Week-3: Code-along

Wong Wei Qi

2023-08-30

# I. Code to edit and execute

**To be submitted on canvas before attending the tutorial**

### Loading packages

# Load package tidyverse  
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.2 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.3 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

### Assigning values to variables

# Example a.: execute this example  
x <- 'A'  
x

## [1] "A"

# Complete the code for Example b and execute it  
y <- "apple"  
y

## [1] "apple"

# Complete the code for Example c and execute it  
c <- FALSE  
c

## [1] FALSE

# Complete the code for Example d and execute it  
d <- 5L  
d

## [1] 5

# Complete the code for Example e and execute it  
e <- "5"  
e

## [1] "5"

# Complete the code for Example f and execute it  
f <- 1i  
f

### Checking the type of variables

# Example a.: execute this example  
x <- 'A'  
typeof(x)

# Complete the code for Example b and execute it  
typeof(y)

# Complete the code for Example c and execute it  
typeof(c)

# Complete the code for Example d and execute it  
typeof(d)

# Complete the code for Example e and execute it  
typeof(e)  
e

# Complete the code for Example f and execute it  
typeof(f)

### Need for data types

# import the cat-lovers data from the csv file you downloaded from canvas  
cat\_lovers <- read.csv("cat-lovers.csv")  
cat\_lovers

## name number\_of\_cats  
## 1 Bernice Warren 0  
## 2 Woodrow Stone 0  
## 3 Willie Bass 1  
## 4 Tyrone Estrada 3  
## 5 Alex Daniels 3  
## 6 Jane Bates 2  
## 7 Latoya Simpson 1  
## 8 Darin Woods 1  
## 9 Agnes Cobb 0  
## 10 Tabitha Grant 0  
## 11 Perry Cross 0  
## 12 Wanda Silva 0  
## 13 Alicia Sims 1  
## 14 Emily Logan 3  
## 15 Woodrow Elliott 3  
## 16 Brent Copeland 2  
## 17 Pedro Carlson 1  
## 18 Patsy Luna 1  
## 19 Brett Robbins 0  
## 20 Oliver George 0  
## 21 Calvin Perry 1  
## 22 Lora Gutierrez 1  
## 23 Charlotte Sparks 0  
## 24 Earl Mack 0  
## 25 Leslie Wade 4  
## 26 Santiago Barker 0  
## 27 Jose Bell 0  
## 28 Lynda Smith 0  
## 29 Bradford Marshall 0  
## 30 Irving Miller 0  
## 31 Caroline Simpson 0  
## 32 Frances Welch 0  
## 33 Melba Jenkins 0  
## 34 Veronica Morales 0  
## 35 Juanita Cunningham 0  
## 36 Maurice Howard 0  
## 37 Teri Pierce 0  
## 38 Phil Franklin 0  
## 39 Jan Zimmerman 0  
## 40 Leslie Price 0  
## 41 Bessie Patterson 0  
## 42 Ethel Wolfe 0  
## 43 Naomi Wright 1  
## 44 Sadie Frank 3  
## 45 Lonnie Cannon 3  
## 46 Tony Garcia 2  
## 47 Darla Newton 1  
## 48 Ginger Clark 1.5 - honestly I think one of my cats is half human  
## 49 Lionel Campbell 0  
## 50 Florence Klein 0  
## 51 Harriet Leonard 1  
## 52 Terrence Harrington 0  
## 53 Travis Garner 1  
## 54 Doug Bass three  
## 55 Pat Norris 1  
## 56 Dawn Young 1  
## 57 Shari Alvarez 1  
## 58 Tamara Robinson 0  
## 59 Megan Morgan 0  
## 60 Kara Obrien 2  
## handedness  
## 1 left  
## 2 left  
## 3 left  
## 4 left  
## 5 left  
## 6 left  
## 7 left  
## 8 left  
## 9 left  
## 10 left  
## 11 left  
## 12 left  
## 13 left  
## 14 right  
## 15 right  
## 16 right  
## 17 right  
## 18 right  
## 19 right  
## 20 right  
## 21 right  
## 22 right  
## 23 right  
## 24 right  
## 25 right  
## 26 right  
## 27 right  
## 28 right  
## 29 right  
## 30 right  
## 31 right  
## 32 right  
## 33 right  
## 34 right  
## 35 right  
## 36 right  
## 37 right  
## 38 right  
## 39 right  
## 40 right  
## 41 right  
## 42 right  
## 43 right  
## 44 right  
## 45 right  
## 46 right  
## 47 right  
## 48 right  
## 49 right  
## 50 right  
## 51 right  
## 52 right  
## 53 right  
## 54 right  
## 55 right  
## 56 ambidextrous  
## 57 ambidextrous  
## 58 ambidextrous  
## 59 ambidextrous  
## 60 ambidextrous

# Compute the mean of the number of cats: execute this command  
mean\_age <- mean(cat\_lovers$Age)

# Get more information about the mean() command using ? operator  
  
?mean()

# Convert the variable number\_of\_cats using as.integer()  
  
 cat\_lovers$number\_of\_cats

## [1] "0"   
## [2] "0"   
## [3] "1"   
## [4] "3"   
## [5] "3"   
## [6] "2"   
## [7] "1"   
## [8] "1"   
## [9] "0"   
## [10] "0"   
## [11] "0"   
## [12] "0"   
## [13] "1"   
## [14] "3"   
## [15] "3"   
## [16] "2"   
## [17] "1"   
## [18] "1"   
## [19] "0"   
## [20] "0"   
## [21] "1"   
## [22] "1"   
## [23] "0"   
## [24] "0"   
## [25] "4"   
## [26] "0"   
## [27] "0"   
## [28] "0"   
## [29] "0"   
## [30] "0"   
## [31] "0"   
## [32] "0"   
## [33] "0"   
## [34] "0"   
## [35] "0"   
## [36] "0"   
## [37] "0"   
## [38] "0"   
## [39] "0"   
## [40] "0"   
## [41] "0"   
## [42] "0"   
## [43] "1"   
## [44] "3"   
## [45] "3"   
## [46] "2"   
## [47] "1"   
## [48] "1.5 - honestly I think one of my cats is half human"  
## [49] "0"   
## [50] "0"   
## [51] "1"   
## [52] "0"   
## [53] "1"   
## [54] "three"   
## [55] "1"   
## [56] "1"   
## [57] "1"   
## [58] "0"   
## [59] "0"   
## [60] "2"

# Display the elements of the column number\_of\_cats   
as.integer(cat\_lovers$number\_of\_cats)

# Display the elements of the column number\_of\_cats after converting it using as.numeric()  
as.numeric(cat\_lovers$number\_of\_cats)

### Create an empty vector

# Empty vector  
library(tidyverse)  
# Type of the empty vector  
typeof(x)

### Create vectors of type logical

# Method 1  
x<-vector("logical",length=5)  
# Display the contents of x  
print(x)  
# Display the type of x  
print(typeof(x))

# Method 2  
x<-logical(5)  
# Display the contents of x  
print(x)

## [1] FALSE FALSE FALSE FALSE FALSE

# Display the type of x  
print(typeof(x))

## [1] "logical"

# Method 3  
x<-c(TRUE,FALSE,TRUE,FALSE,TRUE)  
# Display the contents of x  
print(x)  
# Display the type of x  
print(typeof(x))

### Create vectors of type character

# Method 1  
  
# Display the contents of x  
print(x)  
# Display the type of x  
print(typeof(x))

# Method 2  
  
# Display the contents of x  
print(x)  
# Display the type of x

# Method 3  
  
# Display the contents of x  
  
# Display the type of x

### Create vectors of type integer

# Method 1  
  
# Display the contents of x  
  
# Display the type of x  
print(typeof(x))

# Method 2  
  
# Display the contents of x  
print(x)  
# Display the type of x

# Method 3  
  
# Display the contents of x  
  
# Display the type of x

# Method 4  
  
# Display the contents of x  
  
# Display the type of x

# Method 5  
  
# Display the contents of x  
  
# Display the type of x

### Create vectors of type double

# Method 1  
  
# Display the contents of x  
  
# Display the type of x

# Method 2  
  
# Display the contents of x  
  
# Display the type of x

# Method 3  
  
# Display the contents of x  
  
# Display the type of x

### Implicit coercion

#### Example 1

# Create a vector  
  
# Check the type of x

# Add a character to the vector  
  
# Check the type of x

#### Example 2

# Create a vector  
  
# Check the type of x

# Add a number to the vector  
  
# Check the type of x

#### Example 3

# Create a vector  
  
# Check the type of x

# Add a logical value to the vector  
  
# Check the type of x

#### Example 4

# Create a vector  
  
# Check the type of x

# Add a number to the vector  
  
# Check the type of x

### Explicit coercion

#### Example 1

# Create a vector  
  
# Check the type of x

# Convert the vector to type character  
  
# Check the type of x

#### Example 2

# Create a vector  
  
# Check the type of x

# Convert the vector to type double  
  
# Check the type of x

### Accessing elements of the vector

# Create a vector  
x <- c(1,10,9,8,1,3,5)

# Access one element with index 3

# Access elements with consecutive indices, 2 to 4: 2,3,4

# Access elements with non-consecutive indices, 1,3,5

# Access elements using logical vector  
x[c(TRUE,FALSE,FALSE,TRUE,FALSE,FALSE,TRUE)]

# Access elements using the conditional operator <

### Examining vectors

# Display the length of the vector  
print(length(x))  
# Display the type of the vector  
print(typeof(x))  
# Display the structure of the vector  
print(str(x))

### Lists

# Initialise a named list  
my\_pie = list(type="key lime", diameter=7, is.vegetarian=TRUE)  
# display the list  
my\_pie

# Print the names of the list

# Retrieve the element named type

# Retrieve a truncated list

# Retrieve the element named type

#### Exploring data-sets

# Install package  
install.packages("openintro")  
# Load the package  
library(openintro)  
# Load package  
library(tidyverse)

# Catch a glimpse of the data-set: see how the rows are stacked one below another  
glimpse(loans\_full\_schema)

# Selecting numeric variables  
loans <- loans\_full\_schema %>% # <-- pipe operator  
 select(paid\_total, term, interest\_rate,  
 annual\_income,paid\_late\_fees,debt\_to\_income)  
# View the columns stacked one below another  
glimpse(loans)

# Selecting categoric variables  
loans <- loans\_full\_schema %>%   
 select( ) # type the chosen columns as in the lecture slide  
# View the columns stacked one below another  
glimpse(loans)