
## Linking to ImageMagick 6.9.12.3
## Enabled features: cairo, fontconfig, freetype, heic, lcms, pango, raw, rsvg, webp
## Disabled features: fftw, ghostscript, x11
```

```
input <- function(path, text) {
  meme_pic <- image_scale(image_read(path), "300")
  image_annotate(meme_pic, print(paste0(text)), size = 30, location = "+25+240")
}

input("/Users/eexuan/Downloads/AY2324 Sem1/NM2207/Week-5/meme.jpeg", "Me during NM220
7")
```

```
## [1] "Me during 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
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──
## ✔ ggplot2 3.3.6     ✔ purrr   0.3.4
## ✔ tibble  3.1.8     ✔ dplyr   1.0.9
## ✔ tidyr   1.2.0     ✔ stringr 1.4.0
## ✔ readr   2.1.2     ✔ forcats 0.5.1
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
input <- function(text) {
  words <- strsplit(text, " ")[[1]]
  num_words <- length(words)

  num_char <- nchar(text)

  avg_word_length <- mean(nchar(words))

  print(paste0("Number of words = ", num_words,
                       ", Number of Characters = ", num_char,
                       ", Average Word Length = ", avg_word_length))

  ifelse (avg_word_length >= 5, "Communication Skill Level: Good", "Communication Ski
ll Level: Bad")

}

input("Good Morning Everyone")
```

```
## [1] "Number of words = 3, Number of Characters = 21, Average Word Length = 6.33333
333333333"
```

```
## [1] "Communication Skill Level: Good"
```