Introducción a R

Agosto 2022

Material de trabajo

Carpeta de drive:

https://drive.google.com/drive/u/0/folders/14QDyRnqnLUjosCBlAzBqXL-qglOMjNJL

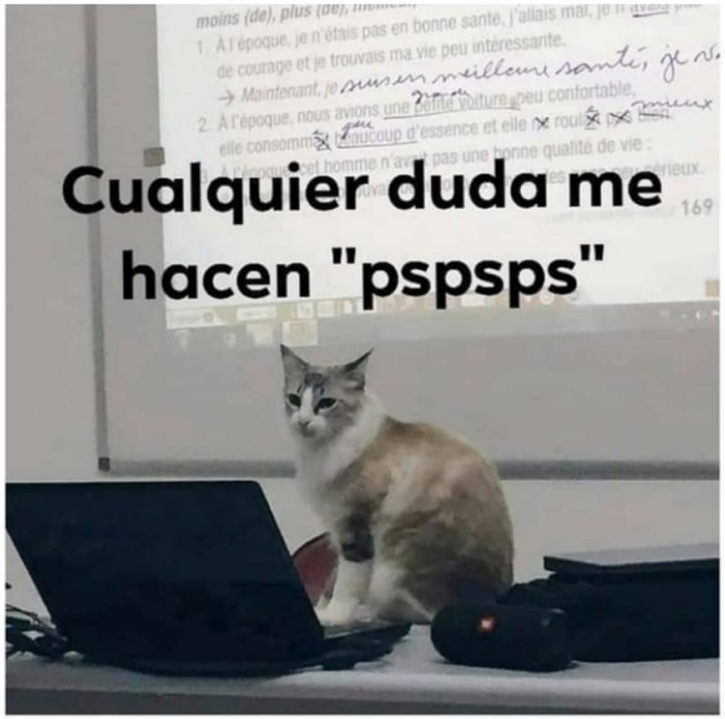
Base a descargar ile_completa

Liga de zoom https://us02web.zoom.us/j/83841692580

ID de reunión: 838 4169 2580

Antes de iniciar

Siéntanse libre de preguntar.



Índice

- Introducción a R y Rstudio.
- tidyverse.
- Interfas de Rstudio.
- Generar un nuevo proyecto.
- Establecer directorio de trabajo.
- Tipo y estructura de datos
- Funciones básicas.
- Importar archivos en R.

R y R studio

R **es un entorno y lenguaje de programación** con un enfoque al análisis estadístico. R nació como una reimplementación de software libre del lenguaje S, adicionado con soporte para ámbito estático.

El lenguaje está compuesto por **símbolos y reglas sintácticas y semánticas, expresadas en forma de instrucciones y relaciones lógicas**, mediante las cuales se construye el código fuente de una aplicación o pieza de software determinado.

RStudio es un **entorno de desarrollo integrado** para el lenguaje de programación R, dedicado a la computación estadística y gráficos.

Ambos especializado en análisis estadístico y visualización de datos.

En resumen

- Software libre.
- Lenguaje de programación.
- Interfaz de una consola

- Software libre.
- Es un programa para manejar R (IDE).
- Interfaz de varios paneles.

O más claro :)

¿Por qué utilizar Rstudio?

- Es un lenguaje bastante adecuado para la estadística, ya que permite manipular los datos rápidamente y de forma precisa.

- Se puede automatizar fácilmente, gracias a la creación de scripts que automatizan procesos, por ejemplo, leer datos o hacer operaciones con los datos, y hacerlo siempre de forma automática.

- Puede leer prácticamente cualquier tipo de datos.

- Hasta cierto punto, es compatible con grandes conjuntos de datos.

- Es gratuito.

- Tiene capacidades avanzadas de gráficos, por lo que nos permite realizar gráficos y dashboards de forma que podamos presentar los resultados de forma vistosa.

- Se ejecuta en muchas plataformas.

Tidyverse

Tidyverse es un **conjunto de paquetes en R** diseñados para ciencia de datos. Ayuda en todo el proceso de importar transformar visualizar modela y comunicar toda la información que normalmente utilizamos en procesos de ciencia de datos.

Ventajas

- Facilita el análisis y manipulación de datos y más rápido.

- Comparten estructura y nombre comunes.

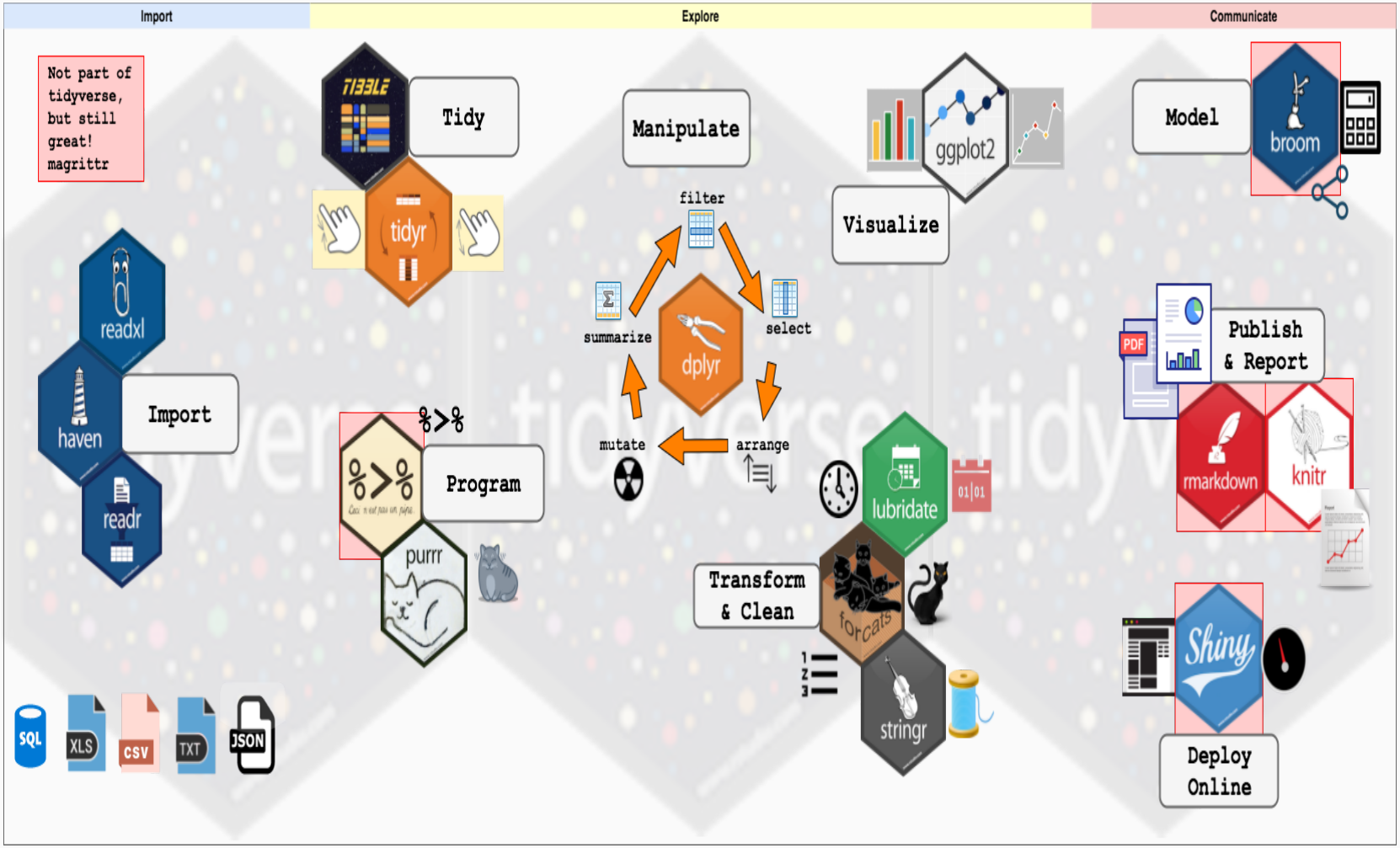
- Sixtaxis, nombre y característica similares compatible entre paquetes.

Desventaja

- Deja de lado la forma usual de programación.

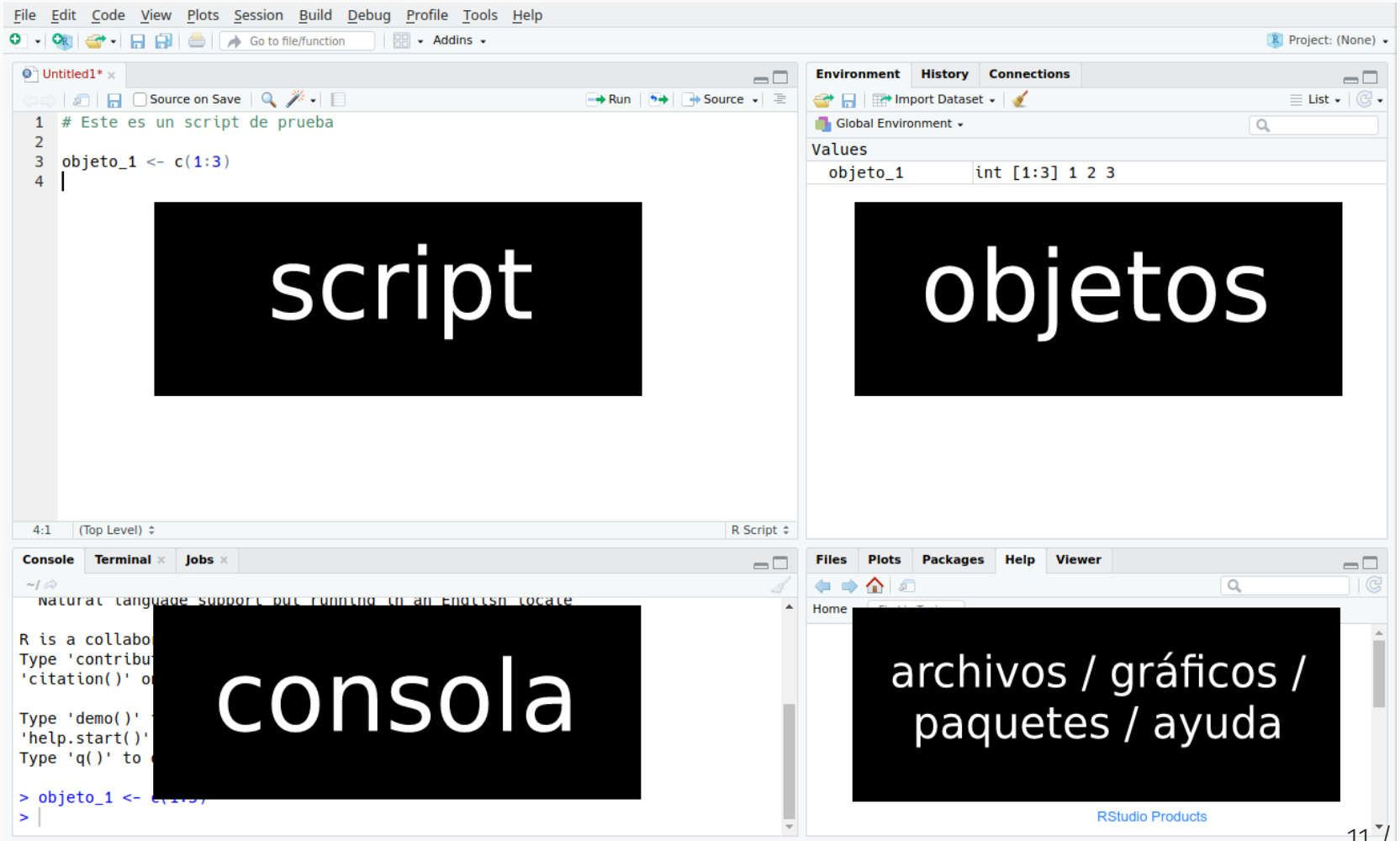
- Uso de pipes %>%.

Tidyverse


Import
Explore
Communicate
Not part of tidyverse, but still great! magrittr
Tidy
tidyr
Manipulate
filter
summarize
select
mutate
arrange
dplyr
Visualize
ggplot2
Model
broom
readxl
haven
readr
Import
%>%
Program
purrr
lubridate
Transform & Clean
forcats
stringr
Publish & Report
PDF
rmarkdown
knitr
Shiny
Deploy Online
SQL
XLS
CSV
TXT
JSON

Interfaz de Rstudio

RStudio está dividido en cuatro paneles:



Paneles

RStudio está dividido en cuatro paneles, que se presentarán a continuación. Vayamos a Rstudio y familiaremonos con estos componentes.

- **Script:** en este panel vas a escribir, editar, ver los R script y los datasets.

- **Consola:** también se conoce como terminal, aquí se ejecutan los comandos redactados en el script.

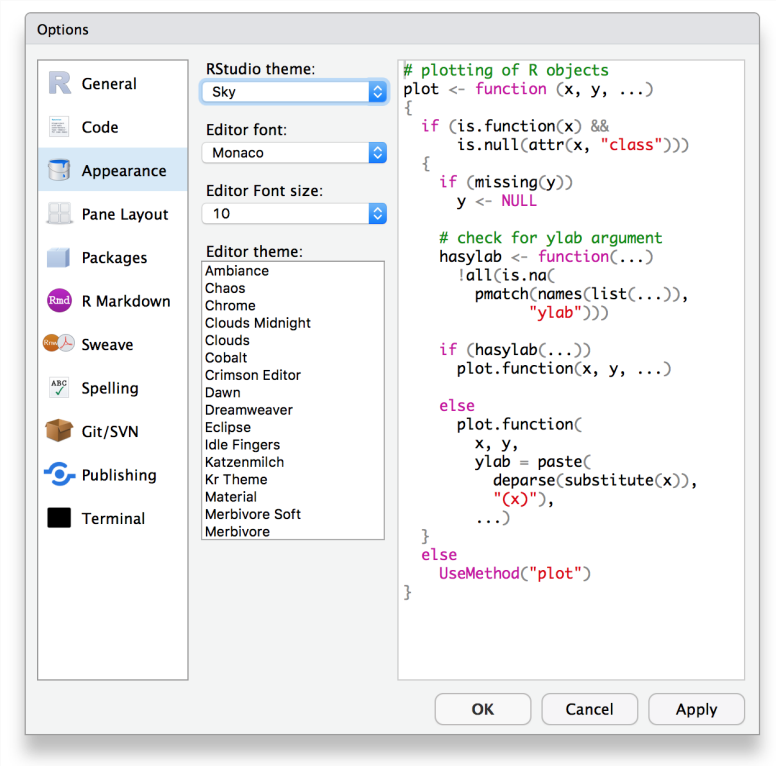
- **Objeto / Environment:** te muestra qué datasets y qué objetos (variables) que has creadas en la memoria.

- **Archivo, gráficos, paqueterias y ayuda**: es un panel multipropósito que devuelve información solicitada.

Recomendación:

Explora los subpaneles **Files**, **Plots**, **Packages** y **Help** observa que elementos por default Rstudio muestra.

Una vez conocido los paneles, puedes personalizar tu interfaz de R, selecciona el menú de **"Tool"**, después clic en **"Global options"** y clic **"Appearance"**.



Proyectos en Rstudio

Los **proyectos** hacen que sea más fácil dividir el trabajo en múltiples contextos, cada uno con su propio directorio de trabajo, espacio de trabajo, historial y los documentos de origen. Los proyectos de RStudio están asociados a los directorios de trabajo. Al trabajar en Rstudio se recomienda tener un orden de los elementos que crearás, que importarás y exportarás por eso la importancia de contar con un proyecto.

Crear un proyecto nuevo

- Hacer clic en el menú **Archivo**, luego en **Nuevo proyecto**.
- Hacer clic en **Nuevo directorio** .
- Hacer clic en **Nuevo proyecto**.
- Introducir el nombre del directorio para guardar tu proyecto, por ejemplo: **feminismo_datos**.

Directorio de trabajo

El directorio o carpeta de trabajo es el lugar en nuestra computadora en el que se encuentran los archivos con los que estamos trabajando en R. Este es el lugar donde R buscara archivos para importarlos y al que serán exportados, a menos que indiquemos otra cosa.

Puedes encontrar cuál es tu directorio de trabajo con la función getwd(). Sólo tienes que escribir la función en la consola y ejecutarla.

Para ejecutar parte del código: presiona **Ctrl + Enter**

```
#Muestra la ruta o path de mi directorio
getwd()
```

```
## [1] "C:/Users/Paola Viridiana/Downloads"
```

Tambien, puedes cambiar el directorio de trabajo usando la función setwd(), dando como argumento la ruta del directorio que quieres usar.

```
#setwd("~/Desktop/Taller/tidyverse/taller_tidy")
```

Antes de iniciar

Debes conocer algunas herramientas en R.

Cometarios

En R puedes **añadir** comentarios en tu script, lo que esta junto al **#** es comenario.

```
# Este es un comentario y no interfiere en los comandos de R.
# Son anotaciones para nosotres.
```

¿Cómo ejecutar comando o líneas?

presiona **Ctrl + Enter** | Seleccionar y dar clic en la viñeta de **Run** el cual se ejecutará en el panel de la consola.

Asignación de variables

Una variable puede almacenar un objeto.

```
# Mi variable x tiene un objeto de valor de 1 | tres manera de asignar nombre
x ← 1
x = 1
1→ x
```

Tipo y estructura de datos

Tipo numérico:

```
#Números decimales
x←1.0
1.0 + 2.0
```

```
## [1] 3
```

Tipo enteros o integer:

```
#Números enteros
int ← 1
```

Tipo character:

```
#    Cadenas de texto
texto←"Hello, world!"
print(texto)
```

```
## [1] "Hello, world!"
```

Tipo y estructura de datos

Tipo factor:

```
#Una variable factor es una variable categórica. Los vectores de caracteres a menudo
#se almacenan como factores para explotar funciones para tratar datos categóricos
"Femenino, Masculino, Otro"
```

```
## [1] "Femenino, Masculino, Otro"
```

Tipo logical:

```
# Verdadero (TRUE) o falso (FALSE). Es a menudo el resultado de operaciones lógicas.
a ← 1
b ← 2
a < b
```

```
## [1] TRUE
```

Tipo de datos

NA y NULL

En R, usamos **NA** para representar datos perdidos, mientras que **NULL** representa la ausencia de datos.

La diferencia entre las dos es que un dato NULL aparece sólo cuando R intenta recuperar un dato y no encuentra nada, mientras que NA es usado para representar explícitamente datos perdidos, omitidos o que por alguna razón son faltantes.

Por ejemplo, si tratamos de recuperar la edad de una persona encuestada que no existe, obtendríamos un NULL, pues no hay ningún dato que corresponda con ello. En cambio, si tratamos de recuperar su estado civil, y la persona encuestada no contestó esta pregunta, obtendríamos un NA.

NA además puede aparecer como resultado de una operación realizada, pero no tuvo éxito en su ejecución.

Validación de datos

Se puede verificar si un dato es de un tipo específico con la familia de funciones is()

Función	**Tipo que verifican**
is.integer()	Entero
is.numeric()	Numerico
is.character()	Cadena de texto
is.factor()	Factor
is.logical()	Lógico
is.na()	NA
is.null()	NULL

Además, con la función **class** podemos saber que tipo de dato es una variable.

Coerción de datos

También podemos hacer coerciones explícitas usando la familia de funciones as().

Función	Tipo al que hace coerción
as.integer()	Entero
as.numeric()	Numerico
as.character()	Cadena de texto
as.factor()	Factor
as.logical()	Lógico
as.na()	NA
as.null()	NULL

Vectores

- Elemento más básico en R.
- Se crea con la función **c()**, que significa 'concatenar' o 'combinar'.
- Un vector puede almacenar varios objetos diferentes en orden en r un vector es un espacio de memoria, estas últimas dos representan estructuras rectangulares de datos.

Creamos vectores con información de algunos estados de México.
Estados con mayor población en Mx:

```
abr_may ← c("EdoMex", "CDMX", "Ver", "Jal") #Vector numeric

entidad_may ← c("Estado México", "Ciudad de México", "Veracruz", "Jalisco") #Vector c

pob_mil_may ←c(17363387, 8811266, 8163963, 8110943)
```

Estados con menor población en Mx

```
abr_men ← c("Nay", "Camp", "BCS", "Col")

entidad_men ← c("Nayarit", "Campeche", "Baja Calfornia S", "Colima")

pob_mil_men ←c(1268460, 935047, 809833, 747801)
```

Funciones de vectores

```
length(entidad_may) #Longitud del vector
```

```
## [1] 4
```

```
entidad_may[2:3] #Extraer información del vector
```

```
## [1] "Ciudad de México" "Veracruz"
```

```
entidad_may[c(2:3)] #Extraer información del vector | Otra forma
```

```
## [1] "Ciudad de México" "Veracruz"
```

```
pob_mil_may - pob_mil_men #Transformar vectores
```

```
## [1] 16094927  7876219  7354130  7363142
```

```
is.vector(entidad_may) #Validar vector
```

```
## [1] TRUE
```

Matrices

Una matriz es una forma de acomodar los datos que tiene renglones o filas y columnas. Continuando con nuestro ejemplo.

```
matrix_ent_may ← matrix(c(abr_may, entidad_may, pob_mil_may),
                        nrow = 4,
                        ncol = 3)
matrix_ent_may
```

```
##      [,1]     [,2]              [,3]
## [1,] "EdoMex" "Estado México"   "17363387"
## [2,] "CDMX"   "Ciudad de México" "8811266"
## [3,] "Ver"    "Veracruz"        "8163963"
## [4,] "Jal"    "Jalisco"         "8110943"
```

```
colnames(matrix_ent_may)←c("Abreviatura", "Entidad", "Población")
rownames(matrix_ent_may)←c(1, 2, 3, 4)
matrix_ent_may
```

```
##   Abreviatura Entidad           Población
## 1 "EdoMex"    "Estado México"   "17363387"
## 2 "CDMX"      "Ciudad de México" "8811266"
## 3 "Ver"       "Veracruz"        "8163963"
```

Selección de elementos de una matriz

```
matrix_ent_may[4,] #Selección de una fila (pob)
```

```
## Abreviatura     Entidad   Población
##       "Jal"   "Jalisco"   "8110943"
```

```
matrix_ent_may[,3] #Selección de una fila (pob)
```

```
##          1          2          3          4
## "17363387"  "8811266"  "8163963"  "8110943"
```

```
matrix_ent_may [4,2] # Seleccionar un elemento en especifico
```

```
## [1] "Jalisco"
```

Hemos visto:

- **Variables:** espacio para guardar un objeto.
- **Vectores:** una o más variable del mismo tipo de datos.
- **Matrices:** varias columnas/vectores del mismo tipo de datos.
  ¿qué sigue?
- **Dataframes:** tabla o columna de diferente tipo de datos.

Dataframes:

En un dataframe podemos tener una columna con caracteres otra con números y otra con variables lógicas.

Añadimos columna TRUE - FALSE

```
ubicacion_may ←c(TRUE, T, F, F)
```

```
df_ent_may ← data.frame(abr_may,
                        entidad_may,
                        pob_mil_may,
                       ubicacion_may)
df_ent_may
```

```
##   abr_may      entidad_may pob_mil_may ubicacion_may
## 1  EdoMex    Estado México    17363387          TRUE
## 2    CDMX Ciudad de México     8811266          TRUE
## 3     Ver          Veracruz     8163963         FALSE
## 4     Jal           Jalisco     8110943         FALSE
```

Comenzamos a utilizar propiedades de los dataframes

```
names(df_ent_may) #Nombre de las columnas
```

```
## [1] "abr_may"       "entidad_may"   "pob_mil_may"   "ubicacion_may"
```

```
head(df_ent_may, 2) #Muestra el encabezado del df la "," después del objeto es el núme
```

```
##   abr_may      entidad_may pob_mil_may ubicacion_may
## 1  EdoMex    Estado México    17363387          TRUE
## 2    CDMX Ciudad de México     8811266          TRUE
```

```
#Muestra la parte final del df la "," después
#del objeto es el número de obsevaciones que deseas ver.

tail(df_ent_may, 1)
```

```
##   abr_may entidad_may pob_mil_may ubicacion_may
## 4     Jal     Jalisco     8110943         FALSE
```

```
dim(df_ent_may) #Muestra la dimension del df
```

```
## [1] 4 4
```

```
class(df_ent_may)
```

```
## [1] "data.frame"
```

La función summary() provee salidas para cada variable dependiendo del tipo de datos. Cuando los valores son numéricos, como en nuestro caso, summary() muestra el minimo, 1er cuartil, mediana, media mean, entre otroe. Para variables categóricas, muestra el número de veces que cada valor aparece en los datos (esto es llamado "level").

```
summary(df_ent_may) #Muestra la clase del df
```

```
##    abr_may          entidad_may         pob_mil_may       ubicacion_may
##  Length:4           Length:4           Min.   : 8110943   Mode :logical
##  Class :character   Class :character   1st Qu.: 8150708   FALSE:2
##  Mode  :character   Mode  :character   Median : 8487614   TRUE :2
##                                        Mean   :10612390
##                                        3rd Qu.:10949296
```

En resumen

Qué hemos visto:

- R y R studio .
  - Tipos y estructura de datos.
  - Extracción de información.

Hasta ahorita nos hemos introducido en dataframes, continuaremos con ellos y más de sus propiedades en complemento con otras paqueterías.

Paquetería Readxl

Readxl nos permite importar archivos de extensión xlxs / excel. Para esto necestamos dos cosas:

1. Tener la paquetería **readxl**.
2. La ruta o path del archivo excel(.xlxs)

Primero:
Instalamos la paquetería escribimos y ejecutamos:
install.packages("readxl") Después carga el paquete con

```
library(readxl)
```

Segundo:
Debemos ubicar donde se encuentra nuestro archivo excel en nuestro ordenador. Se suguire utilizar la función **file.choose**.

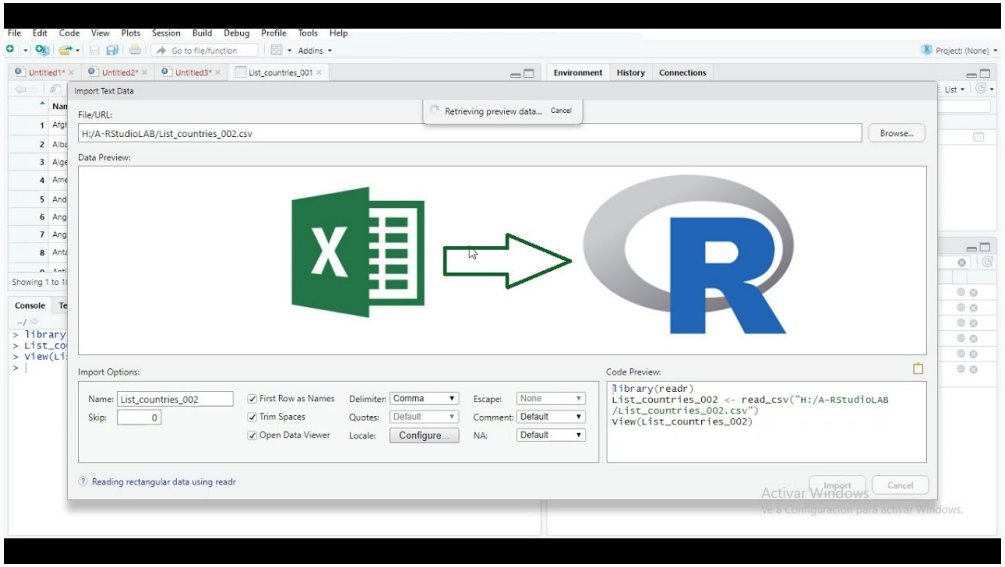
```
datos ← read_excel("~/ile_completa.xlsx")
```

La base importada se mostrará en el panel de **environment** con el nombre que se le asigno.

Paquetería Readxl (segunda opcion)

De manera manual se puede instalar y subir una archivo xlxs.

- Ir al panel de **environment**.
- Clic en **Import Dataset**, seleccionar **From Excel...**
  Se abrirá un cuadro de texto como el siguientte:



Debes llenar el espacio de URL/Ruta/File, **browse** es un buen complemento para encontrar el archivo a importar. En **data preview** se muestra la base. Por último en el cuadro de **Code preview** se mostrará el código que es lo que se presentó en anteriormente.

Exploración de datos

str() para conocer la estructura de los datos. Lo interesante de esta función es que combina varios elementos de otras funciones ya vistas.

```
str(datos)
```

```
## tibble [74,406 × 46] (S3: tbl_df/tbl/data.frame)
##  $ year                 : num [1:74406] 2016 2016 2016 2016 2016 ...
##  $ mes                  : chr [1:74406] "ENERO" "ENERO" "ENERO" "ENERO" ...
##  $ hospital             : chr [1:74406] "C.S.T III beatriz velasco de aleman" "C.S.T III
##  $ fingreso             : POSIXct[1:74406], format: "2016-01-07" "2016-01-07" ...
##  $ autoref              : chr [1:74406] "NA" "NA" "NA" "NA" ...
##  $ edocivil_descripcion: chr [1:74406] "soltera" "uniÃ³n libre" "soltera" "soltera" ...
##  $ edad                 : num [1:74406] 16 35 22 29 23 45 33 30 29 21 ...
##  $ desc_derechohab      : chr [1:74406] "ISSSTE" "ninguna" "ninguna" "IMSS" ...
##  $ nivel_edu            : chr [1:74406] "secundaria" "preparatoria o bachillerato" "prepa
##  $ ocupacion            : chr [1:74406] "estudiante" "empleado" "estudiante" "estudiante
##  $ religion             : chr [1:74406] "catÃ³lica" "ninguna religiÃ³n" "ninguna religiÃ
##  $ parentesco           : chr [1:74406] "NA" "NA" "NA" "NA" ...
##  $ entidad              : chr [1:74406] "distrito federal" "EDO.MEX" "EDO.MEX" "EDO.MEX"
##  $ del_o_municipio      : chr [1:74406] "gustavo a. madero" "tecÃ¡mac" "ecatepec de morel
##  $ menarca              : chr [1:74406] "12" "12" "10" "15" ...
##  $ fsexual              : chr [1:74406] "16" "19" "16" "15" ...
##  $ fmenstrua            : chr [1:74406] "42325" "42323" "42317" "42318" ...
```

Con summary nos devuelve una estadistica rapida del conjunto de datos, sí son variables númericas nos muestra métricas como; mínimo, máximo, quartiles, promedio y mediana. En las variables categóricas muestra el largo o número de observaciones la clase y la moda.

```
summary(datos)
```

```
##       year          mes              hospital
##  Min.   :2016   Length:74406       Length:74406
##  1st Qu.:2017   Class :character   Class :character
##  Median :2018   Mode  :character   Mode  :character
##  Mean   :2018
##  3rd Qu.:2019
##  Max.   :2020
##  NA's   :21
##     fingreso                       autoref          edocivil_descripcion
##  Min.   :1900-01-21 00:00:00.00   Length:74406       Length:74406
##  1st Qu.:2017-01-04 00:00:00.00   Class :character   Class :character
##  Median :2018-01-16 00:00:00.00   Mode  :character   Mode  :character
##  Mean   :2018-02-04 21:59:44.26
##  3rd Qu.:2019-03-13 00:00:00.00
##  Max.   :2048-04-30 00:00:00.00
##  NA's   :1605
```