

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

1) Loading the dataset onto a DataFrame

2) Displaying first few records of the DataFrame

```
In [2]: df = pd.read_csv("IPL IMB381IPL2013.csv")
df.head()
```

Out[2]:

	SI.NO.	PLAYER NAME	AGE	COUNTRY	TEAM	PLAYING ROLE	T-RUNS	T-WKTS	ODI-RUNS-S	ODI-SR-B	...
0	1	Abdulla, YA	2	SA	KXIP	Allrounder	0	0	0	0.00	...
1	2	Abdur Razzak	2	BAN	RCB	Bowler	214	18	657	71.41	...
2	3	Agarkar, AB	2	IND	KKR	Bowler	571	58	1269	80.62	...
3	4	Ashwin, R	1	IND	CSK	Bowler	284	31	241	84.56	...
4	5	Badrinath, S	2	IND	CSK	Batsman	63	0	79	45.93	...

5 rows × 26 columns



3) Finding metadata of the DataFrame

```
In [3]: df.dtypes
```

```
Out[3]: Sl.NO.           int64
        PLAYER NAME      object
        AGE              int64
        COUNTRY          object
        TEAM              object
        PLAYING ROLE     object
        T-RUNS            int64
        T-WKTS            int64
        ODI-RUNS-S       int64
        ODI-SR-B          float64
        ODI-WKTS          int64
        ODI-SR-BL         float64
        CAPTAINCY EXP    int64
        RUNS-S            int64
        HS                int64
        AVE               float64
        SR-B              float64
        SIXERS            int64
        RUNS-C            int64
        WKTS              int64
        AVE-BL            float64
        ECON              float64
        SR-BL              float64
        AUCTION YEAR     int64
        BASE PRICE        int64
        SOLD PRICE        int64
        dtype: object
```

```
In [4]: df.shape
```

```
Out[4]: (130, 26)
```

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 130 entries, 0 to 129
Data columns (total 26 columns):
 #   Column            Non-Null Count  Dtype  
--- 
 0   Sl.NO.             130 non-null    int64  
 1   PLAYER NAME        130 non-null    object  
 2   AGE                130 non-null    int64  
 3   COUNTRY             130 non-null    object  
 4   TEAM               130 non-null    object  
 5   PLAYING ROLE       130 non-null    object  
 6   T-RUNS              130 non-null    int64  
 7   T-WKTS              130 non-null    int64  
 8   ODI-RUNS-S          130 non-null    int64  
 9   ODI-SR-B             130 non-null    float64 
 10  ODI-WKTS             130 non-null    int64  
 11  ODI-SR-BL            130 non-null    float64 
 12  CAPTAINCY EXP        130 non-null    int64  
 13  RUNS-S              130 non-null    int64  
 14  HS                  130 non-null    int64  
 15  AVE                 130 non-null    float64 
 16  SR-B                130 non-null    float64 
 17  SIXERS              130 non-null    int64  
 18  RUNS-C              130 non-null    int64  
 19  WKTS                130 non-null    int64  
 20  AVE-BL              130 non-null    float64 
 21  ECON                130 non-null    float64 
 22  SR-BL               130 non-null    float64 
 23  AUCTION YEAR         130 non-null    int64  
 24  BASE PRICE            130 non-null    int64  
 25  SOLD PRICE            130 non-null    int64  
dtypes: float64(7), int64(15), object(4)
memory usage: 26.5+ KB
```

```
In [6]: df.columns
```

```
Out[6]: Index(['Sl.NO.', 'PLAYER NAME', 'AGE', 'COUNTRY', 'TEAM', 'PLAYING ROLE',
 'T-RUNS', 'T-WKTS', 'ODI-RUNS-S', 'ODI-SR-B', 'ODI-WKTS', 'ODI-SR-BL',
 'CAPTAINCY EXP', 'RUNS-S', 'HS', 'AVE', 'SR-B', 'SIXERS', 'RUNS-C',
 'WKTS', 'AVE-BL', 'ECON', 'SR-BL', 'AUCTION YEAR', 'BASE PRICE',
 'SOLD PRICE'],
 dtype='object')
```

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

```
Out[7]: RangeIndex(start=0, stop=130, step=1)
```

```
In [8]: df.isnull().sum()
```

```
Out[8]: Sl.NO.          0  
        PLAYER NAME      0  
        AGE              0  
        COUNTRY          0  
        TEAM              0  
        PLAYING ROLE     0  
        T-RUNS            0  
        T-WKTS            0  
        ODI-RUNS-S        0  
        ODI-SR-B          0  
        ODI-WKTS          0  
        ODI-SR-BL         0  
        CAPTAINCY EXP     0  
        RUNS-S            0  
        HS                0  
        AVE               0  
        SR-B              0  
        SIXERS            0  
        RUNS-C            0  
        WKTS              0  
        AVE-BL             0  
        ECON               0  
        SR-BL              0  
        AUCTION YEAR      0  
        BASE PRICE         0  
        SOLD PRICE         0  
        dtype: int64
```

4) Finding Summary of the DataFrame

```
In [9]: df.count()
```

```
Out[9]: Sl.NO.          130
        PLAYER NAME    130
        AGE             130
        COUNTRY         130
        TEAM            130
        PLAYING ROLE   130
        T-RUNS          130
        T-WKTS          130
        ODI-RUNS-S     130
        ODI-SR-B       130
        ODI-WKTS        130
        ODI-SR-BL      130
        CAPTAINCY EXP  130
        RUNS-S          130
        HS              130
        AVE             130
        SR-B            130
        SIXERS          130
        RUNS-C          130
        WKTS            130
        AVE-BL          130
        ECON            130
        SR-BL           130
        AUCTION YEAR   130
        BASE PRICE      130
        SOLD PRICE      130
        dtype: int64
```

```
In [10]: df.select_dtypes(include='number').mean()[1:]
```

```
Out[10]: AGE            2.092308
        T-RUNS         2166.715385
        T-WKTS          66.530769
        ODI-RUNS-S     2508.738462
        ODI-SR-B       71.164385
        ODI-WKTS        76.076923
        ODI-SR-BL      34.033846
        CAPTAINCY EXP  0.315385
        RUNS-S          514.246154
        HS              47.430769
        AVE             18.719308
        SR-B            111.053462
        SIXERS          17.692308
        RUNS-C          475.523077
        WKTS            17.169231
        AVE-BL          23.110231
        ECON            6.204462
        SR-BL           17.382615
        AUCTION YEAR   2009.092308
        BASE PRICE      192230.769231
        SOLD PRICE      521223.076923
        dtype: float64
```

```
In [11]: df.select_dtypes(include='number').median()[1:]
```

```
Out[11]: AGE           2.000
          T-RUNS        542.500
          T-WKTS         7.000
          ODI-RUNS-S     835.000
          ODI-SR-B       78.225
          ODI-WKTS        18.500
          ODI-SR-BL      36.600
          CAPTAINCY EXP   0.000
          RUNS-S         172.000
          HS              35.500
          AVE             18.635
          SR-B            118.510
          SIXERS          6.000
          RUNS-C          297.000
          WKTS            8.500
          AVE-BL          24.785
          ECON             7.380
          SR-BL            19.935
          AUCTION YEAR    2008.000
          BASE PRICE      200000.000
          SOLD PRICE      437500.000
          dtype: float64
```

```
In [12]: df.select_dtypes(include='number').std()[1:]
```

```
Out[12]: AGE           0.576627
          T-RUNS        3305.646757
          T-WKTS         142.676855
          ODI-RUNS-S     3582.205625
          ODI-SR-B       25.898440
          ODI-WKTS        111.205070
          ODI-SR-BL      26.751749
          CAPTAINCY EXP   0.466466
          RUNS-S          615.226335
          HS              36.403624
          AVE             11.094224
          SR-B            35.928907
          SIXERS          23.828146
          RUNS-C          558.314049
          WKTS            21.816763
          AVE-BL          20.802057
          ECON             4.941531
          SR-BL            15.273422
          AUCTION YEAR    1.377821
          BASE PRICE      153097.300897
          SOLD PRICE      406807.351419
          dtype: float64
```

```
In [13]: df.select_dtypes(include='number').min()[1:]
```

```
Out[13]: AGE           1.0
          T-RUNS        0.0
          T-WKTS        0.0
          ODI-RUNS-S    0.0
          ODI-SR-B      0.0
          ODI-WKTS      0.0
          ODI-SR-BL     0.0
          CAPTAINCY EXP 0.0
          RUNS-S        0.0
          HS            0.0
          AVE           0.0
          SR-B          0.0
          SIXERS         0.0
          RUNS-C         0.0
          WKTS          0.0
          AVE-BL         0.0
          ECON           0.0
          SR-BL          0.0
          AUCTION YEAR   2008.0
          BASE PRICE     20000.0
          SOLD PRICE     20000.0
          dtype: float64
```

```
In [14]: df.select_dtypes(include='number').max()[1:]
```

```
Out[14]: AGE           3.00
          T-RUNS        15470.00
          T-WKTS        800.00
          ODI-RUNS-S    18426.00
          ODI-SR-B      116.66
          ODI-WKTS      534.00
          ODI-SR-BL     150.00
          CAPTAINCY EXP 1.00
          RUNS-S        2254.00
          HS            158.00
          AVE           50.11
          SR-B          235.49
          SIXERS         129.00
          RUNS-C         1975.00
          WKTS          83.00
          AVE-BL         126.30
          ECON           38.11
          SR-BL          100.20
          AUCTION YEAR   2011.00
          BASE PRICE     1350000.00
          SOLD PRICE     1800000.00
          dtype: float64
```

```
In [15]: df.select_dtypes(include='number').quantile(0.25)[1:]
```

```
Out[15]: AGE           2.0000
          T-RUNS        25.5000
          T-WKTS         0.0000
          ODI-RUNS-S     73.2500
          ODI-SR-B       65.6500
          ODI-WKTS        0.0000
          ODI-SR-BL      0.0000
          CAPTAINCY EXP   0.0000
          RUNS-S         39.0000
          HS              16.0000
          AVE             9.8250
          SR-B            98.2375
          SIXERS          1.0000
          RUNS-C          0.0000
          WKTS            0.0000
          AVE-BL          0.0000
          ECON             0.0000
          SR-BL           0.0000
          AUCTION YEAR    2008.0000
          BASE PRICE      100000.0000
          SOLD PRICE      225000.0000
          Name: 0.25, dtype: float64
```

```
In [16]: df.select_dtypes(include='number').quantile(0.5)[1:]
```

```
Out[16]: AGE           2.000
          T-RUNS        542.500
          T-WKTS         7.000
          ODI-RUNS-S     835.000
          ODI-SR-B       78.225
          ODI-WKTS        18.500
          ODI-SR-BL      36.600
          CAPTAINCY EXP   0.000
          RUNS-S         172.000
          HS              35.500
          AVE             18.635
          SR-B            118.510
          SIXERS          6.000
          RUNS-C          297.000
          WKTS            8.500
          AVE-BL          24.785
          ECON             7.380
          SR-BL           19.935
          AUCTION YEAR    2008.000
          BASE PRICE      200000.000
          SOLD PRICE      437500.000
          Name: 0.5, dtype: float64
```

```
In [17]: df.select_dtypes(include='number').quantile(0.75)[1:]
```

```
Out[17]: AGE           2.0000
          T-RUNS        3002.2500
          T-WKTS         47.5000
          ODI-RUNS-S    3523.5000
          ODI-SR-B       86.7900
          ODI-WKTS        106.0000
          ODI-SR-BL      45.3250
          CAPTAINCY EXP   1.0000
          RUNS-S          925.2500
          HS              73.7500
          AVE             27.8725
          SR-B            129.1025
          SIXERS          29.7500
          RUNS-C          689.2500
          WKTS            23.7500
          AVE-BL          35.5800
          ECON             8.2475
          SR-BL            26.2125
          AUCTION YEAR    2011.0000
          BASE PRICE       225000.0000
          SOLD PRICE       700000.0000
          Name: 0.75, dtype: float64
```

```
In [18]: df.describe(include='object')
```

	PLAYER NAME	COUNTRY	TEAM	PLAYING ROLE
count	130	130	130	130
unique	130	10	17	4
top	Abdulla, YA	IND	CSK	Bowler
freq	1	53	14	44

```
In [19]: df.describe(include='number')
```

Out[19]:

	SI.NO.	AGE	T-RUNS	T-WKTS	ODI-RUNS-S	ODI-SR-B	ODI-W
count	130.000000	130.000000	130.000000	130.000000	130.000000	130.000000	130.000000
mean	65.500000	2.092308	2166.715385	66.530769	2508.738462	71.164385	76.076000
std	37.671829	0.576627	3305.646757	142.676855	3582.205625	25.898440	111.205000
min	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	33.250000	2.000000	25.500000	0.000000	73.250000	65.650000	0.000000
50%	65.500000	2.000000	542.500000	7.000000	835.000000	78.225000	18.500000
75%	97.750000	2.000000	3002.250000	47.500000	3523.500000	86.790000	106.000000
max	130.000000	3.000000	15470.000000	800.000000	18426.000000	116.660000	534.000000

8 rows × 22 columns



5) Slicing and Indexing a dataframe

6) Selecting Columns by Column Names (eg: PLAYER NAME)

In [20]: `df['PLAYER NAME'].head(10)`

Out[20]:

```
0      Abdulla, YA
1      Abdur Razzak
2      Agarkar, AB
3      Ashwin, R
4      Badrinath, S
5      Bailey, GJ
6      Balaji, L
7      Bollinger, DE
8      Botha, J
9      Boucher, MV
Name: PLAYER NAME, dtype: object
```

7) Finding Unique Occurrences of Values in Columns (eg: COUNTRY)

In [21]: `df['COUNTRY'].unique()`

Out[21]: `array(['SA', 'BAN', 'IND', 'AUS', 'WI', 'SL', 'NZ', 'ENG', 'PAK', 'ZIM'],
 dtype=object)`

8) Cross-tabulation between two columns (eg: AGE & PLAYING ROLE)

In [22]: `pd.crosstab(df['AGE'], df['PLAYING ROLE'])`

```
Out[22]: PLAYING ROLE Allrounder Batsman Bowler W. Keeper
```

AGE				
1	4	5	7	0
2	25	21	29	11
3	6	13	8	1

9) Sorting dataframe by column values (eg: SOLD PRICE)

```
In [23]: df['SOLD PRICE'].sort_values()[:25]
```

```
Out[23]: 73      20000
46      24000
0       50000
1       50000
118     50000
5       50000
85      50000
34      50000
61      80000
78      95000
104     100000
102     100000
100     100000
90      100000
63      100000
58      100000
129     110000
109     125000
53      140000
45      150000
40      150000
20      150000
21      150000
115     150000
89      160000
Name: SOLD PRICE, dtype: int64
```

```
In [24]: df['SOLD PRICE'].sort_values(ascending=False)[:25]
```

```
Out[24]: 93      1800000
127     1800000
50      1800000
111     1800000
113     1600000
83      1550000
23      1550000
15      1500000
108     1350000
110     1000000
41      975000
39      950000
59      950000
95      950000
8       950000
81      925000
55      900000
44      900000
101     875000
72      850000
3       850000
30      850000
116     800000
71      800000
26      800000
Name: SOLD PRICE, dtype: int64
```

10) Which player got the maximum premium on the base price?

```
In [25]: df['Premium'] = df['SOLD PRICE'] - df['BASE PRICE']
max_premium_player = df.loc[df['Premium'].idxmax(), 'PLAYER NAME']
max_premium_player
```

Out[25]: 'Kohli, V'

11) Which players got the maximum premium offering on their base price?

```
In [26]: df['Premium'] = df['SOLD PRICE'] - df['BASE PRICE']
max_premium = df['Premium'].max()
players_with_max_premium = df[df['Premium'] == max_premium][['PLAYER NAME', 'BASE P
players_with_max_premium
```

Out[26]:

	PLAYER NAME	BASE PRICE	SOLD PRICE	Premium
50	Kohli, V	150000	1800000	1650000

12) What is the average SOLD PRICE for each age category?

```
In [27]: df.groupby('AGE')['SOLD PRICE'].mean()
```

```
Out[27]: AGE
1    720250.000000
2    484534.883721
3    520178.571429
Name: SOLD PRICE, dtype: float64
```

13) Average SOLD PRICE for Different Playing Roles in Each Age Category?

```
In [28]: df.groupby(['AGE', 'PLAYING ROLE'])['SOLD PRICE'].mean()
```

```
Out[28]: AGE  PLAYING ROLE
1   Allrounder      5.875000e+05
      Batsman        1.110000e+06
      Bowler         5.177143e+05
2   Allrounder      4.494000e+05
      Batsman        6.547619e+05
      Bowler         3.979310e+05
      W. Keeper       4.677273e+05
3   Allrounder      7.666667e+05
      Batsman        4.576923e+05
      Bowler         4.143750e+05
      W. Keeper       7.000000e+05
Name: SOLD PRICE, dtype: float64
```

```
In [ ]:
```