R tidyverse course

# Day 1:

Today is 27/04/2020

library(tidyverse)

## -- Attaching packages -------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.0 v purrr 0.3.4  
## v tibble 3.0.1 v dplyr 0.8.5  
## v tidyr 1.0.2 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.5.0

## -- Conflicts ----------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

## 1. The Pipe " %>% "

Video: <https://www.youtube.com/watch?v=9yjhxvu-pDg&list=PLLxj8fULvXwGOf8uHlL4Tr62oXSB5k_in&index=3>

# First let's make some random data\_frame  
  
x = c(4, 7, 4, 2)  
y = 1:4  
data = data\_frame(x,y)

## Warning: `data\_frame()` is deprecated as of tibble 1.1.0.  
## Please use `tibble()` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_warnings()` to see where this warning was generated.

data

## # A tibble: 4 x 2  
## x y  
## <dbl> <int>  
## 1 4 1  
## 2 7 2  
## 3 4 3  
## 4 2 4

# Looking at the data "classicly":  
  
colSums(data)

## x y   
## 17 10

mean(colSums(data))

## [1] 13.5

# With the pipe:   
data %>% colSums() %>% mean()

## [1] 13.5

#It's just a bit cleaner. Looks even better like this:   
data %>%  
 colSums() %>%  
 mean()

## [1] 13.5

### Keyboard shortcut for "%>%" is cntrl+shift+m

## 2. Vectors, dataframes and lists

Video: <https://www.youtube.com/watch?v=XHAm_V-KZE8&list=PLLxj8fULvXwGOf8uHlL4Tr62oXSB5k_in&index=4>

library(tidyverse)  
mtcars

## mpg cyl disp hp drat wt qsec vs am gear carb  
## Mazda RX4 21.0 6 160.0 110 3.90 2.620 16.46 0 1 4 4  
## Mazda RX4 Wag 21.0 6 160.0 110 3.90 2.875 17.02 0 1 4 4  
## Datsun 710 22.8 4 108.0 93 3.85 2.320 18.61 1 1 4 1  
## Hornet 4 Drive 21.4 6 258.0 110 3.08 3.215 19.44 1 0 3 1  
## Hornet Sportabout 18.7 8 360.0 175 3.15 3.440 17.02 0 0 3 2  
## Valiant 18.1 6 225.0 105 2.76 3.460 20.22 1 0 3 1  
## Duster 360 14.3 8 360.0 245 3.21 3.570 15.84 0 0 3 4  
## Merc 240D 24.4 4 146.7 62 3.69 3.190 20.00 1 0 4 2  
## Merc 230 22.8 4 140.8 95 3.92 3.150 22.90 1 0 4 2  
## Merc 280 19.2 6 167.6 123 3.92 3.440 18.30 1 0 4 4  
## Merc 280C 17.8 6 167.6 123 3.92 3.440 18.90 1 0 4 4  
## Merc 450SE 16.4 8 275.8 180 3.07 4.070 17.40 0 0 3 3  
## Merc 450SL 17.3 8 275.8 180 3.07 3.730 17.60 0 0 3 3  
## Merc 450SLC 15.2 8 275.8 180 3.07 3.780 18.00 0 0 3 3  
## Cadillac Fleetwood 10.4 8 472.0 205 2.93 5.250 17.98 0 0 3 4  
## Lincoln Continental 10.4 8 460.0 215 3.00 5.424 17.82 0 0 3 4  
## Chrysler Imperial 14.7 8 440.0 230 3.23 5.345 17.42 0 0 3 4  
## Fiat 128 32.4 4 78.7 66 4.08 2.200 19.47 1 1 4 1  
## Honda Civic 30.4 4 75.7 52 4.93 1.615 18.52 1 1 4 2  
## Toyota Corolla 33.9 4 71.1 65 4.22 1.835 19.90 1 1 4 1  
## Toyota Corona 21.5 4 120.1 97 3.70 2.465 20.01 1 0 3 1  
## Dodge Challenger 15.5 8 318.0 150 2.76 3.520 16.87 0 0 3 2  
## AMC Javelin 15.2 8 304.0 150 3.15 3.435 17.30 0 0 3 2  
## Camaro Z28 13.3 8 350.0 245 3.73 3.840 15.41 0 0 3 4  
## Pontiac Firebird 19.2 8 400.0 175 3.08 3.845 17.05 0 0 3 2  
## Fiat X1-9 27.3 4 79.0 66 4.08 1.935 18.90 1 1 4 1  
## Porsche 914-2 26.0 4 120.3 91 4.43 2.140 16.70 0 1 5 2  
## Lotus Europa 30.4 4 95.1 113 3.77 1.513 16.90 1 1 5 2  
## Ford Pantera L 15.8 8 351.0 264 4.22 3.170 14.50 0 1 5 4  
## Ferrari Dino 19.7 6 145.0 175 3.62 2.770 15.50 0 1 5 6  
## Maserati Bora 15.0 8 301.0 335 3.54 3.570 14.60 0 1 5 8  
## Volvo 142E 21.4 4 121.0 109 4.11 2.780 18.60 1 1 4 2

class(mtcars)

## [1] "data.frame"

# a dataframe is   
  
# Create a datafram from vectors  
x = c(4, 7, 4, 2)  
y = 1:4  
data = data\_frame(x,y)  
  
data

## # A tibble: 4 x 2  
## x y  
## <dbl> <int>  
## 1 4 1  
## 2 7 2  
## 3 4 3  
## 4 2 4

data$x

## [1] 4 7 4 2

# if the vectors are not the same size, you will run into problems.  
# Instead, you can use a list  
  
w = c(4, 5, 2,1)  
z = 1:10  
  
my\_list = list(w,z)  
  
my\_list

## [[1]]  
## [1] 4 5 2 1  
##   
## [[2]]  
## [1] 1 2 3 4 5 6 7 8 9 10

## 3. Working Directories

video: <https://www.youtube.com/watch?v=lWe7sMmynJk&list=PLLxj8fULvXwGOf8uHlL4Tr62oXSB5k_in&index=5>

A working directory is where you pull your data from and export your stuff to.

You can do that from the top-bar: session %>% set working directory %>% Choose directoy

# To get the working directory:  
getwd()

## [1] "C:/Users/anakar/Desktop/Scratchboard/Tidyverse Course"

# To change the working directory:   
wd.path = "C:/Users/anakar/Desktop/Scratchboard/Tidyverse Course/"  
setwd(wd.path)

## 4. Saving a loading data

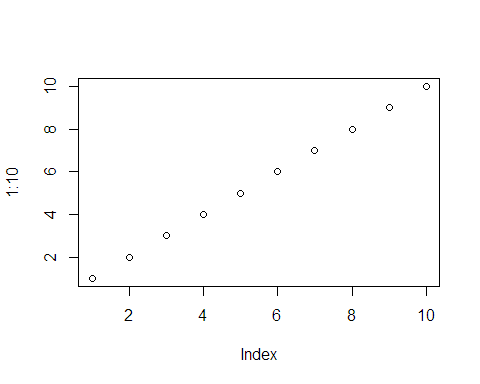
Note: I skipped the part about csv files My notes start here <https://youtu.be/Y3Nzz6WEl_E?t=200>

summary: 1. To read/write data: write.csv / read.csv 2. To read/write R objects: saveRDS() / readRDS() 3. To save/load R environments: save.image() / load()

# Saving dataframes as csvs:   
data = mtcars  
  
class(data)

## [1] "data.frame"

write.csv(data, "mtcars.csv")  
### This just made a new file in my wd called "mtcars.csv"  
  
#Now let's read it:   
import.data = read.csv("mtcars.csv")  
  
# But what if your data is NOT a dataframe?  
# You can save it as an R readable file: .rds  
  
Plot = plot(1:10)



saveRDS(Plot, "plot.rds")  
  
import.rds = readRDS("plot.rds")  
  
### How to save everything, all the stuff in your environment  
  
save.image("tidyverse.Rdata")  
  
load("tidyverse.Rdata")