



R para Finanzas

Guía de Ejercicios 3

Profesor: *Víctor Macías E.*

```
library(readr)      # Pregunta 1 (si se usa Forma 2) y pregunta 5
library(readxl)     # Pregunta 2
library(tidyquant)   # Preguntas 3 y 4
library(lubridate)   # Preguntas 3 y 4
library(WDI)         # Pregunta 5
library(dplyr)       # Todas las preguntas
```

Pregunta 1

Importa datos de los informes financieros anuales de Amazon desde el siguiente sitio:

<https://github.com/rfordatascience/tidytuesday/blob/main/data/2025/2025-03-25/readme.md>

La fuente de estos datos es la siguiente página de Amazon:

<https://ir.aboutamazon.com/annual-reports-proxies-and-shareholder-letters/>

¿Qué variables tiene la base? ¿Qué representa cada variable? ¿Qué años incluye la data?

```
# Forma 1
report_words_clean <-
  readr::read_csv("https://raw.githubusercontent.com/rfordatascience/
                  tidytuesday/main/data/2025/2025-03-25/report_words_clean.csv")
```

```
# Forma 2
report_words_clean2 <-
  read_csv("https://raw.githubusercontent.com/rfordatascience/
            tidytuesday/main/data/2025/2025-03-25/report_words_clean.csv")
```

```
# Número de observaciones por año
report_words_clean |> group_by(year) |> count()
```

Pregunta 2

Importa datos de las centrales generadoras en Chile de la siguiente página del Coordinador Eléctrico Nacional:

<https://infotecnica.coordinador.cl/instalaciones/centrales>

Construye una tabla de frecuencias con el número de centrales por tipo (solares, eólicas, hidroeléctricas, termoeléctricas, etc.) ¿En qué comunas se ubican las centrales termoeléctricas a carbón?

```
centrales <- read_xlsx("datos/reporte_centrales.xlsx", skip = 6)
```

```
# Centrales generadoras por tipo
centrales |> count(`Tipo Central`, sort = TRUE)
```

```
# Renombrando variables
centrales_1 <- centrales |>
  rename(tipo_central = `Tipo Central`,
         tipo_combustible = `11.1.35 Combustible (solo para termoeléctricas)`)
```

```
# Centrales a carbón
centrales_1 |>
  filter(tipo_central == "Termoeléctricas",
         tipo_combustible %in% c("Carbón", "GN / GNL / Carbón / Diesel")) |>
  select(Nombre, `Nombre Propietario`, Comuna, tipo_combustible)
```

Pregunta 3

Baja datos del precio de la acción de Shell (SHEL), BP (BP), Exxon Mobil Corporation (XOM) y Chevron Corporation (CVX) para el período 02/01/2018 hasta el 25/03/2025 desde Yahoo! Finance.

```
stock_prices <- tq_get(c("SHEL", "BP", "XOM", "CVX"),
  get = "stock.prices",
  from = "2018-01-02", to = "2025-03-26")
```

Después de obtener los datos, responde las siguientes preguntas:

- (a) ¿Cuántas observaciones hay en la base de datos? ¿Qué variables se encuentran disponibles? ¿Cuál es el tipo de cada variable?

```
stock_prices |> glimpse()
```

- (b) ¿Cuántas observaciones hay para cada acción?

```
stock_prices |> group_by(symbol) |> count()
```

- (c) Calcula el retorno diario de la acción, usando las siguientes fórmulas:

$$(1) \quad r_t = \frac{p_t - p_{t-1}}{p_{t-1}} \times 100 \quad (1)$$

$$(2) \quad r_t = 100 \times (\log(p_t) - \log(p_{t-1})) \quad (2)$$

donde p_t es el precio de cierre de la acción. ¿Qué diferencias se observan en los retornos calculados usando las fórmulas (1) y (2)?

```
stock_prices <- stock_prices |> group_by() |>
  mutate(retorno_aritmetico = 100*(close - lag(close))/lag(close),
         retorno_logaritmico = 100*(log(close)-log(lag(close)))) |>
  ungroup()
```

- (d) ¿Qué día de la semana se alcanzó el mayor retorno diario? ¿Qué día de la semana se alcanzó el menor retorno diario? ¿Cuál fue el precio de cierre los días anteriores a este aumento?

```
# Días con mayor variación positiva
stock_prices |> group_by(symbol) |>
  slice_max(retorno_aritmetico, n = 1) |>
  select(symbol, date, close, retorno_aritmetico) |>
  ungroup() |>
  mutate(dia_semana = wday(date, label = TRUE, abbr = FALSE))
```

```
# Precios días anteriores a la variación positiva más alta
stock_prices |> filter(date >= ymd("2020-03-20") & date <= ymd("2020-03-24")) |>
  select(symbol, date, close, retorno_aritmetico)
```

```
# Días con mayores pérdidas
stock_prices |> group_by(symbol) |>
  slice_min(retorno_aritmetico, n = 1) |>
  select(symbol, date, close, retorno_aritmetico) |>
  ungroup() |>
  mutate(dia_semana = wday(date, label = TRUE, abbr = FALSE))
```

Pregunta 4

Obtenga datos del precio del bitcoin para el período 02/01/2018 hasta el 25/03/2025 desde la página web de la Federal Reserve Bank of St. Louis (FRED) (<https://fred.stlouisfed.org/series/CBBTCUSD>) ¿Cuál son los mayores 3 precios que alcanzó el bitcoin en el período analizado? ¿Cuáles son los tres más bajos? ¿Qué día de la semana se alcanzaron los máximos y mínimos?

```
bitcoin <- tq_get("CBBTCUSD", get = "economic.data",
  from = "2018-01-02", to = "2025-03-26")
```

```
# Precios máximos
bitcoin |>
  slice_max(price, n = 3) |>
  mutate(dia_semana = wday(date, label = TRUE, abbr = FALSE))
```

```
# Precios mínimos
bitcoin |>
  slice_min(price, n = 3) |>
  mutate(dia_semana = wday(date, label = TRUE, abbr = FALSE))
```

Pregunta 5

Usando el paquete WDI, obtenga datos de las siguientes variables de World Development Indicators del Banco Mundial para el período 1970-2023:

```

# GDP per capita, PPP (constant 2021 international $)
# NY.GDP.PCAP.PP.KD

# Population
# SP.POP.TOTL

# Carbon dioxide (CO2) emissions (total) excluding LULUCF (Mt CO2e)
# EN.GHG.CO2.MT.CE.AR5

# Total greenhouse gas emissions excluding LULUCF (Mt CO2e)
# EN.GHG.ALL.MT.CE.AR5

# Carbon dioxide (CO2) emissions excluding LULUCF per capita (tCO2e/capita)
# EN.GHG.CO2.PC.CE.AR5

# Total greenhouse gas emissions excluding LULUCF per capita (tCO2e/capita)
# EN.GHG.ALL.PC.CE.AR5

# Total greenhouse gas emissions including LULUCF (Mt CO2e)
# EN.GHG.ALL.LU.MT.CE.AR5

```

```

wbData <- WDI(indicator=c("NY.GDP.PCAP.PP.CD",
                          "SP.POP.TOTL",
                          "EN.GHG.CO2.MT.CE.AR5",
                          "EN.GHG.ALL.MT.CE.AR5",
                          "EN.GHG.CO2.PC.CE.AR5",
                          "EN.GHG.ALL.PC.CE.AR5",
                          "EN.GHG.ALL.LU.MT.CE.AR5"),
              country="all", extra = TRUE, start=1970, end=2023)

```

```

write_rds(wbData, "datos/wbData_1970_2023.rds")

```