```
import seaborn as sns
import matplotlib.pyplot as plt

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

df = pd.read_csv("/content/ipl_dataset.csv")
df.head()
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 958,\n  \"fields\": [\
n    {\n      \"column\": \"full_scorecard\",\n      \"properties\":
{\n        \"dtype\": \"string\",\n        \"num_unique_values\":
958,\n        \"samples\": [\n
\"https://stats.espncricinfo.com/ci/engine/match/419162.html\",\n
\"https://stats.espncricinfo.com/ci/engine/match/829791.html\",\n
\"https://stats.espncricinfo.com/ci/engine/match/1082627.html\"\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"team1\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
18,\n        \"samples\": [\n          \"Chennai Super Kings\",\n
\"Mumbai Indians\",\n          \"Delhi Capitals\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"team2\",\n      \"properties\": {\
n        \"dtype\": \"category\",\n        \"num_unique_values\": 18,\
n        \"samples\": [\n          \"Kolkata Knight Riders\",\n
\"Delhi Capitals\",\n          \"Chennai Super Kings\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"team1_score\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
29.695734123174965,\n        \"min\": 67.0,\n        \"max\": 263.0,\n
\"num_unique_values\": 149,\n        \"samples\": [\n          129.0,\
n          162.0,\n          73.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"team2_score\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
30.606164639251055,\n        \"min\": 2.0,\n        \"max\": 226.0,\n
\"num_unique_values\": 148,\n        \"samples\": [\n          175.0,\
n          217.0,\n          93.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"toss_winner\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 20,\n        \"samples\": [\n
\"Kolkata Knight Riders\",\n          \"Deccan Chargers\",\n
\"8 (1 nb)\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"toss_choice\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 3,\n        \"samples\":
[\n          \"Field\",\n          \"Bat\",\n          \"No Toss\"\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"winner\",\n      \"properties\":
```

{\n   \"dtype\": \"category\",\n   \"num_unique_values\": 20,\n   \"samples\": [\n   \"KKR\",\n   \"Warriors\",\n   \"Match\"\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"margin\",\n   \"properties\": {\n   \"dtype\": \"category\",\n   \"num_unique_values\": 107,\n   \"samples\": [\n   \"51 runs\",\n   \"7 wickets\",\n   \"3 wickets\"\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"man_of_the_match\",\n   \"properties\": {\n   \"dtype\": \"category\",\n   \"num_unique_values\": 266,\n   \"samples\": [\n   \"Aditya Tare\",\n   \"Mujeeb Ur Rahman\",\n   \"Manan Vohra\"\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"stadium\",\n   \"properties\": {\n   \"dtype\": \"category\",\n   \"num_unique_values\": 40,\n   \"samples\": [\n   \"Saurashtra Cricket Association Stadium\",\n   \"Punjab Cricket Association IS Bindra Stadium\",\n   \"Rajiv Gandhi International Stadium\"\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"place\",\n   \"properties\": {\n   \"dtype\": \"category\",\n   \"num_unique_values\": 31,\n   \"samples\": [\n   \"East London\",\n   \"Kanpur\",\n   \"Cape Town\"\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   }\n]\n}","type":"dataframe","variable_name":"df"}

```
df.drop(['full_scorecard','place' ],axis =1 , inplace = True)
df
```

{"summary":"{\n \"name\": \"df\",\n \"rows\": 958,\n \"fields\": [\n   {\n   \"column\": \"team1\",\n   \"properties\": {\n   \"dtype\": \"category\",\n   \"num_unique_values\": 18,\n   \"samples\": [\n   \"Chennai Super Kings\",\n   \"Mumbai Indians\",\n   \"Delhi Capitals\"\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"team2\",\n   \"properties\": {\n   \"dtype\": \"category\",\n   \"num_unique_values\": 18,\n   \"samples\": [\n   \"Kolkata Knight Riders\",\n   \"Delhi Capitals\",\n   \"Chennai Super Kings\"\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"team1_score\",\n   \"properties\": {\n   \"dtype\": \"number\",\n   \"std\": 29.695734123174965,\n   \"min\": 67.0,\n   \"max\": 263.0,\n   \"num_unique_values\": 149,\n   \"samples\": [\n   129.0,\n   162.0,\n   73.0\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"team2_score\",\n   \"properties\": {\n   \"dtype\": \"number\",\n   \"std\":

```
30.606164639251055,\n        \"min\": 2.0,\n        \"max\": 226.0,\n
\"num_unique_values\": 148,\n        \"samples\": [\n          175.0,\
n          217.0,\n          93.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"toss_winner\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 20,\n        \"samples\": [\n
\"Kolkata Knight Riders\",\n          \"Deccan Chargers\",\n
\"8 (1 nb)\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"toss_choice\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 3,\n        \"samples\":
[\n          \"Field\",\n          \"Bat\",\n          \"No Toss\"\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"winner\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
20,\n        \"samples\": [\n          \"KKR\",\n
\"Warriors\",\n          \"Match\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"margin\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
107,\n        \"samples\": [\n          \"51 runs\",\n          \"7
wickets\",\n          \"3 wickets\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"man_of_the_match\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 266,\n        \"samples\": [\n
\"Aditya Tare\",\n          \"Mujeeb Ur Rahman\",\n          \"Manan
Vohra\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"stadium\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 40,\n
\"samples\": [\n          \"Saurashtra Cricket Association Stadium\",\
n          \"Punjab Cricket Association IS Bindra Stadium\",\n
\"Rajiv Gandhi International Stadium\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    }\n  ]\n}","type":"dataframe","variable_name":"df"}
```


```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 958 entries, 0 to 957
Data columns (total 10 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   team1             958 non-null    object
 1   team2             950 non-null    object
 2   team1_score       930 non-null    float64
 3   team2_score       948 non-null    float64
 4   toss_winner       950 non-null    object
```

```
 5   toss_choice       958 non-null    object
 6   winner            958 non-null    object
 7   margin            958 non-null    object
 8   man_of_the_match  950 non-null    object
 9   stadium           950 non-null    object
dtypes: float64(2), object(8)
memory usage: 75.0+ KB
```

```
df.shape
```

```
(958, 10)
```

```
df.describe()
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 8,\n  \"fields\": [\n
{\n      \"column\": \"team1_score\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 286.52218286276775,\n
\"min\": 29.695734123174965,\n        \"max\": 930.0,\n
\"num_unique_values\": 8,\n        \"samples\": [\n
163.1494623655914,\n          164.0,\n          930.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"team2_score\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
300.74222379007483,\n        \"min\": 2.0,\n        \"max\": 948.0,\n
\"num_unique_values\": 8,\n        \"samples\": [\n
149.28270042194092,\n          151.0,\n          948.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    }\n  ]\n}","type":"dataframe"}
```


```
df.isnull().sum()
```

```
team1                0
team2                8
team1_score         28
team2_score         10
toss_winner          8
toss_choice          0
winner               0
margin               0
man_of_the_match     8
stadium              8
dtype: int64
```

```
df.iloc[:,:-1]
df.dropna(inplace=True)
```

```
df.isnull().sum()
```

```
team1               0
team2               0
team1_score         0
```

```
team2_score          0
toss_winner          0
toss_choice          0
winner               0
margin               0
man_of_the_match     0
stadium              0
dtype: int64
```

```
df.shape
```

```
(930, 10)
```

```
df['team1'].unique()
```

```
array(['Chennai Super Kings', 'Mumbai Indians',
       'Royal Challengers Bangalore', 'Lucknow Super Giants',
       'Rajasthan Royals', 'Kolkata Knight Riders', 'Punjab Kings',
       'Gujarat Titans', 'Delhi Capitals', 'Sunrisers Hyderabad',
       'Kings XI Punjab', 'Delhi Daredevils', 'Gujarat Lions',
       'Rising Pune Supergiant', 'Rising Pune Supergiants',
       'Pune Warriors', 'Deccan Chargers', 'Kochi Tuskers Kerala'],
      dtype=object)
```

```
df['team1']=df['team1'].str.replace('Delhi Daredevils','Delhi
Capitals')
df['team2']=df['team2'].str.replace('Delhi Daredevils','Delhi
Capitals')
df['toss_winner']=df['toss_winner'].str.replace('Delhi
Daredevils','Delhi Capitals')

df['team1']=df['team1'].str.replace('Deccan Chargers','Sunrisers
Hyderabad')
df['team2']=df['team2'].str.replace('Deccan Chargers','Sunrisers
Hyderabad')
df['toss_winner']=df['toss_winner'].str.replace('Deccan
Chargers','Sunrisers Hyderabad')

df['team1']=df['team1'].str.replace('Rising Pune Supergiants','Rising
Pune Supergiant')
df['team2']=df['team2'].str.replace('Rising Pune Supergiants','Rising
Pune Supergiant')
df['toss_winner']=df['toss_winner'].str.replace('Rising Pune
Supergiants','Rising Pune Supergiant')

df['team1']=df['team1'].str.replace('Kings XI Punjab','Punjab Kings')
df['team2']=df['team2'].str.replace('Kings XI Punjab','Punjab Kings')
df['toss_winner']=df['toss_winner'].str.replace('Kings XI
Punjab','Punjab Kings')

df['team1'].unique()
```

```
array(['Chennai Super Kings', 'Mumbai Indians',
       'Royal Challengers Bangalore', 'Lucknow Super Giants',
       'Rajasthan Royals', 'Kolkata Knight Riders', 'Punjab Kings',
       'Gujarat Titans', 'Delhi Capitals', 'Sunrisers Hyderabad',
       'Gujarat Lions', 'Rising Pune Supergiant', 'Pune Warriors',
       'Kochi Tuskers Kerala'], dtype=object)
```

```
df["winner"].unique()
```

```
array(['KKR', 'Capitals', 'Punjab', 'Titans', 'Royals', 'RCB',
'Super',
       'Sunrisers', 'Mumbai', 'Tied', 'Kings', 'Daredevils',
'Supergiant',
       'Guj', 'Supergiants', 'No', 'Warriors', 'Chargers', 'Kochi'],
      dtype=object)
```

```
df['winner']=df['winner'].str.replace('Capitals','DC')
df['winner']=df['winner'].str.replace('Punjab','PBKS')
df['winner']=df['winner'].str.replace('Titans','GT')
df['winner']=df['winner'].str.replace('Royals','RR')
df['winner']=df['winner'].str.replace('Super','CSK')
df['winner']=df['winner'].str.replace('Sunrisers','SRH')
df['winner']=df['winner'].str.replace('Mumbai','MI')
df['winner']=df['winner'].str.replace('Kings','PBKS')
df['winner']=df['winner'].str.replace('Daredevils','DC')
df['winner']=df['winner'].str.replace('Supergiant','RPS')
df['winner']=df['winner'].str.replace('Guj','GL')
df['winner']=df['winner'].str.replace('Supergiants','RPS')
df['winner']=df['winner'].str.replace('Warriors','PWI')
df['winner']=df['winner'].str.replace('Chargers','SRH')
df['winner']=df['winner'].str.replace('Kochi','KTK')

df['winner']=df['winner'].str.replace('CSKgiants','RPS')
df['winner']=df['winner'].str.replace('CSKgiant','RPS')

df['winner']=df['winner'].str.replace('RPSs','RPS')

df["winner"].unique()
```

```
array(['KKR', 'DC', 'PBKS', 'GT', 'RR', 'RCB', 'CSK', 'SRH', 'MI',
'Tied',
       'RPS', 'GL', 'No', 'PWI', 'KTK'], dtype=object)
```

```
x = ["stadium", "toss_choice", "winner"]
for i in x:
  print("------------")
  print(df[i].unique())
  print(df[i].value_counts())
```

```
------------
['Wankhede Stadium' 'Brabourne Stadium' 'Dr DY Patil Sports Academy'
```

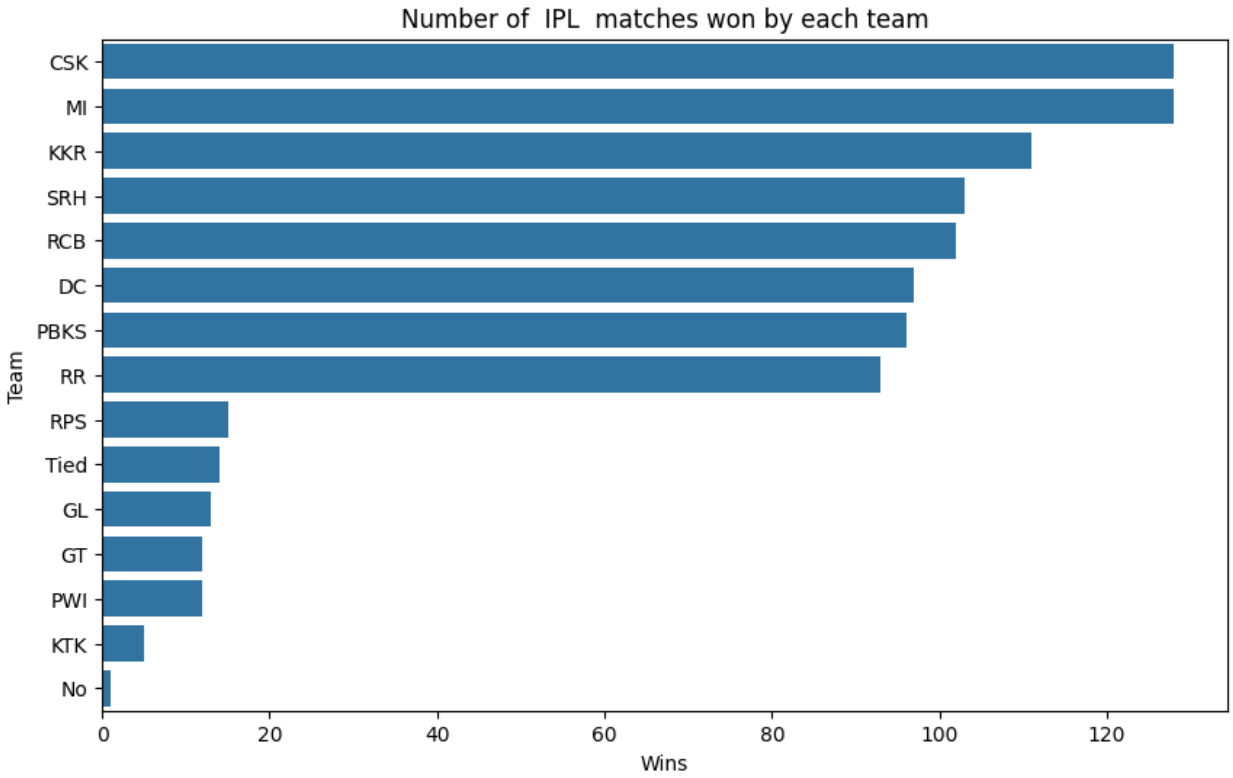
```
 'Maharashtra Cricket Association Stadium' 'Eden Gardens'
 'Narendra Modi Stadium' 'MA Chidambaram Stadium' 'Arun Jaitley
Stadium'
 'Dubai International Cricket Stadium' 'Zayed Cricket Stadium'
 'Sharjah Cricket Stadium' 'Sheikh Zayed Stadium' 'Sawai Mansingh
Stadium'
 'Feroz Shah Kotla' 'M Chinnaswamy Stadium'
 'Rajiv Gandhi International Stadium'
 'Punjab Cricket Association IS Bindra Stadium'
 'Andhra Cricket Association-Visakhapatnam District Cricket
Association Stadium'
 'Holkar Cricket Stadium' 'Saurashtra Cricket Association Stadium'
 'Green Park' 'Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Stadium'
 'Shaheed Veer Narayan Singh International Stadium' 'Sardar Patel
Stadium'
 'Punjab Cricket Association Stadium' 'JSCA International Stadium
Complex'
 'Barabati Stadium' 'Subrata Roy Sahara Stadium'
 'Himachal Pradesh Cricket Association Stadium' 'Nehru Stadium'
 'Vidarbha Cricket Association Stadium' 'Newlands' "St George's Park"
 'Kingsmead' 'SuperSport Park' 'Buffalo Park' 'New Wanderers Stadium'
 'De Beers Diamond Oval' 'OUTsurance Oval']
Wankhede Stadium
104
Eden Gardens
76
M Chinnaswamy Stadium
73
Feroz Shah Kotla
71
MA Chidambaram Stadium
67
Rajiv Gandhi International Stadium
63
Sawai Mansingh Stadium
46
Dubai International Cricket Stadium
46
Dr DY Patil Sports Academy
37
Maharashtra Cricket Association Stadium
34
Punjab Cricket Association Stadium
34
Sharjah Cricket Stadium
28
Zayed Cricket Stadium
28
Brabourne Stadium
```

```
27
Punjab Cricket Association IS Bindra Stadium
21
Subrata Roy Sahara Stadium
17
Kingsmead
15
Sardar Patel Stadium
12
Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Stadium
11
SuperSport Park
11
Saurashtra Cricket Association Stadium
10
Holkar Cricket Stadium
9
Sheikh Zayed Stadium
9
Himachal Pradesh Cricket Association Stadium
9
New Wanderers Stadium
8
Barabati Stadium
7
St George's Park
7
Narendra Modi Stadium
7
JSCA International Stadium Complex
6
Shaheed Veer Narayan Singh International Stadium
6
Newlands
6
Green Park
4
Nehru Stadium
4
Arun Jaitley Stadium
4
De Beers Diamond Oval
3
Vidarbha Cricket Association Stadium
3
Buffalo Park
3
Andhra Cricket Association-Visakhapatnam District Cricket Association
Stadium      2
```

```
OUTsurance Oval
2
Name: stadium, dtype: int64
------------
['Field' 'Bat']
Field    586
Bat      344
Name: toss_choice, dtype: int64
------------
['KKR' 'DC' 'PBKS' 'GT' 'RR' 'RCB' 'CSK' 'SRH' 'MI' 'Tied' 'RPS' 'GL'
'No'
 'PWI' 'KTK']
CSK     128
MI      128
KKR     111
SRH     103
RCB     102
DC       97
PBKS     96
RR       93
RPS      15
Tied     14
GL       13
GT       12
PWI      12
KTK       5
No        1
Name: winner, dtype: int64
```

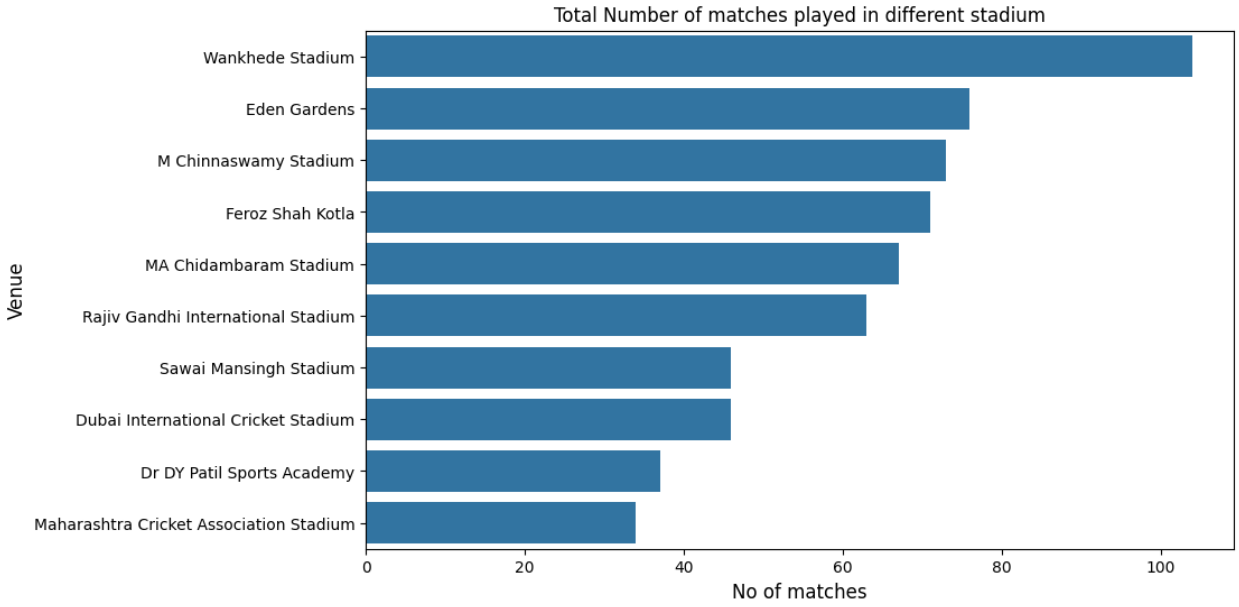
```
plt.figure(figsize = (10,6))
sns.countplot(y = 'winner',data = df,order=
df['winner'].value_counts().index)
plt.xlabel('Wins')
plt.ylabel('Team')
plt.title('Number of  IPL  matches won by each team')
```

```
Text(0.5, 1.0, 'Number of  IPL  matches won by each team')
```



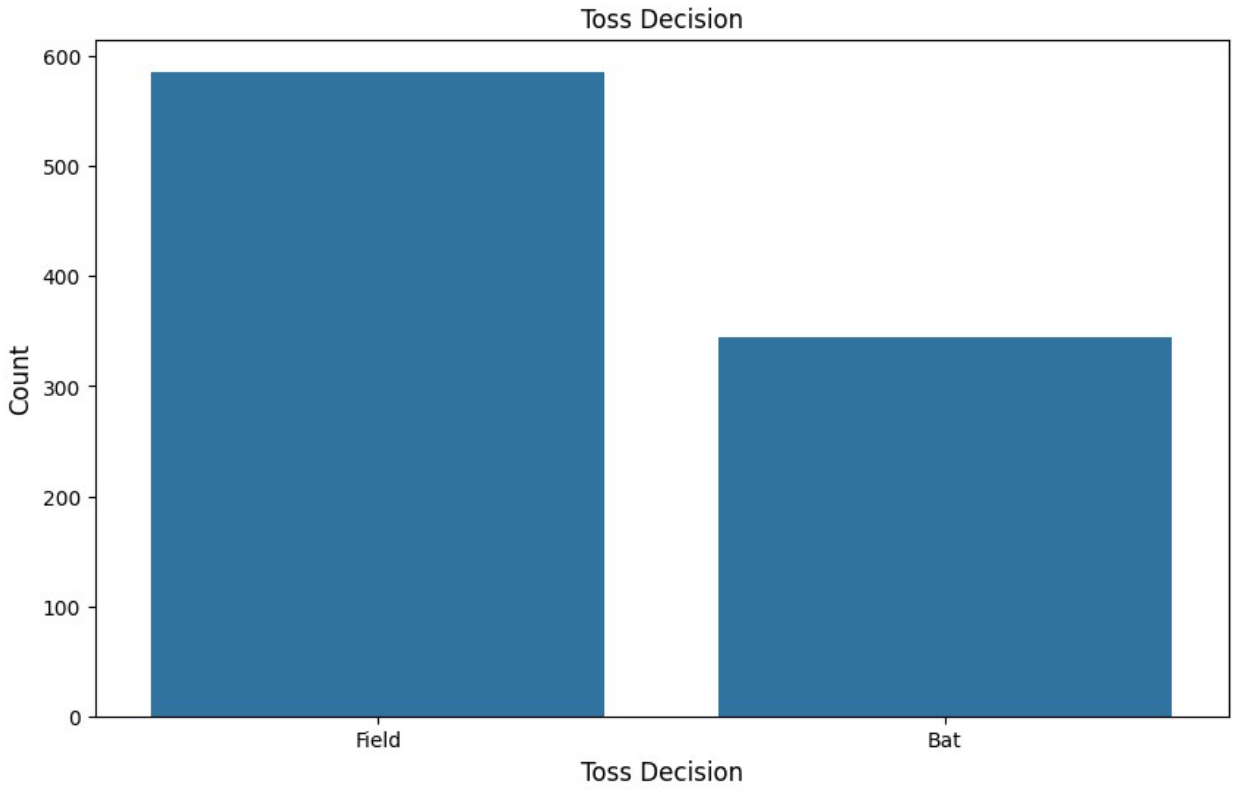

```
plt.figure(figsize = (10,6))
sns.countplot(y = 'stadium',data = df,order =
df['stadium'].value_counts().iloc[:10].index)
plt.xlabel('No of matches',fontsize=12)
plt.ylabel('Venue',fontsize=12)
plt.title('Total Number of matches played in different stadium')
```

```
Text(0.5, 1.0, 'Total Number of matches played in different stadium')
```




```
plt.figure(figsize = (10,6))
sns.countplot(x = "toss_choice", data=df)
plt.xlabel('Toss Decision',fontsize=12)
plt.ylabel('Count',fontsize=12)
plt.title('Toss Decision')
```

```
Text(0.5, 1.0, 'Toss Decision')
```



df


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 930,\n  \"fields\": [\n    {\n      \"column\": \"team1\",\n      \"properties\": {\n        \"dtype\": \"category\",\n        \"num_unique_values\": 14,\n        \"samples\": [\n          \"Sunrisers Hyderabad\",\n          \"Rising Pune Supergiant\",\n          \"Chennai Super Kings\"\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"team2\",\n      \"properties\": {\n        \"dtype\": \"category\",\n        \"num_unique_values\": 14,\n        \"samples\": [\n          \"Rajasthan Royals\",\n          \"Gujarat Lions\",\n          \"Kolkata Knight Riders\"\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"team1_score\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 29.695734123174965,\n        \"min\": 67.0,\n        \"max\": 263.0,\n        \"num_unique_values\": 149,\n        \"samples\": [\n          129.0,\n          162.0,\n          73.0\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"team2_score\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 29.807725198313946,\n        \"min\": 2.0,\n        \"max\": 226.0,\n        \"num_unique_values\": 146,\n        \"samples\": [\n          137.0,\n          99.0,\n          148.0\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
```

```
n    },\n    {\n      \"column\": \"toss_winner\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 14,\n        \"samples\": [\n
\"Rajasthan Royals\",\n          \"Gujarat Lions\",\n
\"Kolkata Knight Riders\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"toss_choice\",\n      \"properties\": {\n
\"dtype\": \"category\",\n        \"num_unique_values\": 2,\n
\"samples\": [\n          \"Bat\",\n          \"Field\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"winner\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
15,\n        \"samples\": [\n          \"Tied\",\n          \"GL\"\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"margin\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
106,\n        \"samples\": [\n          \"92 runs\",\n          \"7
wickets\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"man_of_the_match\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 261,\n
\"samples\": [\n          \"Shikhar Dhawan\",\n          \"Praveen
Kumar\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"stadium\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 39,\n
\"samples\": [\n          \"Kingsmead\",\n          \"New Wanderers
Stadium\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    }\n  ]\
n}","type":"dataframe","variable_name":"df"}
```


```
x=
df.drop(['team1_score','team2_score','man_of_the_match','stadium','win
ner'],axis=1)
x
```


```
{"summary":"{\n  \"name\": \"x\",\n  \"rows\": 930,\n  \"fields\": [\n
{\n      \"column\": \"team1\",\n      \"properties\": {\n
\"dtype\": \"category\",\n        \"num_unique_values\": 14,\n
\"samples\": [\n          \"Sunrisers Hyderabad\",\n          \"Rising
Pune Supergiant\",\n          \"Chennai Super Kings\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"team2\",\n      \"properties\": {\
n        \"dtype\": \"category\",\n        \"num_unique_values\": 14,\
n        \"samples\": [\n          \"Rajasthan Royals\",\n
\"Gujarat Lions\",\n          \"Kolkata Knight Riders\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"toss_winner\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 14,\n        \"samples\": [\n
```

\"Rajasthan Royals\",\n   \"Gujarat Lions\",\n
\"Kolkata Knight Riders\"\n   ],\n   \"semantic_type\":
\"\",\n   \"description\": \"\"\n   }\n   },\n   {\n
\"column\": \"toss_choice\",\n   \"properties\": {\n
\"dtype\": \"category\",\n   \"num_unique_values\": 2,\n
\"samples\": [\n   \"Bat\",\n   \"Field\"\n   ],\n
\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\
n   },\n   {\n   \"column\": \"margin\",\n   \"properties\":
{\n   \"dtype\": \"category\",\n   \"num_unique_values\":
106,\n   \"samples\": [\n   \"92 runs\",\n   \"7
wickets\"\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   }\n]\
n}","type":"dataframe","variable_name":"x"}

```
x = pd.get_dummies(x, ["team1","team2", "toss_winner", "toss_choice",
"result"], drop_first = True)
```

```
y = df["winner"]
y
```

```
0       KKR
1        DC
2      PBKS
3        GT
4        RR
       ...
952     CSK
954    PBKS
955      RR
956     CSK
957      RR
Name: winner, Length: 930, dtype: object
```

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
y = le.fit_transform(y)
```

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, train_size =
0.8)
```

```
from sklearn.ensemble import RandomForestClassifier
model =
RandomForestClassifier(n_estimators=200,min_samples_split=3,max_featur
es = "auto")
model
```

```
RandomForestClassifier(max_features='auto', min_samples_split=3,
                       n_estimators=200)
```

```
model.fit(x_train, y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/
_forest.py:424: FutureWarning: `max_features='auto'` has been
deprecated in 1.1 and will be removed in 1.3. To keep the past
behaviour, explicitly set `max_features='sqrt'` or remove this
parameter as it is also the default value for RandomForestClassifiers
and ExtraTreesClassifiers.
  warn(
```

```
RandomForestClassifier(max_features='auto', min_samples_split=3,
                       n_estimators=200)
```

```
y_pred = model.predict(x_test)
```

```
from sklearn.metrics import accuracy_score
ac = accuracy_score(y_pred, y_test)
ac
```

```
0.6182795698924731
```