In [1]:
```
import pandas as pd
import numpy as np
```

In [2]:
```
df = pd.read_csv("income.csv", names=["name","income"], skiprows=[0])
df
```

Out[2]:

	name	income
0	Rob	5000
1	Rafiq	6000
2	Nina	4000
3	Sofia	7500
4	Mohan	8000
5	Tao	7000
6	Elon Musk	10000000

In [12]:
```
df.income.describe()
```

Out[12]:
```
count    7.000000e+00
mean     1.433929e+06
std      3.777283e+06
min      4.000000e+03
25%      5.500000e+03
50%      7.000000e+03
75%      7.750000e+03
max      1.000000e+07
Name: income, dtype: float64
```

In [54]:
```
df.income.quantile(0)
```

Out[54]:
```
4000.0
```

In [63]:
```
df.income.quantile(0.25,interpolation="higher")
```

Out[63]:
```
6000
```

In [62]:
```
df.income.quantile(0.5,interpolation="higher")
```

Out[62]:
```
7000
```

In [53]:
```
df.income.quantile(0.75)
```

Out[53]:
```
7750.0
```

In [55]:
```
df.income.quantile(1)
```

Out[55]:
```
10000000.0
```

In [72]:
```
percentile_99 = df.income.quantile(0.99)
percentile_99
```

Out[72]:
```
9400479.999999994
```

In [73]:
```
df[df.income>percentile_99]
```

Out[73]:

	name	income
6	Elon Musk	10000000

In [74]:
```
df
```

Out[74]:

	name	income
0	Rob	5000
1	Rafiq	6000
2	Nina	4000
3	Sofia	7500
4	Mohan	8000
5	Tao	7000
6	Elon Musk	10000000

In [108…
```
df['income'][3]=np.NaN
```

```
<ipython-input-108-61710ea50a82>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/us
er_guide/indexing.html#returning-a-view-versus-a-copy
  df['income'][3]=np.NaN
C:\Program Files\Python38\lib\site-packages\pandas\core\indexing.py:205: SettingWith
CopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/us
er_guide/indexing.html#returning-a-view-versus-a-copy
  self._setitem_with_indexer(indexer, value)
```

In [109…
```
df
```

Out[109…

	name	income
0	Rob	5000.0
1	Rafiq	6000.0
2	Nina	4000.0
3	Sofia	NaN
4	Mohan	8000.0
5	Tao	7000.0
6	Elon Musk	10000000.0

In [112…
```
df.income.mean()
```

Out[112… 1671666.6666666667

In [110…
```
df_new = df.fillna(df.income.mean())
df_new
```

Out[110…

	name	income

	name	income
0	Rob	5.000000e+03
1	Rafiq	6.000000e+03
2	Nina	4.000000e+03
3	Sofia	1.671667e+06
4	Mohan	8.000000e+03
5	Tao	7.000000e+03
6	Elon Musk	1.000000e+07

In [111…

```
df_new = df.fillna(df.income.median())
df_new
```

Out[111…

	name	income
0	Rob	5000.0
1	Rafiq	6000.0
2	Nina	4000.0
3	Sofia	6500.0
4	Mohan	8000.0
5	Tao	7000.0
6	Elon Musk	10000000.0