

Introducción a R y Tidyverse

Sesión 01

Laboratorio de Innovación en Salud

2022-06-03

 @healthinnovation
 @innovalab_imt
 innovalab.info

Acerca del curso

Este curso busca introducir al estudiante en el uso de R y el uso del metapaqute **tidyverse** para el manejo inicial de datos durante un proceso de investigación.

Al finalizar el curso de manera satisfactoria, el participante podrá:

- Importar conjuntos de datos de los principales formatos que se trabajan en proyectos de investigación.
- Dar formato inicial al conjunto de datos mediante reconocimiento de variables y modificaciones en las mismas.
- Ejecutar modificaciones en variables pre-existentes y nuevas variables que se introduzcan.
- Trabajar con múltiples bases de datos y unificarlas de acuerdo al objetivo de investigación planteado.

Contenidos

- Introducción a ciencia de datos
- Manejo y reconocimiento inicial de Rstudio
- Creación y manejo de estructura de proyectos en rstudio
- Creación de vectores

Exploración competencial

¿Qué es Ciencia de Datos?

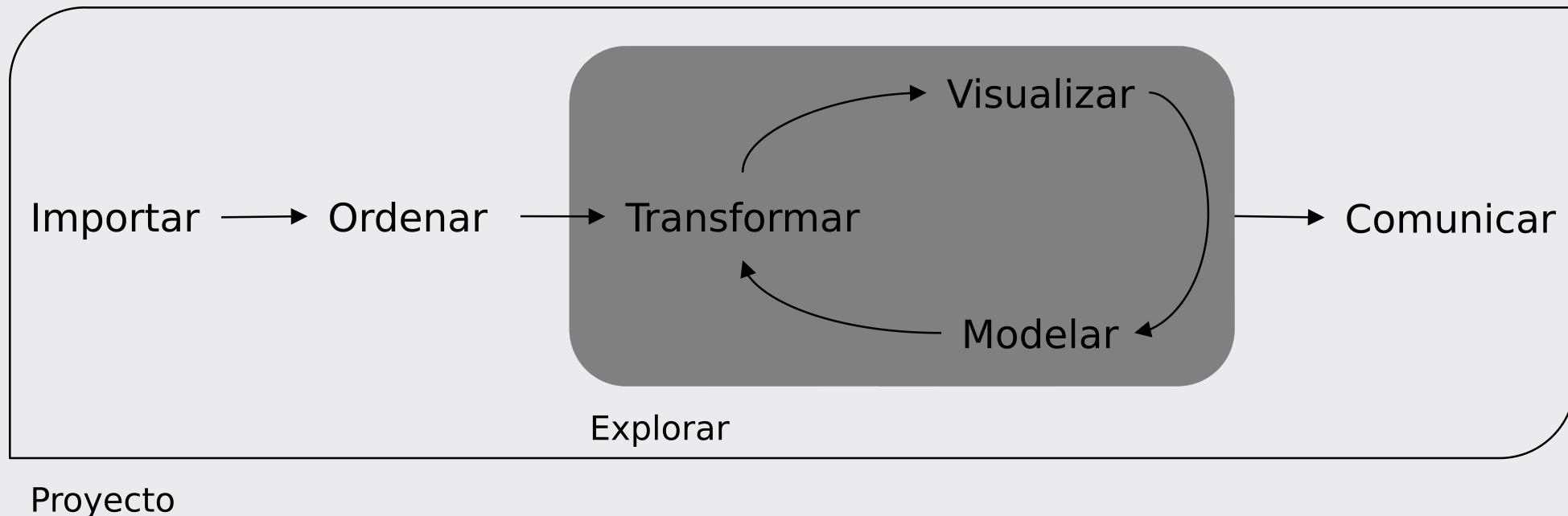
La Ciencia de Datos es una fusión entre múltiples disciplinas, incluyendo matemáticas, estadística, informática, y tecnología de la información.

La Ciencia de Datos permite extraer información relevante de los datos.



¿Por qué usar R para Ciencia de Datos?

R cuenta con las herramientas necesarias (entorno, librerías, y funciones) para desarrollar proyectos de Ciencia de Datos.



Reconocimiento Rstudio



The screenshot displays the RStudio interface with several key components:

- Script Editor:** On the left, a code editor window titled "Untitled1" shows the word "script".
- Environment Pane:** In the top right, the "Environment" tab is active, showing the message "Environment is empty".
- Console:** At the bottom left, the R console shows the prompt ">".
- File Browser:** On the right, the "Files" tab of the sidebar shows a directory structure:

Name	Size	Modified
.r	117 B	Oct 28, 2021, 1:33 PM
.Renvironment	8.1 KB	Nov 12, 2021, 9:34 PM
.Rhistory		
bin		
Blender		
Descargas		
Documentos		
Documents		
- Text Overlays:** Several bold, black, sans-serif text overlays are placed on the interface:
 - "script" is centered above the code editor.
 - "Ambiente de trabajo, historial, conexión y tutoriales" is centered above the environment pane.
 - "consola" is centered below the console.
 - "Archivos, gráficos, paquetes, sistema de ayuda y visor." is centered below the file browser.

Reconocimiento Rstudio

SCRIPT

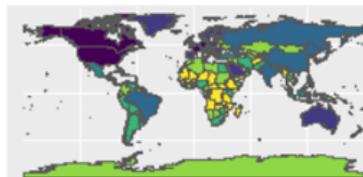
```
1  ### MI PRIMER SCRIPT
2
3  print('Hola mundo spatial')
4
5  library(ggplot2)
6  library(sf)
7  library(rnaturalearth)
8  vignette("rnaturalearth")
9
10 world <- ne_countries(scale = "medium", returnclass = "sf")
11 class(world)
12 |
13 ggplot(world)+ 
14   geom_sf(aes(fill = economy))+
15   scale_fill_viridis_d()
16
12:1 # MI PRIMER SCRIPT
```

CONSOLA

```
> ### MI PRIMER SCRIPT
> print('Hola mundo spatial')
[1] "Hola mundo spatial"
>
```

ENTORNO, HISTORIAL, CONEXIONES

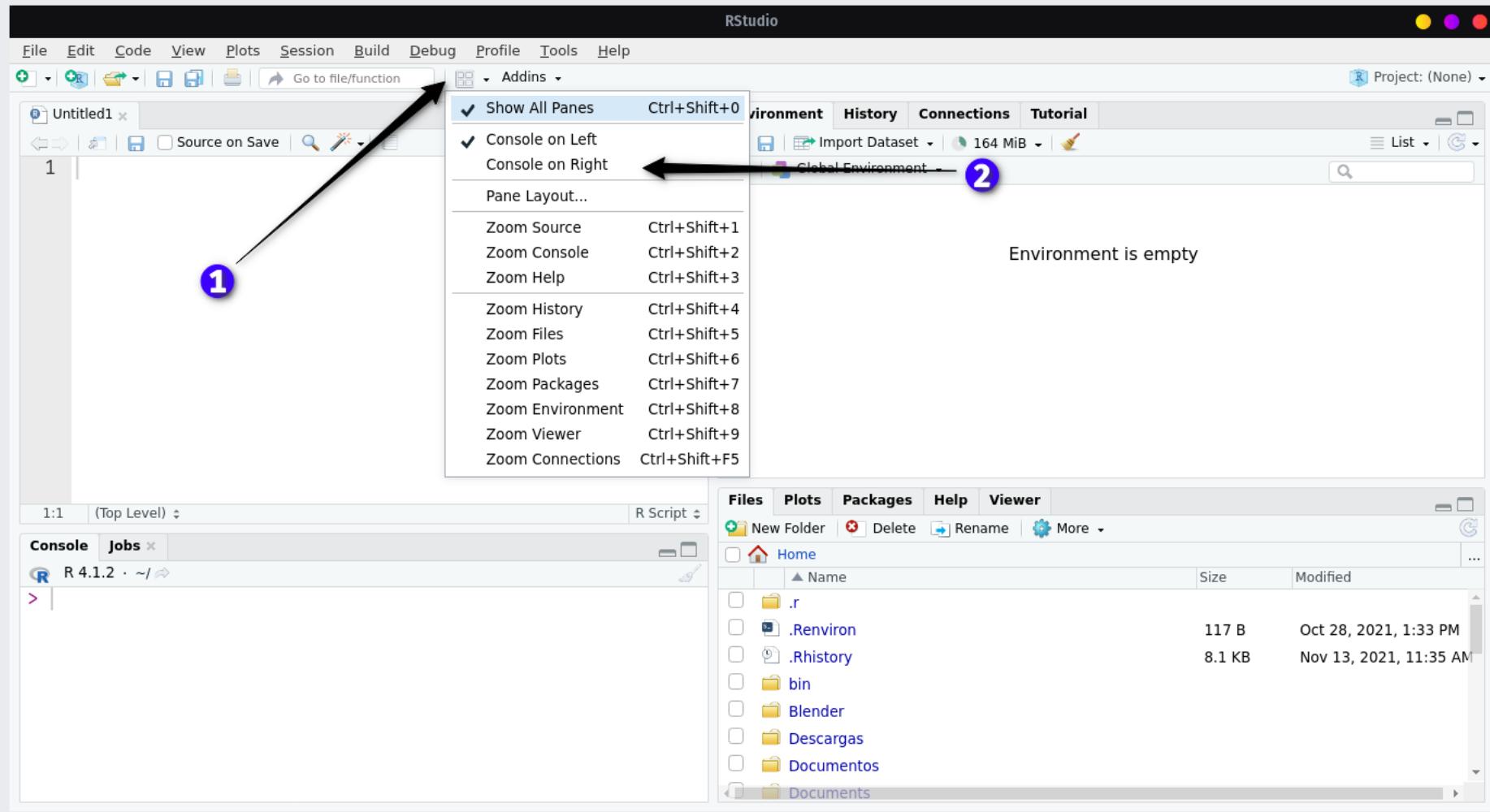
GRÁFICOS, ARCHIVOS, SISTEMA DE AYUDA, VISUALIZADOR INTERACTIVO



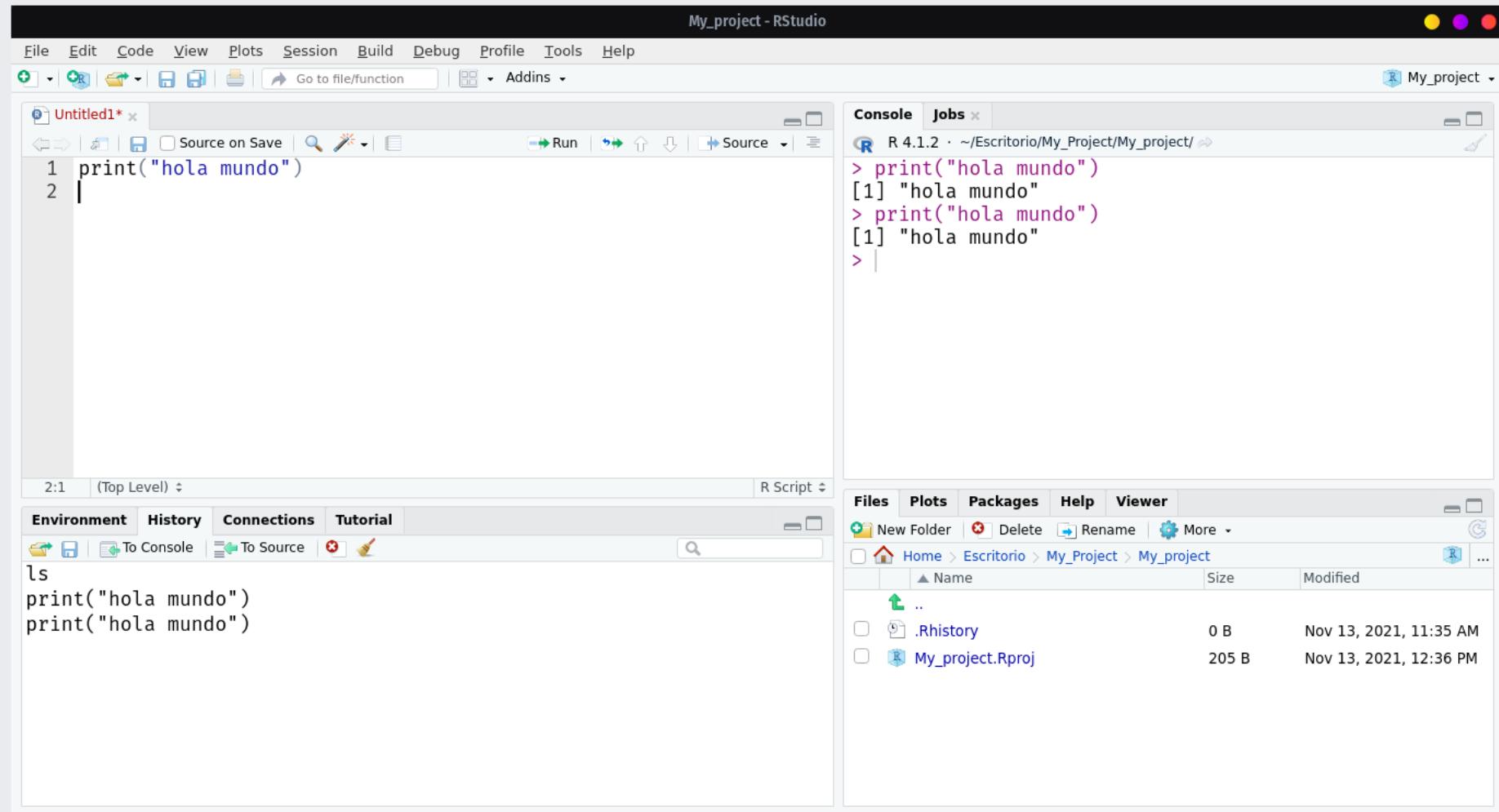
economy

- 1. Developed region: G7
- 2. Developed region: nonG7
- 3. Emerging region: BRIC
- 4. Emerging region: MIKT
- 5. Emerging region: G20
- 6. Developing region
- 7. Least developed region

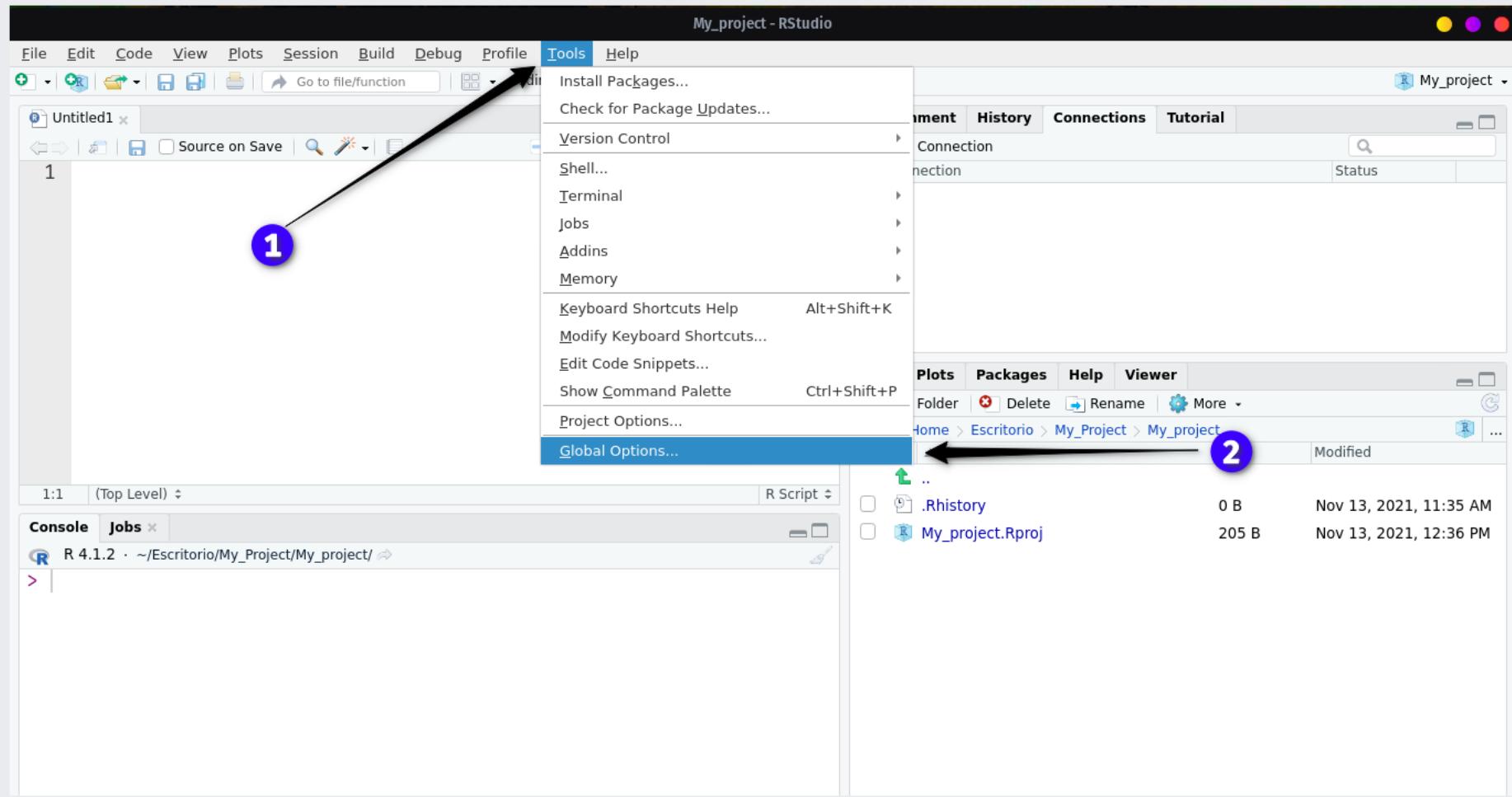
Mover paneles



Mover paneles



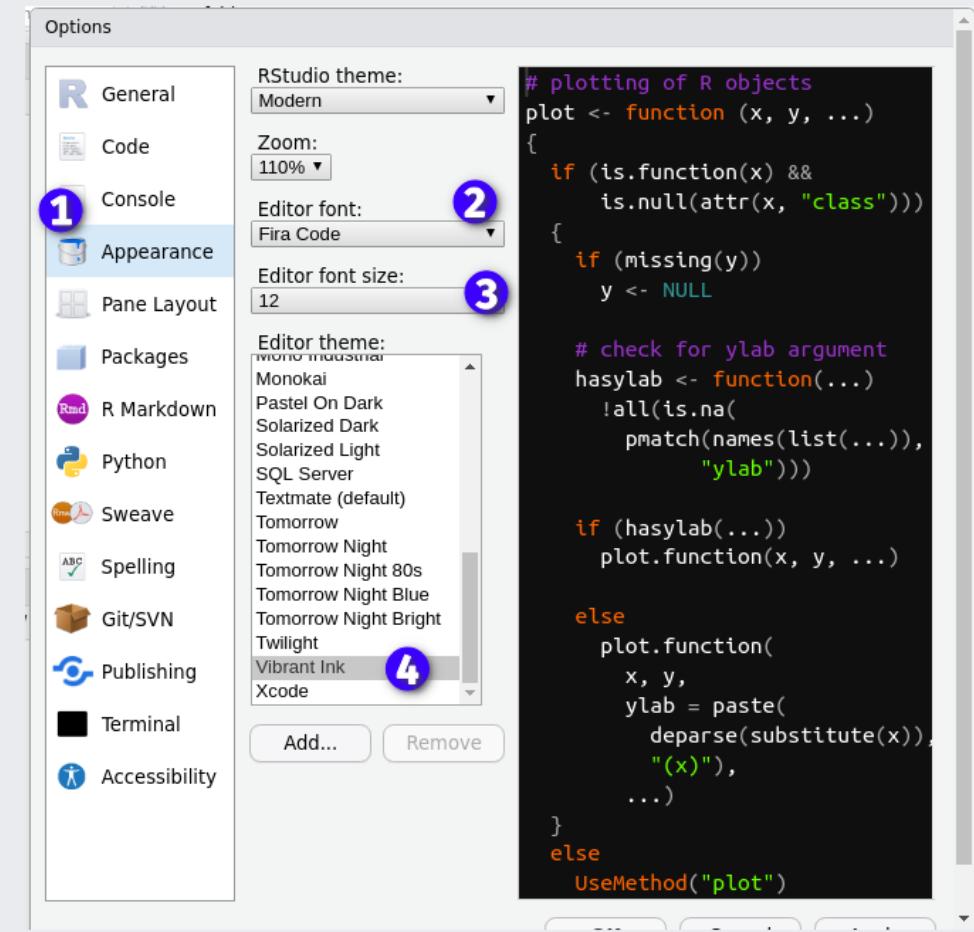
Personalización de Rstudio



Personalización de Rstudio

Modificaciones sobre la interfaz de Rstudio

- Aumentar el zoom o utilizar el atajo:
Ctrl y +
- Cambiar tipo de letra puede hacer un tanto más agradable la codificación. La letra Fira Code es bastante recomendable, pero se requiere instalar.
- El tema también puede ayudar en que la codificación sea más agradable. El paquete `rstthemes`, contiene muchos temas extras.



¡Hazlo tú mismo!

1. Mueve la consola a la derecha
2. Cambia el tema de tu rstudio
3. Cambia el zoom a tu agrado
4. Cambia el tamaño y tipo de letra a tu agrado

08 : 00

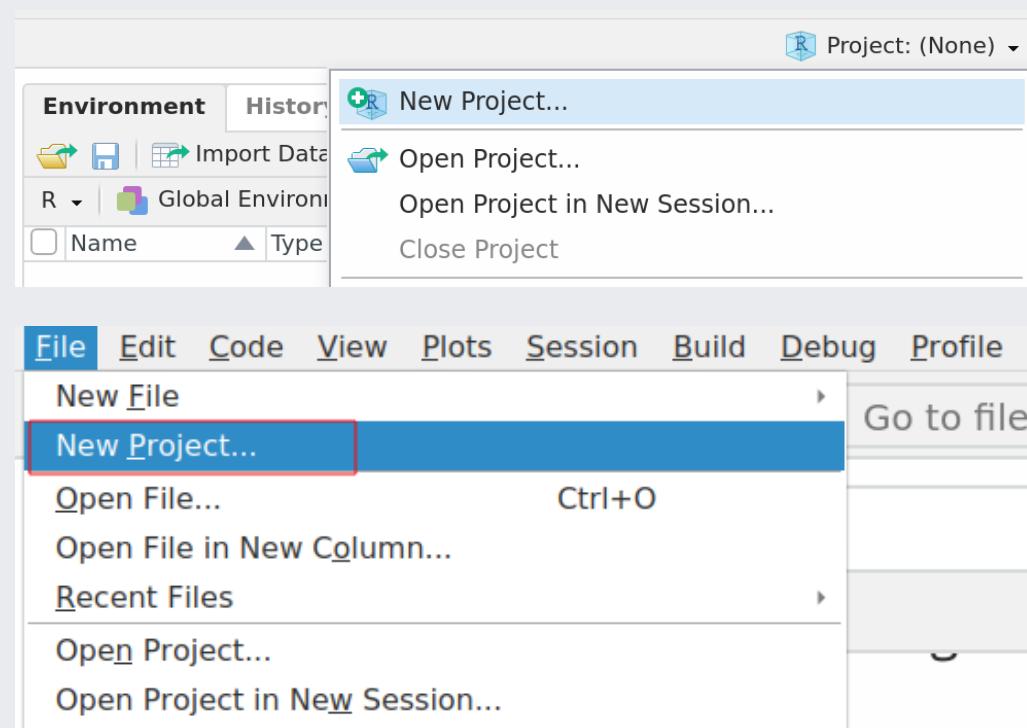
¿Por qué utilizar proyectos?

- Es más fácil poder compartir los proyectos y estos se encuentran listos para que otras personas puedan colaborar contigo.
- Cada proyecto se encuentra aislado. Los códigos en un proyecto no afectarán a ningún otro. Puedes tener muchos proyectos abiertos y los códigos del proyecto 1 no afectarán al proyecto 2.
- Muy útil para facilitar la importación de data.
- Mejora tanto la reproducibilidad como la colaboración.

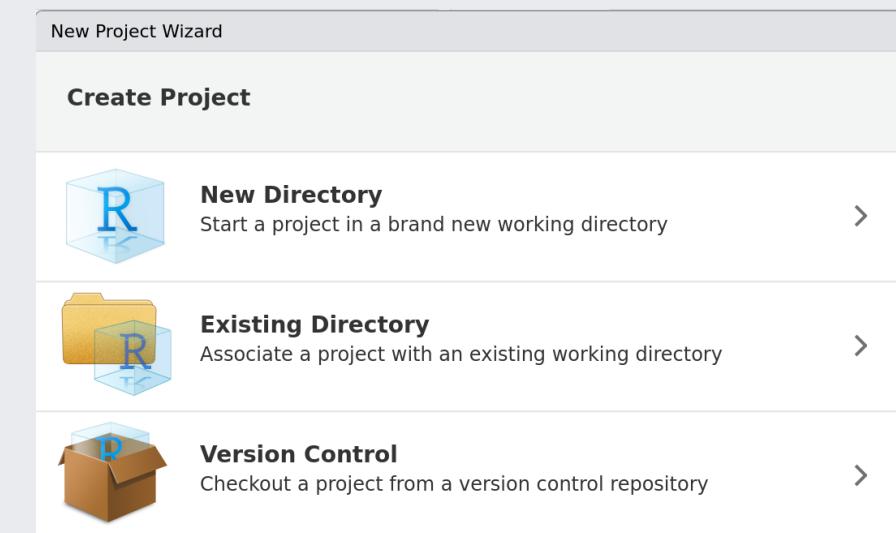
Creación de proyectos

PASOS:

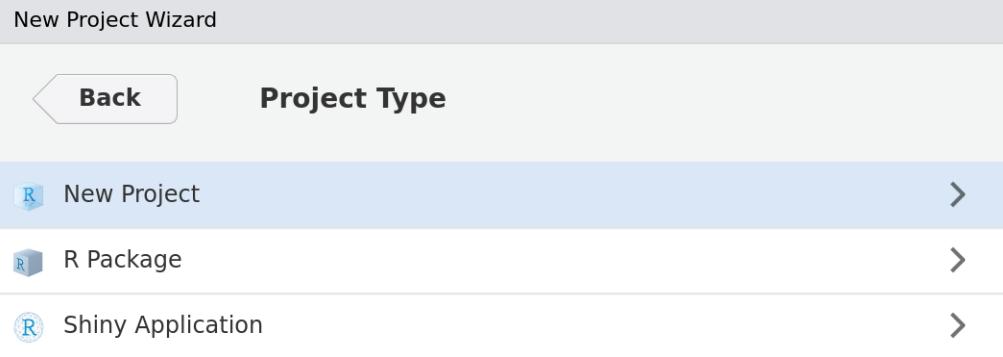
1. Seleccionamos “Project (None)” o “File” y luego, new project.



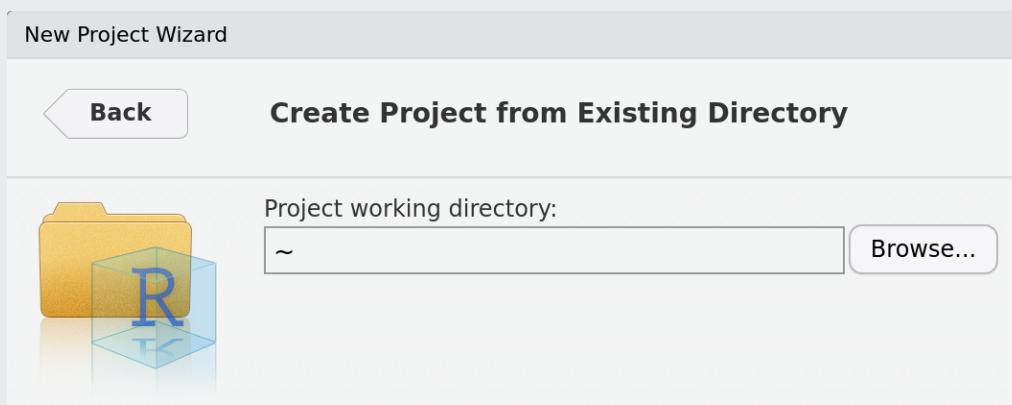
2. “New Directory” se utiliza para indicar dónde voy a almacenar mis archivos y para que R cree una nueva carpeta para mi proyecto, mientras que “Existing Directory” se utiliza si ya tengo una carpeta en la cual voy a almacenar mis archivos. Seleccionamos “New Directory”.



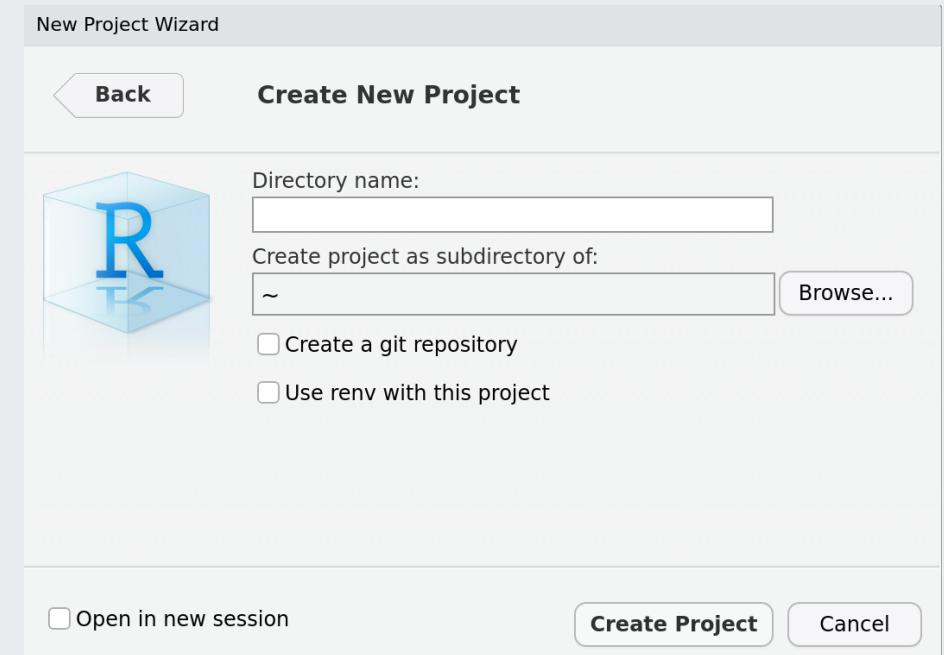
3. Aparecerán más opciones y seleccionamos nuevamente “New Project”



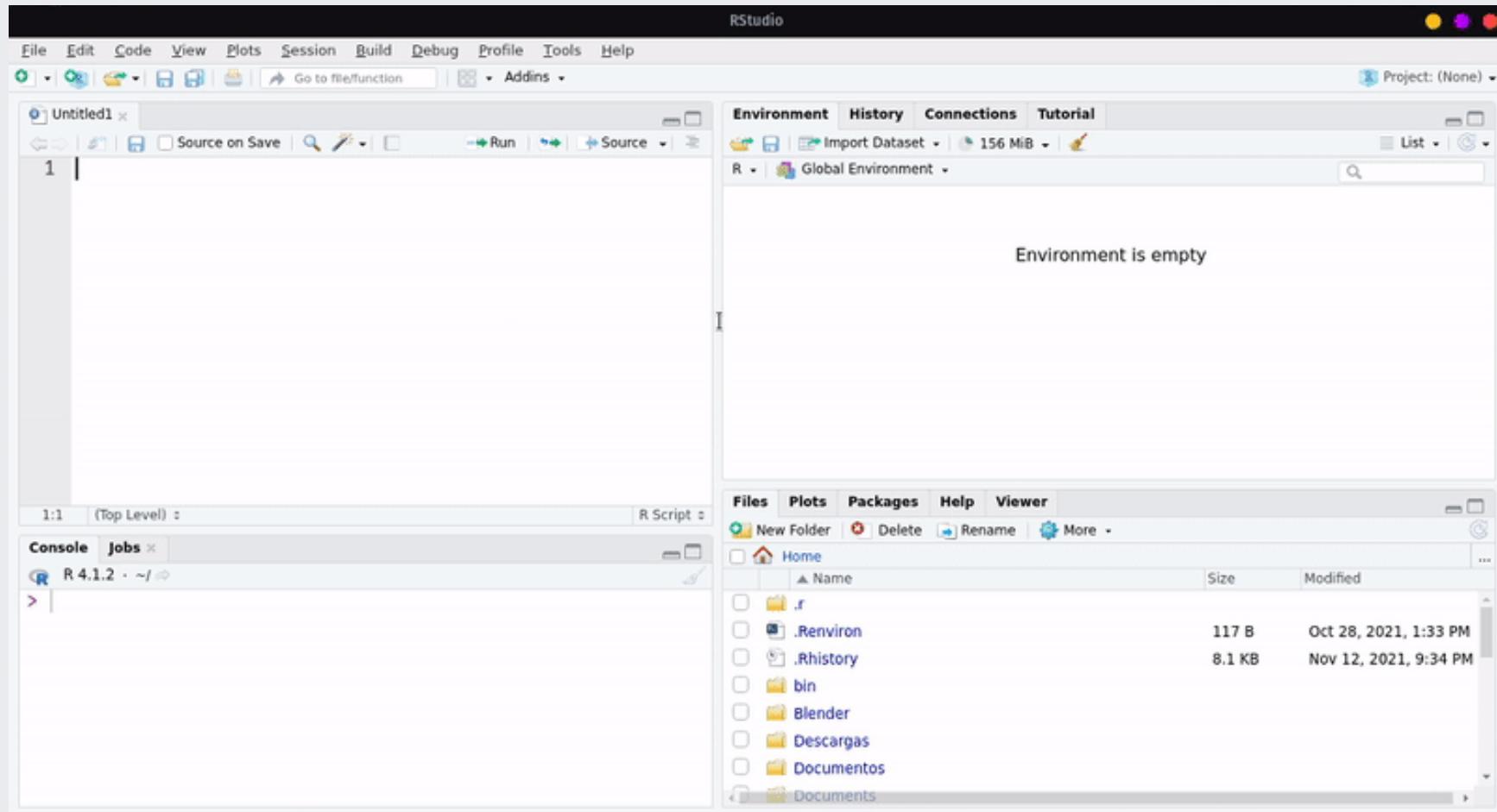
3. En caso de haber seleccionado “Existing Directory” aparecerá esto y buscamos el nombre de la carpeta que utilizaremos.



4. En “Directory name” ponemos el nombre de la carpeta que contendrá el archivo del proyecto, mientras que en “Create project as subdirectory of” seleccionamos dónde está la carpeta en la que trabajaremos.



Creación de proyectos



¡Hazlo tú mismo!

1. Crea una carpeta para todo el curso
2. Dentro crea una carpeta para la primera sesión.
3. Crea un proyecto dentro de esa carpeta creada, por ej. Sesión 01.RProj.

04 : 00

Vectores atómicos

- Los vectores contienen información homogénea o siempre de un solo tipo de datos.
- Un vector puede contener 1 solo elemento o n-elementos.
- Existen hasta 6 tipos de datos que puede contener un vector:
 - Logical: TRUE, FALSE
 - Integer: 1, 5, 7
 - Double: 3.15, 10, 12.86
 - Character: "Marcos", "Laptop".
 - Complex: 3 + 2i

```
v_numeric <- 5  
v_numeric
```

```
## [1] 5
```

```
v_numeric <- c(5, 10, 15)  
v_numeric
```

```
## [1] 5 10 15
```

```
typeof(v_numeric)
```

```
## [1] "double"
```

```
v_character <- c("Laptop", "Rstudio", "4.2")  
v_character
```

```
## [1] "Laptop" "Rstudio" "4.2"
```

Vectores atómicos

Los vectores solo pueden tener **un solo tipo de información a la vez**, así que si dentro de un vector se ingresa un elemento tipo `numeric`, este inmediatamente será transformado a `character`.

```
v_character ← c("Laptop", "Rstudio", 4.2)  
v_character
```

```
## [1] "Laptop"  "Rstudio" "4.2"
```

```
typeof(v_character)
```

```
## [1] "character"
```

Vectores atómicos

Crearemos 2 vectores llamados `nombres` y `edades`.

```
nombres ← c("Luis", "Mateo", "Carlos", "Eduardo")
edades ← c(28, 30, 40, 35)
```

A partir de esto, podemos construir algo con lo que probablemente estemos más familiarizados.

```
data.frame(nombres,
            edades)
```

```
##   nombres edades
## 1 Luis     28
## 2 Mateo    30
## 3 Carlos    40
## 4 Eduardo   35
```

¡Hazlo tú mismo!

1. Se formarán grupos de 6 personas en salas de **Zoom**.
2. Cada uno de los participantes crearán n vectores llamados: **nombres** y otros que sean de su interés, como por ej. **profesion**, **edades**, **universidad**, **distrito**, etc.
3. El contenido de los elementos de los vectores a crear será real y a partir de las preguntas que se puedan hacer entre sí, dentro del mismo grupo de personas.
4. Por último, dentro de **data.frame()** pondrán los vectores que hayan creado. Deberían tener como mínimo, 2 columnas, e idealmente 6 filas.

10 : 00

Retroalimentación

¡Gracias!

✉ imt.innovlab@oficinas-upch.pe

🗣 [@healthinnovation](https://twitter.com/@healthinnovation)

🐦 [@innovalab_imt](https://twitter.com/@innovalab_imt)

Estas diapositivas fueron creadas mediante el paquete `xaringan` y `xaringanthemer`.