File Edit Selection View Go ··· TECLAB

analisis_acciones1.ipynb | analisis_iol.ipynb | Release Notes: 1.103.0 | API2.ipynb U | hogares_viviendas_superficie.csv U | analisis_acciones3.ipynb | analisis_acciones4.ipynb | analisis_acciones5.ipynb | analisis_acciones2.ipynb

Analisis_de_datos > API2.ipynb > Análisis y visualización de datos. > Actividad Nro. 2 > 3 - Revisión de los datos > 3.5 - Muestra un resumen estadistico de cada DataFrame

Generate + Code + Markdown | Run All | Restart | Clear All Outputs | Jupyter Variables | Outline ··· Python 3.13.3

Generate + Code + Markdown

Análisis y visualización de datos.

Actividad Nro. 2

1 - Librerias a utilizar

```
import pandas as pd
import numpy as np
import os
```

[3] ✓ 0.0s Python

2 - Carga de Bases de Datos

```
poblacion = pd.read_csv('poblacion.csv', encoding='latin-1')
esperanza = pd.read_csv('esperanza_de_vida.csv', encoding='latin-1')
hogares = pd.read_csv('hogares_viviendas_superficie.csv', encoding='latin-1')
```

[5] ✓ 0.0s Python

3 - Revisión de los datos

3.1 - Muestra las primeras filas de cada DataFrame

```
poblacion.head()
```

[8] ✓ 0.0s Python

	provincia_id	provincia	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2
0	2	Capital Federal	1150134	1423973	1082998	200
1	6	Buenos Aires	4789484	5377786	4425193	307571
2	10	Catamarca	96001	113634	89376	102602
3	14	Córdoba	1031843	1232211	978553	165321
4	18	Corrientes	267797	292644	248844	88199

main* Launchpad ⊗ 1 ⚠ 3 | Ln 1, Col 16 | Spaces: 4 | Spaces: 4 | CRLF | Cell 21 of 42 | Go Live

Search web & PC | ESP LAA | 13:29 5/9/2025

3 - Revisión de los datos

3.1 - Muestra las primeras filas de cada DataFrame

```
poblacion.head()
```

[8] ✓ 0.0s Python

	provincia_id	provincia	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2
0	2	Capital Federal	1150134	1423973	1082998	200
1	6	Buenos Aires	4789484	5377786	4425193	307571
2	10	Catamarca	96001	113634	89376	102602
3	14	Córdoba	1031843	1232211	978553	165321
4	18	Corrientes	267797	292644	248844	88199

```
esperanza.head()
```

[9] ✓ 0.0s Python

	provincia	anio	mujeres	varones
0	Buenos Aires	2015	80.22	73.54
1	Buenos Aires	2020	81.34	74.74
2	Buenos Aires	2025	82.32	75.80
3	Buenos Aires	2030	83.20	76.76
4	Buenos Aires	2035	83.98	77.60

```
hogares.head()
```

[10] ✓ 0.0s Python

	provincia_id	provincia	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2
0	2	Capital Federal	1150134	1423973	1082998	200
1	6	Buenos Aires	4789484	5377786	4425193	307571
2	10	Catamarca	96001	113634	89376	102602
3	14	Córdoba	1031843	1232211	978553	165321
4	18	Corrientes	267797	292644	248844	88199

3 - Revisión de los datos

3.1 - Muestra las primeras filas de cada DataFrame

```
poblacion.head()
```

[8] ✓ 0.0s Python

	provincia_id	provincia	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2
0	2	Capital Federal	1150134	1423973	1082998	200
1	6	Buenos Aires	4789484	5377786	4425193	307571
2	10	Catamarca	96001	113634	89376	102602
3	14	Córdoba	1031843	1232211	978553	165321
4	18	Corrientes	267797	292644	248844	88199

```
esperanza.head()
```

[9] ✓ 0.0s Python

	provincia	anio	mujeres	varones
0	Buenos Aires	2015	80.22	73.54
1	Buenos Aires	2020	81.34	74.74
2	Buenos Aires	2025	82.32	75.80
3	Buenos Aires	2030	83.20	76.76
4	Buenos Aires	2035	83.98	77.60

```
hogares.head()
```

[10] ✓ 0.0s Python

	provincia_id	provincia	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2
0	2	Capital Federal	1150134	1423973	1082998	200
1	6	Buenos Aires	4789484	5377786	4425193	307571
2	10	Catamarca	96001	113634	89376	102602
3	14	Córdoba	1031843	1232211	978553	165321
4	18	Corrientes	267797	292644	248844	88199

```
hogares.head()
```

[10] ✓ 0.0s Python

	provincia_id	provincia	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2
0	2	Capital Federal	1150134	1423973	1082998	200
1	6	Buenos Aires	4789484	5377786	4425193	307571
2	10	Catamarca	96001	113634	89376	102602
3	14	Córdoba	1031843	1232211	978553	165321
4	18	Corrientes	267797	292644	248844	88199

3.2 - Muestra las ultimas filas de cada DataFrame

```
poblacion.tail()
```

[11] ✓ 0.0s Python

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres
770	Tierra del Fuego	2036	241593	122567	119026
771	Tierra del Fuego	2037	245734	124625	121109
772	Tierra del Fuego	2038	249853	126670	123183
773	Tierra del Fuego	2039	253948	128702	125246
774	Tierra del Fuego	2040	258020	130721	127299

```
esperanza.tail()
```

[12] ✓ 0.0s Python

	provincia	anio	mujeres	varones
139	Tucumán	2020	81.05	75.11
140	Tucumán	2025	82.11	76.15
141	Tucumán	2030	83.03	77.07
142	Tucumán	2035	83.84	77.88
143	Tucumán	2040	84.54	78.58

```
esperanza.dtypes
```

[20] ✓ 0.0s — Python

```
provincia     object
anio           int64
mujeres      float64
varones      float64
dtype: object
```

```
hogares.dtypes
```

[22] ✓ 0.0s — Python

```
provincia_id                    int64
provincia                      object
hogares                         int64
viviendas_particulares          int64
viviendas_particulares_habitadas    int64
superficie_km2                  int64
dtype: object
```

3.5 - Muestra un resumen estadistico de cada DataFrame

```
poblacion.describe()
```

[23] ✓ 0.3s — Python

	anio	poblacion_total	poblacion_varones	poblacion_mujeres
count	775.000000	7.750000e+02	7.750000e+02	7.750000e+02
mean	2025.000000	3.777746e+06	1.856888e+06	1.920858e+06
std	8.950048	9.560571e+06	4.699604e+06	4.861043e+06
min	2010.000000	1.316610e+05	6.723500e+04	6.442600e+04
25%	2017.000000	5.845510e+05	2.906740e+05	2.934905e+05
50%	2025.000000	1.017731e+06	5.061010e+05	5.161370e+05
75%	2033.000000	1.855285e+06	9.138865e+05	9.404745e+05
max	2040.000000	5.277848e+07	2.603809e+07	2.674038e+07

```
esperanza.describe()
```

```
esperanza.dtypes
```

[20] ✓ 0.0s — Python

```
provincia     object
anio           int64
mujeres      float64
varones      float64
dtype: object
```

```
hogares.dtypes
```

[22] ✓ 0.0s — Python

```
provincia_id                    int64
provincia                      object
hogares                         int64
viviendas_particulares          int64
viviendas_particulares_habitadas    int64
superficie_km2                  int64
dtype: object
```

3.5 - Muestra un resumen estadistico de cada DataFrame

```
poblacion.describe()
```

[23] ✓ 0.3s — Python

	anio	poblacion_total	poblacion_varones	poblacion_mujeres
count	775.000000	7.750000e+02	7.750000e+02	7.750000e+02
mean	2025.000000	3.777746e+06	1.856888e+06	1.920858e+06
std	8.950048	9.560571e+06	4.699604e+06	4.861043e+06
min	2010.000000	1.316610e+05	6.723500e+04	6.442600e+04
25%	2017.000000	5.845510e+05	2.906740e+05	2.934905e+05
50%	2025.000000	1.017731e+06	5.061010e+05	5.161370e+05
75%	2033.000000	1.855285e+06	9.138865e+05	9.404745e+05
max	2040.000000	5.277848e+07	2.603809e+07	2.674038e+07

```
esperanza.describe()
```

File Edit Selection View Go ··· TECLAB

analisis_acciones1.ipynb | analisis_iol.ipynb | Release Notes: 1.103.0 | API2.ipynb U | hogares_viviendas_superficie.csv U | analisis_acciones3.ipynb | analisis_acciones4.ipynb | analisis_acciones5.ipynb | analisis_acciones2.ipynb

Analisis_de_datos > API2.ipynb > Análisis y visualización de datos. > Actividad Nro. 2 > 3 - Revisión de los datos > 3.5 - Muestra un resumen estadistico de cada DataFrame

Generate + Code + Markdown | Run All | Restart | Clear All Outputs | Jupyter Variables | Outline ··· Python 3.13.3

3.6 - Aplicamos filtrado en DataFrame 'población' para excluir 'Total país'

```
poblacion.groupby('provincia')['provincia'].count()
```

[30] ✓ 0.0s Python

```
provincia
Buenos Aires          31
Capital Federal       31
Catamarca             31
Chaco                 31
Chubut                31
Corrientes            31
Córdoba               31
Entre Ríos            31
Formosa               31
Jujuy                 31
La Pampa              31
La Rioja              31
Mendoza               31
Misiones              31
Neuquén               31
Río Negro             31
Salta                 31
San Juan              31
San Luis              31
Santa Cruz            31
Santa Fe              31
Santiago del Estero   31
Tierra del Fuego      31
Total País            31
Tucumán               31
Name: provincia, dtype: int64
```

```
poblacion_filtrado = poblacion[poblacion['provincia'] != 'Total País']
poblacion_filtrado
```

[34] ✓ 0.0s Python

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres
31	Capital Federal	2010	3028481	1405566	1622915
32	Capital Federal	2011	3033639	1409835	1623804

main* Launchpad ⊗ 1 ⚠ 3 | Ln 1, Col 19 | Spaces: 4 | Spaces: 4 | CRLF | Cell 21 of 42 | Go Live

Search web & PC | ESP LAA | 13:36 5/9/2025

File Edit Selection View Go ··· TECLAB

analisis_acciones1.ipynb | analisis_iol.ipynb | Release Notes: 1.103.0 | API2.ipynb U | hogares_viviendas_superficie.csv U | analisis_acciones3.ipynb | analisis_acciones4.ipynb | analisis_acciones5.ipynb | analisis_acciones2.ipynb

Analisis_de_datos > API2.ipynb > Análisis y visualización de datos. > Actividad Nro. 2 > 3 - Revisión de los datos > 3.5 - Muestra un resumen estadistico de cada DataFrame

Generate + Code + Markdown | Run All | Restart | Clear All Outputs | Jupyter Variables | Outline ··· Python 3.13.3

3.6 - Aplicamos filtrado en DataFrame 'población' para excluir 'Total país'

```
poblacion.groupby('provincia')['provincia'].count()
```

[30] ✓ 0.0s Python

```
provincia
Buenos Aires          31
Capital Federal       31
Catamarca             31
Chaco                 31
Chubut                31
Corrientes            31
Córdoba               31
Entre Ríos            31
Formosa               31
Jujuy                 31
La Pampa              31
La Rioja              31
Mendoza               31
Misiones              31
Neuquén               31
Río Negro             31
Salta                 31
San Juan              31
San Luis              31
Santa Cruz            31
Santa Fe              31
Santiago del Estero   31
Tierra del Fuego      31
Total País            31
Tucumán               31
Name: provincia, dtype: int64
```

```
poblacion_filtrado = poblacion[poblacion['provincia'] != 'Total País']
poblacion_filtrado
```

[34] ✓ 0.0s Python

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres
31	Capital Federal	2010	3028481	1405566	1622915
32	Capital Federal	2011	3033639	1409835	1623804

main* Launchpad ⊗ 1 ⚠ 3 | Ln 1, Col 19 | Spaces: 4 | Spaces: 4 | CRLF | Cell 21 of 42 | Go Live

Search web & PC | ESP LAA | 13:36 5/9/2025

3.7 - Creación un nuevo DataFrame por medio de la union de población_filtrado y hogares

```
poblac_superficie = pd.merge(poblacion_filtrado, hogares, left_on='provincia', right_on='provincia', how='left')
poblac_superficie
```

[35] ✓ 0.1s Python

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres	provincia_id	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2
0	Capital Federal	2010	3028481	1405566	1622915	2	1150134	1423973	1082998	200
1	Capital Federal	2011	3033639	1409835	1623804	2	1150134	1423973	1082998	200
2	Capital Federal	2012	3038860	1414105	1624755	2	1150134	1423973	1082998	200
3	Capital Federal	2013	3044076	1418339	1625737	2	1150134	1423973	1082998	200
4	Capital Federal	2014	3049229	1422507	1626722	2	1150134	1423973	1082998	200
...	...	...	...	...	...	...	...	...	...	...
739	Tierra del Fuego	2036	241593	122567	119026	94	38956	43360	36689	1002445
740	Tierra del Fuego	2037	245734	124625	121109	94	38956	43360	36689	1002445
741	Tierra del Fuego	2038	249853	126670	123183	94	38956	43360	36689	1002445
742	Tierra del Fuego	2039	253948	128702	125246	94	38956	43360	36689	1002445
743	Tierra del Fuego	2040	258020	130721	127299	94	38956	43360	36689	1002445

744 rows × 10 columns

3.8 - Calculo de la densidad y se agrega resultado al DataFrame (población total / superficie)

```
poblac_superficie['densidad_poblacion'] = poblac_superficie['poblacion_total'] / poblac_superficie['superficie_km2']
poblac_superficie
```

[37] ✓ 0.0s Python

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres	provincia_id	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2	densidad_poblacion
0	Capital Federal	2010	3028481	1405566	1622915	2	1150134	1423973	1082998	200	15142.405000
1	Capital Federal	2011	3033639	1409835	1623804	2	1150134	1423973	1082998	200	15168.195000
2	Capital Federal	2012	3038860	1414105	1624755	2	1150134	1423973	1082998	200	15194.300000
3	Capital Federal	2013	3044076	1418339	1625737	2	1150134	1423973	1082998	200	15220.380000
4	Capital Federal	2014	3049229	1422507	1626722	2	1150134	1423973	1082998	200	15246.145000

3.7 - Creación un nuevo DataFrame por medio de la union de población_filtrado y hogares

```
poblac_superficie = pd.merge(poblacion_filtrado, hogares, left_on='provincia', right_on='provincia', how='left')
poblac_superficie
```

[35] ✓ 0.1s Python

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres	provincia_id	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2
0	Capital Federal	2010	3028481	1405566	1622915	2	1150134	1423973	1082998	200
1	Capital Federal	2011	3033639	1409835	1623804	2	1150134	1423973	1082998	200
2	Capital Federal	2012	3038860	1414105	1624755	2	1150134	1423973	1082998	200
3	Capital Federal	2013	3044076	1418339	1625737	2	1150134	1423973	1082998	200
4	Capital Federal	2014	3049229	1422507	1626722	2	1150134	1423973	1082998	200
...	...	...	...	...	...	...	...	...	...	...
739	Tierra del Fuego	2036	241593	122567	119026	94	38956	43360	36689	1002445
740	Tierra del Fuego	2037	245734	124625	121109	94	38956	43360	36689	1002445
741	Tierra del Fuego	2038	249853	126670	123183	94	38956	43360	36689	1002445
742	Tierra del Fuego	2039	253948	128702	125246	94	38956	43360	36689	1002445
743	Tierra del Fuego	2040	258020	130721	127299	94	38956	43360	36689	1002445

744 rows × 10 columns

3.8 - Calculo de la densidad y se agrega resultado al DataFrame (población total / superficie)

```
poblac_superficie['densidad_poblacion'] = poblac_superficie['poblacion_total'] / poblac_superficie['superficie_km2']
poblac_superficie
```

[37] ✓ 0.0s Python

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres	provincia_id	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2	densidad_poblacion
0	Capital Federal	2010	3028481	1405566	1622915	2	1150134	1423973	1082998	200	15142.405000
1	Capital Federal	2011	3033639	1409835	1623804	2	1150134	1423973	1082998	200	15168.195000
2	Capital Federal	2012	3038860	1414105	1624755	2	1150134	1423973	1082998	200	15194.300000
3	Capital Federal	2013	3044076	1418339	1625737	2	1150134	1423973	1082998	200	15220.380000
4	Capital Federal	2014	3049229	1422507	1626722	2	1150134	1423973	1082998	200	15246.145000

3.8 - Calculo de la densidad y se agrega resultado al DataFrame (población total / superficie)

```
poblac_superficie['densidad_poblacion'] = poblac_superficie['poblacion_total'] / poblac_superficie['superficie_km2']
poblac_superficie
```

[37] ✓ 0.0s Python

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres	provincia_id	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2	densidad_poblacion
0	Capital Federal	2010	3028481	1405566	1622915	2	1150134	1423973	1082998	200	15142.405000
1	Capital Federal	2011	3033639	1409835	1623804	2	1150134	1423973	1082998	200	15168.195000
2	Capital Federal	2012	3038860	1414105	1624755	2	1150134	1423973	1082998	200	15194.300000
3	Capital Federal	2013	3044076	1418339	1625737	2	1150134	1423973	1082998	200	15220.380000
4	Capital Federal	2014	3049229	1422507	1626722	2	1150134	1423973	1082998	200	15246.145000
...	...	...	...	...	...	...	...	...	...	...	...
739	Tierra del Fuego	2036	241593	122567	119026	94	38956	43360	36689	1002445	0.241004
740	Tierra del Fuego	2037	245734	124625	121109	94	38956	43360	36689	1002445	0.245135
741	Tierra del Fuego	2038	249853	126670	123183	94	38956	43360	36689	1002445	0.249244
742	Tierra del Fuego	2039	253948	128702	125246	94	38956	43360	36689	1002445	0.253329
743	Tierra del Fuego	2040	258020	130721	127299	94	38956	43360	36689	1002445	0.257391

744 rows × 11 columns

3.8 - Comprobar si hay datos nulos

```
poblac_superficie.isnull().sum()
```

[38] ✓ 0.0s Python

```
provincia                       0
anio                            0
poblacion_total                 0
poblacion_varones               0
poblacion_mujeres               0
provincia_id                    0
hogares                         0
viviendas_particulares          0
viviendas_particulares_habitadas    0
superficie_km2                  0
```

3.8 - Comprobar si hay datos nulos

```
poblac_superficie.isnull().sum()
```

[38] ✓ 0.0s Python

```
provincia                       0
anio                            0
poblacion_total                 0
poblacion_varones               0
poblacion_mujeres               0
provincia_id                    0
hogares                         0
viviendas_particulares          0
viviendas_particulares_habitadas    0
superficie_km2                  0
densidad_poblacion              0
dtype: int64
```

3.9 - Calculo del z-score de la densidad_población y agregar resultado en una nueva columna 'Fuera de rango'

```
poblac_superficie['fuera_de_rango'] = (poblac_superficie['densidad_poblacion'] - np.mean(poblac_superficie['densidad_poblacion'])) / np.std(poblac_superficie['densidad_poblacion'])
poblac_superficie
```

[39] ✓ 0.0s Python

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres	provincia_id	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2	densidad_poblacion	fuera_de_rango
0	Capital Federal	2010	3028481	1405566	1622915	2	1150134	1423973	1082998	200	15142.405000	4.731966
1	Capital Federal	2011	3033639	1409835	1623804	2	1150134	1423973	1082998	200	15168.195000	4.740389
2	Capital Federal	2012	3038860	1414105	1624755	2	1150134	1423973	1082998	200	15194.300000	4.748916
3	Capital Federal	2013	3044076	1418339	1625737	2	1150134	1423973	1082998	200	15220.380000	4.757434
4	Capital Federal	2014	3049229	1422507	1626722	2	1150134	1423973	1082998	200	15246.145000	4.765849
...	...	...	...	...	...	...	...	...	...	...	...	...
739	Tierra del Fuego	2036	241593	122567	119026	94	38956	43360	36689	1002445	0.241004	-0.213774
740	Tierra del Fuego	2037	245734	124625	121109	94	38956	43360	36689	1002445	0.245135	-0.213773
741	Tierra del Fuego	2038	249853	126670	123183	94	38956	43360	36689	1002445	0.249244	-0.213771
742	Tierra del Fuego	2039	253948	128702	125246	94	38956	43360	36689	1002445	0.253329	-0.213770
743	Tierra del Fuego	2040	258020	130721	127299	94	38956	43360	36689	1002445	0.257391	-0.213769

744 rows × 12 columns

3.10 - Calculo el percentil de la columna 'Fuera de rango'

```
[41] p99 = np.percentile(poblac_superficie['fuera_de_rango'],99)
     p99
     ✓ 0.1s                                                    Python
```

```
np.float64(4.821306724666531)
```

```
[43] poblac_superficie[poblac_superficie['fuera_de_rango'] >= p99]
     ✓ 0.0s                                                    Python
```

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres	provincia_id	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2	densidad_poblacion	fuera_de_rango
13	Capital Federal	2023	3083770	1452588	1631182	2	1150134	1423973	1082998	200	15418.850	4.822258
14	Capital Federal	2024	3085483	1454716	1630767	2	1150134	1423973	1082998	200	15427.415	4.825056
15	Capital Federal	2025	3086680	1456560	1630120	2	1150134	1423973	1082998	200	15433.400	4.827011
16	Capital Federal	2026	3087338	1458111	1629227	2	1150134	1423973	1082998	200	15436.690	4.828085
17	Capital Federal	2027	3087434	1459359	1628075	2	1150134	1423973	1082998	200	15437.170	4.828242
18	Capital Federal	2028	3086973	1460309	1626664	2	1150134	1423973	1082998	200	15434.865	4.827489
19	Capital Federal	2029	3085971	1460970	1625001	2	1150134	1423973	1082998	200	15429.855	4.825853
20	Capital Federal	2030	3084450	1461355	1623095	2	1150134	1423973	1082998	200	15422.250	4.823369

4 - Genera un informe breve sobre una de las variables originales (por ejemplo, superficie), con algunas medidas resumen (percentiles, promedio, etc.) y tus respuestas.

```
[44] import numpy as np

     # Seleccionamos la variable de interés
     superficie = poblac_superficie["superficie_km2"]

     # Medidas descriptivas
     resumen = {
         "Promedio": superficie.mean(),
         "Mediana (P50)": np.percentile(superficie, 50),
         "Percentil 25 (Q1)": np.percentile(superficie, 25),
         "Percentil 75 (Q3)": np.percentile(superficie, 75),
         "Mínimo": superficie.min(),
         "Máximo": superficie.max(),
```

File Edit Selection View Go ··· TECLAB

analisis_acciones1.ipynb | analisis_iol.ipynb | Release Notes: 1.103.0 | API2.ipynb U | hogares_viviendas_superficie.csv U | analisis_acciones3.ipynb | analisis_acciones4.ipynb | analisis_acciones5.ipynb | analisis_acciones2.ipynb

Analisis_de_datos > API2.ipynb > Análisis y visualización de datos. > Actividad Nro. 2 > 3 - Revisión de los datos > 3.5 - Muestra un resumen estadistico de cada DataFrame

Generate | Code | Markdown | Run All | Restart | Clear All Outputs | Jupyter Variables | Outline | ··· | Python 3.13.3

4 - Genera un informe breve sobre una de las variables originales (por ejemplo, superficie), con algunas medidas resumen (percentiles, promedio, etc.) y tus respuestas.

```python
import numpy as np

# Seleccionamos la variable de interés
superficie = poblac_superficie["superficie_km2"]

# Medidas descriptivas
resumen = {
    "Promedio": superficie.mean(),
    "Mediana (P50)": np.percentile(superficie, 50),
    "Percentil 25 (Q1)": np.percentile(superficie, 25),
    "Percentil 75 (Q3)": np.percentile(superficie, 75),
    "Mínimo": superficie.min(),
    "Máximo": superficie.max(),
    "Desviación estándar": superficie.std()
}

print(resumen)
```

[44] ✓ 0.0s Python

```
{'Promedio': np.float64(156719.75), 'Mediana (P50)': np.float64(101117.5), 'Percentil 25 (Q1)': np.float64(78272.75), 'Percentil 75 (Q3)': np.float64(157946.25), 'Mínimo': np.int64(200), 'Máximo': np.int64(1002445), 'Desviación estándar': np
```

```python
poblac_superficie["superficie_km2"].describe()
```

[45] ✓ 0.2s Python

```
count    7.440000e+02
mean     1.567198e+05
std      1.901313e+05
min      2.000000e+02
25%      7.827275e+04
50%      1.011175e+05
75%      1.579462e+05
max      1.002445e+06
Name: superficie_km2, dtype: float64
```

📊 Informe de la variable superficie_km2

main* | Launchpad | ⊗ 1 ⚠ 3 | Ln 2, Col 18 | Spaces: 4 | Spaces: 4 | CRLF | Cell 21 of 42 | Go Live

Search web & PC | ESP LAA | 13:37 5/9/2025