```
import pandas as pd
import numpy as np
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
import pandas as pd
file_path = '/content/drive/MyDrive/ColabFiles/matches.csv'
df = pd.read_csv(file_path)
df.head()
```

	id	Season	city	date	team1	team2	toss_winner	toss_decisi
0	1	IPL-2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	fi
1	2	IPL-2017	Pune	06-04-2017	Mumbai Indians	Rising Pune Supergiant	Rising Pune Supergiant	fi
2	3	IPL-2017	Rajkot	07-04-2017	Gujarat Lions	Kolkata Knight Riders	Kolkata Knight Riders	fi
3	4	IPL-2017	Indore	08-04-2017	Rising Pune Supergiant	Kings XI Punjab	Kings XI Punjab	fi
4	5	IPL-2017	Bangalore	08-04-2017	Royal Challengers Bangalore	Delhi Daredevils	Royal Challengers Bangalore	b

```
### Problem 1: Find the total number of matches played in the IPL dataset.

total_matches = df.shape[0]
print("Total number of matches played:", total_matches)
```

```
Total number of matches played: 756
```

```
# ### Problem 2: Find the team with the highest number of wins in IPL history.

top_team = df['winner'].value_counts().idxmax()
print("Team with the highest number of wins:", top_team)
```

```
Team with the highest number of wins: Mumbai Indians
```

```
# ### Problem 3: Find the player with the most "Player of the Match" awards.

top_player = df['player_of_match'].value_counts().idxmax()
print("Player with the most 'Player of the Match' awards:", top_player)
```

```
Player with the most 'Player of the Match' awards: CH Gayle
```

```
# ### Problem 4: Find the season with the most matches played.

most_matches_season = df['Season'].value_counts().idxmax()
print("Season with the most matches played:", most_matches_season)
```

```
Season with the most matches played: IPL-2013
```

```
# ### Problem 5: Find the venue that hosted the most IPL matches.

top_venue = df['venue'].value_counts().idxmax()
print("Venue with the most matches hosted:", top_venue)
```

```
Venue with the most matches hosted: Eden Gardens
```

```
# ### Problem 6: Find the top 5 teams with the highest total wins.

top_5_teams = df['winner'].value_counts().head(5)
print("Top 5 teams with highest total wins:")
print(top_5_teams)
```

```
Top 5 teams with highest total wins:
winner
Mumbai Indians                109
Chennai Super Kings           100
Kolkata Knight Riders          92
Royal Challengers Bangalore    84
Kings XI Punjab                82
Name: count, dtype: int64
```

```
# ### Problem 7: Find the team that won the toss the most number of times.

top_toss_winner = df['toss_winner'].value_counts().idxmax()
print("Team that won the toss most often:", top_toss_winner)
```

```
Team that won the toss most often: Mumbai Indians
```

```
# ### Problem 8: Find how often the team that won the toss also won the match.

same_winner = df[df['toss_winner'] == df['winner']].shape[0]
print("Number of times toss winner also won the match:", same_winner)
```

```
Number of times toss winner also won the match: 393
```

```
# ### Problem 9: Find the match with the largest victory margin by runs.

max_run_win = df.loc[df['win_by_runs'].idxmax()]
print("Match with largest victory margin by runs:")
print(max_run_win)
```

```
Match with largest victory margin by runs:
id                              44
Season                     IPL-2017
city                          Delhi
date                     06-05-2017
team1                Mumbai Indians
team2              Delhi Daredevils
toss_winner        Delhi Daredevils
toss_decision                 field
result                       normal
dl_applied                        0
winner               Mumbai Indians
win_by_runs                     146
win_by_wickets                    0
player_of_match         LMP Simmons
venue              Feroz Shah Kotla
umpire1                 Nitin Menon
umpire2                   CK Nandan
umpire3                         NaN
Name: 43, dtype: object
```

```
# ### Problem 10: Find the match with the largest victory margin by wickets.

max_wicket_win = df.loc[df['win_by_wickets'].idxmax()]
print("Match with largest victory margin by wickets:")
print(max_wicket_win)
```

```
Match with largest victory margin by wickets:
id                                                    3
Season                                          IPL-2017
city                                              Rajkot
date                                          07-04-2017
team1                                       Gujarat Lions
team2                             Kolkata Knight Riders
toss_winner                       Kolkata Knight Riders
toss_decision                                     field
result                                           normal
dl_applied                                            0
winner                            Kolkata Knight Riders
win_by_runs                                           0
win_by_wickets                                       10
player_of_match                                CA Lynn
venue              Saurashtra Cricket Association Stadium
umpire1                                      Nitin Menon
umpire2                                        CK Nandan
umpire3                                              NaN
Name: 2, dtype: object
```

```
# ### Problem 11: Find the player who received the most "Player of the Match" a

finals = df[df['result'] == 'normal']
finals_awards = finals['player_of_match'].value_counts().idxmax()
print("Player with most 'Player of the Match' awards in finals:", finals_awards
```

```
Player with most 'Player of the Match' awards in finals: CH Gayle
```

```
# ### Problem 12: Find the total number of matches decided by super over.

super_over_matches = df[df['result'] == 'tie'].shape[0]
print("Total number of matches decided by super over:", super_over_matches)
```

```
Total number of matches decided by super over: 9
```

```
# ### Problem 13: Find the team with the most wins while chasing (batting secor

bat_second_wins = df[df['win_by_wickets'] > 0]['winner'].value_counts().idxmax(
print("Team with most wins while chasing:", bat_second_wins)
```

```
Team with most wins while chasing: Kolkata Knight Riders
```

```
# ### Problem 14: Find the team with the most wins while defending (batting fir

bat_first_wins = df[df['win_by_runs'] > 0]['winner'].value_counts().idxmax()
print("Team with most wins while defending:", bat_first_wins)
```

```
Team with most wins while defending: Mumbai Indians
```

```
# ### Problem 15: Find the number of matches where no result was declared.

no_result_matches = df[df['result'] == 'no result'].shape[0]
print("Number of matches with no result:", no_result_matches)
```

```
Number of matches with no result: 4
```

```
# ### Problem 16: Find the number of matches played in each city.

matches_per_city = df['city'].value_counts()
print("Number of matches played in each city:")
print(matches_per_city)
```

```
Number of matches played in each city:
city
Mumbai            101
Kolkata            77
Delhi              74
Bangalore          66
Hyderabad          64
Chennai            57
Jaipur             47
Chandigarh         46
Pune               38
Durban             15
Bengaluru          14
Visakhapatnam      13
Centurion          12
Ahmedabad          12
Mohali             10
Rajkot             10
Dharamsala          9
Indore              9
Johannesburg        8
Port Elizabeth      7
Abu Dhabi           7
Cape Town           7
Ranchi              7
Cuttack             7
Sharjah             6
Raipur              6
Kochi               5
Kanpur              4
Kimberley           3
Nagpur              3
East London         3
Bloemfontein        2
Name: count, dtype: int64
```

```
# ### Problem 17: Find the most successful city for Mumbai Indians (where they

mi_wins = df[df['winner'] == 'Mumbai Indians']
best_city_mi = mi_wins['city'].value_counts().idxmax()
print("City where Mumbai Indians won the most matches:", best_city_mi)
```

```
City where Mumbai Indians won the most matches: Mumbai
```

```
# ### Problem 18: Find the match where the margin of victory was the smallest b

smallest_run_win = df[df['win_by_runs'] > 0].nsmallest(1, 'win_by_runs')
print("Match with smallest victory margin by runs:")
print(smallest_run_win)
```

```
Match with smallest victory margin by runs:
      id    Season       city        date           team1  \
58  59  IPL-2017  Hyderabad  21-05-2017  Mumbai Indians

                       team2     toss_winner toss_decision  result  dl_applie
58  Rising Pune Supergiant  Mumbai Indians           bat  normal

             winner  win_by_runs  win_by_wickets player_of_match  \
58  Mumbai Indians            1               0       KH Pandya

                                        venue   umpire1 umpire2 umpire3
58  Rajiv Gandhi International Stadium, Uppal  NJ Llong  S Ravi     NaN
```

```
# ### Problem 19: Find the team that won the most tosses in finals matches.

finals_matches = df[df['result'] == 'normal']
top_toss_final = finals_matches['toss_winner'].value_counts().idxmax()
print("Team with most toss wins in finals:", top_toss_final)
```

```
Team with most toss wins in finals: Mumbai Indians
```

```
# ### Problem 20: Find the city which hosted the most number of finals.

finals_cities = df[df['result'] == 'normal']['city'].value_counts().idxmax()
print("City that hosted the most finals:", finals_cities)
```

```
City that hosted the most finals: Mumbai
```

Start coding or generate with AI.