

<!--Estudio IMIBIC-->

R para Todos:  
El poder para  
explotar tus Datos {

<Por="Adrián Santiago Ortiz"/>

}

# Contenidos

01 Introducción a R y Rstudio

02 Tipos de datos

03 Operadores en R

04 Tipos de objetos y operaciones

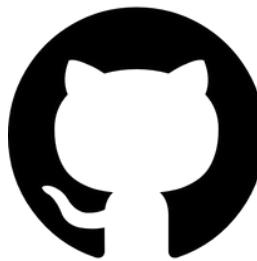
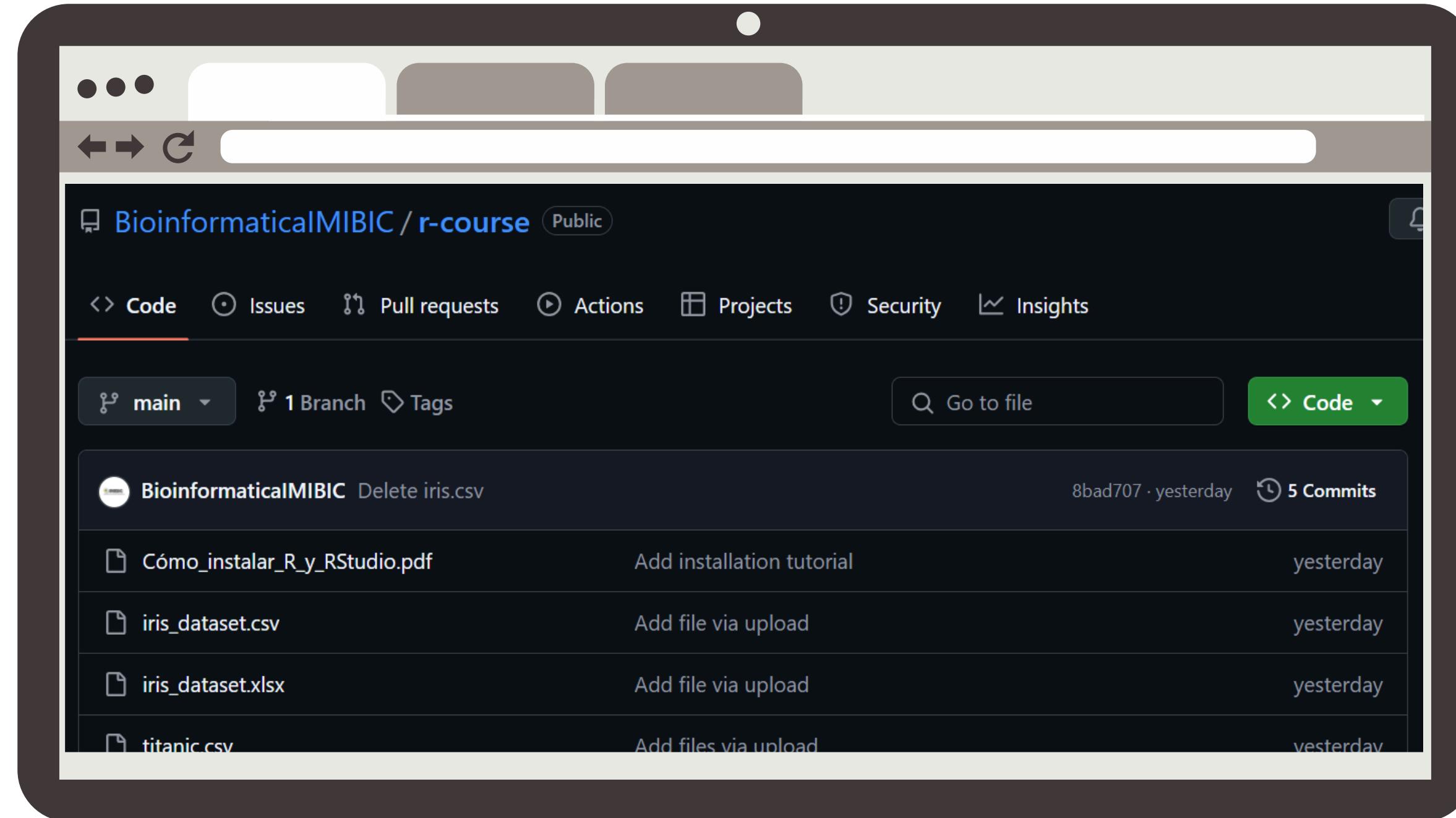
05 Estructuras de control y flujo de ejecución

06 Creación de funciones

07 Visualización de resultados en gráficos

Sesión 2

# Material disponible



<https://github.com/BioinformaticalMIBIC/r-course>

# Repaso de la sesión 1



## R es un lenguaje orientado a objetos

	Dimensions	Mode (data "type")	Example		Dimensions	Mode (data "type")	Example
<b>Vector</b>	1	Identical	<code>c(10,0.2,34,48,53)</code>				
				<b>Data frame</b>		Can be different	<code>data.frame(x = 1:3, y = 5:7)</code>

# Repaso de la sesión 1



## R es un lenguaje orientado a objetos

	Dimensions	Mode (data "type")	Example		Dimensions	Mode (data "type")	Example
<b>Vector</b>	1	Identical	c(10,0.2,34,48,53)				
				<b>Data frame</b>		Can be different	data.frame(x = 1:3, y = 5:7)



## Instalar paquetes y cargar librerías



```
install.packages("package1")
library(package1)
```

# Repaso de la sesión 1



## R es un lenguaje orientado a objetos

	Dimensions	Mode (data "type")	Example		Dimensions	Mode (data "type")	Example
<b>Vector</b>	1	Identical	c(10,0.2,34,48,53)				
				<b>Data frame</b>		Can be different	data.frame(x = 1:3, y = 5:7)



## Instalar paquetes y cargar librerías



```
install.packages("package1")
library(package1)
```



## Cargar nuestros datos (tablas)



```
library(readxl)
read_excel("file.xlsx", ...)
```



```
read.table("file.txt", ...)
```

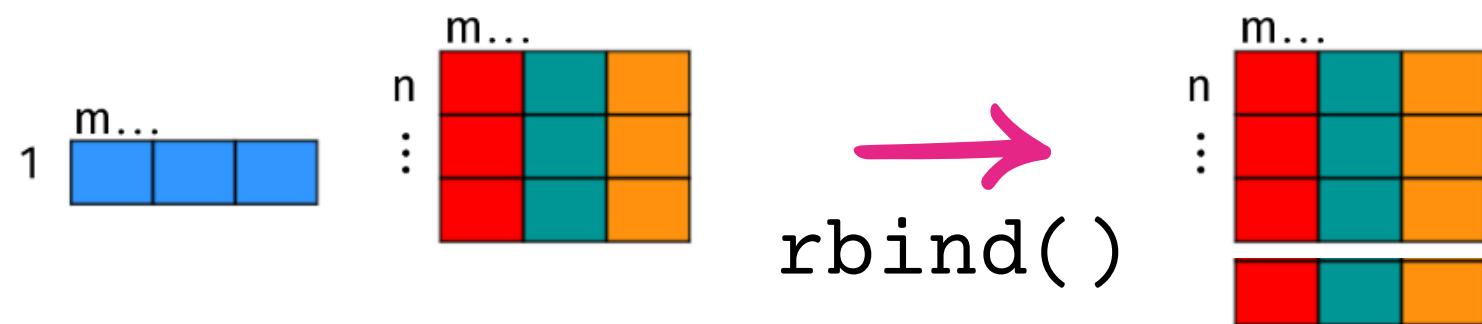


```
read.csv("file.csv", ...)
```

# Repaso de la sesión 1



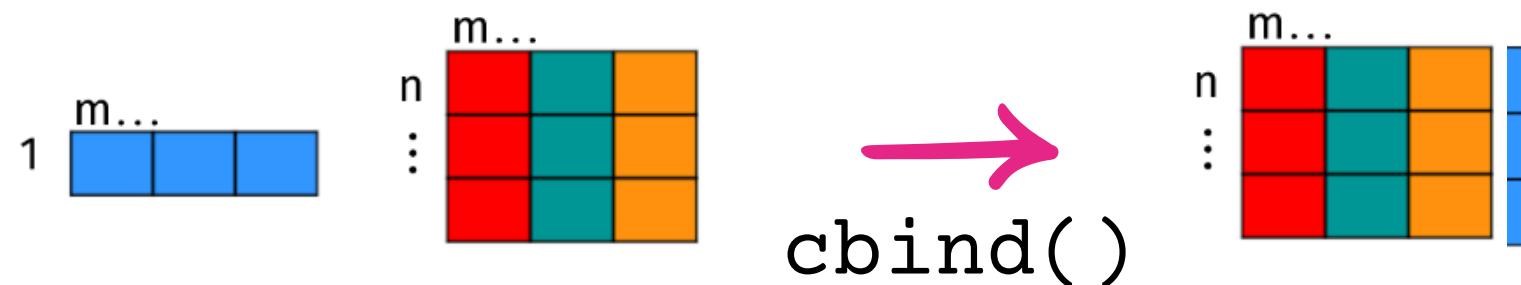
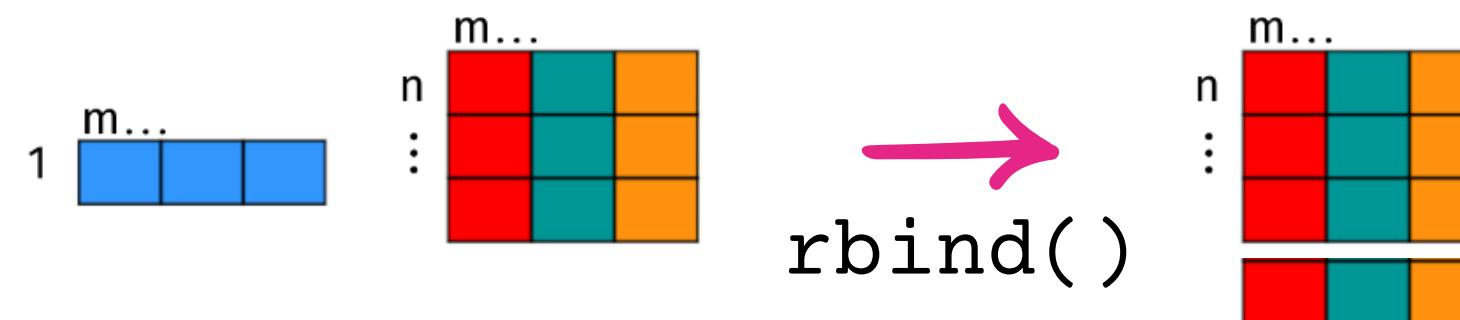
**Añadir filas/columnas de un `data.frame`**



# Repaso de la sesión 1



## Añadir filas/columnas de un data.frame



```
cbind(df, c("Spain", "France",  
          "Italy", "France"))
```



```
df$Country <- c("Spain", "France",  
                 "Italy", "France")
```

Petal.Width	Species
0.2	setosa

Petal.Width	Species	Country
0.2	setosa	Spain
0.2	setosa	France
0.2	setosa	Italy
0.2	setosa	France

# Contenidos

01 Introducción a R y Rstudio

02 Tipos de datos

03 Operadores en R

04 Tipos de objetos y operaciones

05 Estructuras de control y flujo de ejecución

06 Creación de funciones

07 Visualización de resultados en gráficos

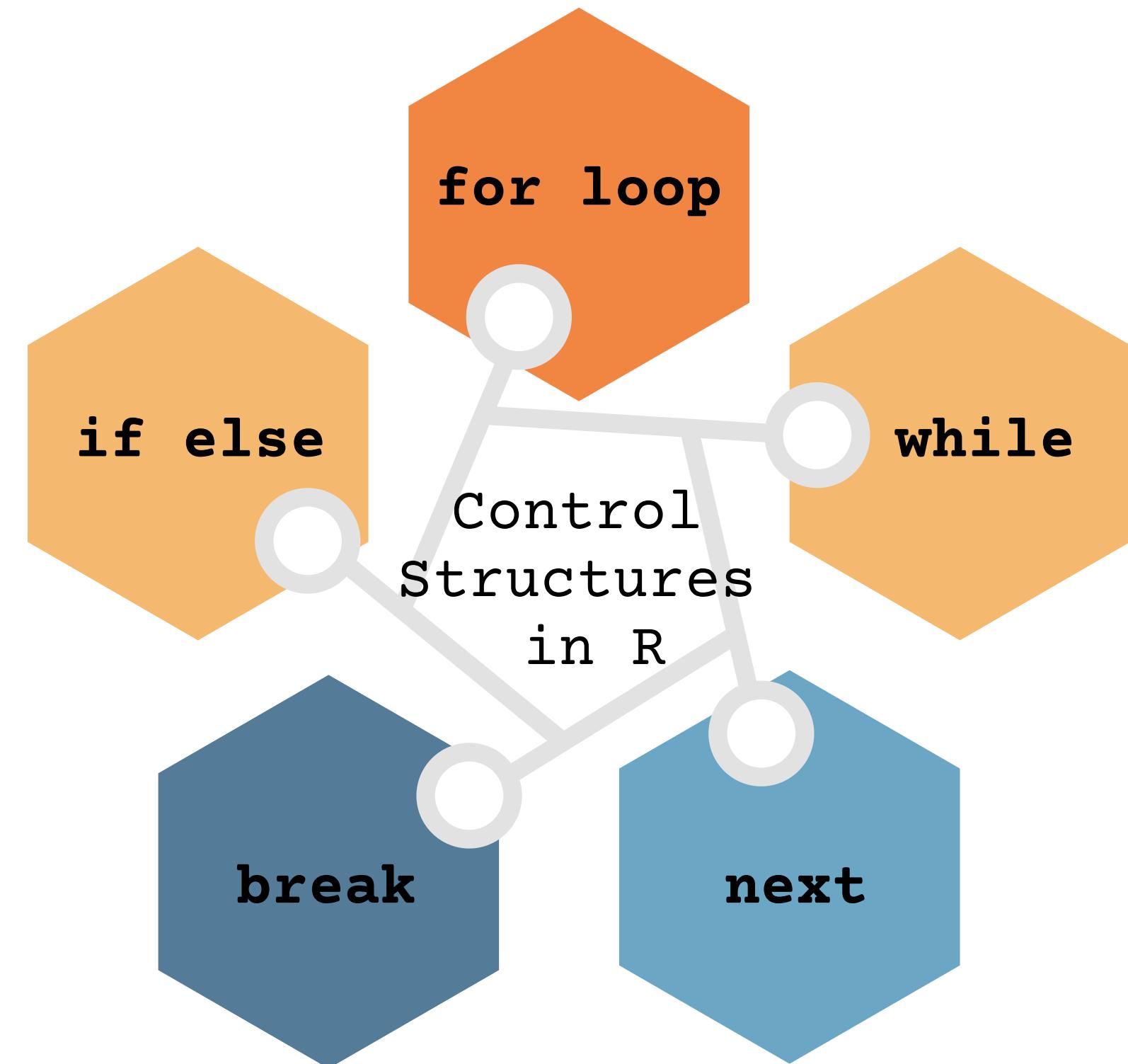
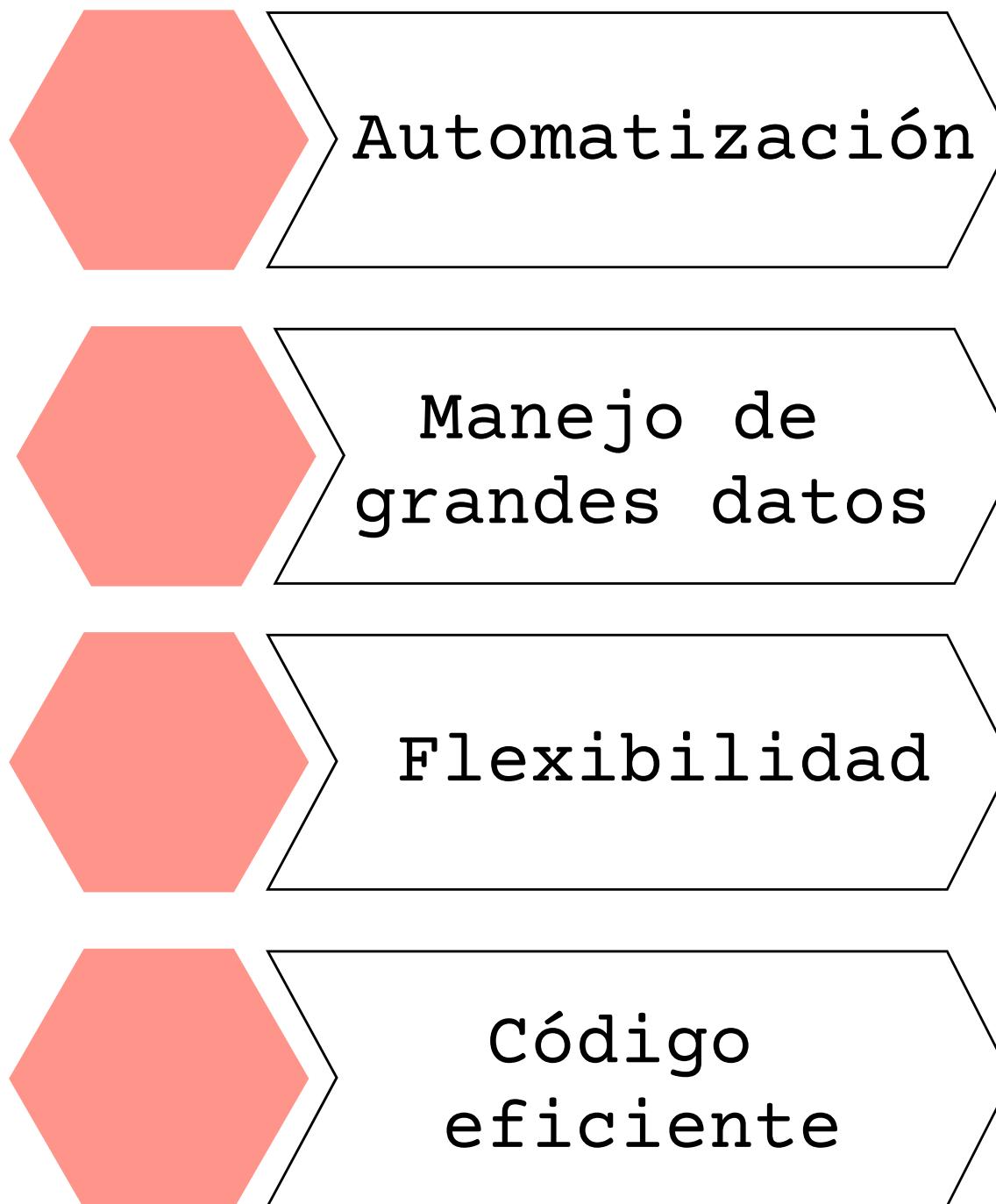
Sesión 2

# Contenidos

01  
02  
03  
04  
05 Estructuras de control y flujo de ejecución  
06  
07

Sesión 2

# • ¿Por qué utilizarlas?

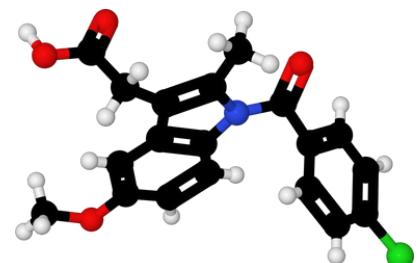


# Para practicar...

ERCP (Endoscopic Retrograde Cholangio-Pancreatography)

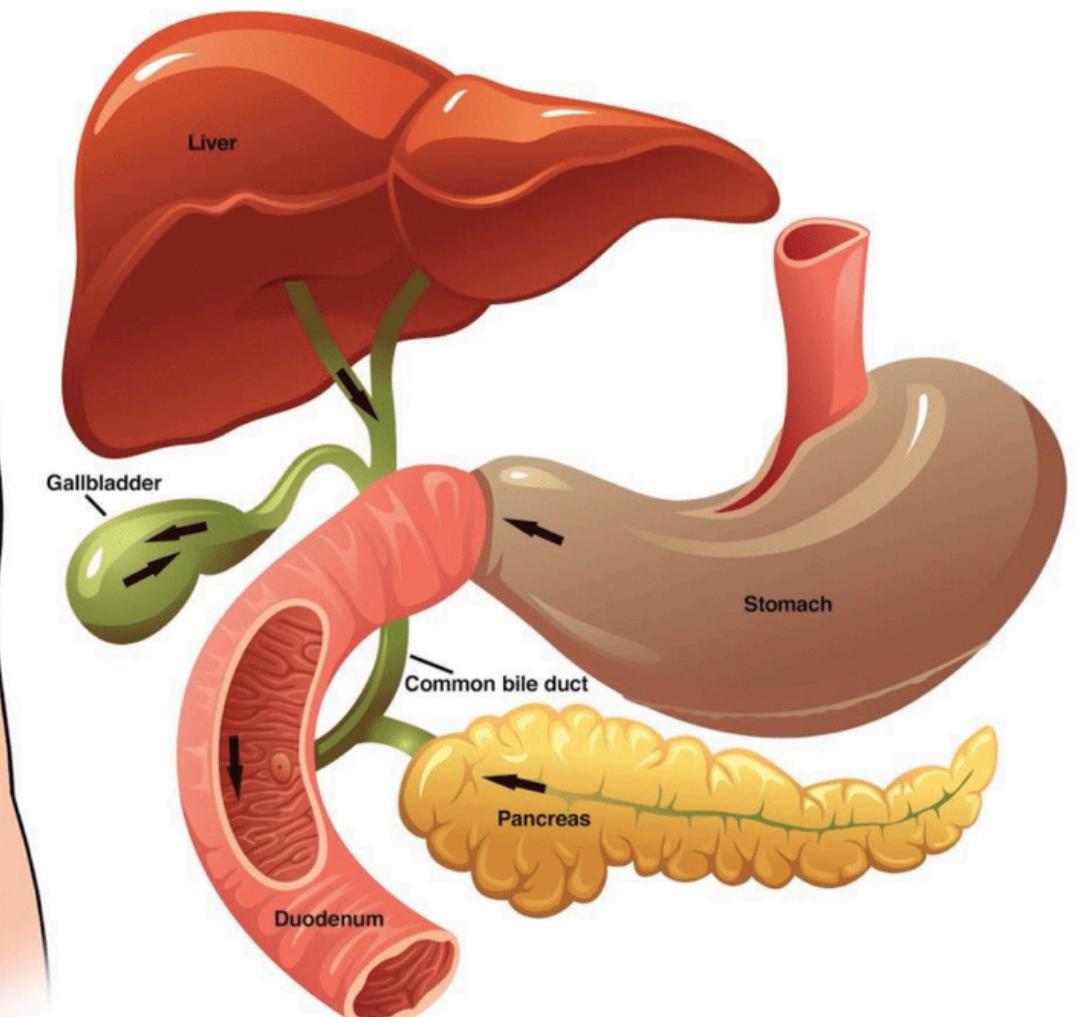
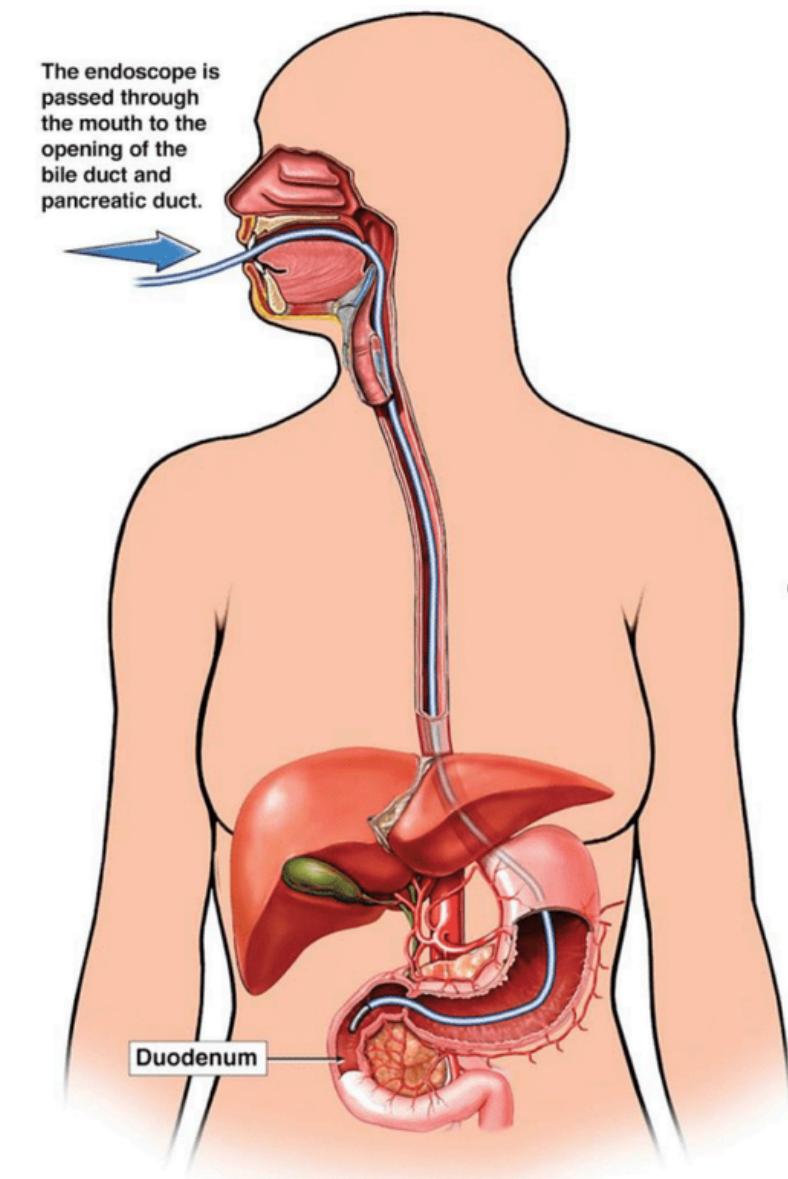
Data frame: indo\_rct

- 602 observaciones
- 33 variables



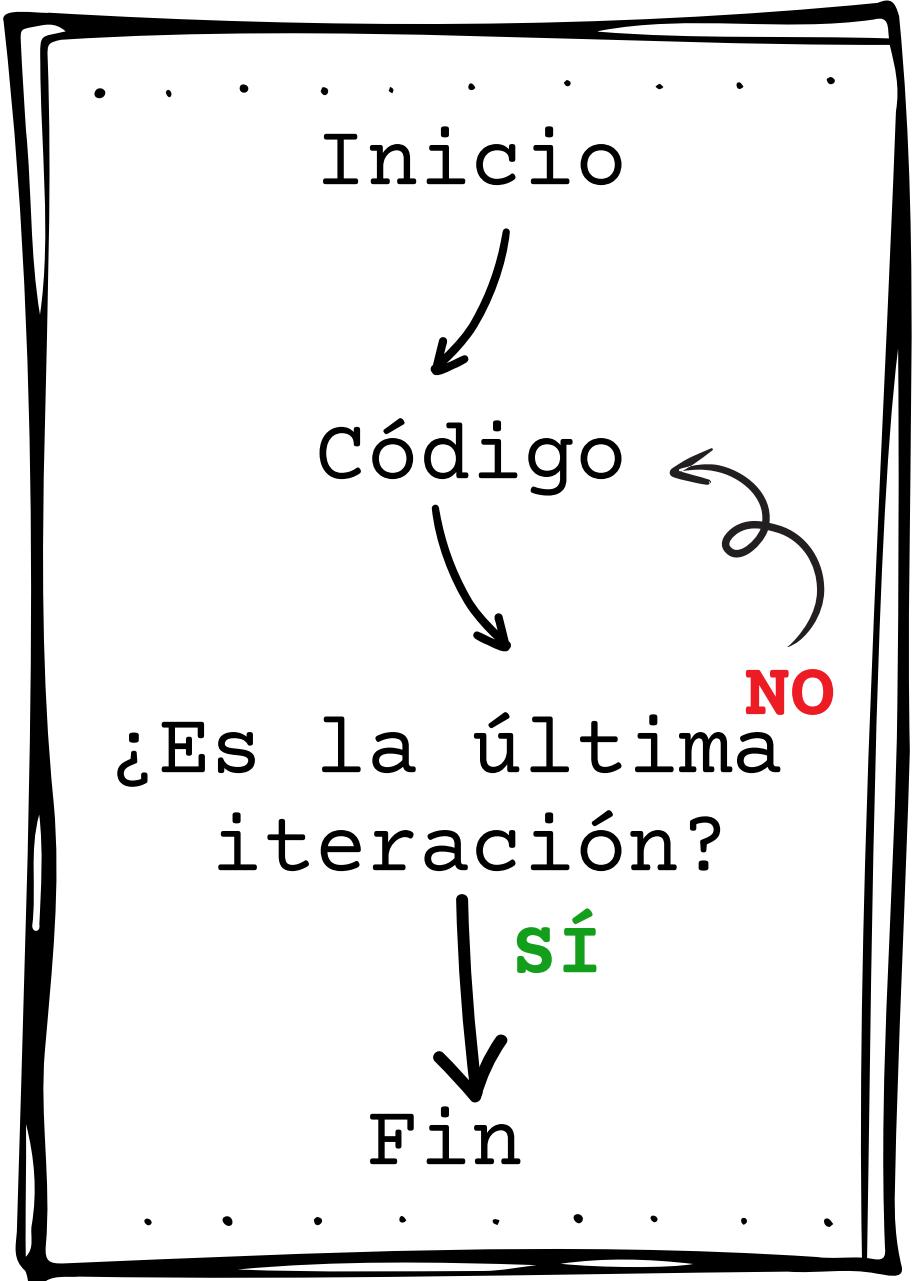
```
install.packages("medicalexdata")
library(medicalexdata)
```

```
data(indo_rct)
```



# Sintaxis del bucle for

Se utiliza para repetir un bloque de código un número específico de veces



# Sintaxis del bucle for

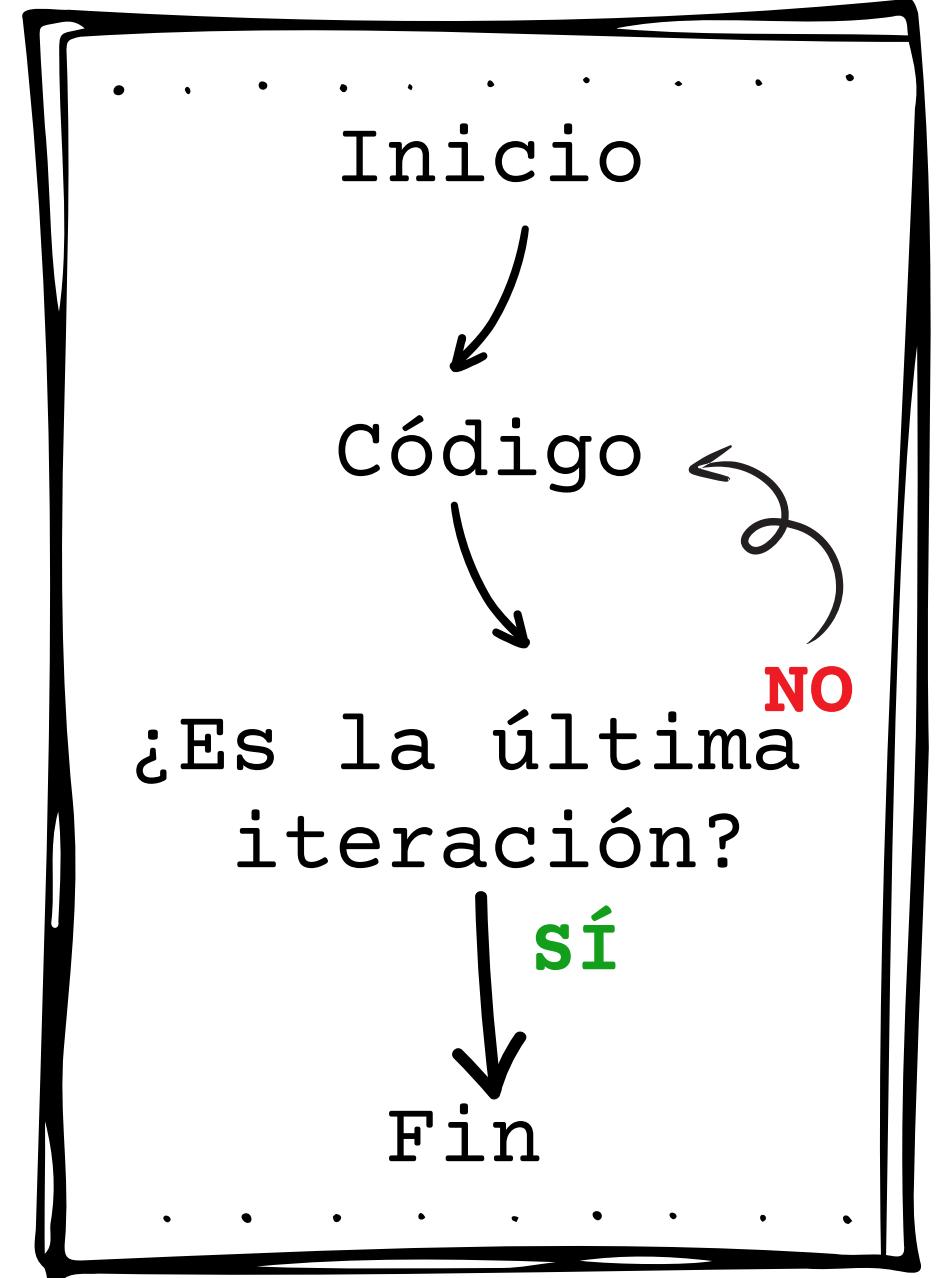
Se utiliza para repetir un bloque de código un número específico de veces

`var`  
Se toman elementos  
de **iterable**

```
for (var in iterable) {  
    statement  
    statement  
    ...  
}
```

**iterable**  
Objeto R  
(vector, lista, etc)

**cuerpo**  
líneas de código  
a ejecutar



# Ejercicio

Muestra la información de los 10 primeros pacientes por pantalla  
(Opcional: solo las 5 primeras columnas)

Usa el comando **print** para mostrar un valor por pantalla

`var`  
Se toman elementos  
de **iterable**

```
for (var in iterable) {  
  statement  
  statement  
  ...  
}
```

**iterable**  
Objeto R  
(**vector**, lista, etc)

**cuerpo**  
líneas de código  
a ejecutar

# Ejercicio

Muestra la información de los 10 primeros pacientes por pantalla  
(Opcional: solo las 5 primeras columnas)

Usa el comando **print** para mostrar un valor por pantalla

i valdrá 1,2,3,...,10  
en cada iteración

```
for (i in 1:10) {  
  print (indo_rct[i,])  
}
```

var  
Se toman elementos  
de iterable

```
for (var in iterable) {  
  statement  
  statement  
  ...  
}
```

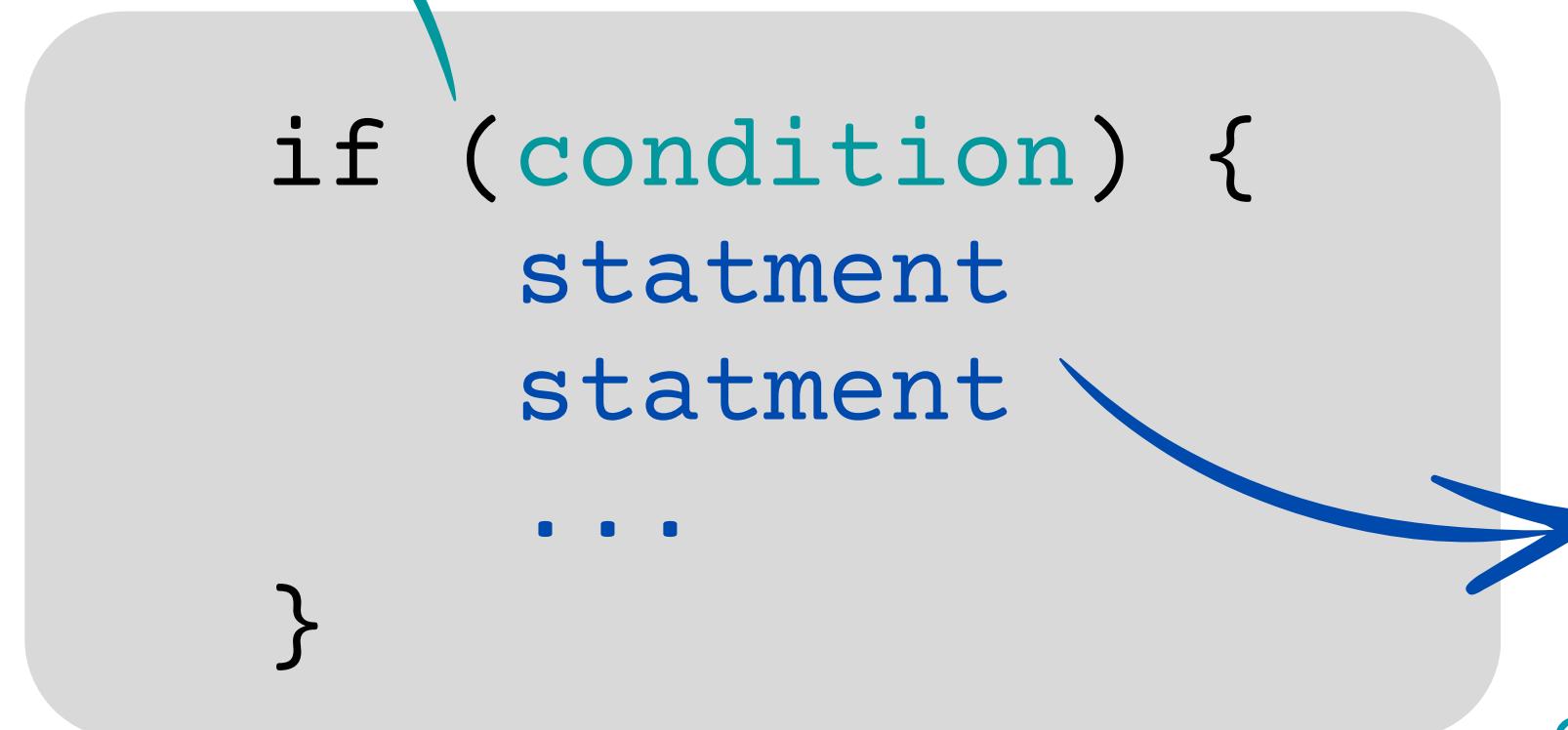
iterable  
Objeto R  
(vector, lista, etc)

cuerpo  
líneas de código  
a ejecutar

# IF / ELSE / ELSE IF /

Para ejecutar un bloque de código **solo si** se cumple una condición específica

**condition**  
evalúa si es  
TRUE o FALSE



TRUE or FALSE?

Operador	Descripción
< y >	menor/mayor a
<= y >=	menor/mayor o igual a
==	exactamente igual a
!=	no es igual a
x   y	x OR y
x & y	x AND y
x %in% y	x IN y

cuerpo  
se ejecuta si  
condition es TRUE

# IF / ELSE / ELSE IF /

Evalúa si el paciente en la fila 22 es mayor de 60 años y muestra un mensaje si se cumple

**condition**  
evalúa si es  
TRUE o FALSE

```
fila_paciente <- 22  
  
if (indo_rctfila_paciente, "age"] > 60) {  
  print("Es mayor de 60 años")  
}
```

TRUE or FALSE?

Operador	Descripción
< y >	menor/mayor a
<= y >=	menor/mayor o igual a
==	exactamente igual a
!=	no es igual a
x   y	x OR y
x & y	x AND y
x %in% y	x IN y

cuerpo  
se ejecuta si  
condition es TRUE

# • IF / ELSE / ELSE IF /

**condition**  
evalúa si es  
TRUE o FALSE

```
if (condition) {  
    statement  
    statement  
    ...  
}
```

→ **Rama TRUE**  
Se ejecuta si  
**condition** es TRUE

# IF / ELSE / ELSE IF /

**condition**  
evalúa si es  
TRUE o FALSE

```
if (condition) {  
    statement  
    statement  
    ...  
} else {  
    statement  
    statement  
    ...  
}
```

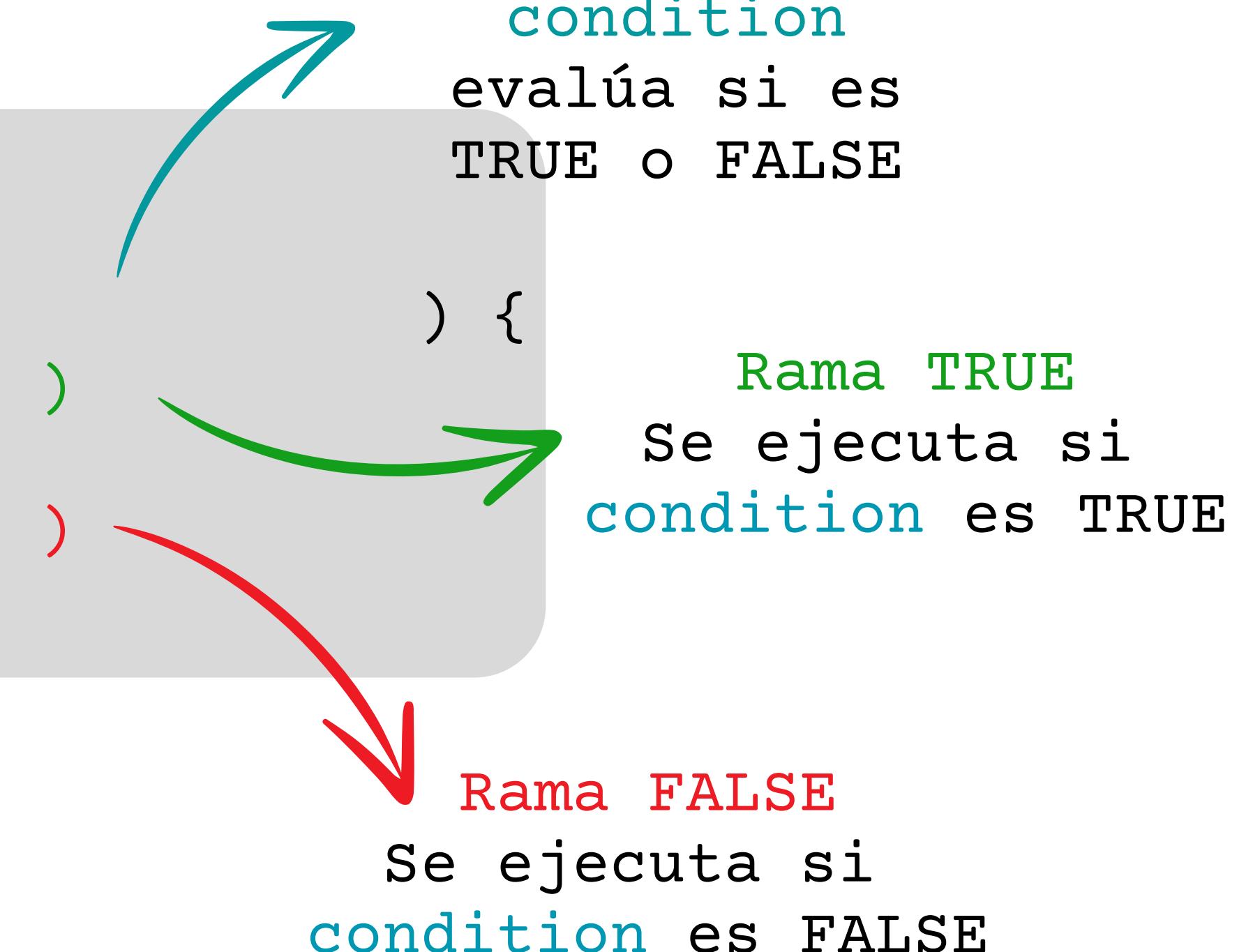
→ **Rama TRUE**  
Se ejecuta si  
**condition** es TRUE

→ **Rama FALSE**  
Se ejecuta si  
**condition** es FALSE

## Ejercicio

Evalúa si el paciente en la fila 22 es mayor de 60 años y muestra un mensaje para las dos situaciones

```
fila_paciente <- 22  
  
if (  
  print(  
} else {  
  print(  
}
```



## Ejercicio

Evalúa si el paciente en la fila 22 es mayor de 60 años y muestra un mensaje para las dos situaciones

```
fila_paciente <- 22

if (indo_rctfila_paciente, "age"] > 60) {
  print("Es mayor de 60 años")
} else {
  print("Es menor de 60 años")
}
```

condition  
evalúa si es TRUE o FALSE

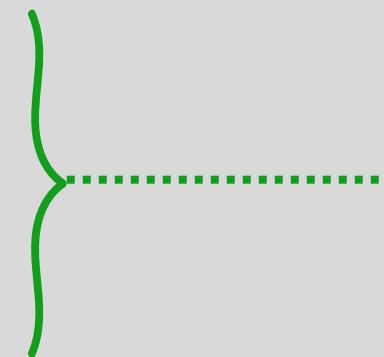
Rama TRUE  
Se ejecuta si condition es TRUE

Rama FALSE  
Se ejecuta si condition es FALSE

# • IF / ELSE / ELSE IF /

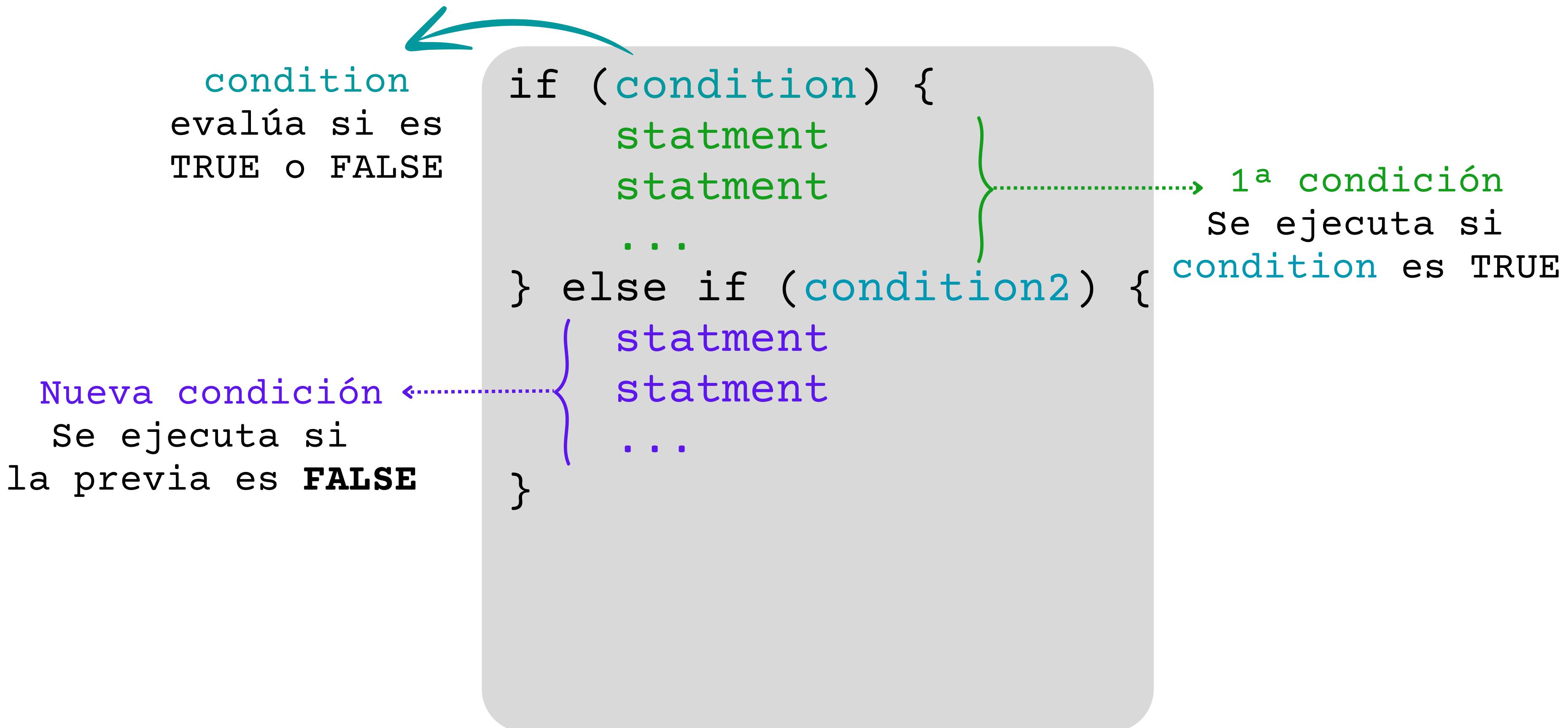
condition  
evalúa si es  
TRUE o FALSE

```
if (condition) {  
    statement  
    statement  
    ...  
}
```



1<sup>a</sup> condición  
Se ejecuta si  
**condition** es TRUE

# IF / ELSE / ELSE IF /



# IF / ELSE / ELSE IF /

condition  
evalúa si es  
TRUE o FALSE

Nueva condición  
Se ejecuta si  
la previa es FALSE

```
if (condition) {  
    statement  
    statement  
    ...  
} else if (condition2) {  
    statement  
    statement  
    ...  
} else {  
    statement  
    statement  
    ...  
}
```

1<sup>a</sup> condición  
Se ejecuta si  
condition es TRUE

Rama FALSE  
Se ejecuta si  
ninguna es TRUE

## Ejercicios

Evalúa si el paciente en la fila 22 es mayor de 60 años y muestra un mensaje para cualquier situación

```
indo_rct[22, "age"] <- 60

fila_paciente <- 22

if (indo_rctfila_paciente, "age") > 60) {
  print("Es mayor de 60 años")

} else if (indo_rctfila_paciente, "age") < 60) {
  print("Es menor de 60 años")

} else {
  print("Tiene 60 años")
}
```

<!--Estudio IMIBIC-->

# Enhorabuena {

<"Has superado la segunda  
sesión"/>

}

UCAIB BIOINFORMÁTICA Y  
BIOESTADÍSTICA  
(IMIBIC)

