

Week-5: Code-along

Lok

2023-09-10

II. Code to edit and execute using the Code-along.Rmd file

A. Writing a function

1. Write a function to print a “Hello” message (Slide #14)

```
say_hello_to <- function(name) {  
  print(paste0("Hello", name, "!"))  
}
```

2. Function call with different input names (Slide #15)

```
say_hello_to('lok')
```

```
## [1] "Hello!lok!"
```

3. typeof primitive functions (Slide #16)

```
typeof(sum)
```

```
## [1] "builtin"
```

4. typeof user-defined functions (Slide #17)

```
typeof(say_hello_to)
```

```
## [1] "closure"
```

5. Function to calculate mean of a sample (Slide #19)

```
calc_sample_mean <- function(sample_size) {  
  mean(rnorm(sample_size))  
}
```

6. Test your function (Slide #22)

```
calc_sample_mean(1000)
```

```
## [1] 0.009837235
```

```
calc_sample_mean(c(100,300,3000))
```

```
## [1] 0.7655766
```

7. Customizing the function to suit input (Slide #23)

```
library(tidyverse)  
sample_tribble <- (sample_sizes = c(100,300,3000))  
  
sample_tribble %>%  
  group_by(sample_sizes) %>%  
  mutate(sample_means =  
    calc_sample_mean(sample_sizes))
```

8. Setting defaults (Slide #25)

```
calc_sample_mean <- function(sample_size,  
                              our_mean=0,  
                              our_sd=1) {  
  sample <- rnorm(sample_size,  
                  mean=our_mean,  
                  sd=our_sd)  
  mean(sample)  
}  
  
calc_sample_mean(sample_size = 10)
```

```
## [1] 0.5252375
```

9. Different input combinations (Slide #26)

```
calc_sample_mean(10,our_sd = 2)
```

```
## [1] -0.07960109
```

```
calc_sample_mean(10,our_mean = 6)
```

```
## [1] 5.787467
```

```
calc_sample_mean(10,6,2)
```

```
## [1] 4.966614
```

10. Different input combinations (Slide #27)

```
## [1] 5.094682
```

11. Some more examples (Slide #28)

```
add_two <- function(x) {  
  x+2  
}  
add_two(4)
```

```
## [1] 6
```

```
add_two(-34)
```

```
## [1] -32
```

```
add_two(5.654)
```

```
## [1] 7.654
```

B. Scoping

12. Multiple assignment of z (Slide #36)

```
z <- 1  
sprintf("the value assigned to z outside the funtion is %d",z)
```

```
## [1] "the value assigned to z outside the funtion is 1"
```

```
foo <- function(z = 2) {  
  z<-3  
  return(z+3)  
}  
foo()
```

```
## [1] 6
```

13. Multiple assignment of z (Slide #37)

```
foo(z=4)
```

```
## [1] 6
```

```
sprintf("the value assigned to z outside the funtion is %d",z)
```

```
## [1] "the value assigned to z outside the funtion is 1"
```