



# Armando un proyecto de trabajo

Big Data e Inteligencia Territorial

---

# Hoja de ruta

- ✓ Paquetes
- ✓ Lectura / escritura de archivos
- ✓ Proyectos

# Paquetes

---

# Paquetes

Con R podemos aumentar las funciones para realizar operaciones a través de los **paquetes**.

# Paquetes

Para poder usar un paquete, primero debemos **instalarlo**:

```
`install.packages("nombre_del_paquete")`
```

- Tener en cuenta:
  - El nombre del paquete va **entre comillas**.
  - Esta función se corre una sóla vez (por computadora)

Cada vez que queremos utilizar una función del paquete, debemos **convocarlo**:

```
`library(nombre_del_paquete)`
```

- Tener en cuenta:
  - El nombre del paquete va **sin comillas**.
  - Se corre **cada vez** que quiero usar una función del paquete en cuestión.

# Proyectos y rutas de trabajo

---

# Proyectos y rutas de trabajo

Supongamos que descargo la base de datos de registros de certificados **CUIDAR-VERANO** en formato separado por comas (.csv) y la quiero importar con R.

la función que me permite importar una base de datos de tipo .csv se llama `read.csv()`:

```
base_cuidar <- read.csv(  
  file = "C:/Users/pablo/Documents/Pablo/Cursos R/R+BigData/entradas/certificados-personas-por-f  
  header = TRUE,  
  sep = ",",  
  dec = ".")
```

# Proyectos y rutas de trabajo

- A tener en cuenta respecto a la ruta donde se ubica el archivo (nuestra base):

```
base_cuidar <- read.csv(  
  file = "C:/Users/pablo/Documents/Pablo/Cursos R/R+BigData/entradas/certificados-personas-por-f  
  header = TRUE,  
  sep = ",",  
  dec = ".")
```

- Si se la compartimos a otra persona, **se rompe**
- Si cambiamos de computadora, **se rompe**
- Si lo cambiamos de lugar, **se rompe**



# Proyectos y rutas de trabajo

Armando un proyecto con Rstudio, la función se escribiría de esta forma:

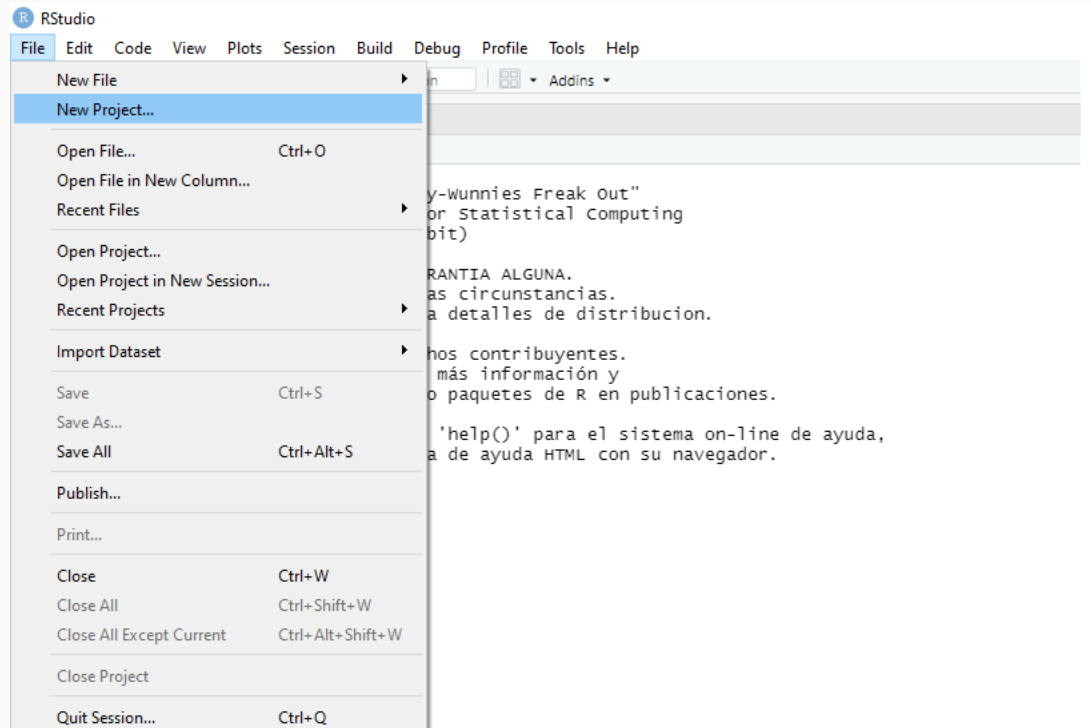
```
base_cuidar <- read.csv(  
  file = "entradas/certificados-personas-por-fecha-ingreso-provincia-localidad.csv",  
  header = TRUE,  
  sep = ",",  
  dec = ".")
```

## Ventajas:

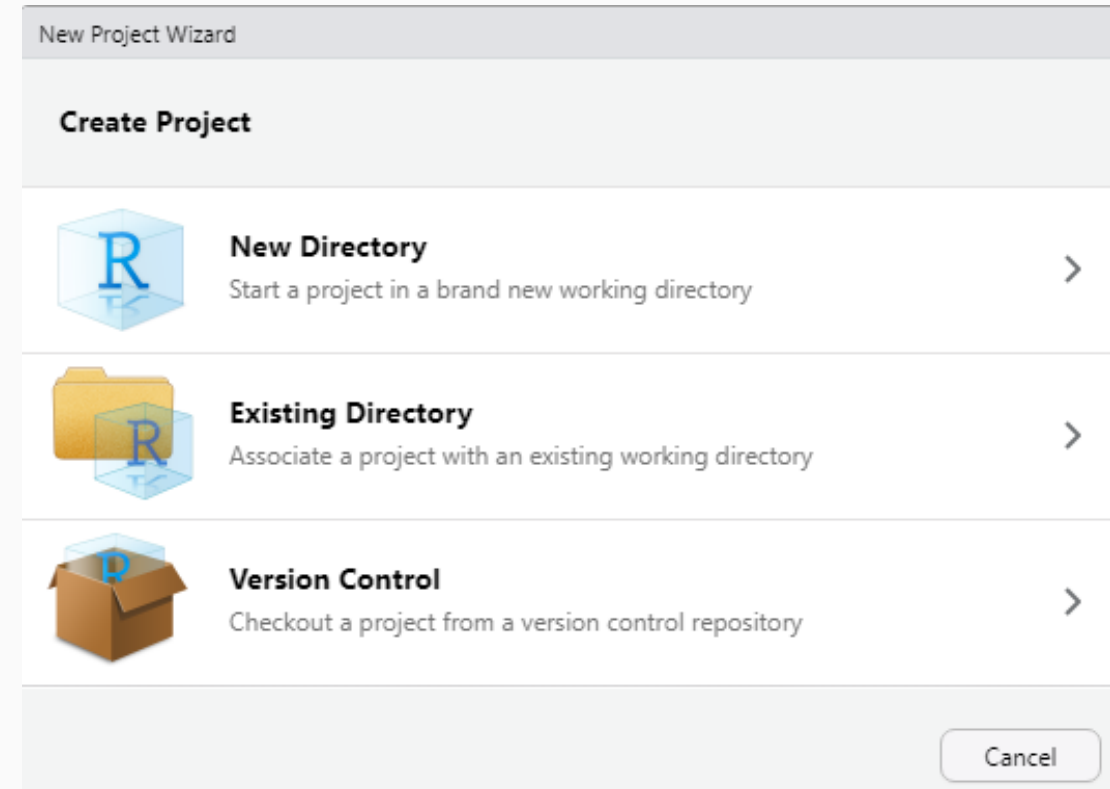
- Le indicamos a R dónde "vive" nuestra carpeta de trabajo
- La ruta ya no contiene ninguna referencia personal que impida ser abierto en otra computadora

# Armar un nuevo proyecto

## Paso 1:

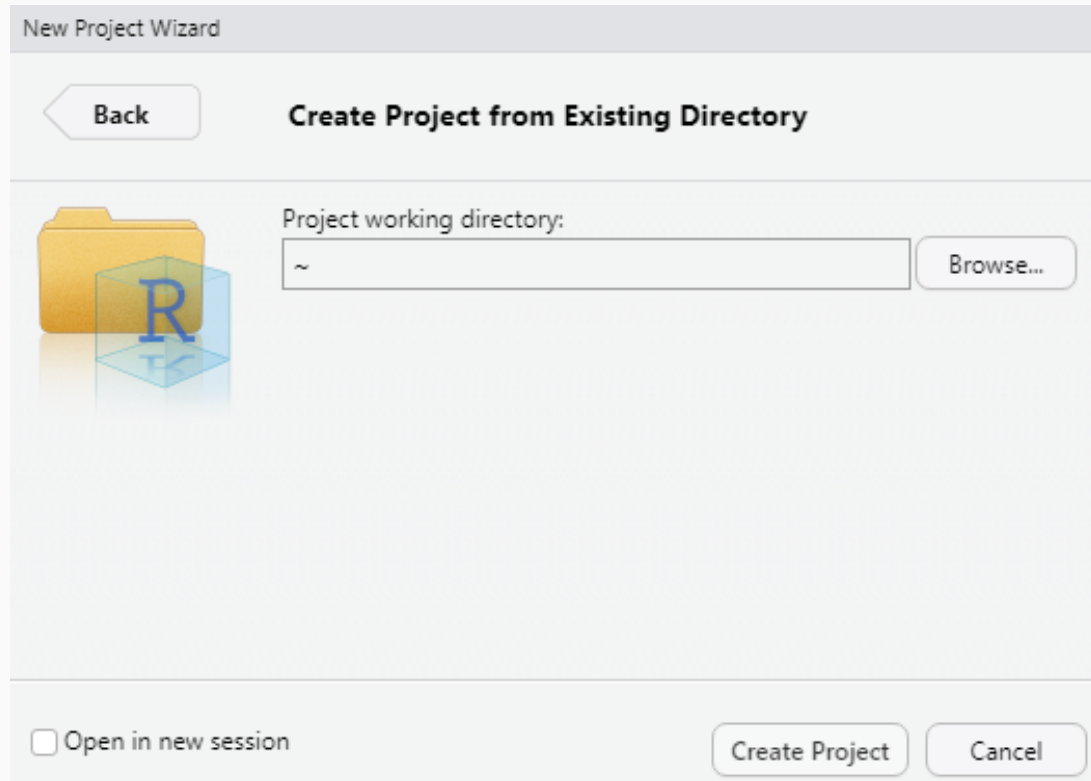


## Paso 2:

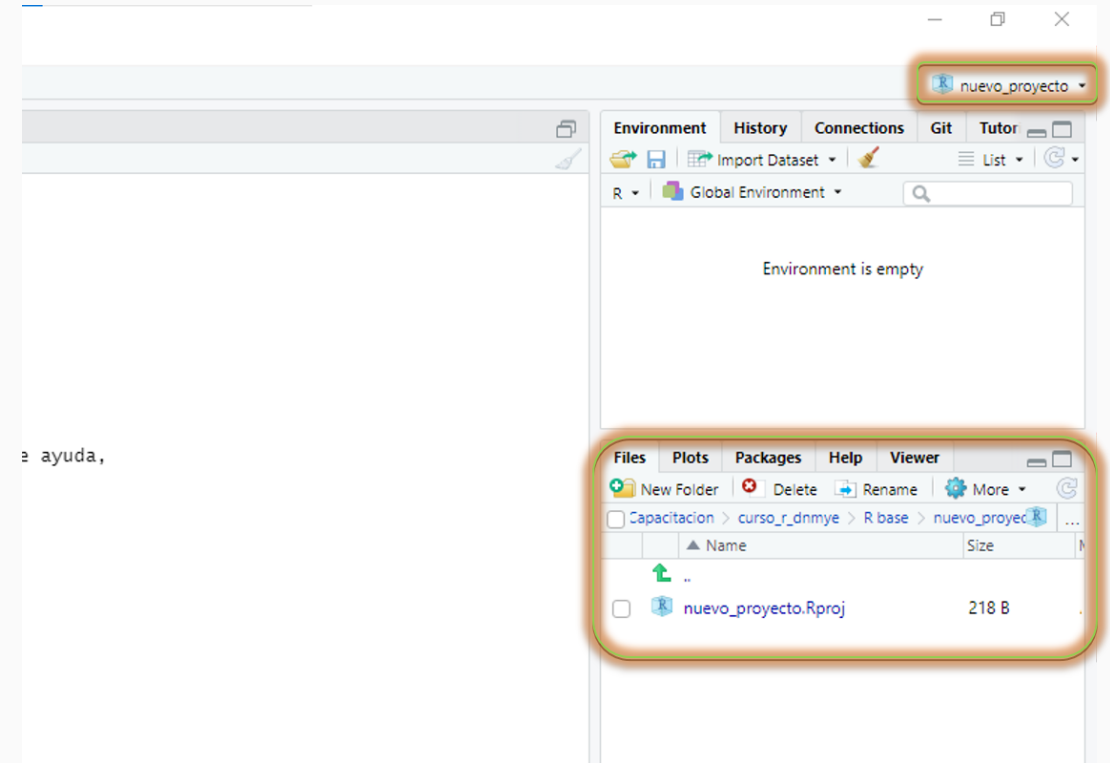


# Armar un nuevo proyecto

## Paso 3:



## Resultado:



# Lectura / escritura de datos

---

# Lectura / escritura de datos

R base tiene una función para cada extensión de archivo:

- `read.table()` --> *.txt*
- `read.csv()` --> *.CSV*
- `readRDS()` --> *.rds*

# Lectura / escritura de datos

Pero... ¿no están todas, no?

Para ello, instalaremos algunos paquetes que nos permiten incorporar más funciones para leer más tipos de datos.

Por ejemplo, el paquete `haven` incorpora:

- `read_sas()` --> *.sas* (SAS)
- `read_spss()` --> *.sav* (SPSS)
- `read_stata()` --> *.dta* (STATA)

y el paquete `readxl`:

- `read_xlsx()` --> *.xlsx* (Excel)

# Lectura / escritura de datos

Supongamos que queremos comenzar un nuevo proyecto para analizar el conjunto de certificados solicitados para vacacionar durante la temporada verano del año 2021







El flujo de trabajo sería el siguiente:

1) Armo la carpeta del proyecto, por ejemplo con la estructura siguiente:

-  cuidar
  -  entradas
  -  salidas
  -  scripts
  -  docs\_metodologicos

# Lectura / escritura de datos








2) Ubico la base de datos en la carpeta correspondiente (**entradas**):

-  cuidar
  -  entradas
    -  certificados-personas-por-fecha-ingreso-provincia-localidad.csv
  -  salidas
  -  scripts
  -  docs\_metodologicos











# Lectura / escritura de datos

3) Abro Rstudio y creo un proyecto, ubicado en la carpeta principal (**cuidar**)

-  cuidar
  -  *proyecto\_cuidar.Rproj*
  -  entradas
    -  *certificados-personas-por-fecha-ingreso-provincia-localidad.csv*
  -  salidas
  -  scripts
  -  docs\_metodologicos

# Lectura / escritura de datos

4) Creo un script llamado *levantar\_datos.R* para trabajar en R y lo guardo en la carpeta **script**

-  cuidar
  -  *proyecto\_cuidar.Rproj*
  -  entradas
    -  *certificados-personas-por-fecha-ingreso-provincia-localidad.csv*
  -  salidas
  -  scripts
    -  *1\_levantar\_datos.R*
  -  docs\_metodologicos

# Lectura / escritura de datos

5) En el script, cargo los paquetes cuyas funciones voy a utilizar. En este caso, el paquete `readxl` para poder importar bases de datos de una extensión `.xlsx` (Excel):

```
library(readxl)
```

6) Creo un objeto en el cual voy a importar la base de datos y utilizo la función `read_xlsx()` para hacerlo.

**Recordar:** *Estamos en un proyecto, por lo que la ruta es a partir de la carpeta en la que se encuentra el mismo*


```
base_cuidar <- read_xlsx(file = "entradas/cuidar_verano_excel.xlsx")
```


# Funciones para una primera mirada a la base de datos:

Función	Acción
<code>dim()</code>	<i>Nos dice cuántas filas y columnas tiene la base</i>
<code>colnames()</code>	<i>Devuelve todos los nombres de las variables de la base</i>
<code>str()</code>	<i>Devuelve una vista preliminar de la estructura de las variables</i>
<code>summary()</code>	<i>Devuelve un resumen de medidas centrales para variables numéricas y características para variables de texto</i>
<code>class()</code>	<i>Chequeo el tipo de objeto</i>
<code>length()</code>	<i>Devuelve la cantidad de elementos de un objeto</i>
<code>table()</code>	<i>Frecuencia simple o bivariada de una variable</i>
<code>is.na()</code>	<i>Verifica la presencia de valores NA (missinh) en la base/variable</i>
<code>mean()</code>	<i>Calcula el valor promedio de una variable numérica</i>

# Chusmeando la base

Code

 Start Over

 Run Code

```
1  
2  
3
```

# PRÁCTICA

---

# Práctica

1. Crear una carpeta donde alojar el proyecto de trabajo
2. Crear la estructura de carpetas propuesta
3. En la carpeta **entradas** incorporar una base de datos, preferentemente que usemos cotidianamente (no importa el formato)
4. Crear un script de trabajo y alojarlo en la carpeta **scripts**
5. Importar la base de datos y aplicar alguna de las funciones descritas previamente

05:00