1. Data Loading

In [2]:
```
import pandas as pd
```

In [3]:
```
df=pd.read_csv("data.csv")
```

In [4]:
```
df.head(5)
```

Out[4]:

	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	vie
0	5/2/2014 0:00	313000.0	3	1.50	1340.0	7912.0	1.5	0.0	0
1	5/2/2014 0:00	2384000.0	5	2.50	3650.0	9050.0	2.0	0.0	4
2	5/2/2014 0:00	342000.0	3	2.00	1930.0	11947.0	1.0	0.0	0
3	5/2/2014 0:00	420000.0	3	2.25	2000.0	8030.0	1.0	0.0	0
4	5/2/2014 0:00	550000.0	4	NaN	NaN	NaN	NaN	NaN	0

In [10]:
```
df.tail()
```

Out[10]:

	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfro
4595	7/9/2014 0:00	308166.6667	3	1.75	1510.0	6360.0	1.0	0
4596	7/9/2014 0:00	534333.3333	3	2.50	1460.0	7573.0	2.0	0
4597	7/9/2014 0:00	416904.1667	3	2.50	3010.0	7014.0	2.0	0
4598	7/10/2014 0:00	203400.0000	4	2.00	2090.0	6630.0	1.0	0
4599	7/10/2014 0:00	220600.0000	3	2.50	1490.0	8102.0	2.0	0

1. Data Exploration

In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4600 entries, 0 to 4599
Data columns (total 18 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   date           4600 non-null   object 
 1   price          4600 non-null   float64
 2   bedrooms       4600 non-null   int64  
 3   bathrooms      4553 non-null   float64
 4   sqft_living    4553 non-null   float64
 5   sqft_lot       4553 non-null   float64
 6   floors         4553 non-null   float64
 7   waterfront     4553 non-null   float64
 8   view           4552 non-null   float64
 9   condition      4552 non-null   float64
 10  sqft_above     4552 non-null   float64
 11  sqft_basement  4552 non-null   float64
 12  yr_built       4552 non-null   float64
 13  yr_renovated   4552 non-null   float64
 14  street         4552 non-null   object 
 15  city           4600 non-null   object 
 16  statezip       4600 non-null   object 
 17  country        4600 non-null   object 
dtypes: float64(12), int64(1), object(5)
memory usage: 647.0+ KB
```

In [7]:
```
df.shape
```

Out[7]:
```
(4600, 18)
```

In [8]:
```
df.describe()
```

Out[8]:

	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors
count	4.600000e+03	4600.000000	4553.000000	4553.000000	4.553000e+03	4553.000000
mean	5.519630e+05	3.400870	2.161871	2140.734900	1.485242e+04	1.513068
std	5.638347e+05	0.908848	0.785562	965.330337	3.602707e+04	0.538393
min	0.000000e+00	0.000000	0.000000	370.000000	6.380000e+02	1.000000
25%	3.228750e+05	3.000000	1.750000	1460.000000	5.000000e+03	1.000000
50%	4.609435e+05	3.000000	2.250000	1980.000000	7.680000e+03	1.500000
75%	6.549625e+05	4.000000	2.500000	2620.000000	1.100000e+04	2.000000
max	2.659000e+07	9.000000	8.000000	13540.000000	1.074218e+06	3.500000

In [9]:
```
df.isnull().sum()
```

Out[9]:
```
date             0
price            0
bedrooms         0
bathrooms       47
sqft_living     47
sqft_lot        47
floors          47
waterfront      47
view            48
condition       48
sqft_above      48
sqft_basement   48
yr_built        48
yr_renovated    48
street          48
city             0
statezip         0
country          0
dtype: int64
```

In [11]:
```
df.count()
```

Out[11]:
```
date             4600
price            4600
bedrooms         4600
bathrooms        4553
sqft_living      4553
sqft_lot         4553
floors           4553
waterfront       4553
view             4552
condition        4552
sqft_above       4552
sqft_basement    4552
yr_built         4552
yr_renovated     4552
street           4552
city             4600
statezip         4600
country          4600
dtype: int64
```