

NOMBRE COMPLETO: Ing. Hilaría Adima Vásquez Durán

NAO ID: 3033

FECHA: 2 de agosto de 2024

NOMBRE DE LA TRAYECTORIA

EN LA QUE ESTÁS ENROLADO: DATA ANALYST CORE

Título del Reto:

Programación R al elaborar reportes estadísticos

SPRINT 1 DESARROLLO R1

Crea un conjunto de programas en R para explorar los datos de Rossmann Pharma a través de tablas con resúmenes estadísticos y visualizaciones que cumplan con las siguientes directrices:

EJERCICIO 1

Desarrolla un programa ("prop_sales_by_country_by_year_month.R") que calcule la proporción que han representado las ventas de los países a lo largo de los meses de 2013 a 2015 (Ejemplo: Enero 2013, Italia 10%, Francia 60%, Italia, 30%, Febrero 2013) Italia 15%, Francia 52%, Italia, 33%. El resultado de este script deberá ser una tabla denominada "prop_sales_by_country_by_year_month.csv".

Desarrollo del ejercicio

Carga las bibliotecas necesarias para la manipulación de datos (tidyverse), la lectura de archivos Excel (readxl) y la manipulación de fechas (lubridate).

```
` ``{r}
library(tidyverse)
library(readxl)
library(lubridate)
` ``
```

```

> library(tidyverse)
— Attaching core tidyverse packages — tidyverse 2.0.0 —
✓ dplyr      1.1.4    ✓ readr      2.1.5
✓ forcats    1.0.0    ✓ stringr    1.5.1
✓ ggplot2    3.5.1    ✓ tibble     3.2.1
✓ lubridate  1.9.3    ✓ tidyr      1.3.1
✓ 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
> library(readxl)
> library(lubridate)

```

Establecer el directorio de trabajo a la ruta especificada en:

`DATA_PATH`

```

DATA_PATH <- "/Users/Usuario/Desktop/Help ME"
setwd(DATA_PATH)

```

Cargar los datos correspondientes de para el desarrollo

```

SALES_DATA <- 'sales.csv'
sales <- read.csv2(SALES_DATA, sep=";", header=T)

STORES_INFO_DATA <- 'stores_info.xlsx'
stores <- read_excel(STORES_INFO_DATA, sheet = "data")

```

```

>
> DATA_PATH <- "/Users/Usuario/Desktop/Help ME"
> setwd(DATA_PATH)
> SALES_DATA <- 'sales.csv'
> sales <- read.csv2(SALES_DATA, sep=";", header=T)
> STORES_INFO_DATA <- 'stores_info.xlsx'
> stores <- read_excel(STORES_INFO_DATA, sheet = "data")
>

```

Unión de los datos de ventas (`sales`) con la información de las tiendas (`stores`) usando la columna `store` como clave.

```
sales_all <- sales %>% left_join(stores, by=c("store"))
```

Convierte la columna `date` a un objeto de fecha, agrega columnas para el año y el mes, y filtra los datos para los años 2013 a 2015.

```
sales_all$date <- ymd(sales_all$date) sales_all  
<- sales_all %>% mutate(year = year(date),  
month = month(date)) %>% filter(year >= 2013  
& year <= 2015)
```

Agrupar los datos por país, año y mes, y se calcula la suma total de ventas para cada grupo.

```
total_sales_monthly <- sales_all %>%  
  group_by(year, month) %>%  
  summarise(total_sales_all_countries = sum(sales), .groups =
```

```
> sales_all <- sales %>% left_join(stores, by=c("store"))  
> sales_all$date <- ymd(sales_all$date)  
> sales_all <- sales_all %>%  
+   mutate(year = year(date), month = month(date)) %>%  
+   filter(year >= 2013 & year <= 2015)  
> monthly_sales <- sales_all %>%  
+   group_by(country, year, month) %>%  
+   summarise(total_sales = sum(sales), .groups = 'drop')  
> total_sales_monthly <- sales_all %>%  
+   group_by(year, month) %>%  
+   summarise(total_sales_all_countries = sum(sales), .groups = 'drop')
```

Unión de los datos de ventas mensuales (`monthly_sales`) con las ventas
totales mensuales (`total_sales_monthly`), y se

calcula la proporción de las ventas totales de cada país con respecto a las ventas
totales de todos los países en cada mes. Finalmente, se seleccionan las

columnas
country, year, month, proportion

```
prop_sales <- monthly_sales  
  left_join(total_sales_monthly, by = c("year", "month"))  
  mutate(proportion = total_sales /  
    total_sales_all_countries select(country, year, month,  
    proportion)
```

```
> prop_sales <- monthly_sales %>%  
+ left_join(total_sales_monthly, by = c("year", "month")) %>%  
+ mutate(proportion = total_sales / total_sales_all_countries * 100) %>%  
+ select(country, year, month, proportion)
```

Mostrar resultados en consola

```
print(prop_sales)
```

```
> print(prop_sales)  
# A tibble: 93 × 4  
  country year month proportion  
  <chr>   <dbl> <dbl>      <dbl>  
1 france  2013     1      52.5  
2 france  2013     2      52.0  
3 france  2013     3      51.4  
4 france  2013     4      52.0  
5 france  2013     5      51.4  
6 france  2013     6      51.2  
7 france  2013     7      51.1  
8 france  2013     8      50.2  
9 france  2013     9      50.9  
10 france 2013    10      51.2  
# i 83 more rows  
# i Use `print(n = ...)` to see more rows  
>
```

Crea un archivo llamado `prop_sales_by_country_by_year_month.csv`
con los datos que sacamos de `prop_sales`

```
write.csv(prop_sales,  
"prop_sales_by_country_by_year_month.csv"
```

```
> write.csv(prop_sales, "prop_sales_by_country_by_year_month.csv", row.names = $  
> cat("La tabla 'prop_sales_by_country_by_year_month.csv' ha sido creada exito$  
La tabla 'prop_sales_by_country_by_year_month.csv' ha sido creada exitosamente.  
> "
```

