

```
# loading the required libraries
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns

# Loading the ipl matches dataset
ipl = pd.read_csv('data.csv')

# having a glance at first five dataset
ipl.head()
```

	id	season	city	date	team1	team2	toss_winner	toss_decision	r
0	1	2008	Bangalore	2008-04-18	Kolkata Knight Riders	Royal Challengers Bangalore	Royal Challengers Bangalore	field	r
1	2	2008	Chandigarh	2008-04-19	Chennai Super Kings	Kings XI Punjab	Chennai Super Kings	bat	r
2	3	2008	Delhi	2008-04-19	Rajasthan Royals	Delhi Daredevils	Rajasthan Royals	bat	r
3	4	2008	Mumbai	2008-04-20	Mumbai Indians	Royal Challengers Bangalore	Mumbai Indians	bat	r
4	5	2008	Kolkata	2008-04-20	Deccan Chargers	Kolkata Knight Riders	Deccan Chargers	bat	r

```
# Looking at the rows and columns of dataset
ipl.shape
```

```
(577, 18)
```

```
# Getting the frequency of most man of the match awards
ipl['player_of_match'].value_counts()
```

```
CH Gayle          17
YK Pathan         16
AB de Villiers    15
DA Warner         14
RG Sharma         13
..
MN Samuels        1
S Badrinath       1
```

```
R Bhatia          1
RE Levi           1
BCJ Cutting       1
Name: player_of_match, Length: 187, dtype: int64
```

```
ipl['player_of_match'].value_counts()[0:10] # Getting the top 10 players with most man of
```

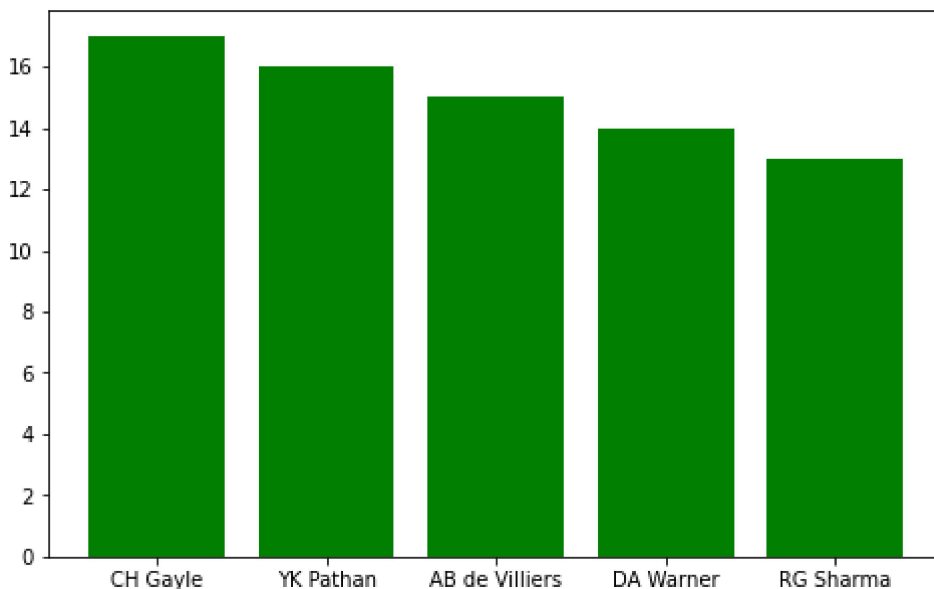
```
CH Gayle          17
YK Pathan         16
AB de Villiers    15
DA Warner         14
RG Sharma         13
SK Raina          13
AM Rahane         12
MEK Hussey        12
G Gambhir         12
MS Dhoni          12
Name: player_of_match, dtype: int64
```

```
ipl['player_of_match'].value_counts()[0:5] #Getting the top 5 players with most man of tr
```

```
CH Gayle          17
YK Pathan         16
AB de Villiers    15
DA Warner         14
RG Sharma         13
Name: player_of_match, dtype: int64
```

```
li = list(ipl['player_of_match'].value_counts()[0:5].keys())
```

```
plt.figure(figsize=(8,5))
plt.bar(li, list(ipl['player_of_match'].value_counts()[0:5]),color = "g")
plt.show()
```



```
# Getting the frequency of result column
ipl['result'].value_counts()
```

```
normal      568
tie          6
no result    3
Name: result, dtype: int64
```

```
# Finding the number of toss wins w.r.t each team
ipl['toss_winner'].value_counts()
```

```
Mumbai Indians      74
Kolkata Knight Riders 69
Chennai Super Kings  66
Kings XI Punjab      64
Delhi Daredevils     64
Rajasthan Royals     63
Royal Challengers Bangalore 61
Deccan Chargers      43
Sunrisers Hyderabad  30
Pune Warriors        20
Kochi Tuskers Kerala  8
Gujarat Lions        8
Rising Pune Supergiants 7
Name: toss_winner, dtype: int64
```

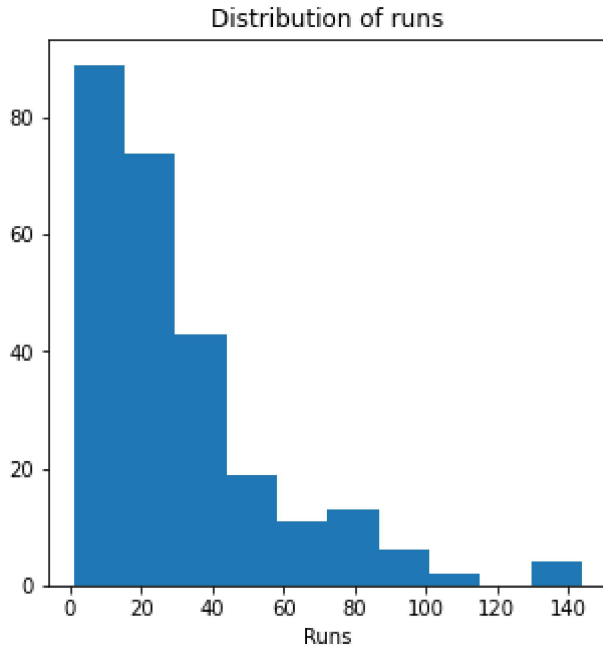
```
# Extracting the records where a team won by batting first
batting_first = ipl[ipl['win_by_runs']!=0]
```

```
# Looking at the head
batting_first.head()
```

```

# Making a histogram
plt.figure(figsize=(5,5))
plt.hist(batting_first['win_by_runs'])
plt.title("Distribution of runs")
plt.xlabel("Runs")
plt.show()

```



```

# Finding the number of wins w.r.t each team after batting first
batting_first['winner'].value_counts()

```

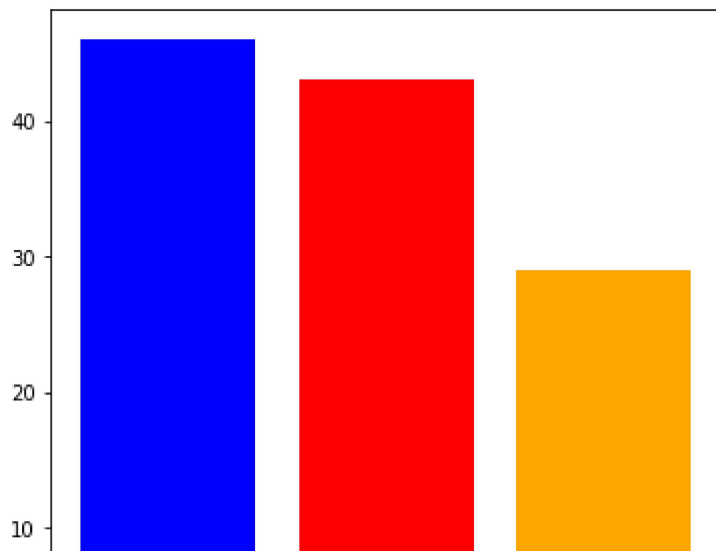
Chennai Super Kings	46
Mumbai Indians	43
Kolkata Knight Riders	29
Kings XI Punjab	28
Royal Challengers Bangalore	27
Rajasthan Royals	23
Delhi Daredevils	18
Deccan Chargers	18
Sunrisers Hyderabad	18
Pune Warriors	6
Kochi Tuskers Kerala	2
Rising Pune Supergiants	2
Gujarat Lions	1

Name: winner, dtype: int64

```

# Making bar plot of top 3 with most wins after batting first
plt.figure(figsize=(6,6))
plt.bar(list(batting_first['winner'].value_counts()[0:3].keys()),list(batting_first['winner'].value_counts()[0:3].values()))
plt.show()

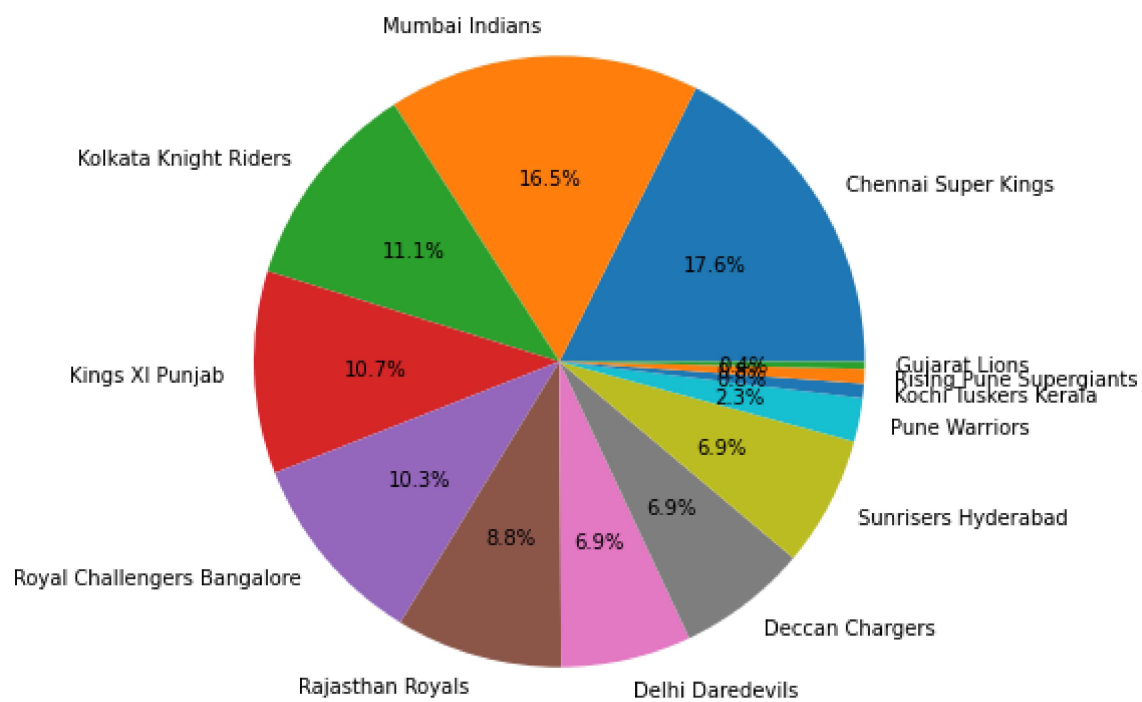
```



```
# Making a pie-chart distribution
```

```
plt.figure(figsize=(7,7))
```

```
plt.pie(list(batting_first['winner'].value_counts()),labels=list(batting_first['winner']).\nplt.show()
```



```
# Extracting records where team has won after batting second
```

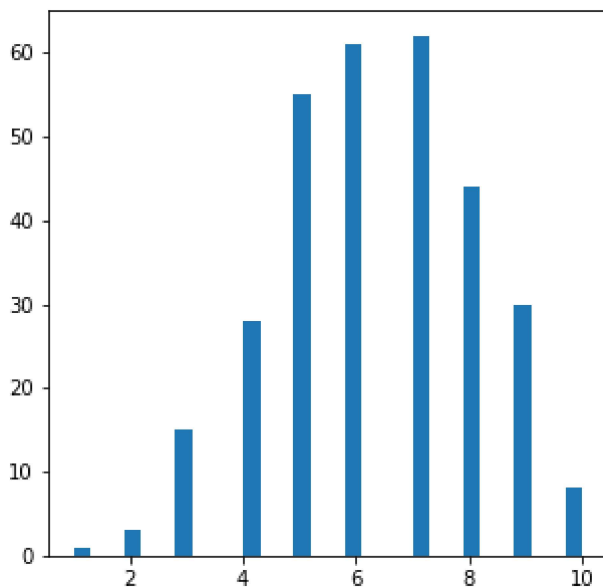
```
batting_second=ipl[ipl['win_by_wickets']!=0]
```

```
# Looking at the head
```

```
batting_second.head()
```

	id	season	city	date	team1	team2	toss_winner	toss_decision	re
2	3	2008	Delhi	2008-04-19	Rajasthan Royals	Delhi Daredevils	Rajasthan Royals	bat	n
3	4	2008	Mumbai	2008-04-20	Mumbai Indians	Royal Challengers Bangalore	Mumbai Indians	bat	n
4	5	2008	Kolkata	2008-04-20	Deccan Chargers	Kolkata Knight Riders	Deccan Chargers	bat	n
5	6	2008	Jaipur	2008-04-21	Kings XI Punjab	Rajasthan Royals	Kings XI Punjab	bat	n
6	7	2008	Hyderabad	2008-	Deccan	Delhi	Deccan	bat	n

```
# Making a histogram
plt.figure(figsize=(5,5))
plt.hist(batting_second['win_by_wickets'],bins = 30)
plt.show()
```



```
# Findinf out the frequency of wins w.r.t each time after batting second
batting_second['winner'].value_counts()
```

```
Royal Challengers Bangalore    42
Kolkata Knight Riders          39
Delhi Daredevils               38
Rajasthan Royals               38
Mumbai Indians                 37
Chennai Super Kings            33
Kings XI Punjab                33
Sunrisers Hyderabad            15
Deccan Chargers                11
Gujarat Lions                  8
Pune Warriors                   6
```

```

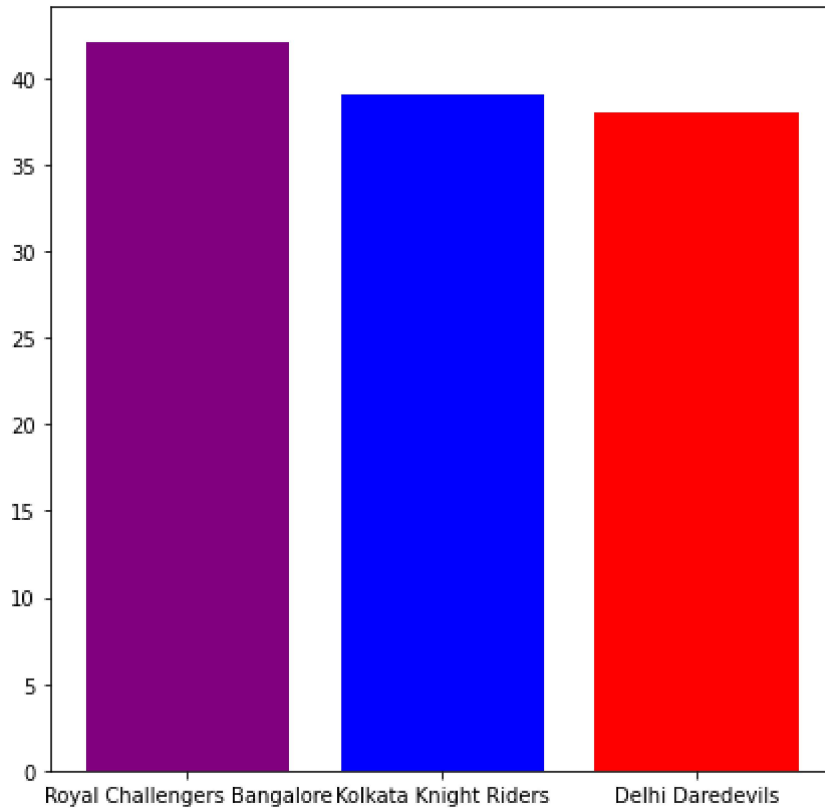
Kochi Tuskers Kerala      4
Rising Pune Supergiants   3
Name: winner, dtype: int64

```

```

# Making a bar plot for top 3 teams with most wins after batting second
plt.figure(figsize=(7,7))
plt.bar(list(batting_second['winner'].value_counts()[0:3].keys()),list(batting_second['winner'].value_counts()[0:3].values()))
plt.show()

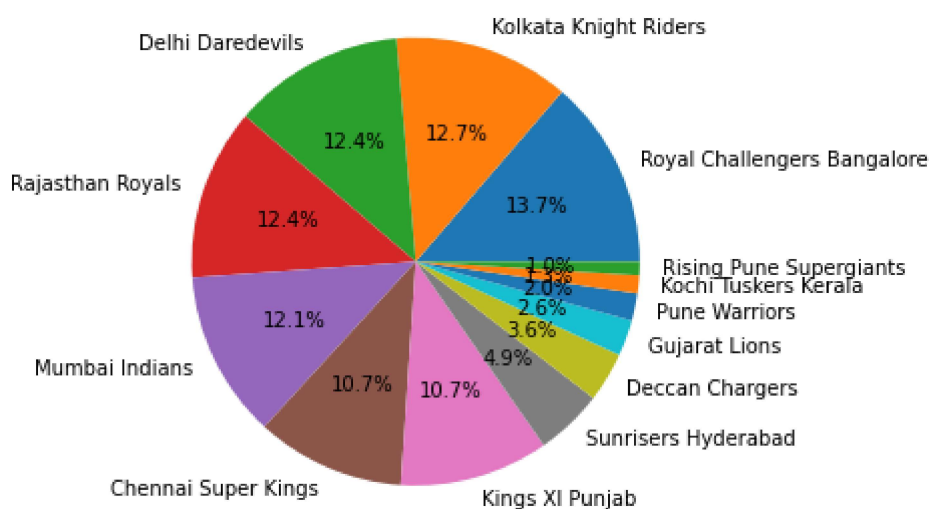
```



```

# Making a pie-chart distribution
plt.figure(figsize=(5,7))
plt.pie(list(batting_second['winner'].value_counts()),labels=list(batting_second['winner'].value_counts().keys()),autopct='%1.1f%%')
plt.show()

```



```
# Looking at number of matches played each season
```

```
ipl['season'].value_counts()
```

```
2013    76
2012    74
2011    73
2010    60
2014    60
2016    60
2015    59
2008    58
2009    57
Name: season, dtype: int64
```

```
# Looking at the number of matches played in each city
```

```
ipl['city'].value_counts()
```

```
Mumbai          77
Bangalore       58
Kolkata         54
Delhi           53
Chennai         48
Chandigarh      42
Hyderabad       41
Jaipur          33
Pune            25
Durban          15
Centurion       12
Ahmedabad       12
Visakhapatnam   11
Dharamsala      9
Johannesburg    8
Cuttack         7
Port Elizabeth  7
Cape Town       7
Ranchi          7
Abu Dhabi       7
Sharjah         6
Raipur          6
Kochi           5
Rajkot          5
Kimberley       3
Nagpur          3
East London     3
Bloemfontein    2
Indore          2
Kanpur          2
Name: city, dtype: int64
```

```
#Finding how many times a team has won after winning toss
```

```
np.sum(ipl['toss_winner']==ipl['winner'])
```

```
291
```

```
delivery = pd.read_csv('delivery.csv')
```



```
delivery.head() # Getting first five entries
```

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker
0	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	1	1	SC Ganguly	BB McCullum
1	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	1	2	BB McCullum	SC Ganguly
2	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	1	3	BB McCullum	SC Ganguly
3	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	1	4	BB McCullum	SC Ganguly
4	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	1	5	BB McCullum	SC Ganguly

5 rows × 21 columns

```
delivery.shape
```

(1000, 21)

```
delivery.tail()
```

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker
995	5	1	Deccan Chargers	Kolkata Knight Riders	9	6	SB Styris	A Symonds
996	5	1	Deccan Chargers	Kolkata Knight Riders	10	1	SB Styris	A Symonds
997	5	1	Deccan Chargers	Kolkata Knight Riders	10	2	A Symonds	SB Styris
998	5	1	Deccan Chargers	Kolkata Knight Riders	10	3	SB Styris	A Symonds
999	5	1	Deccan Chargers	Kolkata Knight Riders	10	4	A Symonds	SB Styris

5 rows × 21 columns

```
delivery.describe()
```

	match_id	inning	over	ball	is_super_over	wide_runs
<b>count</b>	1000.000000	1000.000000	1000.000000	1000.000000	1000.0	1000.000000
<b>mean</b>	2.672000	1.445000	9.658000	3.633000	0.0	0.051000
<b>std</b>	1.239335	0.497214	5.562012	1.823084	0.0	0.310637
<b>min</b>	1.000000	1.000000	1.000000	1.000000	0.0	0.000000
<b>25%</b>	2.000000	1.000000	5.000000	2.000000	0.0	0.000000
<b>50%</b>	3.000000	1.000000	9.000000	4.000000	0.0	0.000000
<b>75%</b>	4.000000	2.000000	14.000000	5.000000	0.0	0.000000
<b>max</b>	5.000000	2.000000	20.000000	8.000000	0.0	5.000000

```
delivery['match_id'].unique()
```

```
array([1, 2, 3, 4, 5])
```

```
match_1=delivery[delivery['match_id']==1]
```

```
match_1.head()
```

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker
	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	1	1	SC Ganguly	BB McCullum

match\_1.shape

(225, 21)

srh=match\_1[match\_1['inning']==1]

2	1	1	Kolkata Knight Riders	Challengers	1	3	BB McCullum	SC Ganguly
---	---	---	-----------------------	-------------	---	---	-------------	------------

srh['batsman\_runs'].value\_counts()

0 45  
1 39  
4 15  
6 14  
2 11

Name: batsman\_runs, dtype: int64

5 rows x 9 columns

srh['dismissal\_kind'].value\_counts()

caught 3  
Name: dismissal\_kind, dtype: int64

rcb = match\_1[match\_1['inning']==2]

rcb['batsman\_runs'].value\_counts()

0 65  
1 27  
2 3  
6 3  
4 3  
Name: batsman\_runs, dtype: int64

rcb['dismissal\_kind'].value\_counts()

caught 6  
bowled 3  
run out 1  
Name: dismissal\_kind, dtype: int64

# End of Project 1

---

✓ 0s completed at 2:40 PM

● ✕