Challenge-5

Esther Kho Yining

2024-02-13

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
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:

```
# Enter code here
total <- 1
accumulate_total <- function(value) {
  total <<- total + value}

accumulate_total(5)
accumulate_total(10)
accumulate_total(15)

print(total)
```

```
## [1] 31
```

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

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

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

```
result1 <- add_to_total(25)
cat("After adding 25, total becomes:", result1, "\n")
```

```
## After adding 25, total becomes: 125
```

```
result2 <- add_to_total(50)
cat("After adding 50, total becomes:", result2, "\n")
```

```
## After adding 50, total becomes: 175
```

```
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:

```
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:

```
install.packages("magick")
```

```
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
```

```
generate_meme <- function(text_input, image_path) {
  image <- image_read(image_path)
  image <- image_scale(image, "200")
  image_with_text <- image_annotate(image, text_input, gravity = "center", size = 50,
                                    color = "black", font = "Impact")
  print(image_with_text)
}

text_input <- "when you got work to clear during CNY"
image_path <- "Squidward_meme.jpg"
generate_meme(text_input, image_path)
```

```
## Error in eval(expr, envir, enclos): R: NonconformingDrawingPrimitiveDefinition ‘text’ @ error/draw.c
```

##Note: I’m a macbook user

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 <- readline("Enter a sentence: ")
  sentence <- trimws(sentence)
  words <- strsplit(sentence, "\\s+")[[1]]
  num_words <- length(words)
  num_chars <- nchar(sentence)
  avg_word_length <- mean(nchar(words))

skill_level <- ifelse(num_words >=10 & avg_word_length >=5, "Advanced",
ifelse(num_words >= 5 & avg_word_length >= 4, "Intermediate", "Beginner"))

stats <- list(
  num_words = num_words,
  num_chars = num_chars,
```

```r
  avg_word_length = avg_word_length)

print(paste("Number of words:", stats$num_words))
print(paste("Number of characters:", stats$num_chars))
print(paste("Average word length:", stats$avg_word_length))
print(paste("Skill Level:", skill_level))
return(list(stats=stats, skill_level=skill_level)) }

text_analysis_game()
```

```
## Enter a sentence:
## [1] "Number of words: 0"
## [1] "Number of characters: 0"
## [1] "Average word length: NaN"
## [1] "Skill Level: Beginner"

## $stats
## $stats$num_words
## [1] 0
##
## $stats$num_chars
## [1] 0
##
## $stats$avg_word_length
## [1] NaN
##
##
## $skill_level
## [1] "Beginner"
```