

EDA

Librerías:

```
import pandas as pd
```

MÉTODOS:

`pd.set_option('display.max_columns', None)`: visualizar el DataFrame con todas sus columnas

`df.head()`: muestra 5 **primeras filas**, o el número indicado entre los paréntesis

`df.tail()`: muestra 5 **últimas filas**

`df.sample(6)`: muestra 6 de forma **aleatoria**

`df.info()`: **información** del DataFrame

`df.duplicated().sum()`: indica los valores **duplicados, filas** con la misma info

`df.describe().T`: aporta **datos** estadísticos de la columnas numéricas como la media, la desviación estándar, los valores mínimo y máximo, los percentiles y más:

- count: número de valores no nulos
- mean: media
- std: desviación estándar, dispersión de los datos
- min: valor mínimo
- 25%: valor por debajo del que se encuentran el 25% del valor de la columna
- 50%: mediana, divide al conjunto de datos en dos mitades iguales
- 75%: valor por debajo del que se encuentran el 75% del valor de la columna
- max: valor máximo

`df.describe(include = "object").T`: aporta **datos** sobre las **columnas categóricas**:

- count: número de valores no nulos
- unique: cantidad valores únicos
- top: valor más común de la columna
- freq: frecuencia del valor más común

`df.shape[0]`: **número** de **filas**

`df.shape[1]`: **número** de **columnas**

`df.columns`: **nombres** de las **columnas**

`df['columna']`: saca todos los **datos** de la **columna**

`df['columna'].unique()`: saca los **datos únicos** de la columna

`df['columna'].value_counts()`: total de cada valor único de la columna

`df['columna'].info()`: **información** de la **columna**

`df.columns.get_loc("columna")`: **posición** de la **columna**

`df.select_dtypes(include=None, exclude=None)`: seleccionar **columnas** de un DataFrame **por su tipo de datos**

- include: tipos de datos a incluir
- exclude: tipos de datos a excluir
- Son opcionales
- Tipos de datos: 'int', 'float', 'object'

`df.drop(labels, axis=0, inplace=False)`: para **eliminar columnas** de un DataFrame

- labels: nombres de las filas o columnas a eliminar, un valor o lista
- axis: 0 filas, 1 columnas
- inplace: True eliminación en el DataFrame original, False nuevo DataFrame con los cambios. Opcional.

Datos nulos:

`df.isnull()`: devuelve una **serie booleana** con **datos nulos**

`df.isnull().sum()`: **número** total de **datos nulos** en el DataFrame

`df['columna'].isnull()`

`df['columna'].isnull().sum()`

`.isna()`: igual que `isnull`

`.notnull()`: muestra los **datos no nulos**

Valores duplicados:

`df.duplicated()`: devuelve una **serie booleana** con **datos duplicados**

`df.duplicated().sum()`: **número** total de **datos duplicados** en el DataFrame

`df.duplicated(subset = "columna").sum()`: **número** de **valores duplicados** en una **columna**