

## **Ejercicios Python**

Solana Brun, Delfina Linck, Pilar Solorzano

BIG DREAMS

**Ejercicio 1.** Carga el archivo heart\_disease.csv en un DataFrame llamado d\_hd.

URL:

[https://raw.githubusercontent.com/fpineyro/homework-0/master/heart\\_disease.csv](https://raw.githubusercontent.com/fpineyro/homework-0/master/heart_disease.csv)

```
##Ejercicio 1
import pandas as pd
d_hd =
pd.read_csv('https://raw.githubusercontent.com/fpineyro/homework-0/master/heart_disease.csv')
d_hd
```

**Ejercicio 2.** Muestra los primeros 5 y los últimos 5 registros de d\_hd.

```
##Ejercicio 2
d_hd.head(5)
d_hd.tail(5)
```

**Ejercicio 3.** Selecciona e imprime las columnas age y has\_heart\_disease.

```
##Ejercicio 3
print(d_hd[['age', 'has_heart_disease']])
```

**Ejercicio 4.** Crea un filtro para encontrar todos los registros donde colesterol (serum\_cholesterol) sea mayor que 250.

```
##Ejercicio 4
filtro_colesterol = d_hd['serum_cholesterol'] > 250
d_hd[filtro_colesterol]
```

**Ejercicio 5.** Usa la función status() para identificar qué variables tienen valores nulos.

```
##Ejercicio 5
```

```
d_hd.isnull().sum()
```

**Ejercicio 6.** Reemplaza los valores nulos de la columna thal por un valor de 999.

```
##Ejercicio 6
d_hd['thal'] = d_hd['thal'].fillna(999)
print(d_hd['thal'].isnull().sum())
```

**Ejercicio 7.** Filtrar por múltiples condiciones:

Utiliza el DataFrame d\_hd y filtra para encontrar todos los registros de pacientes que tienen más de 60 años Y un nivel de colesterol (serum\_cholesterol) superior a 250.

```
##Ejercicio 7
filtro_edad = d_hd['age'] > 60
filtro_colesterol = d_hd['serum_cholesterol'] > 250
d_hd_filtrado = d_hd[filtro_edad & filtro_colesterol]
print(d_hd_filtrado)
```

**Ejercicio 8.** Reemplazar valores en una columna:

Selecciona la columna slope del DataFrame d\_hd. Reemplaza el valor 1 por el string 'ascendente' y el valor 2 por 'descendente'. Muestra el resultado (no necesitas guardarlo en el DataFrame original).

```
##Ejercicio 8
d_hd['slope'] = d_hd['slope'].replace({1: 'ascendente', 2:
'descendente'})
print(d_hd['slope'])
```

**Ejercicio 9.** Crear una nueva columna:

En el DataFrame d\_hd, crea una nueva columna llamada is\_healthy. Asigna el valor True si el paciente no tiene enfermedad del corazón (es decir, el valor en heart\_disease\_severity es 0) y False en caso contrario. Muestra el DataFrame con la nueva columna

```
##Ejercicio 9
d_hd['is_healthy'] = d_hd['heart_disease_severity'] == 0
```

```
print(d_hd)
```

**Ejercicio 10.** Eliminar registros con nulos:

Crea una copia del DataFrame `d_hd` llamada `d_hd_clean`. Usa el método `dropna()` para eliminar todos los registros que contengan al menos un valor nulo. Luego, usa `status()` para verificar que ya no hay valores nulos en `d_hd_clean`.

```
##Ejercicio 10
from funpymodeling.exploratory import status
d_hd_clean = d_hd.copy()
d_hd_clean.dropna(inplace=True)
status(d_hd_clean)
```