

Repaso de R & Python

Sesión 2

Manipulación de Datos con DataFrames

Ricard Sierra Calls

Índice

- ➊ Introducción a los DataFrames
- ➋ Operaciones Básicas de Manipulación de Datos
- ➌ Creación, Modificación y Eliminación de Columnas
- ➍ Comparación entre Pandas y dplyr/tidyr

The diagram illustrates the structure of a data table. It features a table with 4 rows and 4 columns. The columns are labeled 'Nombre', 'Edad', 'Grado', and 'Correo'. The rows are numbered 1 to 4. Red arrows point from labels to the corresponding parts of the table: 'Nombres Filas' points to the row numbers, 'Nombres Columnas' points to the column headers, and 'Columnas' points to the column headers. 'Filas' points to the row numbers. The table is enclosed in a red dashed border.

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

ACTIVIDADES

