In [3]:

```
import pandas as pd
import numpy as np
df=pd.read_csv(r"C:\Users\unmes\Downloads\Dataset-20220307\NSE1.csv")
print(data)
```

```
       company     Open     High      Low  Volume  Turnover      LTP
0   ASIANPAINT  3101.00  3167.35  3091.00   10.29    322.53  3138.00
1     AXISBANK   669.00   674.90   660.45  102.53    684.00   661.00
2   BAJAJ-AUTO  3370.00  3383.50  3320.00    3.42    114.59  3335.00
3         BPCL   397.15   397.20   375.00  100.23    383.54   377.40
4        CIPLA   892.00   976.05   890.65  144.59   1380.90   965.00
5         HDFC  2820.35  2856.00  2723.00   33.53    927.88  2745.00
6    ICICIBANK   739.00   742.05   718.60  189.88   1385.86   720.45
7         INFY  1702.55  1718.35  1684.00   44.94    764.67  1689.55
8         ONGC   152.25   152.25   146.25  231.36    344.33   147.75
9    POWERGRID   204.05   204.95   200.80   96.11    195.09   202.50
10    RELIANCE  2467.80  2477.60  2401.50   72.75   1770.19  2405.10
11  TATAMOTORS   486.00   486.75   458.00  517.88   2430.36   459.40
12         TCS  3425.00  3490.00  3411.90   19.41    670.58  3439.20
13       TITAN  2377.80  2385.10  2285.05   12.89    298.54  2293.00
14       WIPRO   632.00   634.40   619.65   41.39    259.37   621.30
```

In [7]:

```
df['company']
```

Out[7]:

```
0     ASIANPAINT
1       AXISBANK
2     BAJAJ-AUTO
3           BPCL
4          CIPLA
5           HDFC
6      ICICIBANK
7           INFY
8           ONGC
9      POWERGRID
10      RELIANCE
11    TATAMOTORS
12           TCS
13         TITAN
14         WIPRO
Name: company, dtype: object
```

In [8]:

```
df['company'][3]
```

Out[8]:

```
'BPCL'
```

In [9]:

```
df.loc[0]  # 0th Row
```

Out[9]:

```
company      ASIANPAINT
Open             3101.0
High            3167.35
Low              3091.0
Volume            10.29
Turnover         322.53
LTP              3138.0
Name: 0, dtype: object
```

In [10]:

```
df.loc[[0,10,14],:]
```

Out[10]:

	company	Open	High	Low	Volume	Turnover	LTP
0	ASIANPAINT	3101.0	3167.35	3091.00	10.29	322.53	3138.0
10	RELIANCE	2467.8	2477.60	2401.50	72.75	1770.19	2405.1
14	WIPRO	632.0	634.40	619.65	41.39	259.37	621.3

In [11]:

```
df.loc[5:8,['company','Turnover']]
```

Out[11]:

	company	Turnover
5	HDFC	927.88
6	ICICIBANK	1385.86
7	INFY	764.67
8	ONGC	344.33

In [13]:

```
df.loc[df.Turnover>=1000]
```

Out[13]:

	company	Open	High	Low	Volume	Turnover	LTP
4	CIPLA	892.0	976.05	890.65	144.59	1380.90	965.00
6	ICICIBANK	739.0	742.05	718.60	189.88	1385.86	720.45
10	RELIANCE	2467.8	2477.60	2401.50	72.75	1770.19	2405.10
11	TATAMOTORS	486.0	486.75	458.00	517.88	2430.36	459.40

In [14]:

```
df.loc[df.Turnover>=1000,['company','Open']]
```

Out[14]:

	company	Open
4	CIPLA	892.0
6	ICICIBANK	739.0
10	RELIANCE	2467.8
11	TATAMOTORS	486.0

In [15]:

```
df.loc[df['company']=='CIPLA']
```

Out[15]:

	company	Open	High	Low	Volume	Turnover	LTP
4	CIPLA	892.0	976.05	890.65	144.59	1380.9	965.0

In [21]:

```
print(df.iloc[[1,3,4,5]])
```

```
    company      Open      High       Low  Volume  Turnover      LTP
1  AXISBANK    669.00    674.90    660.45  102.53    684.00    661.0
3      BPCL    397.15    397.20    375.00  100.23    383.54    377.4
4     CIPLA    892.00    976.05    890.65  144.59   1380.90    965.0
5      HDFC   2820.35   2856.00   2723.00   33.53    927.88   2745.0
```

In [22]:

```
df.iloc[2:5]  # Access row:2-4
```

Out[22]:

	company	Open	High	Low	Volume	Turnover	LTP
2	BAJAJ-AUTO	3370.00	3383.50	3320.00	3.42	114.59	3335.0
3	BPCL	397.15	397.20	375.00	100.23	383.54	377.4
4	CIPLA	892.00	976.05	890.65	144.59	1380.90	965.0

In [23]:

```
df.loc[2:5]  # Access row:2-4
```

Out[23]:

	company	Open	High	Low	Volume	Turnover	LTP
2	BAJAJ-AUTO	3370.00	3383.50	3320.00	3.42	114.59	3335.0
3	BPCL	397.15	397.20	375.00	100.23	383.54	377.4
4	CIPLA	892.00	976.05	890.65	144.59	1380.90	965.0
5	HDFC	2820.35	2856.00	2723.00	33.53	927.88	2745.0

In [24]:

```
df.iloc[5:11,0:4]
```

Out[24]:

	company	Open	High	Low
5	HDFC	2820.35	2856.00	2723.00
6	ICICIBANK	739.00	742.05	718.60
7	INFY	1702.55	1718.35	1684.00
8	ONGC	152.25	152.25	146.25
9	POWERGRID	204.05	204.95	200.80
10	RELIANCE	2467.80	2477.60	2401.50

In [31]:

```
df.sort_values('Low')
```

Out[31]:

	company	Open	High	Low	Volume	Turnover	LTP
8	ONGC	152.25	152.25	146.25	231.36	344.33	147.75
9	POWERGRID	204.05	204.95	200.80	96.11	195.09	202.50
3	BPCL	397.15	397.20	375.00	100.23	383.54	377.40
11	TATAMOTORS	486.00	486.75	458.00	517.88	2430.36	459.40
14	WIPRO	632.00	634.40	619.65	41.39	259.37	621.30
1	AXISBANK	669.00	674.90	660.45	102.53	684.00	661.00
6	ICICIBANK	739.00	742.05	718.60	189.88	1385.86	720.45
4	CIPLA	892.00	976.05	890.65	144.59	1380.90	965.00
7	INFY	1702.55	1718.35	1684.00	44.94	764.67	1689.55
13	TITAN	2377.80	2385.10	2285.05	12.89	298.54	2293.00
10	RELIANCE	2467.80	2477.60	2401.50	72.75	1770.19	2405.10
5	HDFC	2820.35	2856.00	2723.00	33.53	927.88	2745.00
0	ASIANPAINT	3101.00	3167.35	3091.00	10.29	322.53	3138.00
2	BAJAJ-AUTO	3370.00	3383.50	3320.00	3.42	114.59	3335.00
12	TCS	3425.00	3490.00	3411.90	19.41	670.58	3439.20

In [32]:

```
df.sort_values('company')
```

Out[32]:

	company	Open	High	Low	Volume	Turnover	LTP
0	ASIANPAINT	3101.00	3167.35	3091.00	10.29	322.53	3138.00
1	AXISBANK	669.00	674.90	660.45	102.53	684.00	661.00
2	BAJAJ-AUTO	3370.00	3383.50	3320.00	3.42	114.59	3335.00
3	BPCL	397.15	397.20	375.00	100.23	383.54	377.40
4	CIPLA	892.00	976.05	890.65	144.59	1380.90	965.00
5	HDFC	2820.35	2856.00	2723.00	33.53	927.88	2745.00
6	ICICIBANK	739.00	742.05	718.60	189.88	1385.86	720.45
7	INFY	1702.55	1718.35	1684.00	44.94	764.67	1689.55
8	ONGC	152.25	152.25	146.25	231.36	344.33	147.75
9	POWERGRID	204.05	204.95	200.80	96.11	195.09	202.50
10	RELIANCE	2467.80	2477.60	2401.50	72.75	1770.19	2405.10
11	TATAMOTORS	486.00	486.75	458.00	517.88	2430.36	459.40
12	TCS	3425.00	3490.00	3411.90	19.41	670.58	3439.20
13	TITAN	2377.80	2385.10	2285.05	12.89	298.54	2293.00
14	WIPRO	632.00	634.40	619.65	41.39	259.37	621.30

In [35]:

```
df.sort_values('Turnover',ascending=False)
```

Out[35]:

	company	Open	High	Low	Volume	Turnover	LTP
11	TATAMOTORS	486.00	486.75	458.00	517.88	2430.36	459.40
10	RELIANCE	2467.80	2477.60	2401.50	72.75	1770.19	2405.10
6	ICICIBANK	739.00	742.05	718.60	189.88	1385.86	720.45
4	CIPLA	892.00	976.05	890.65	144.59	1380.90	965.00
5	HDFC	2820.35	2856.00	2723.00	33.53	927.88	2745.00
7	INFY	1702.55	1718.35	1684.00	44.94	764.67	1689.55
1	AXISBANK	669.00	674.90	660.45	102.53	684.00	661.00
12	TCS	3425.00	3490.00	3411.90	19.41	670.58	3439.20
3	BPCL	397.15	397.20	375.00	100.23	383.54	377.40
8	ONGC	152.25	152.25	146.25	231.36	344.33	147.75
0	ASIANPAINT	3101.00	3167.35	3091.00	10.29	322.53	3138.00
13	TITAN	2377.80	2385.10	2285.05	12.89	298.54	2293.00
14	WIPRO	632.00	634.40	619.65	41.39	259.37	621.30
9	POWERGRID	204.05	204.95	200.80	96.11	195.09	202.50
2	BAJAJ-AUTO	3370.00	3383.50	3320.00	3.42	114.59	3335.00

In [36]:

```
df.describe()
```

Out[36]:

	Open	High	Low	Volume	Turnover	LTP
count	15.000000	15.000000	15.000000	15.000000	15.000000	15.000000
mean	1562.396667	1583.096667	1532.390000	108.080000	795.495333	1546.643333
std	1234.388418	1248.575247	1218.183633	132.037375	671.672718	1225.831035
min	152.250000	152.250000	146.250000	3.420000	114.590000	147.750000
25%	559.000000	560.575000	538.825000	26.470000	310.535000	540.350000
50%	892.000000	976.050000	890.650000	72.750000	670.580000	965.000000
75%	2644.075000	2666.800000	2562.250000	123.560000	1154.390000	2575.050000
max	3425.000000	3490.000000	3411.900000	517.880000	2430.360000	3439.200000

In [37]:

```
df.mean()
```

```
C:\Users\unmes\AppData\Local\Temp/ipykernel_9032/3698961737.py:1: FutureWarn
ing: Dropping of nuisance columns in DataFrame reductions (with 'numeric_onl
y=None') is deprecated; in a future version this will raise TypeError.  Sele
ct only valid columns before calling the reduction.
  df.mean()
```

Out[37]:

```
Open        1562.396667
High        1583.096667
Low         1532.390000
Volume       108.080000
Turnover     795.495333
LTP         1546.643333
dtype: float64
```

In [38]:

```
df.sum()
```

Out[38]:

```
company     ASIANPAINTAXISBANKBAJAJ-AUTOBPCLCIPLAHDFCICICI...
Open                                                23435.95
High                                                23746.45
Low                                                 22985.85
Volume                                                1621.2
Turnover                                            11932.43
LTP                                                 23199.65
dtype: object
```

In [39]:

```
df.max()
```

Out[39]:

```
company       WIPRO
Open         3425.0
High         3490.0
Low          3411.9
Volume        517.88
Turnover     2430.36
LTP           3439.2
dtype: object
```

In [40]:

```
df.min()
```

Out[40]:

```
company     ASIANPAINT
Open            152.25
High            152.25
Low             146.25
Volume            3.42
Turnover        114.59
LTP             147.75
dtype: object
```

In [41]:

```
df.count()
```

Out[41]:

```
company     15
Open        15
High        15
Low         15
Volume      15
Turnover    15
LTP         15
dtype: int64
```

In [42]:

```
df.std()
```

```
C:\Users\unmes\AppData\Local\Temp/ipykernel_9032/3390915376.py:1: FutureWarn
ing: Dropping of nuisance columns in DataFrame reductions (with 'numeric_onl
y=None') is deprecated; in a future version this will raise TypeError.  Sele
ct only valid columns before calling the reduction.
  df.std()
```

Out[42]:

```
Open        1234.388418
High        1248.575247
Low         1218.183633
Volume       132.037375
Turnover     671.672718
LTP         1225.831035
dtype: float64
```

In [43]:

```
df.var()
```

```
C:\Users\unmes\AppData\Local\Temp/ipykernel_9032/1568254755.py:1: FutureWarn
ing: Dropping of nuisance columns in DataFrame reductions (with 'numeric_onl
y=None') is deprecated; in a future version this will raise TypeError.  Sele
ct only valid columns before calling the reduction.
  df.var()
```

Out[43]:

```
Open        1.523715e+06
High        1.558940e+06
Low         1.483971e+06
Volume      1.743387e+04
Turnover    4.511442e+05
LTP         1.502662e+06
dtype: float64
```

In [ ]:

```

```