Exploratory Data Analysis(EDA) of IPL Data

The Data

In [3]:

```
# Loading the required libraries

import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
```

In [4]:

```
# Loading the ipl macthes dataset
ipl=pd.read_csv('Documents\matches.csv')
```

In [5]:

```
# Having a glance of first 5 records of the dataset
ipl.head()
```

Out[5]:

	id	season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets
0	1	2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Sunrisers Hyderabad	35	0
1	2	2017	Pune	06-04-2017	Mumbai Indians	Rising Pune Supergiant	Rising Pune Supergiant	field	normal	0	Rising Pune Supergiant	0	7
2	3	2017	Rajkot	07-04-2017	Gujarat Lions	Kolkata Knight Riders	Kolkata Knight Riders	field	normal	0	Kolkata Knight Riders	0	10
3	4	2017	Indore	08-04-2017	Rising Pune Supergiant	Kings XI Punjab	Kings XI Punjab	field	normal	0	Kings XI Punjab	0	6
4	5	2017	Bangalore	08-04-2017	Royal Challengers Bangalore	Delhi Daredevils	Royal Challengers Bangalore	bat	normal	0	Royal Challengers Bangalore	15	0

In [6]:

```
# Lookin at the number of rows and column in the dataset
ipl.shape
```

Out[6]:

```
(636, 18)
```

In [7]:

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

Out[7]:

```
CH Gayle          18
YK Pathan         16
DA Warner         15
AB de Villiers    15
RG Sharma         14
                  ..
AD Mathews         1
LR Shukla          1
R Bhatia           1
A Singh            1
BCJ Cutting        1
Name: player_of_match, Length: 201, dtype: int64
```

In [8]:

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

Out[8]:

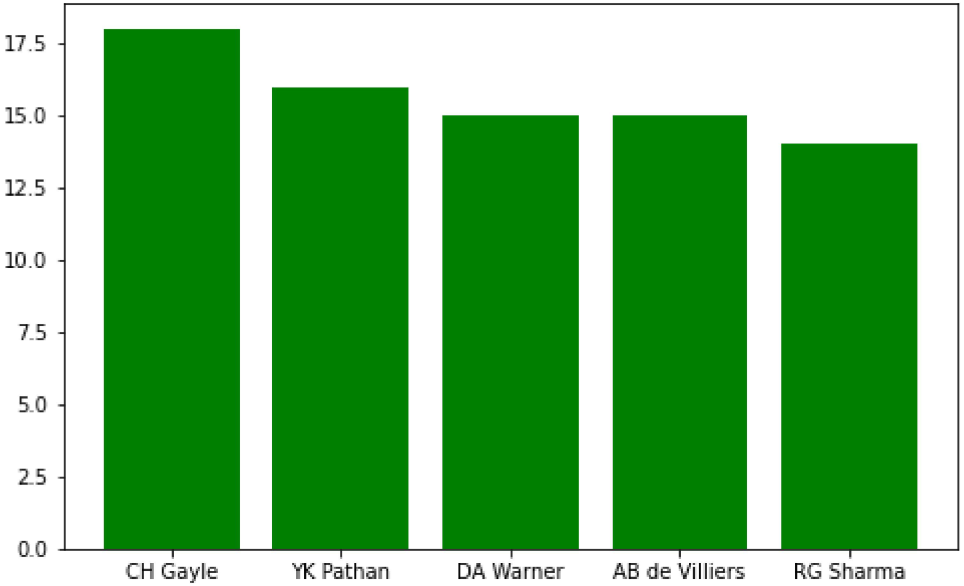
```
CH Gayle          18
YK Pathan         16
DA Warner         15
AB de Villiers    15
RG Sharma         14
SK Raina          14
MS Dhoni          13
G Gambhir         13
AM Rahane         12
MEK Hussey        12
Name: player_of_match, dtype: int64
```

In [9]:
```
# Getting top 10 players with most man of the match awards
ipl['player_of_match'].value_counts()[0:5]
```

Out[9]:
```
CH Gayle          18
YK Pathan         16
DA Warner         15
AB de Villiers    15
RG Sharma         14
Name: player_of_match, dtype: int64
```

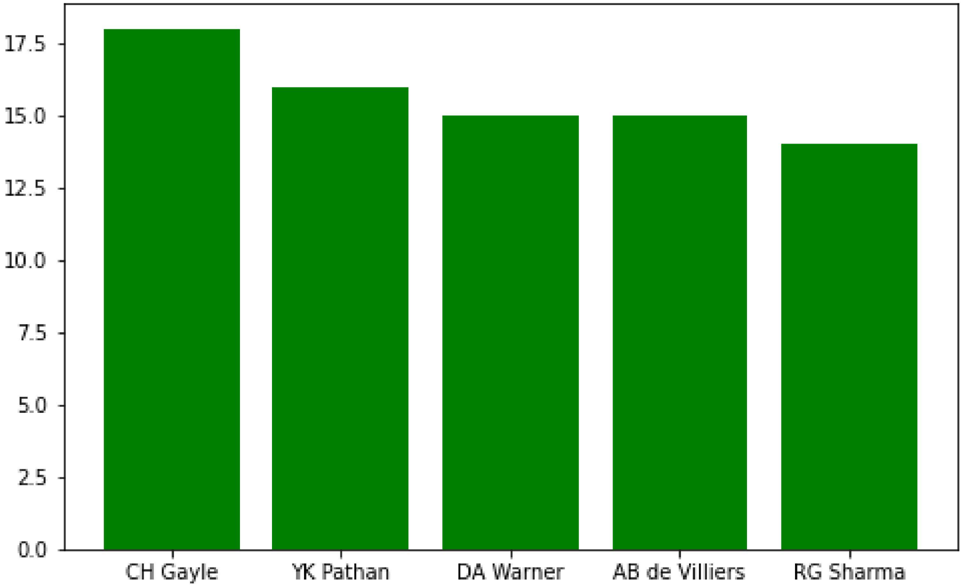
Bar Plot

In [10]:
```
plt.figure(figsize=(8,5))
plt.bar(list(ipl['player_of_match'].value_counts()[0:5].keys()),list(ipl['player_of_match'].value_counts()[0:5]),color='
plt.show()
```



In [11]:

```
# Making a bar plot for the top 5 players with most man of the match awards
plt.figure(figsize=(8,5))
plt.bar(list(ipl['player_of_match'].value_counts()[0:5].keys()),list(ipl['player_of_match'].value_counts()[0:5]),color='
plt.show()
```



In [12]:

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

Out[12]:

```
normal       626
tie            7
no result      3
Name: result, dtype: int64
```

In [13]:

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

Out[13]:

```
Mumbai Indians                 85
Kolkata Knight Riders          78
Delhi Daredevils               72
Royal Challengers Bangalore    70
Kings XI Punjab                68
Chennai Super Kings            66
Rajasthan Royals               63
Deccan Chargers                43
Sunrisers Hyderabad            35
Pune Warriors                  20
Gujarat Lions                  15
Kochi Tuskers Kerala            8
Rising Pune Supergiants         7
Rising Pune Supergiant          6
Name: toss_winner, dtype: int64
```

In [14]:

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

In [15]:

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

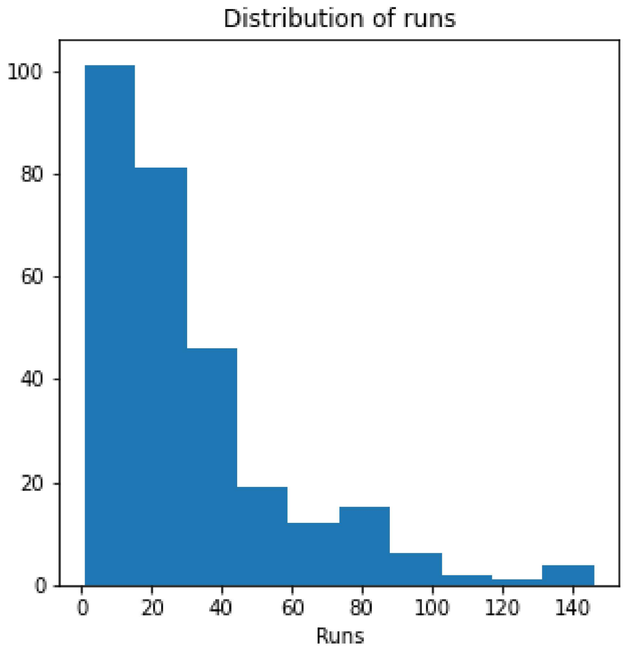
Out[15]:

	id	season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets
0	1	2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Sunrisers Hyderabad	35	0
4	5	2017	Bangalore	08-04-2017	Royal Challengers Bangalore	Delhi Daredevils	Royal Challengers Bangalore	bat	normal	0	Royal Challengers Bangalore	15	0
8	9	2017	Pune	11-04-2017	Delhi Daredevils	Rising Pune Supergiant	Rising Pune Supergiant	field	normal	0	Delhi Daredevils	97	0
13	14	2017	Kolkata	15-04-2017	Kolkata Knight Riders	Sunrisers Hyderabad	Sunrisers Hyderabad	field	normal	0	Kolkata Knight Riders	17	0
14	15	2017	Delhi	15-04-2017	Delhi Daredevils	Kings XI Punjab	Delhi Daredevils	bat	normal	0	Delhi Daredevils	51	0

Histogram

In [16]:

```
# 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()
```



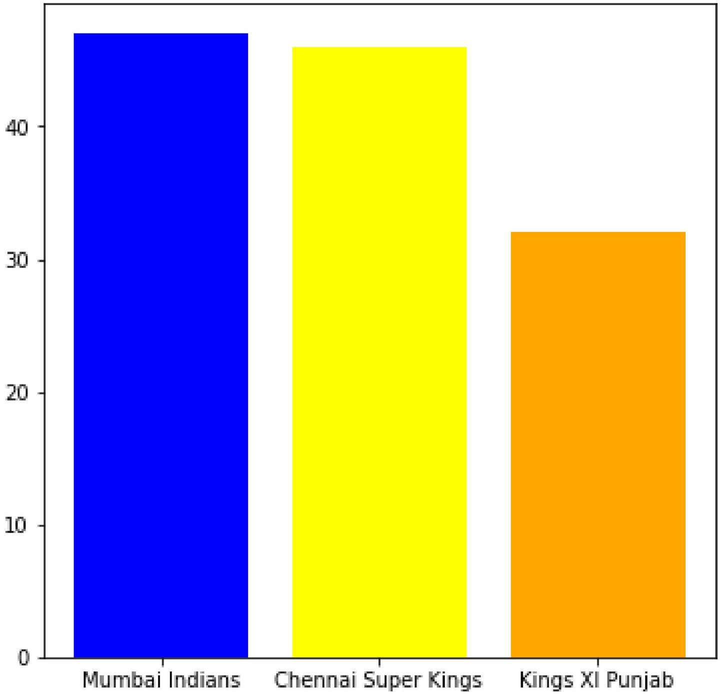
In [17]:

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

Out[17]:

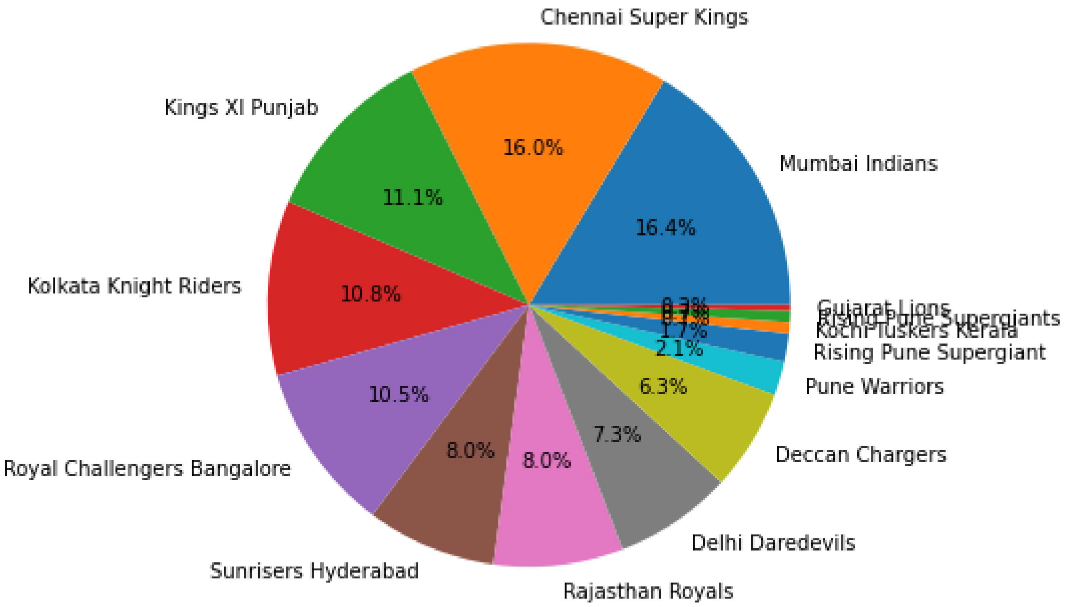
```
Mumbai Indians                 47
Chennai Super Kings            46
Kings XI Punjab                32
Kolkata Knight Riders          31
Royal Challengers Bangalore    30
Sunrisers Hyderabad            23
Rajasthan Royals               23
Delhi Daredevils               21
Deccan Chargers                18
Pune Warriors                   6
Rising Pune Supergiant          5
Kochi Tuskers Kerala            2
Rising Pune Supergiants         2
Gujarat Lions                   1
Name: winner, dtype: int64
```

```
In [18]: # Making a bar plot for top 3 teams 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]),color
         plt.show()
```



Pie chart

In [19]:
```
# Making a Pie Chart for top 3 teams with most wins after batting first
plt.figure(figsize=(6,6))
plt.pie(list(batting_first['winner'].value_counts()),labels=list(batting_first['winner'].value_counts().keys()),autopct=
plt.show()
```



In [20]:
```
# Extracting those records where a team has won after batting second
batting_second=ipl[ipl['win_by_wickets']!=0]
```

In [21]:

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

Out[21]:

	id	season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets	pla
1	2	2017	Pune	06-04-2017	Mumbai Indians	Rising Pune Supergiant	Rising Pune Supergiant	field	normal	0	Rising Pune Supergiant	0	7	
2	3	2017	Rajkot	07-04-2017	Gujarat Lions	Kolkata Knight Riders	Kolkata Knight Riders	field	normal	0	Kolkata Knight Riders	0	10	
3	4	2017	Indore	08-04-2017	Rising Pune Supergiant	Kings XI Punjab	Kings XI Punjab	field	normal	0	Kings XI Punjab	0	6	
5	6	2017	Hyderabad	09-04-2017	Gujarat Lions	Sunrisers Hyderabad	Sunrisers Hyderabad	field	normal	0	Sunrisers Hyderabad	0	9	
6	7	2017	Mumbai	09-04-2017	Kolkata Knight Riders	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0	4	

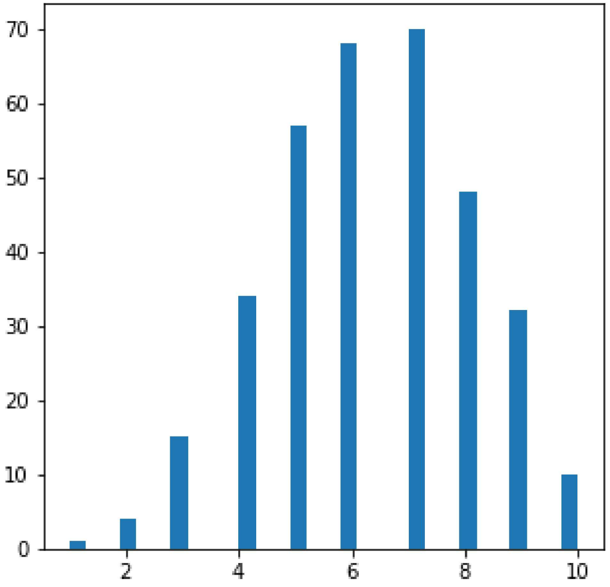
In [23]:

```
# Making a histogram for frequency of wins w.r.t. number of wickets

plt.figure(figsize=(5,5))
plt.hist(batting_second['win_by_wickets'],bins=30)
```

Out[23]:

```
(array([ 1.,  0.,  0.,  4.,  0.,  0., 15.,  0.,  0.,  0., 34.,  0.,  0.,
        57.,  0.,  0., 68.,  0.,  0.,  0., 70.,  0.,  0., 48.,  0.,  0.,
        32.,  0.,  0., 10.]),
 array([ 1. ,  1.3,  1.6,  1.9,  2.2,  2.5,  2.8,  3.1,  3.4,  3.7,  4. ,
         4.3,  4.6,  4.9,  5.2,  5.5,  5.8,  6.1,  6.4,  6.7,  7. ,  7.3,
         7.6,  7.9,  8.2,  8.5,  8.8,  9.1,  9.4,  9.7, 10. ]),
 <BarContainer object of 30 artists>)
```



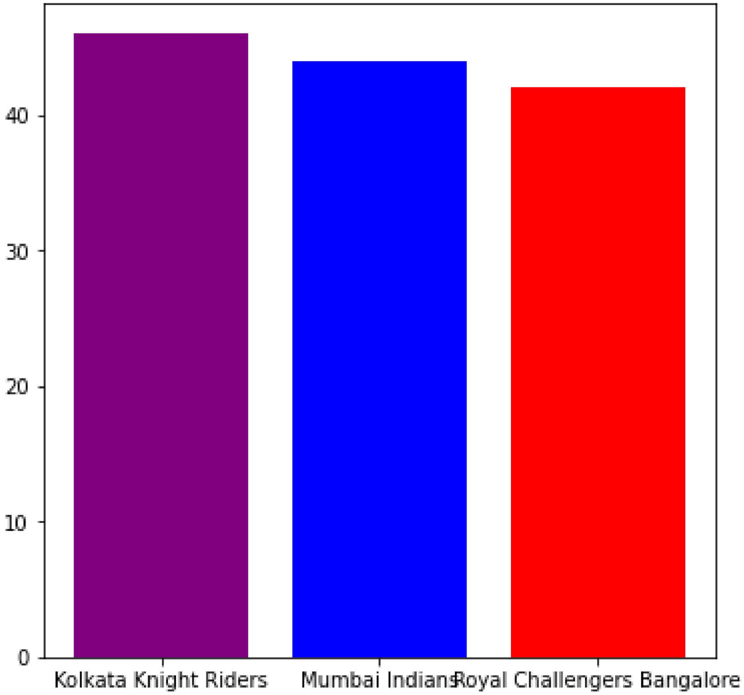
In [24]:

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

Out[24]:

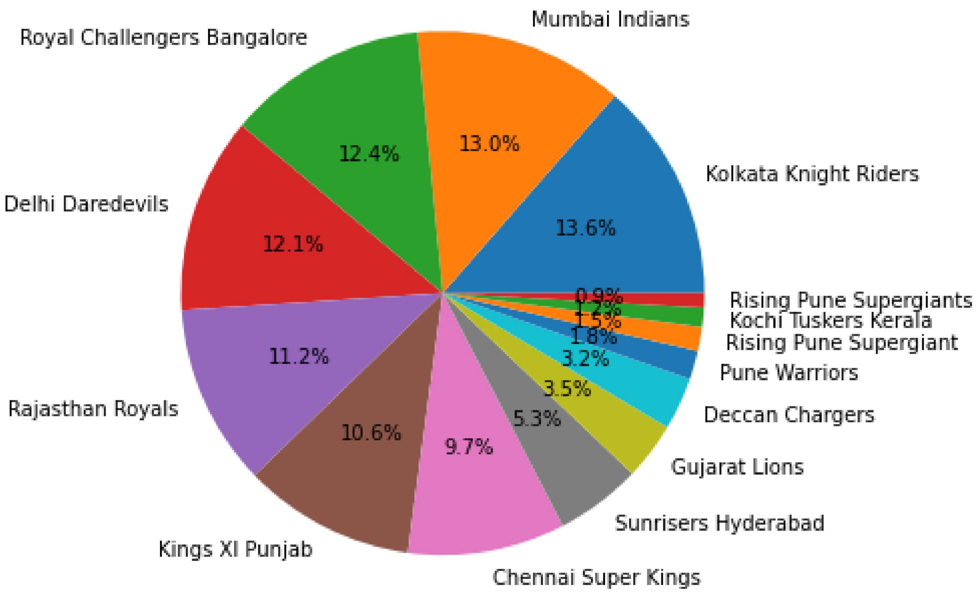
```
Kolkata Knight Riders          46
Mumbai Indians                 44
Royal Challengers Bangalore    42
Delhi Daredevils               41
Rajasthan Royals               38
Kings XI Punjab                36
Chennai Super Kings            33
Sunrisers Hyderabad            18
Gujarat Lions                  12
Deccan Chargers                11
Pune Warriors                   6
Rising Pune Supergiant          5
Kochi Tuskers Kerala            4
Rising Pune Supergiants         3
Name: winner, dtype: int64
```

In [26]:
```
# Making a bar plot for top 3 teams with most wins after batting second
plt.figure(figsize=(6,6))
plt.bar(list(batting_second['winner'].value_counts()[0:3].keys()),list(batting_second['winner'].value_counts()[0:3]),col
plt.show()
```



In [31]:

```
# Making a pie chart for distribution of most wins after batting second
plt.figure(figsize=(6,6))
plt.pie(list(batting_second['winner'].value_counts()),labels=list(batting_second['winner'].value_counts().keys()),autopc
plt.show()
```



In [32]:

```
# Looking at the number of matches played each season
ipl['season'].value_counts()
```

Out[32]:

```
2013    76
2012    74
2011    73
2010    60
2014    60
2016    60
2017    59
2015    59
2008    58
2009    57
Name: season, dtype: int64
```

In [33]:

```
# Looking at the number of matches played in each city
ipl['city'].value_counts()
```

Out[33]:

```
Mumbai            85
Bangalore         66
Kolkata           61
Delhi             60
Hyderabad         49
Chennai           48
Chandigarh        46
Jaipur            33
Pune              32
Durban            15
Ahmedabad         12
Centurion         12
Visakhapatnam     11
Rajkot            10
Dharamsala         9
Johannesburg       8
Cape Town          7
Abu Dhabi          7
Ranchi             7
Port Elizabeth     7
Cuttack            7
Raipur             6
Sharjah            6
Kochi              5
Indore             5
Kanpur             4
Nagpur             3
Kimberley          3
East London        3
Bloemfontein       2
Name: city, dtype: int64
```

In [34]:

```
# Finding out how many times a team has won the match after winning the loss
import numpy as np
np.sum(ipl['toss_winner']==ipl['winner'])
```

Out[34]:

```
325
```

In [35]:
```
325/636
```

Out[35]:
```
0.5110062893081762
```

In [36]:
```
deliveries=pd.read_csv('Documents\deliveries.csv')
```

In [37]:
```
deliveries.head()
```

Out[37]:

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker	bowler	is_super_over	...	bye_runs	legbye_runs	noball_runs
0	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	1	DA Warner	S Dhawan	TS Mills	0	...	0	0	0
1	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	2	DA Warner	S Dhawan	TS Mills	0	...	0	0	0
2	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	3	DA Warner	S Dhawan	TS Mills	0	...	0	0	0
3	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	4	DA Warner	S Dhawan	TS Mills	0	...	0	0	0
4	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	5	DA Warner	S Dhawan	TS Mills	0	...	0	0	0

5 rows × 21 columns

In [38]:

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

Out[38]:

```
array([    1,     2,     3,     4,     5,     6,     7,     8,     9,
          10,    11,    12,    13,    14,    15,    16,    17,    18,
          19,    20,    21,    22,    23,    24,    25,    26,    27,
          28,    29,    30,    31,    32,    33,    34,    35,    36,
          37,    38,    39,    40,    41,    42,    43,    44,    45,
          46,    47,    48,    49,    50,    51,    52,    53,    54,
          55,    56,    57,    58,    59,    60,    61,    62,    63,
          64,    65,    66,    67,    68,    69,    70,    71,    72,
          73,    74,    75,    76,    77,    78,    79,    80,    81,
          82,    83,    84,    85,    86,    87,    88,    89,    90,
          91,    92,    93,    94,    95,    96,    97,    98,    99,
         100,   101,   102,   103,   104,   105,   106,   107,   108,
         109,   110,   111,   112,   113,   114,   115,   116,   117,
         118,   119,   120,   121,   122,   123,   124,   125,   126,
         127,   128,   129,   130,   131,   132,   133,   134,   135,
         136,   137,   138,   139,   140,   141,   142,   143,   144,
         145,   146,   147,   148,   149,   150,   151,   152,   153,
         154,   155,   156,   157,   158,   159,   160,   161,   162,
         163,   164,   165,   166,   167,   168,   169,   170,   171,
         172,   173,   174,   175,   176,   177,   178,   179,   180,
         181,   182,   183,   184,   185,   186,   187,   188,   189,
         190,   191,   192,   193,   194,   195,   196,   197,   198,
         199,   200,   201,   202,   203,   204,   205,   206,   207,
         208,   209,   210,   211,   212,   213,   214,   215,   216,
         217,   218,   219,   220,   221,   222,   223,   224,   225,
         226,   227,   228,   229,   230,   231,   232,   233,   234,
         235,   236,   237,   238,   239,   240,   241,   242,   243,
         244,   245,   246,   247,   248,   249,   250,   251,   252,
         253,   254,   255,   256,   257,   258,   259,   260,   261,
         262,   263,   264,   265,   266,   267,   268,   269,   270,
         271,   272,   273,   274,   275,   276,   277,   278,   279,
         280,   281,   282,   283,   284,   285,   286,   287,   288,
         289,   290,   291,   292,   293,   294,   295,   296,   297,
         298,   299,   300,   301,   302,   303,   304,   305,   306,
         307,   308,   309,   310,   311,   312,   313,   314,   315,
         316,   317,   318,   319,   320,   321,   322,   323,   324,
         325,   326,   327,   328,   329,   330,   331,   332,   333,
         334,   335,   336,   337,   338,   339,   340,   341,   342,
         343,   344,   345,   346,   347,   348,   349,   350,   351,
         352,   353,   354,   355,   356,   357,   358,   359,   360,
         361,   362,   363,   364,   365,   366,   367,   368,   369,
         370,   371,   372,   373,   374,   375,   376,   377,   378,
         379,   380,   381,   382,   383,   384,   385,   386,   387,
         388,   389,   390,   391,   392,   393,   394,   395,   396,
         397,   398,   399,   400,   401,   402,   403,   404,   405,
         406,   407,   408,   409,   410,   411,   412,   413,   414,
         415,   416,   417,   418,   419,   420,   421,   422,   423,
         424,   425,   426,   427,   428,   429,   430,   431,   432,
         433,   434,   435,   436,   437,   438,   439,   440,   441,
         442,   443,   444,   445,   446,   447,   448,   449,   450,
         451,   452,   453,   454,   455,   456,   457,   458,   459,
         460,   461,   462,   463,   464,   465,   466,   467,   468,
         469,   470,   471,   472,   473,   474,   475,   476,   477,
         478,   479,   480,   481,   482,   483,   484,   485,   486,
         487,   488,   489,   490,   491,   492,   493,   494,   495,
         496,   497,   498,   499,   500,   501,   502,   503,   504,
         505,   506,   507,   508,   509,   510,   511,   512,   513,
         514,   515,   516,   517,   518,   519,   520,   521,   522,
         523,   524,   525,   526,   527,   528,   529,   530,   531,
         532,   533,   534,   535,   536,   537,   538,   539,   540,
         541,   542,   543,   544,   545,   546,   547,   548,   549,
         550,   551,   552,   553,   554,   555,   556,   557,   558,
         559,   560,   561,   562,   563,   564,   565,   566,   567,
         568,   569,   570,   571,   572,   573,   574,   575,   576,
         577,   578,   579,   580,   581,   582,   583,   584,   585,
         586,   587,   588,   589,   590,   591,   592,   593,   594,
         595,   596,   597,   598,   599,   600,   601,   602,   603,
         604,   605,   606,   607,   608,   609,   610,   611,   612,
         613,   614,   615,   616,   617,   618,   619,   620,   621,
         622,   623,   624,   625,   626,   627,   628,   629,   630,
         631,   632,   633,   634,   635,   636,  7894,  7895,  7896,
        7897,  7898,  7899,  7900,  7901,  7902,  7903,  7904,  7905,
        7906,  7907,  7908,  7909,  7910,  7911,  7912,  7913,  7914,
        7915,  7916,  7917,  7918,  7919,  7920,  7921,  7922,  7923,
        7924,  7925,  7926,  7927,  7928,  7929,  7930,  7931,  7932,
        7933,  7934,  7935,  7936,  7937,  7938,  7939,  7940,  7941,
        7942,  7943,  7944,  7945,  7946,  7947,  7948,  7949,  7950,
        7951,  7952,  7953, 11137, 11138, 11139, 11140, 11141, 11142,
       11143, 11144, 11145, 11146, 11147, 11148, 11149, 11150, 11151,
       11152, 11153, 11309, 11310, 11311, 11312, 11313, 11314, 11315,
       11316, 11317, 11318, 11319, 11320, 11321, 11322, 11323, 11324,
        11325, 11326, 11327, 11328, 11329, 11330, 11331, 11332, 11333,
        11334, 11335, 11336, 11337, 11338, 11339, 11340, 11341, 11342,
        11343, 11344, 11345, 11346, 11347, 11412, 11413, 11414, 11415],
       dtype=int64)
```

In [39]:

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

In [40]:

```
match_1.head()
```

Out[40]:

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker	bowler	is_super_over	...	bye_runs	legbye_runs	noball_runs
0	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	1	DA Warner	S Dhawan	TS Mills	0	...	0	0	0
1	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	2	DA Warner	S Dhawan	TS Mills	0	...	0	0	0
2	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	3	DA Warner	S Dhawan	TS Mills	0	...	0	0	0
3	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	4	DA Warner	S Dhawan	TS Mills	0	...	0	0	0
4	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	5	DA Warner	S Dhawan	TS Mills	0	...	0	0	0

5 rows × 21 columns

In [41]:

```
match_1.shape
```

Out[41]:

```
(248, 21)
```

In [43]:

```
srh = match_1[match_1['inning']==1]
```

```
In [44]: srh['batsman_runs'].value_counts()
```

```
Out[44]: 1    57
         0    32
         4    17
         6     9
         2     9
         3     1
         Name: batsman_runs, dtype: int64
```

```
In [45]: srh['dismissal_kind'].value_counts()
```

```
Out[45]: caught    3
         bowled    1
         Name: dismissal_kind, dtype: int64
```

```
In [46]: rcb=match_1[match_1['inning']==2]
```

```
In [49]: rcb['batsman_runs'].value_counts()
```

```
Out[49]: 0    49
         1    44
         4    15
         6     8
         2     7
         Name: batsman_runs, dtype: int64
```

```
In [51]: rcb['dismissal_kind'].value_counts()
```

```
Out[51]: caught     6
         bowled     2
         run out    2
         Name: dismissal_kind, dtype: int64
```

Analyzed by

Md Raiesh, Enrollment number : **19UME116**, Registration number : **1911345**, B Tech,7**th** semester,Section : **A**, Mechanical Engineering Department, National Institute of Technology Agartala, Tripura 799046,

In [ ]:

```

```