THE SPARKS FOUNDATION

DATA SCIENCE AND BUSINESS ANALYTICS INTERNSHIP

TASK 4 - Performing Exploratory Data Analysis on dataset 'Indian Premier League'

AUTHOR: MEGHA

In [2]:
```
#importing required libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings(action='ignore')
```

In [3]:
```
df_deliv=pd.read_csv('deliveries.csv')
df_matches=pd.read_csv('matches.csv')
```

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

Out[4]:
```
(179078, 21)
```

In [5]:
```
df_matches.shape
```

Out[5]:
```
(756, 18)
```

In [6]:
```
df_deliv.isnull().sum()
```

Out[6]:
```
match_id          0
inning            0
batting_team      0
bowling_team      0
over              0
ball              0
batsman           0
```

```
non_striker              0
bowler                   0
is_super_over            0
wide_runs                0
bye_runs                 0
legbye_runs              0
noball_runs              0
penalty_runs             0
batsman_runs             0
extra_runs               0
total_runs               0
player_dismissed    170244
dismissal_kind      170244
fielder             172630
dtype: int64
```

In [9]:
```
df_matches.isnull().sum()
```

Out[9]:
```
id                   0
season               0
city                 7
date                 0
team1                0
team2                0
toss_winner          0
toss_decision        0
result               0
dl_applied           0
winner               4
win_by_runs          0
win_by_wickets       0
player_of_match      4
venue                0
umpire1              2
umpire2              2
umpire3            637
dtype: int64
```

In [11]:
```
df_matches.drop(['umpire1','umpire2','umpire3'],axis=1,inplace=True)
```

In [12]:
```
df_matches['city'].unique()
```

Out[12]:
```
array(['Hyderabad', 'Pune', 'Rajkot', 'Indore', 'Bangalore', 'Mumbai',
       'Kolkata', 'Delhi', 'Chandigarh', 'Kanpur', 'Jaipur', 'Chennai',
```

```
       'Cape Town', 'Port Elizabeth', 'Durban', 'Centurion',
       'East London', 'Johannesburg', 'Kimberley', 'Bloemfontein',
       'Ahmedabad', 'Cuttack', 'Nagpur', 'Dharamsala', 'Kochi',
       'Visakhapatnam', 'Raipur', 'Ranchi', 'Abu Dhabi', 'Sharjah', nan,
       'Mohali', 'Bengaluru'], dtype=object)
```

In [13]:
```
df_matches[df_matches.isnull()][['city','date']]
```

Out[13]:

	city	date
0	NaN	NaN
1	NaN	NaN
2	NaN	NaN
3	NaN	NaN
4	NaN	NaN
...	...	...
751	NaN	NaN
752	NaN	NaN
753	NaN	NaN
754	NaN	NaN
755	NaN	NaN

756 rows × 2 columns

In [16]:
```
df_matches.city=df_matches.city.fillna('Dubai')
df_matches.isnull().sum()
```

Out[16]:
```
id               0
season           0
city             0
date             0
team1            0
team2            0
toss_winner      0
toss_decision    0
result           0
```

```
dl_applied         0
winner             4
win_by_runs        0
win_by_wickets     0
player_of_match    4
venue              0
dtype: int64
```

In [17]:
```
df_matches[df_matches.winner.isnull()][['result','winner']]
```

Out[17]:

	result	winner
300	no result	NaN
545	no result	NaN
570	no result	NaN
744	no result	NaN

In [18]:
```
df_deliv.shape
```

Out[18]:
```
(179078, 21)
```

In [19]:
```
df_matches.shape
```

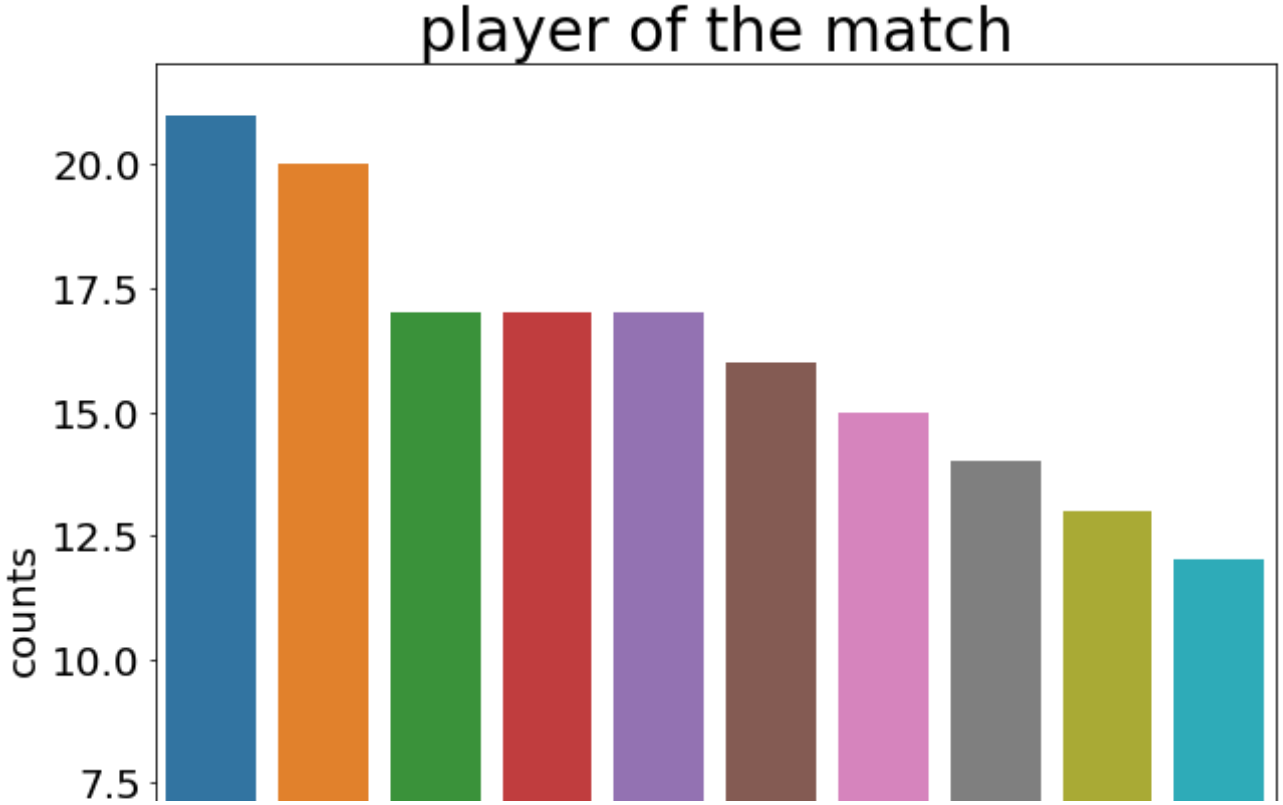
Out[19]:
```
(756, 15)
```

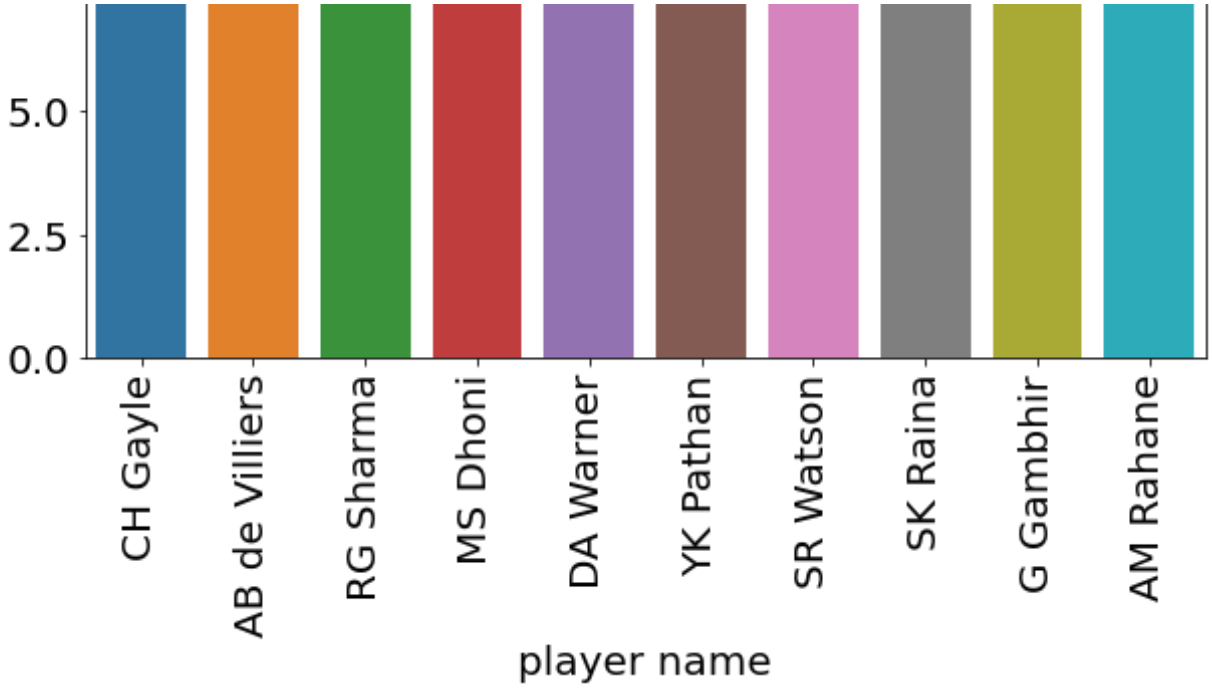
In [20]:
```
df_matches.player_of_match.value_counts()
```

Out[20]:
```
CH Gayle          21
AB de Villiers    20
RG Sharma         17
MS Dhoni          17
DA Warner         17
                  ..
SA Asnodkar        1
GD McGrath         1
J Botha            1
S Nadeem           1
DP Nannes          1
Name: player_of_match, Length: 226, dtype: int64
```

In [45]:
```
plt.figure(figsize=(10,10))
#sns.set(style='whitegrid')
player1=df_matches["player_of_match"].value_counts().index
print(player1)
sns.countplot(df_matches.player_of_match,order=player1[:10])
plt.title("player of the match",size=30)
plt.xlabel("player name", size=20)
plt.ylabel("counts",size=20)
plt.xticks(size=20, rotation=90)
plt.yticks(size=20)
plt.show()
```

```
Index(['CH Gayle', 'AB de Villiers', 'RG Sharma', 'MS Dhoni', 'DA Warner',
       'YK Pathan', 'SR Watson', 'SK Raina', 'G Gambhir', 'AM Rahane',
       ...
       'LJ Wright', 'EJG Morgan', 'TG Southee', 'M Kartik', 'Shoaib Akhtar',
       'SA Asnodkar', 'GD McGrath', 'J Botha', 'S Nadeem', 'DP Nannes'],
      dtype='object', length=226)
```





In [27]:
```
df_deliv2 =df_deliv.groupby("batsman").sum("batsman_runs")
```

In [28]:
```
df_deliv3 =df_deliv2.sort_values("batsman_runs",ascending=False)
df_deliv4 =df_deliv3.reset_index()
```

In [29]:
```
df_deliv4.head(5)
```

Out[29]:

	batsman	match_id	inning	over	ball	is_super_over	wide_runs	bye_runs	legbye_runs	noball_runs	penalty_runs	batsman_runs	extra_run
0	V Kohli	8090018	6195	39005	14728	2	118	5	45	14	0	5434	18
1	SK Raina	7509946	5755	38851	14677	3	149	4	64	19	0	5415	23
2	RG Sharma	6448276	5526	40942	13479	0	78	2	50	13	0	4914	14
3	DA Warner	6673034	4936	24904	11899	1	127	28	63	16	0	4741	23

	batsman	match_id	inning	over	ball	is_super_over	wide_runs	bye_runs	legbye_runs	noball_runs	penalty_runs	batsman_runs	extra_run
4	S Dhawan	8446314	5352	26567	13428	0	141	12	79	12	0	4632	24

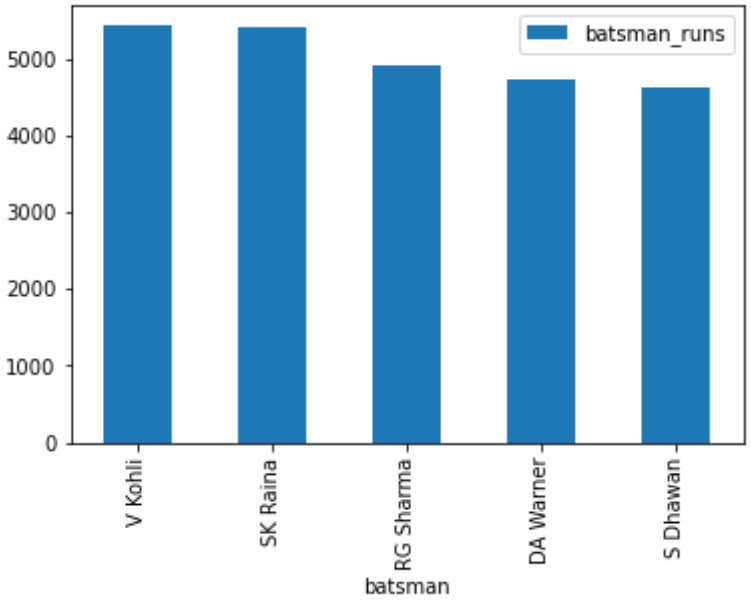
In [30]:
```
df_deliv5 = df_deliv4[["batsman","batsman_runs"]]
df_deliv5.head(5)
```

Out[30]:

	batsman	batsman_runs
0	V Kohli	5434
1	SK Raina	5415
2	RG Sharma	4914
3	DA Warner	4741
4	S Dhawan	4632

In [31]:
```
df_deliv5.head().plot.bar(x="batsman",y="batsman_runs")
```

Out[31]:
```
<AxesSubplot:xlabel='batsman'>
```



In [32]:
```
teams=df_matches["winner"].value_counts()
teams
```

Out[32]:
```
Mumbai Indians                109
Chennai Super Kings           100
Kolkata Knight Riders          92
Royal Challengers Bangalore    84
Kings XI Punjab                82
Rajasthan Royals               75
Delhi Daredevils               67
Sunrisers Hyderabad            58
Deccan Chargers                29
Gujarat Lions                  13
Pune Warriors                  12
Rising Pune Supergiant         10
Delhi Capitals                 10
Kochi Tuskers Kerala            6
Rising Pune Supergiants         5
Name: winner, dtype: int64
```

In [34]:
```
df_matches["winner"].replace(to_replace='Rising Pune Supergiants',value='Rising Pune Supergiants',inplace=True)
```

In [35]:
```
teams=df_matches["winner"].value_counts()
teams
```

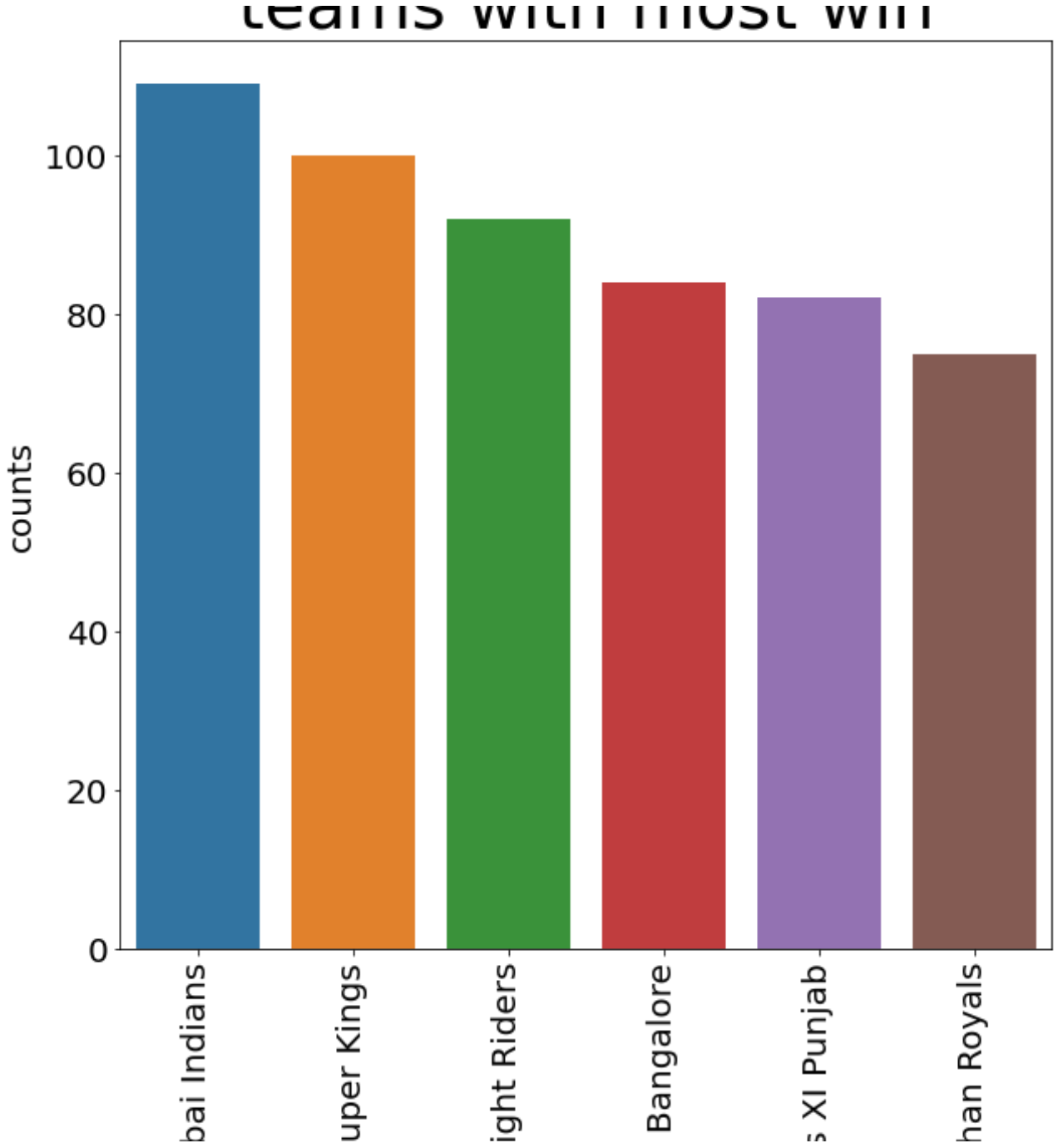
Out[35]:
```
Mumbai Indians                109
Chennai Super Kings           100
Kolkata Knight Riders          92
Royal Challengers Bangalore    84
Kings XI Punjab                82
Rajasthan Royals               75
Delhi Daredevils               67
Sunrisers Hyderabad            58
Deccan Chargers                29
Gujarat Lions                  13
Pune Warriors                  12
Rising Pune Supergiant         10
Delhi Capitals                 10
Kochi Tuskers Kerala            6
Rising Pune Supergiants         5
Name: winner, dtype: int64
```

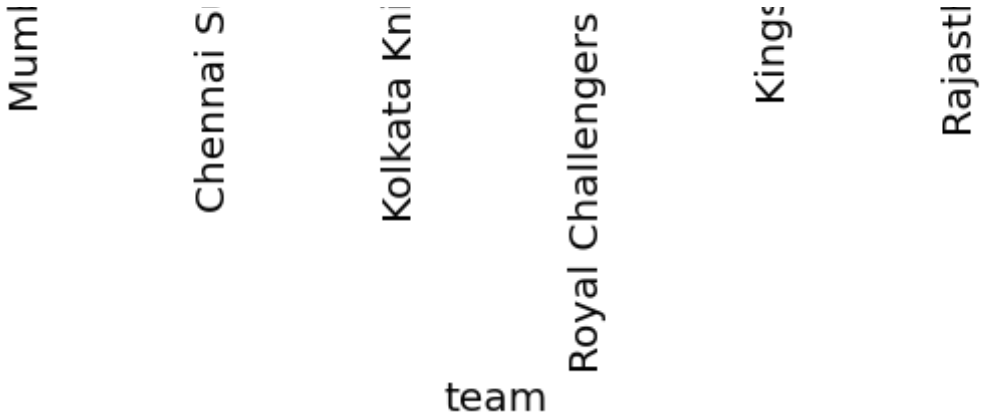
In [36]:
```
len(teams)
```

Out[36]:
```
15
```

In [43]:
```
plt.figure(figsize=(10,10))
#sns.set(style='whitegrid')
team_wins=df_matches["winner"].value_counts().index
print(team_wins)
sns.countplot(df_matches.winner,order=team_wins[:6])
plt.title("teams with most win",size=40)
plt.xlabel("team", size=20)
plt.ylabel("counts",size=20)
plt.xticks(size=20, rotation=90)
plt.yticks(size=20)
plt.show()
```

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





In [49]:
```
final_matches=df_matches.drop_duplicates(subset=['season'],keep='last')
k=final_matches[['winner','season']].reset_index(drop=True).sort_values('season')
print(k)
k.winner.unique()
```

```
                   winner  season
1        Rajasthan Royals    2008
2         Deccan Chargers    2009
3     Chennai Super Kings    2010
4     Chennai Super Kings    2011
5   Kolkata Knight Riders    2012
6          Mumbai Indians    2013
7   Kolkata Knight Riders    2014
8          Mumbai Indians    2015
9     Sunrisers Hyderabad    2016
0          Mumbai Indians    2017
10    Chennai Super Kings    2018
11         Mumbai Indians    2019
```

Out[49]:
```
array(['Rajasthan Royals', 'Deccan Chargers', 'Chennai Super Kings',
       'Kolkata Knight Riders', 'Mumbai Indians', 'Sunrisers Hyderabad'],
      dtype=object)
```

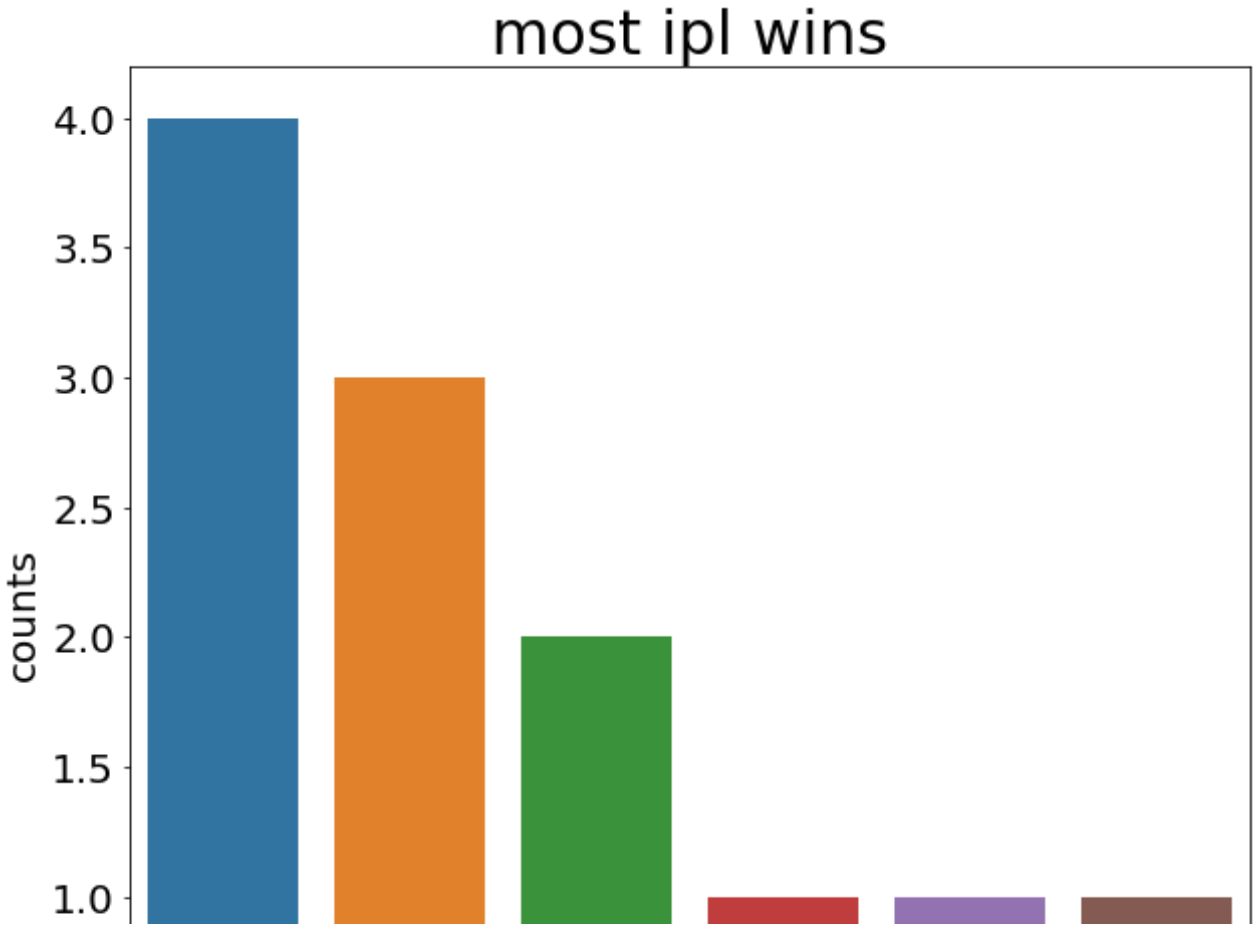
In [50]:
```
k['winner'].value_counts()
```

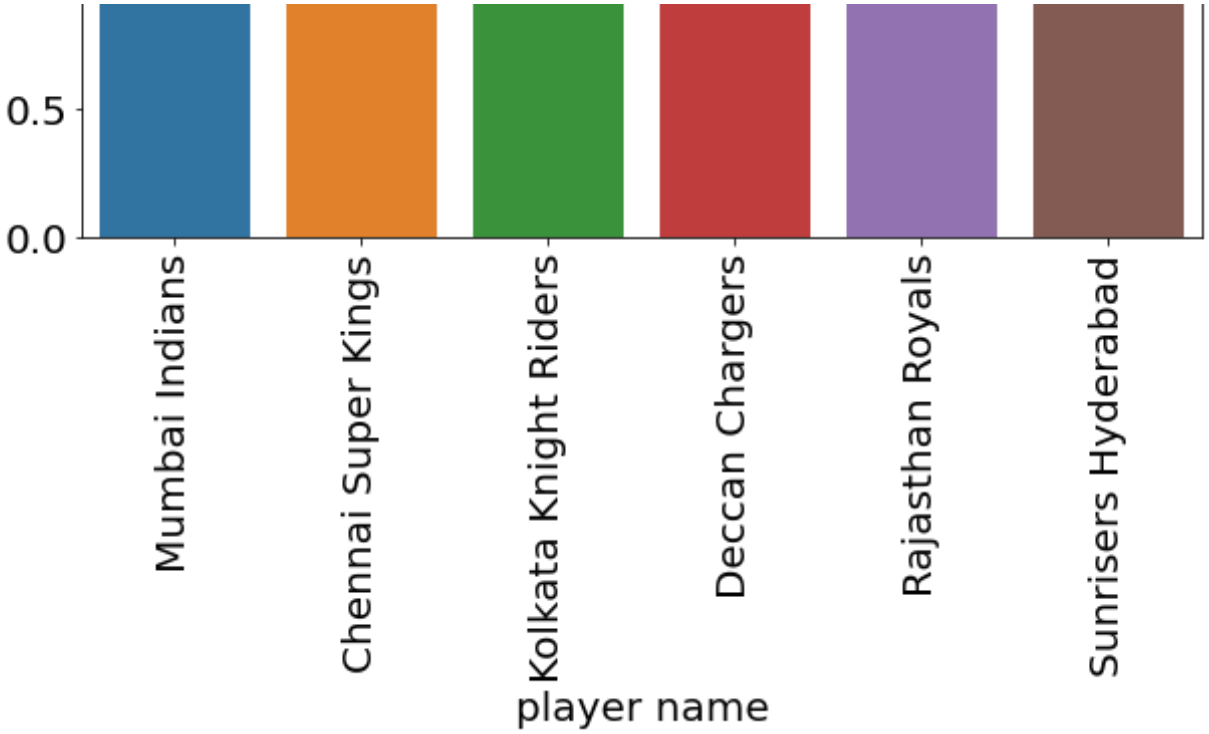
Out[50]:
```
Mumbai Indians           4
Chennai Super Kings      3
Kolkata Knight Riders    2
Deccan Chargers          1
Rajasthan Royals         1
```

```
Sunrisers Hyderabad      1
Name: winner, dtype: int64
```

In [54]:

```
plt.figure(figsize=(10,10))
#sns.set(style='whitegrid')
k2=k["winner"].value_counts().index
sns.countplot(k.winner,order=k2[:6])
plt.title("most ipl wins",size=30)
plt.xlabel("player name", size=20)
plt.ylabel("counts",size=20)
plt.xticks(size=20, rotation=90)
plt.yticks(size=20)
plt.show()
```





In [56]:
```
most_toss_wins=df_matches['toss_winner'].value_counts()
most_toss_wins
```

Out[56]:
```
Mumbai Indians                98
Kolkata Knight Riders         92
Chennai Super Kings           89
Kings XI Punjab               81
Royal Challengers Bangalore   81
Delhi Daredevils              80
Rajasthan Royals              80
Sunrisers Hyderabad           46
Deccan Chargers               43
Pune Warriors                 20
Gujarat Lions                 15
Delhi Capitals                10
Kochi Tuskers Kerala           8
Rising Pune Supergiants        7
Rising Pune Supergiant         6
Name: toss_winner, dtype: int64
```

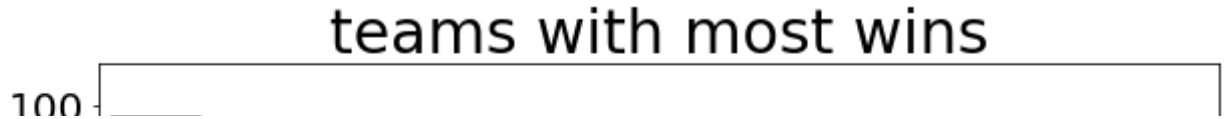
In [57]:
```
df_matches['toss_winner'].replace(to_replace='Rising Pune Supergiants',inplace=True)
```

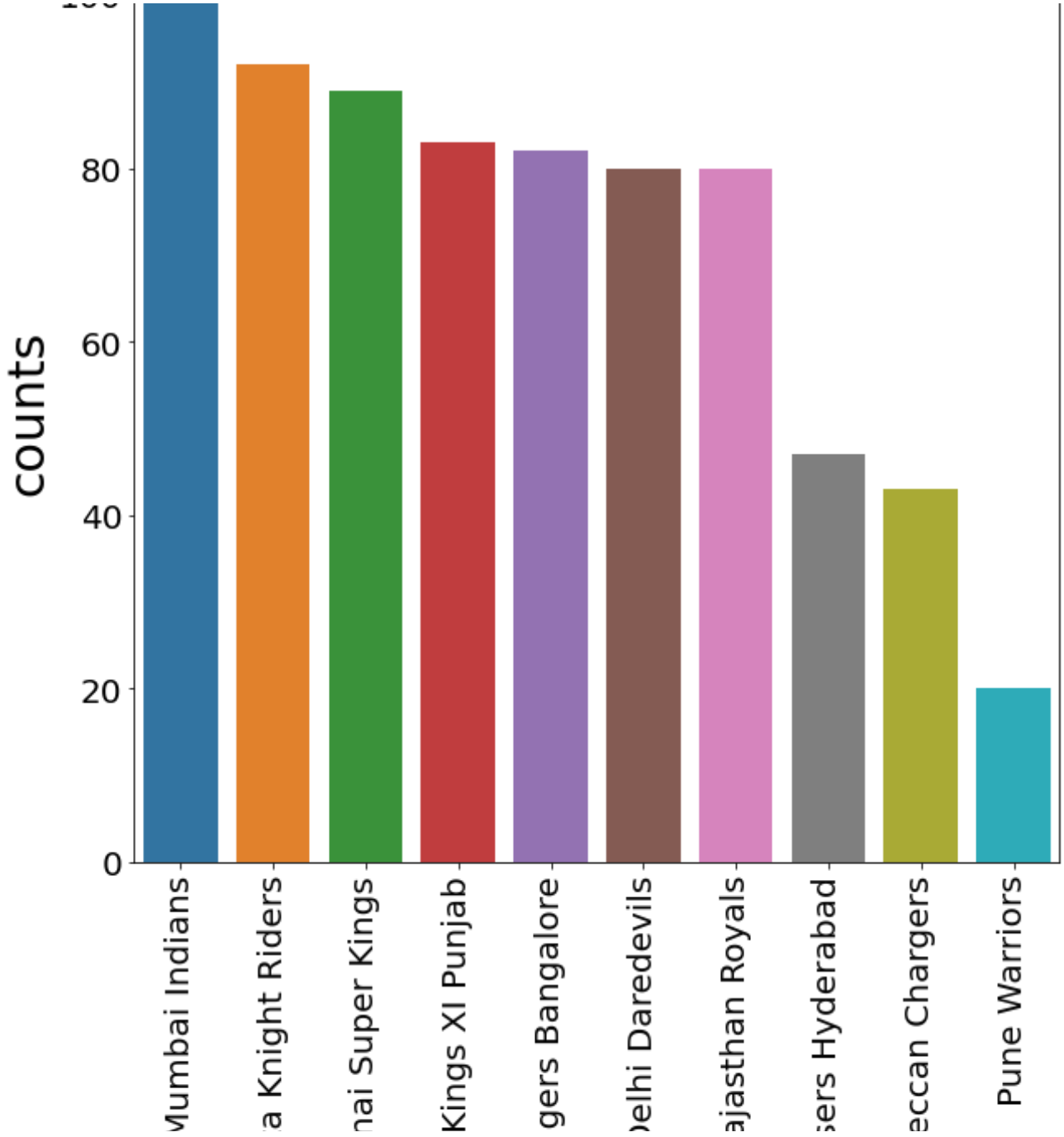
In [58]:
```
most_toss_wins=df_matches['toss_winner'].value_counts()
most_toss_wins
```

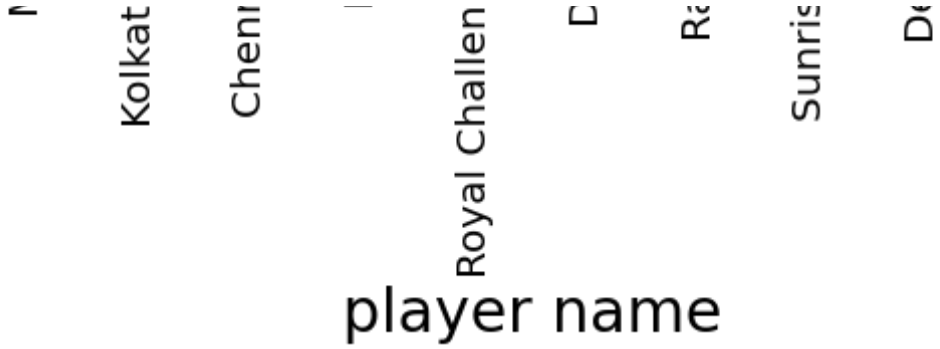
Out[58]:
```
Mumbai Indians                 99
Kolkata Knight Riders          92
Chennai Super Kings            89
Kings XI Punjab                83
Royal Challengers Bangalore    82
Delhi Daredevils               80
Rajasthan Royals               80
Sunrisers Hyderabad            47
Deccan Chargers                43
Pune Warriors                  20
Gujarat Lions                  17
Delhi Capitals                 10
Kochi Tuskers Kerala            8
Rising Pune Supergiant          6
Name: toss_winner, dtype: int64
```

In [63]:
```
plt.figure(figsize=(10,10))
#sns.set(style='whitegrid')
team_toss=df_matches["toss_winner"].value_counts().index
print(team_toss)
sns.countplot(df_matches.toss_winner,order=team_toss[:10])
plt.title("teams with most wins",size=30)
plt.xlabel("player name", size=30)
plt.ylabel("counts",size=30)
plt.xticks(size=20, rotation=90)
plt.yticks(size=20)
plt.show()
```

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







In [64]:
```
toss_decision=df_matches['toss_decision'].value_counts()
```

In [65]:
```
toss_decision
```

Out[65]:
```
field    463
bat      293
Name: toss_decision, dtype: int64
```

In [66]:
```
most_wins_batting_first =df_matches.winner[df_matches.win_by_runs!=0].value_counts().index
most_wins_batting_first
```

Out[66]:
```
Index(['Mumbai Indians', 'Chennai Super Kings', 'Kings XI Punjab',
       'Kolkata Knight Riders', 'Royal Challengers Bangalore',
       'Sunrisers Hyderabad', 'Rajasthan Royals', 'Delhi Daredevils',
       'Deccan Chargers', 'Pune Warriors', 'Rising Pune Supergiant',
       'Delhi Capitals', 'Kochi Tuskers Kerala', 'Rising Pune Supergiants',
       'Gujarat Lions'],
      dtype='object')
```

In [69]:
```
plt.figure(figsize=(10,10))
#sns.set(style='whitegrid')
most_wins_batting_first=df_matches.winner[df_matches.win_by_runs!=0].value_counts().index
print(most_wins_batting_first)
sns.countplot(df_matches.winner[df_matches.win_by_runs!=0],order=most_wins_batting_first[:10])
plt.title("teams with first batting and wins",size=30)
plt.xlabel("player name", size=30)
plt.ylabel("counts",size=30)
plt.xticks(size=20, rotation=90)
plt.yticks(size=20)
plt.show()
```

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