

Importing Libraries

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

Importing Datasets

```
In [4]: df=pd.read_csv(r"C:\Users\user\Downloads\stations.csv")
df
```

```
Out[4]:
```

	id	name	address	lon	lat	elevation
0	28079004	Pza. de España	Plaza de España	-3.712247	40.423853	635
1	28079008	Escuelas Aguirre	Entre C/ Alcalá y C/ O' Donell	-3.682319	40.421564	670
2	28079011	Avda. Ramón y Cajal	Avda. Ramón y Cajal esq. C/ Príncipe de Vergara	-3.677356	40.451475	708
3	28079016	Arturo Soria	C/ Arturo Soria esq. C/ Vizconde de los Asilos	-3.639233	40.440047	693
4	28079017	Villaverde	C/. Juan Peñalver	-3.713322	40.347139	604
5	28079018	Farolillo	Calle Farolillo - C/Ervigio	-3.731853	40.394781	630
6	28079024	Casa de Campo	Casa de Campo (Terminal del Teleférico)	-3.747347	40.419356	642
7	28079027	Barajas Pueblo	C/. Júpiter, 21 (Barajas)	-3.580031	40.476928	621
8	28079035	Pza. del Carmen	Plaza del Carmen esq. Tres Cruces.	-3.703172	40.419208	659
9	28079036	Moratalaz	Avd. Moratalaz esq. Camino de los Vinateros	-3.645306	40.407947	685
10	28079038	Cuatro Caminos	Avda. Pablo Iglesias esq. C/ Marqués de Lema	-3.707128	40.445544	698
11	28079039	Barrio del Pilar	Avd. Betanzos esq. C/ Monforte de Lemos	-3.711542	40.478228	674
12	28079040	Vallecas	C/ Arroyo del Olivar esq. C/ Río Grande.	-3.651522	40.388153	677
13	28079047	Mendez Alvaro	C/ Juan de Mariana / Pza. Amanecer Mendez Alvaro	-3.686825	40.398114	599
14	28079048	Castellana	C/ Jose Gutierrez Abascal	-3.690367	40.439897	676
15	28079049	Parque del Retiro	Paseo Venezuela- Casa de Vacas	-3.682583	40.414444	662
16	28079050	Plaza Castilla	Plaza Castilla (Canal)	-3.688769	40.465572	728
17	28079054	Ensanche de Vallecas	Avda La Gavia / Avda. Las Suertes	-3.612117	40.372933	627

	id	name	address	lon	lat	elevation
18	28079055	Urb. Embajada	C/ Riaño (Barajas)	-3.580747	40.462531	618
19	28079056	Pza. Fernández Ladreda	Pza. Fernández Ladreda - Avda. Oporto	-3.718728	40.384964	604
20	28079057	Sanchinarro	C/ Princesa de Eboli esq C/ Maria Tudor	-3.660503	40.494208	700
21	28079058	El Pardo	Avda. La Guardia	-3.774611	40.518058	615
22	28079059	Juan Carlos I	Parque Juan Carlos I (frente oficinas mantenim...	-3.609072	40.465250	660
23	28079060	Tres Olivos	Plaza Tres Olivos	-3.689761	40.500589	715

Data Cleaning and Data Preprocessing

In [5]: `df=df.dropna()`

In [6]: `df.columns`

Out[6]: Index(['id', 'name', 'address', 'lon', 'lat', 'elevation'], dtype='object')

In [7]: `df.info()`

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 24 entries, 0 to 23
Data columns (total 6 columns):
#   Column      Non-Null Count  Dtype
---  -
0   id          24 non-null     int64
1   name        24 non-null     object
2   address     24 non-null     object
3   lon         24 non-null     float64
4   lat         24 non-null     float64
5   elevation   24 non-null     int64
dtypes: float64(2), int64(2), object(2)
memory usage: 1.3+ KB
```

In [11]: `data=df[['id','elevation']]`
`data`

Out[11]:

	id	elevation
--	----	-----------

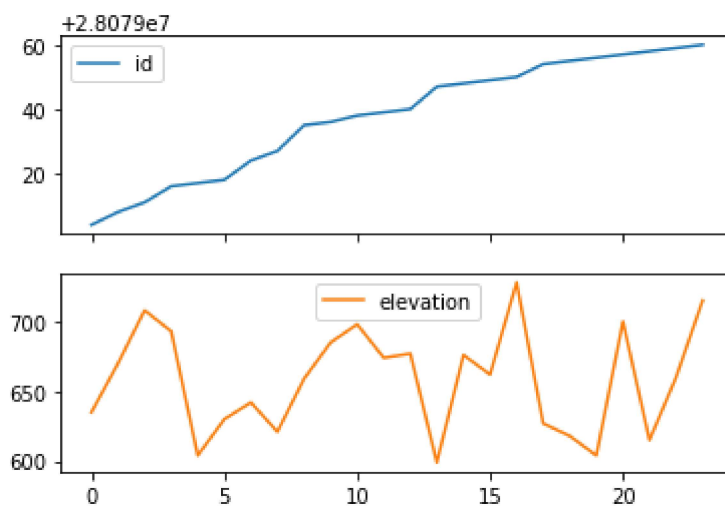
0	28079004	635
1	28079008	670
2	28079011	708
3	28079016	693
4	28079017	604
5	28079018	630

	id	elevation
6	28079024	642
7	28079027	621
8	28079035	659
9	28079036	685
10	28079038	698
11	28079039	674
12	28079040	677
13	28079047	599
14	28079048	676
15	28079049	662
16	28079050	728
17	28079054	627
18	28079055	618
19	28079056	604
20	28079057	700
21	28079058	615
22	28079059	660
23	28079060	715

Line chart

```
In [12]: data.plot.line(subplots=True)
```

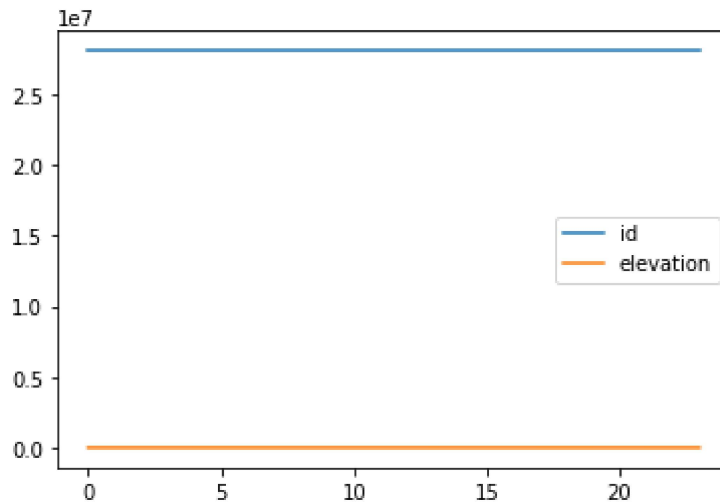
```
Out[12]: array([<AxesSubplot:~>, <AxesSubplot:~>], dtype=object)
```



Line chart

```
In [13]: data.plot.line()
```

```
Out[13]: <AxesSubplot:>
```

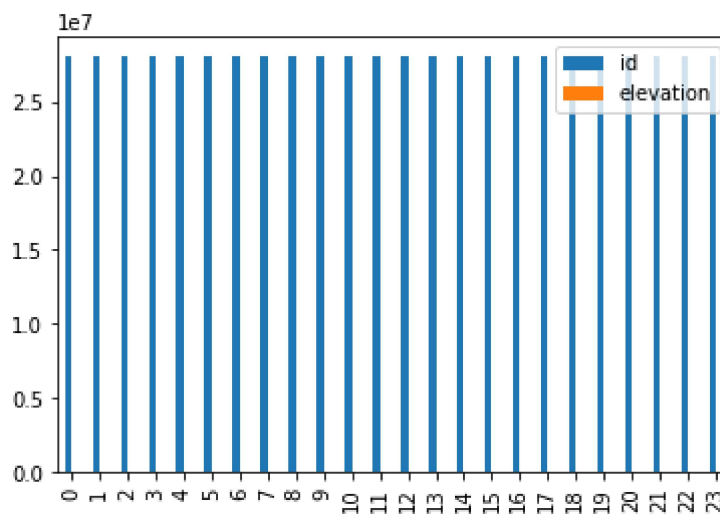


Bar chart

```
In [14]: b=data[0:50]
```

```
In [15]: b.plot.bar()
```

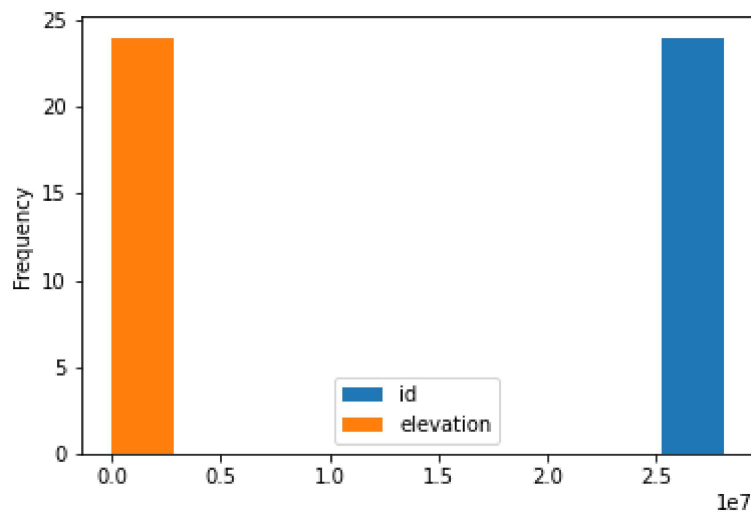
```
Out[15]: <AxesSubplot:>
```



Histogram

```
In [16]: data.plot.hist()
```

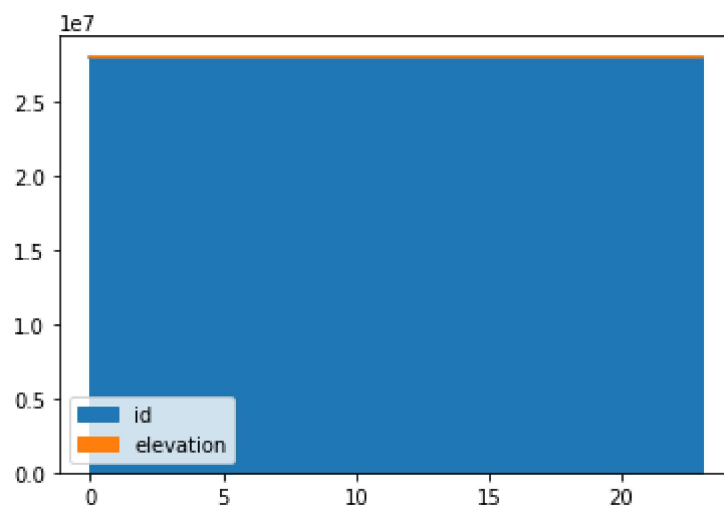
```
Out[16]: <AxesSubplot:ylabel='Frequency'>
```



Area chart

```
In [17]: data.plot.area()
```

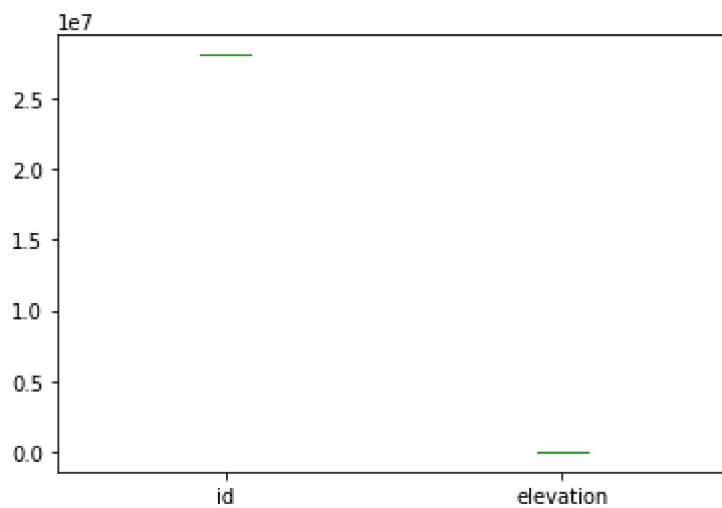
```
Out[17]: <AxesSubplot:>
```



Box chart

```
In [18]: data.plot.box()
```

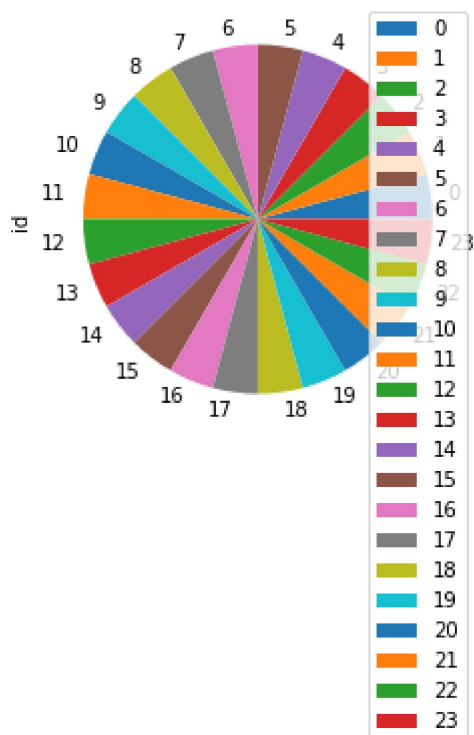
```
Out[18]: <AxesSubplot:>
```



Pie chart

In [21]: `b.plot.pie(y='id')`

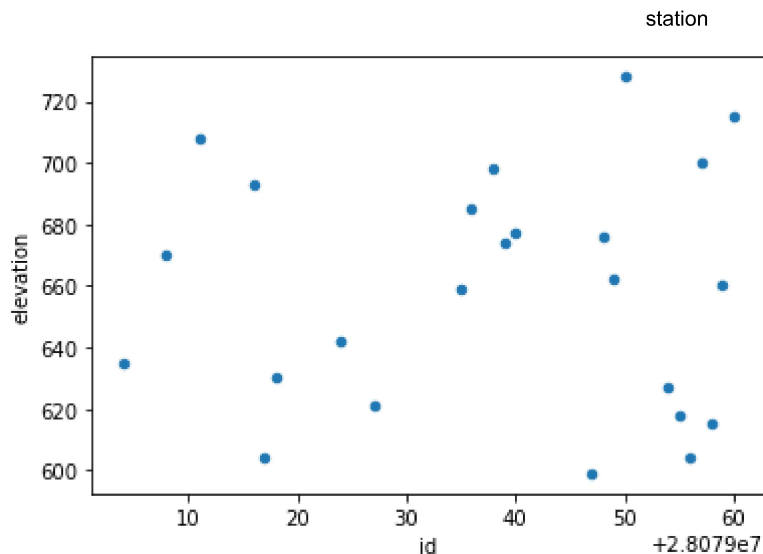
Out[21]: `<AxesSubplot:ylabel='id'>`



Scatter chart

In [23]: `data.plot.scatter(x='id' ,y='elevation')`

Out[23]: `<AxesSubplot:xlabel='id', ylabel='elevation'>`



In [24]: `df.info()`

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 24 entries, 0 to 23
Data columns (total 6 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   id          24 non-null    int64
1   name        24 non-null    object
2   address     24 non-null    object
3   lon         24 non-null    float64
4   lat         24 non-null    float64
5   elevation   24 non-null    int64
dtypes: float64(2), int64(2), object(2)
memory usage: 1.3+ KB
```

In [25]: `df.columns`

Out[25]: `Index(['id', 'name', 'address', 'lon', 'lat', 'elevation'], dtype='object')`

In [26]: `df.describe()`

Out[26]:

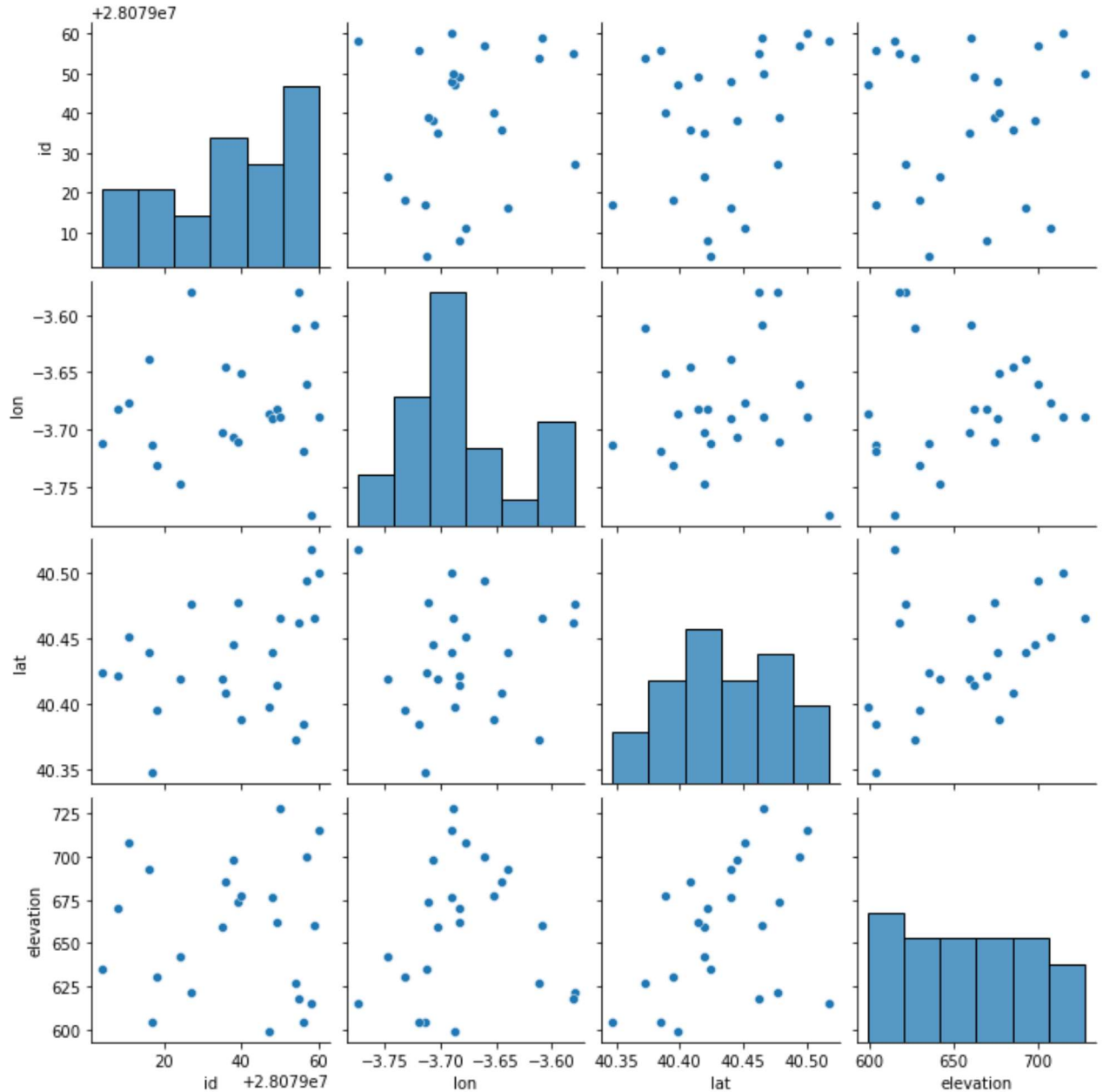
	id	lon	lat	elevation
count	2.400000e+01	24.000000	24.000000	24.000000
mean	2.807904e+07	-3.679019	40.434616	658.333333
std	1.799094e+01	0.049324	0.043022	38.295949
min	2.807900e+07	-3.774611	40.347139	599.000000
25%	2.807902e+07	-3.711718	40.405489	625.500000
50%	2.807904e+07	-3.687797	40.431875	661.000000
75%	2.807905e+07	-3.649968	40.465331	687.000000
max	2.807906e+07	-3.580031	40.518058	728.000000

In [27]: `df1=df[['id', 'name', 'address', 'lon', 'lat', 'elevation']]`

EDA AND VISUALIZATION

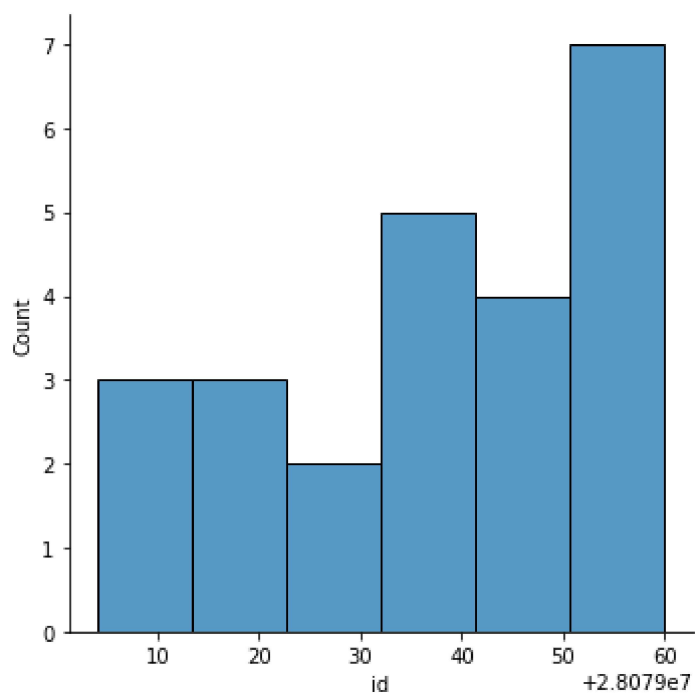
```
In [28]: sns.pairplot(df1[0:50])
```

```
Out[28]: <seaborn.axisgrid.PairGrid at 0x2481992b9d0>
```



```
In [31]: sns.displot(df['id'])
```

```
Out[31]: <seaborn.axisgrid.FacetGrid at 0x2481a5e05e0>
```

```
In [32]: sns.heatmap(df1.corr())
```

```
Out[32]: <AxesSubplot:>
```

