Challenge-5

Insert your name here

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:

```
x <- 5
shadowing <- function() {
  x <- 10
  print(paste("Inside the function is", x))}

shadowing()
```

```
## [1] "Inside the function is 10"
```

```
print(paste("Outside the function is", x))
```

```
## [1] "Outside the function 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:

```
total <- 0

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

total <- add_to_total(3)
total <- add_to_total(10)
total <- add_to_total(7)

total
```

```
## [1] 20
```

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(value) {
  total <<- total + value
  return(total)
}

# Demonstrate how the function interacts with the global variable
print(paste("Initial total is", total))
```

```
## [1] "Initial total is 100"
```

```
new_total <- add_to_total(50)
print(paste("After adding 50 is", new_total))
```

```
## [1] "After adding 50 is 150"
```

```
new_total <- add_to_total(30)
print(paste("After adding 30 is", new_total))
```

```
## [1] "After adding 30 is 180"
```

```
print(paste("Global total after function calls is", total))
```

```
## [1] "Global total after function calls is 180"
```

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(paste("Inner function shows outer:", x))
  }

  inner_function()
}

outer_function()
```

```
## [1] "Inner function shows outer: 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:

```
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
```

```
image <- image_read("10yhq0.jpg")

  text <- "LOL"
font <- "Arial-Bold"
size <- 50
color <- "white"
position <- "+100+100"

image <- image_annotate(image, text, size = size, color = color, location = position)

image_write(image, path = "output_meme.jpg")

image
```

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:

```
# Function to analyze the user's input and provide statistics and communication skill level
analyze_text_game <- function() {
  message("Y u wanna play???")
  message("Anyhow, enter a word, may the longest win.")

  user_input <- readline(prompt = "Enter here")

  words <- unlist(strsplit(user_input, " "))

  num_characters <- nchar(user_input)
  num_words <- length(words)
  average_word_length <- num_characters / num_words

  skill_level <- ifelse(
    average_word_length < 4,
    "B0000",
    ifelse(
      average_word_length < 6,
      "Ok ba",
      "Not bad!!"
    )
  )

  message("\nStatistics:")
  message("Number of characters:", num_characters)
  message("Number of words:", num_words)
  message("Average word length:", round(average_word_length, 2))

  message("\nCommunication Skill Level:", skill_level)
}

analyze_text_game()
```

```
## Y u wanna play???
```

```
## Anyhow, enter a word, may the longest win.
```

```
## Enter here
```

```
##
## Statistics:
```

```
## Number of characters:0
```

```
## Number of words:0
```

```
## Average word length:NaN
```

```
##
## Communication Skill Level:NA
```