

Repaso de R & Python

Sesión 2

Manipulación de Datos con DataFrames

Ricard Sierra Calls

Índice

- 1 Introducción a los DataFrames
- 2 Operaciones Básicas de Manipulación de Datos
- 3 Creación, Modificación y Eliminación de Columnas
- 4 Comparación entre Pandas y dplyr/tidyr

The diagram illustrates a tabular data structure with four rows and four columns. The columns are labeled 'Nombre', 'Edad', 'Grado', and 'Correo'. The rows are indexed from 1 to 4. Red arrows point to specific parts of the table: one arrow points to the first column with the label 'Nombres Columnas'; another arrow points to the first row with the label 'Nombres Filas'; a third arrow points to the first cell of the second row with the label 'Filas'; and a fourth arrow points to the last column with the label 'Columnas'.

	Nombre	Edad	Grado	Correo
1	María	18	Economía	maria@gmail.com
2	Luis	22	Medicina	luis@yahoo.es
3	Carmen	20	Arquitectura	carmen@gmail.com
4	Antonio	21	Economía	antonio@gmail.com

Estructuras de datos tabulares (**filas y columnas**), que son centrales en el análisis de datos.



(dplyr/tidyr)



Pandas



Cargar libreria

```
library(readr)
```

Leer un archivo CSV

```
df <- read_csv("datos.csv")
```

Guardar el DataFrame modificado

```
write_csv(df, "datos_modificados.csv")
```



Cargar libreria

```
import pandas as pd
```

Leer un archivo CSV

```
df = pd.read_csv('datos.csv')
```

Guardar el DataFrame modificado

```
df.to_csv('datos_modificados.csv',  
          index=False)
```



```
library(dplyr)

# Seleccionar columnas "nombre" y
"edad"

df_seleccion1 <- df$nombre
df_seleccion2 <- select(df, nombre,
edad)
# Filtrar filas donde la edad es mayor a 30
df_filtrado <- filter(df, edad > 30)
```



```
# Seleccionar columnas "nombre" y
"edad"

df_seleccion1 = df['nombre']
df_seleccion2 = df[['nombre', 'edad']]

# Filtrar filas donde la edad es mayor a 30
df_filtrado = df[df['edad'] > 30]
```



```
# Crear una nueva columna "ingreso_anual" a partir  
# de "ingreso_mensual"  
df <- df %>%  
  mutate(ingreso_anual = ingreso_mensual * 12)  
  
# Modificar la columna "nombre" para que esté en  
# mayúsculas  
df <- df %>%  
  mutate(nombre = toupper(nombre))  
  
# Eliminar la columna "ingreso_mensual"  
df <- df %>%  
  select(-ingreso_mensual)
```



```
# Crear una nueva columna "ingreso_anual" a  
# partir de la columna "ingreso_mensual"  
df['ingreso_anual'] = df['ingreso_mensual'] * 12  
  
# Modificar la columna "nombre" para que esté  
# en mayúsculas  
df['nombre'] = df['nombre'].str.upper()  
  
# Eliminar la columna "ingreso_mensual"  
df = df.drop(columns=['ingreso_mensual'])
```

dplyr/tidyr vs Pandas

Operación	Python (Pandas)	R (dplyr/tidyr)
Leer CSV	pd.read_csv('archivo.csv')	read_csv('archivo.csv')
Guardar CSV	df.to_csv('archivo.csv')	write_csv(df, 'archivo.csv')
Seleccionar columnas	df[['col1', 'col2']]	select(df, col1, col2)
Filtrar filas	df[df['col'] > valor]	filter(df, col > valor)
Crear/Modificar columna	df['new_col'] = ...	mutate(df, new_col = ...)
Eliminar columna	df.drop(columns=['col'])	select(df, -col)

ACTIVIDADES



Studio[®]



pythonTM