IPL PROJECT

Loading libraries and Data Sets

In [70]:
```
# importing libraries
import numpy as np
import pandas as pd
import seaborn as sb
import matplotlib.pyplot as plt


# loading data
ipl_data = pd.read_csv("IPL.csv")
```

Basic Information

In [74]:
```
# Size of data
print(f" no. of rows and col {ipl_data.shape}\n")


# how many col have null values
print(ipl_data.isnull().sum())
```

```
no. of rows and col (74, 20)

match_id              0
date                  0
venue                 0
team1                 0
team2                 0
stage                 0
toss_winner           0
toss_decision         0
first_ings_score      0
first_ings_wkts       0
second_ings_score     0
second_ings_wkts      0
match_winner          0
won_by                0
margin                0
player_of_the_match   0
top_scorer            0
highscore             0
best_bowling          0
best_bowling_figure   0
dtype: int64
```

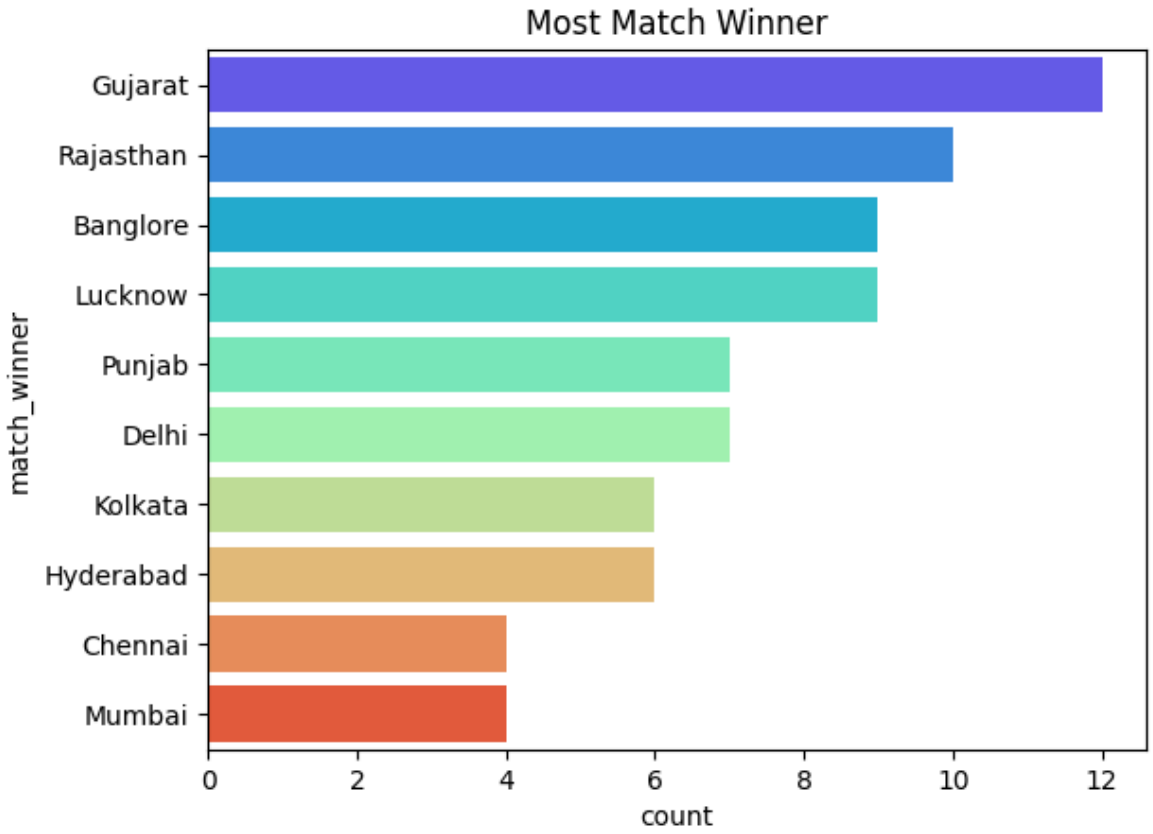
Some basic Question

In [75]:
```
# which team won the most matches
match_winner = ipl_data["match_winner"].value_counts()
print(match_winner)
```

```
# Find the team with the most wins
most_match_winner = match_winner.value_counts().max() # result = 12
most_match_winner_team_name = match_winner.value_counts().idxmax()

print(f"Team with most wins: {most_match_winner_team_name} with {most_match_winn

print(f"\nPlot represenation")
sb.barplot(x=match_winner, y = match_winner.index,palette="rainbow")
plt.title("Most Match Winner")
```

```
match_winner
Gujarat      12
Rajasthan    10
Banglore      9
Lucknow       9
Punjab        7
Delhi         7
Kolkata       6
Hyderabad     6
Chennai       4
Mumbai        4
Name: count, dtype: int64
Team with most wins: 7 with 2 wins


Plot represenation
C:\Users\usman\AppData\Local\Temp\ipykernel_2664\2107393702.py:12: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effe
ct.

  sb.barplot(x=match_winner, y = match_winner.index,palette="rainbow")
```

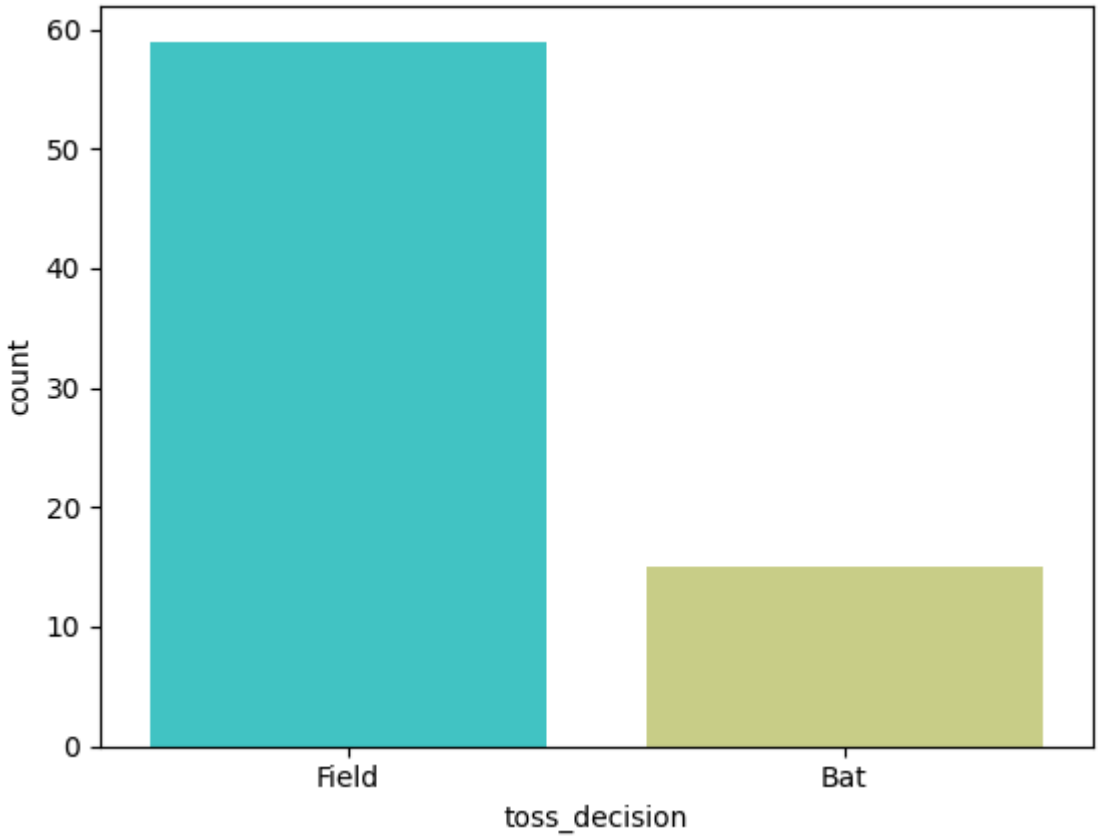
Out[75]: Text(0.5, 1.0, 'Most Match Winner')



In [76]:
```
# Toss Decision Trends
Toss_descision = ipl_data["toss_decision"].value_counts()
field_percentage = (Toss_descision.Field * 100) / (Toss_descision.Field + Toss_d
bat_percentage = (Toss_descision.Bat * 100) / (Toss_descision.Field + Toss_desci
print(f"field percentage : {field_percentage}")
print(f"bat percentage : {bat_percentage}")

sb.countplot(x=ipl_data["toss_decision"],palette="rainbow")
```

```
field percentage : 79.72972972972973
bat percentage : 20.27027027027027
C:\Users\usman\AppData\Local\Temp\ipykernel_2664\350882548.py:8: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sb.countplot(x=ipl_data["toss_decision"],palette="rainbow")
```

Out[76]: <Axes: xlabel='toss_decision', ylabel='count'>



In [118…

```
# Toss Winner vs Match Winner



match_and_toss_winner = ipl_data[(ipl_data["match_winner"]) == ipl_data["toss_wi
match_loss = ipl_data[(ipl_data["match_winner"]) != ipl_data["toss_winner"]]["ma
# match_winner.value_counts()
print(f"Teams who won toss and  matches both are : {match_winner}")
print(f"Teams who loss toss and won match: {match_winner}")
percentage_winner = (match_and_toss_winner * 100)/ipl_data.shape[0]
percentage_opp = (match_loss * 100)/ipl_data.shape[0]

print(f"percentage of match and toss winner : {percentage_winner}")
print(f"percentage of match and toss winner : {percentage_opp}")
```

```
Teams who won toss and  matches both are : 36
Teams who loss toss and won match: 36
percentage of match and toss winner : 48.648648648648646
percentage of match and toss winner : 51.351351351351354
```

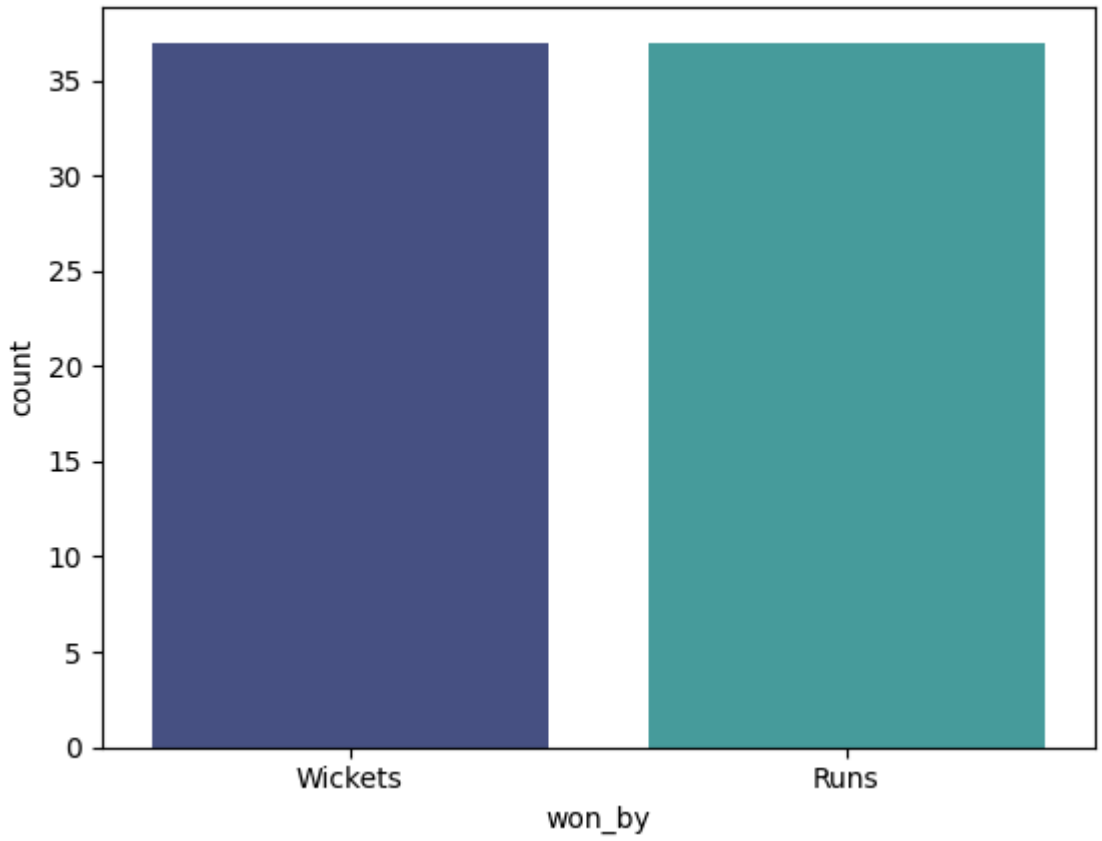
In [141…

```
### how to teams win? (Run or wickets)

won_by = ipl_data["won_by"].value_counts()
print(won_by)
sb.countplot(x=ipl_data["won_by"],palette="mako")
```

```
won_by
Wickets    37
Runs       37
Name: count, dtype: int64
```

```
C:\Users\usman\AppData\Local\Temp\ipykernel_2664\1113998296.py:5: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sb.countplot(x=ipl_data["won_by"],palette="mako")
```

Out[141… <Axes: xlabel='won_by', ylabel='count'>



Key Player Performance

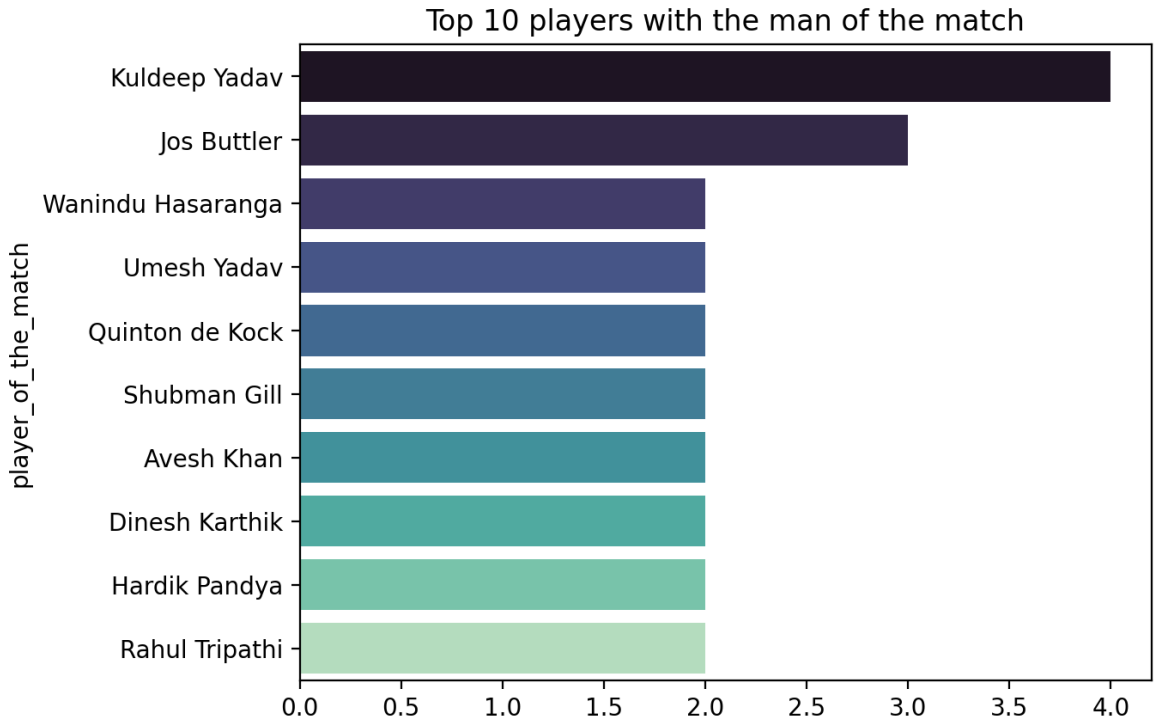
In [142…

```
## Most player of the match
count = most_player_of_match_name = ipl_data["player_of_the_match"].value_counts

most_player_of_match_name = ipl_data["player_of_the_match"].value_counts().idxma
most_player_of_match_count = ipl_data["player_of_the_match"].value_counts().max(
print(f"Most player of the match award goes to {most_player_of_match_name} with

print("\nPlot Representation")
sb.barplot(x=count.values,y=count.index ,palette="mako")
plt.gcf().set_dpi(200)
plt.title("Top 10 players with the man of the match")
```

```
Most player of the match award goes to Kuldeep Yadav with 4 wins

Plot Representation
```

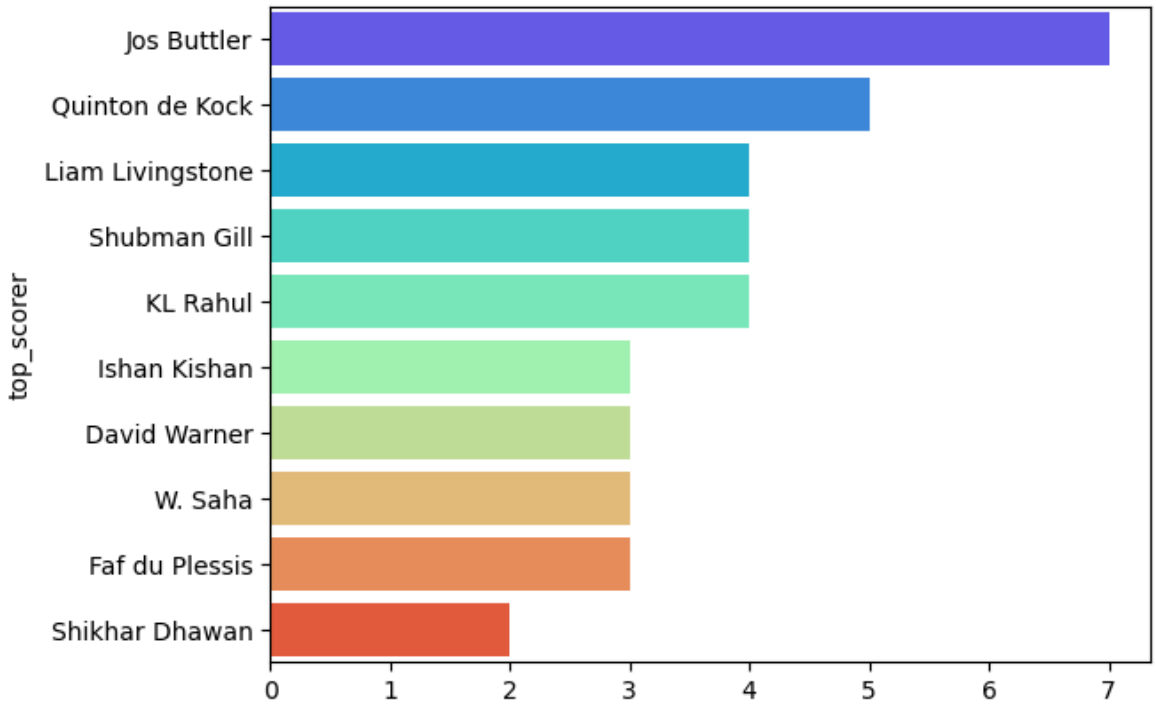
```
C:\Users\usman\AppData\Local\Temp\ipykernel_2664\2580911350.py:9: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effe
ct.

  sb.barplot(x=count.values,y=count.index ,palette="mako")
```

Out[142… Text(0.5, 1.0, 'Top 10 players with the man of the match')



In [ ]:
```
# 2 Top Scorers
top_scorers = ipl_data["top_scorer"].value_counts().head(10)
print(top_scorers.head(2))

sb.barplot(x=top_scorers.values,y=top_scorers.index,palette="rainbow")
```

```
top_scorer
Jos Buttler        7
Quinton de Kock    5
Name: count, dtype: int64

Top 2 scorer with respect to score and high performance:
top_scorer
Jos Buttler        651
Quinton de Kock    377
Name: highscore, dtype: int64
C:\Users\usman\AppData\Local\Temp\ipykernel_2664\2982965586.py:5: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effe
ct.

  sb.barplot(x=top_scorers.values,y=top_scorers.index,palette="rainbow")
```

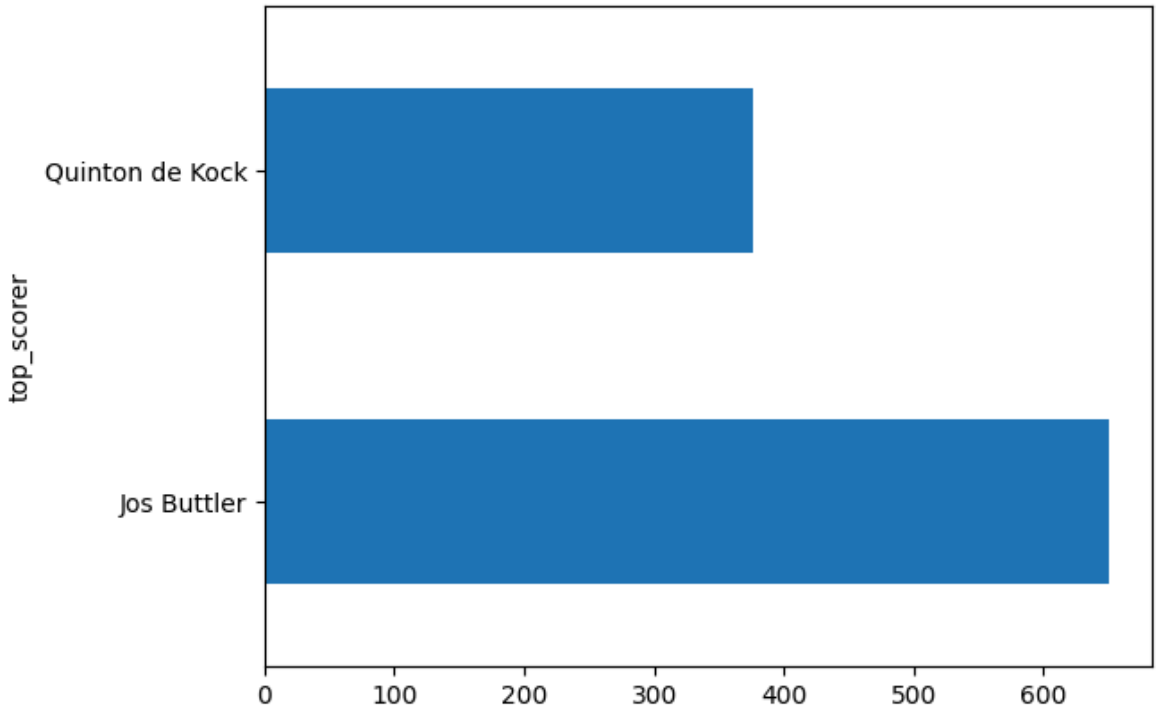


In [163…

```
###  TOP Scorer with high score
print(f"\nTop 2 scorer with respect to score and high performance: ")
high = ipl_data.groupby("top_scorer")["highscore"].sum().sort_values(ascending=F
print(high)

high.plot(kind="barh")
```

```
Top 2 scorer with respect to score and high performance:
top_scorer
Jos Buttler        651
Quinton de Kock    377
Name: highscore, dtype: int64
```

Out[163… <Axes: ylabel='top_scorer'>

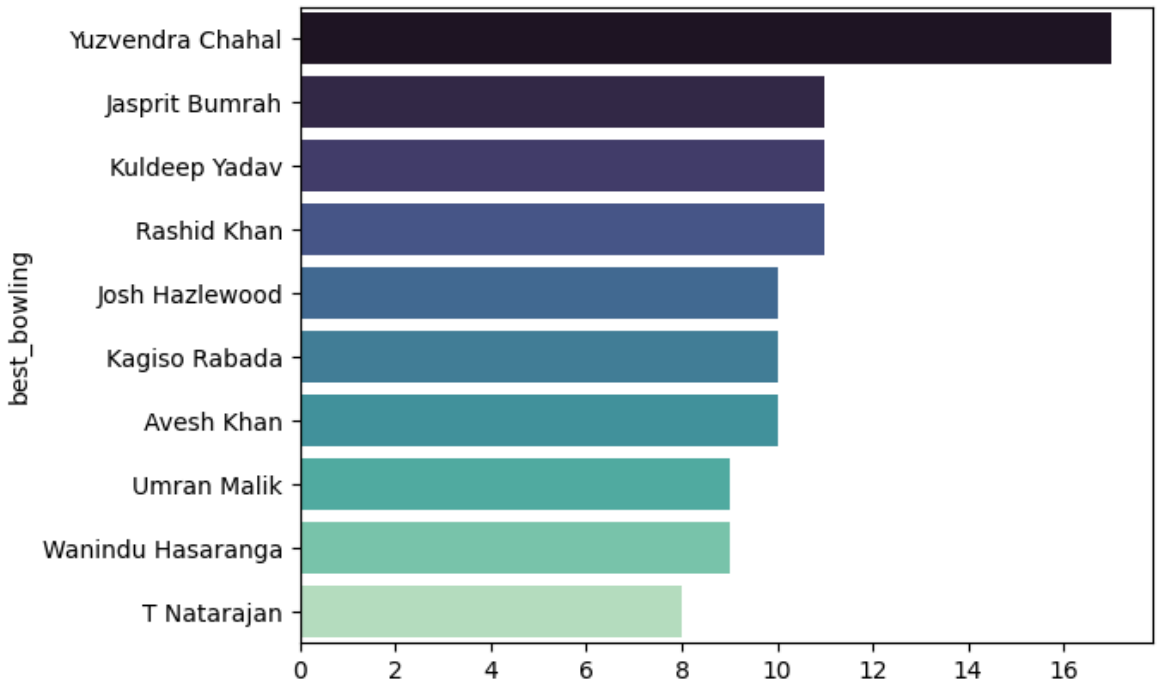


In [179…

```
# Top 10 bowling figures by best bowling and their wickets

# so first we extract wickets from bowling figures

ipl_data["highest_wickets"] = ipl_data["best_bowling_figure"].apply(lambda x: x.
ipl_data["highest_wickets"]=ipl_data["highest_wickets"].astype(int)

high_bowling_figures = ipl_data.groupby("best_bowling")["highest_wickets"].sum()
print(high_bowling_figures)

print("\n plot representation: ")
sb.barplot(x=high_bowling_figures.values ,y=high_bowling_figures.index,palette="
```

```
best_bowling
Yuzvendra Chahal     17
Jasprit Bumrah       11
Kuldeep Yadav        11
Rashid Khan          11
Josh Hazlewood       10
Kagiso Rabada        10
Avesh Khan           10
Umran Malik           9
Wanindu Hasaranga     9
T Natarajan           8
Name: highest_wickets, dtype: int64

 plot representation:
C:\Users\usman\AppData\Local\Temp\ipykernel_2664\4122862788.py:12: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effe
ct.

  sb.barplot(x=high_bowling_figures.values ,y=high_bowling_figures.index,palette
="mako")
```

Out[179… <Axes: ylabel='best_bowling'>

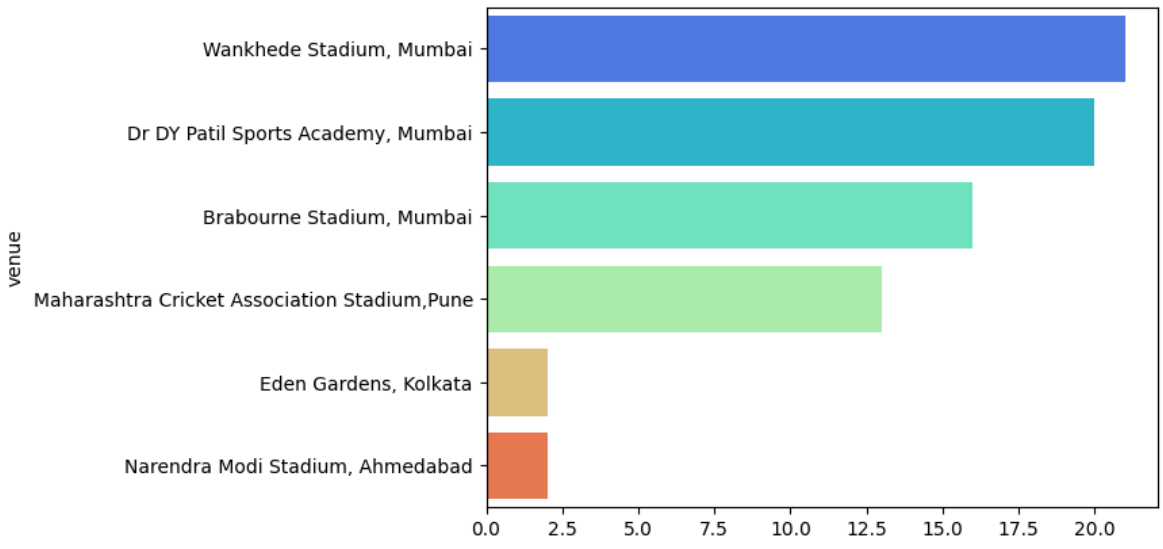


In [181…
```
# Venue Analysis

venue_count = ipl_data["venue"].value_counts()
print(venue_count)

print("\n Plot representation : ")
sb.barplot(x=venue_count.values, y= venue_count.index ,palette="rainbow" )
```

```
venue
Wankhede Stadium, Mumbai                        21
Dr DY Patil Sports Academy, Mumbai              20
Brabourne Stadium, Mumbai                       16
Maharashtra Cricket Association Stadium,Pune    13
Eden Gardens, Kolkata                            2
Narendra Modi Stadium, Ahmedabad                 2
Name: count, dtype: int64

 Plot representation :
```

```
C:\Users\usman\AppData\Local\Temp\ipykernel_2664\3740986067.py:7: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effe
ct.

  sb.barplot(x=venue_count.values, y= venue_count.index ,palette="rainbow" )
```

Out[181…
```
<Axes: ylabel='venue'>
```



Custom Qustions and Insights

In [ ]:
```
# Q1 who won the highest margin by runs

ipl_data.head(10)

won_by_runs = ipl_data[ipl_data["won_by"]=="Runs"].sort_values(by="margin",ascen

won_by_runs[["margin" , "match_winner"]]
```

Out[ ]:

	margin	match_winner
54	91	Chennai

In [ ]:
```
# Q2  which player had the highest individual score?

highest_score = ipl_data[ipl_data["highscore"] == ipl_data["highscore"].max()]
highest_score[["top_scorer","highscore"]]
```

Out[ ]:

	top_scorer	highscore
65	Quinton de Kock	140

In [222…
```
# Q3 which bowler has the best bowling figures
wickets  =  ipl_data["best_bowling_figure"].apply(lambda x: x.split("--")[0])
runs =  ipl_data["best_bowling_figure"].apply(lambda x: x.split("--")[1])

best_bowling_figure = ipl_data[ (wickets==wickets.max()) & (runs==runs.min())]
best_bowling_figure[["best_bowling","best_bowling_figure"]]
```

Out[222…

	best_bowling	best_bowling_figure
55	Jasprit Bumrah	5--10