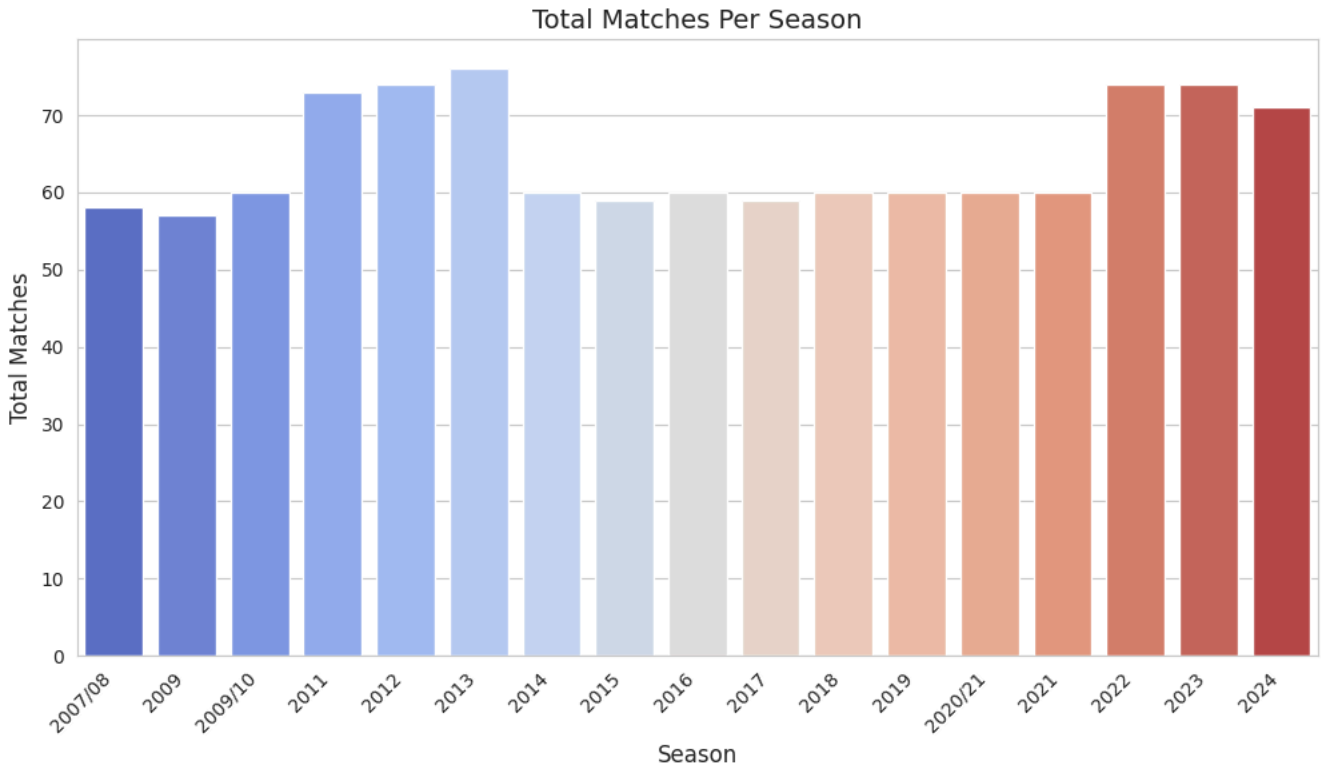
UPDATED PROJECT

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

df = pd.read_csv("/content/drive/MyDrive/matches.csv")

print("Available Columns:", df.columns)

plt.figure(figsize=(12, 6))


season_matches = df['season'].value_counts().sort_index()

ax = sns.barplot(x=season_matches.index, y=season_matches.values, palette="coolwarm")

plt.title("Total Matches Per Season", fontsize=14)
plt.xlabel("Season", fontsize=12)
plt.ylabel("Total Matches", fontsize=12)
plt.xticks(rotation=45, ha='right')

plt.show()
```

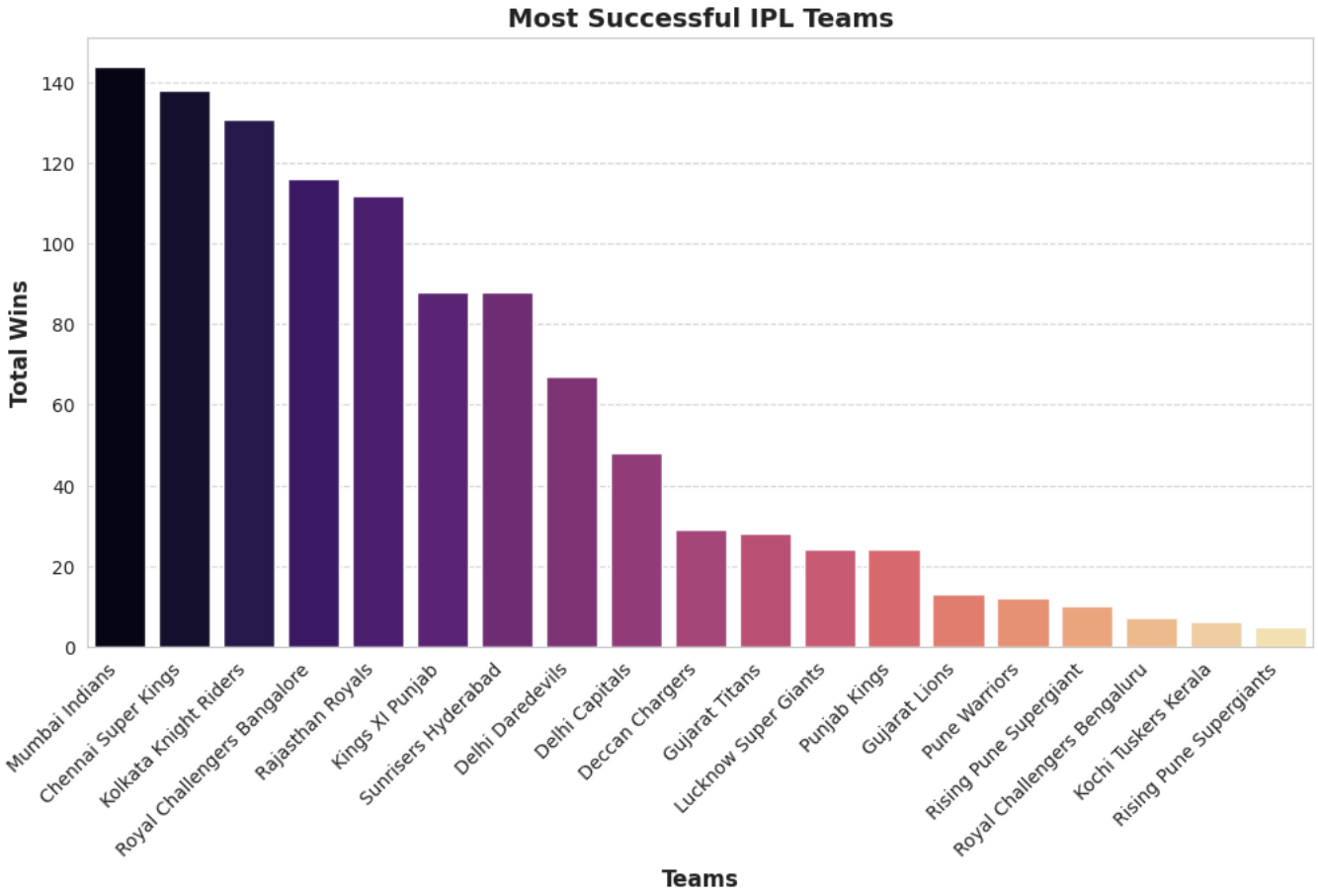
```
Available Columns: Index(['id', 'season', 'city', 'date', 'match_type', 'player_of_match',
       'venue', 'team1', 'team2', 'toss_winner', 'toss_decision', 'winner',
       'result', 'result_margin', 'target_runs', 'target_overs', 'super_over',
       'method', 'umpire1', 'umpire2'],
      dtype='object')
<ipython-input-33-80d480058625>:14: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend

  ax = sns.barplot(x=season_matches.index, y=season_matches.values, palette="coolwarm")
```




```
import pandas as pd
import matplotlib.pyplot as plt
```

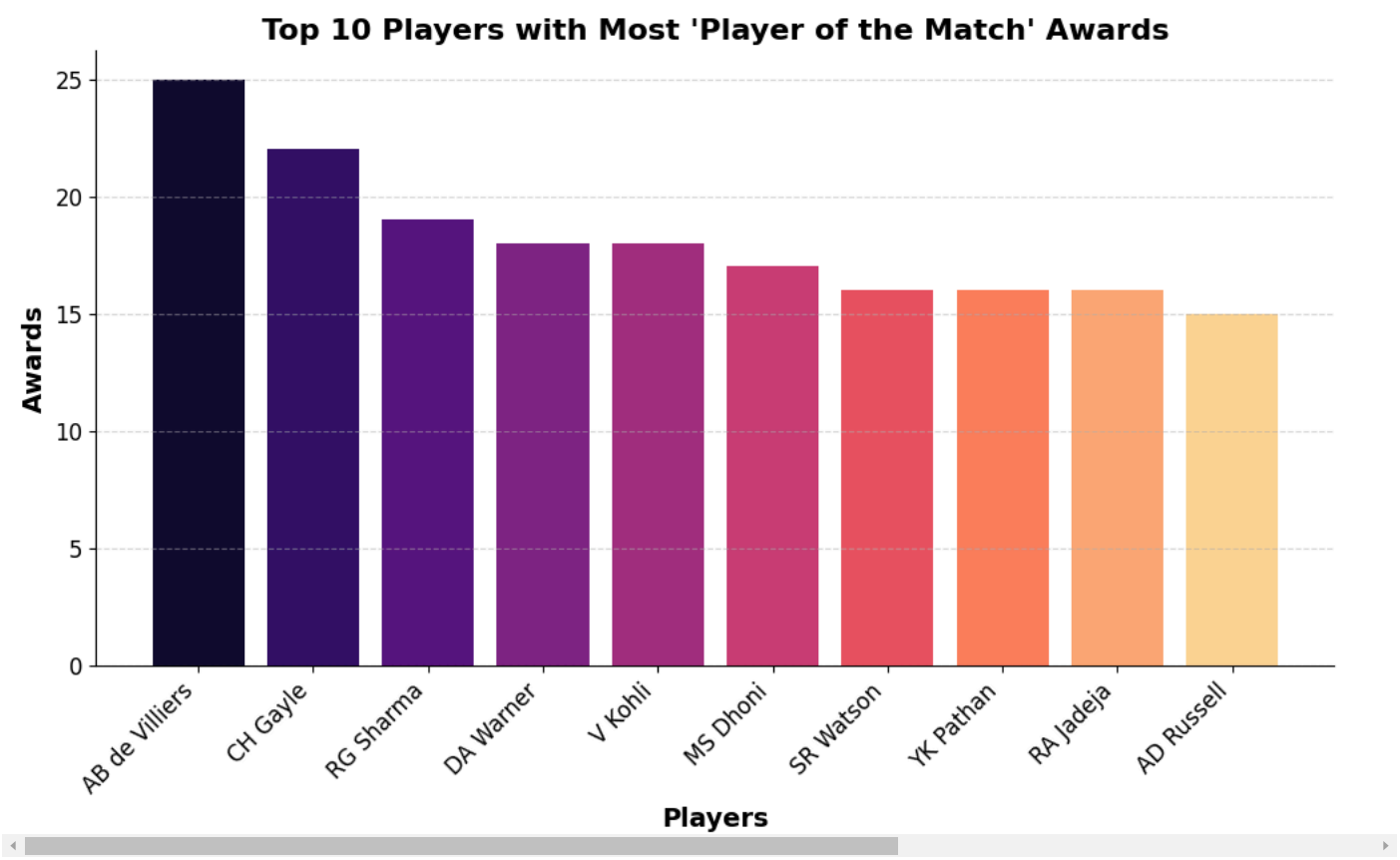
```
import seaborn as sns

# Load your dataset (replace with your actual file path)
df = pd.read_csv("/content/drive/MyDrive/matches.csv")

# Assuming your dataset has a 'winner' column representing the winning team
# Calculate team wins
team_wins = df['winner'].value_counts()

# Extract teams and wins for the barplot
teams = team_wins.index
wins = team_wins.values

# Now you can create the barplot
plt.figure(figsize=(12, 6))
plt.xticks(rotation=45, ha="right", fontsize=10)
plt.title("Most Successful IPL Teams", fontsize=14, fontweight="bold")
plt.xlabel("Teams", fontsize=12, fontweight="bold")
plt.ylabel("Total Wins", fontsize=12, fontweight="bold")
plt.grid(axis='y', linestyle="--", alpha=0.7)
sns.barplot(x=teams, y=wins, palette="magma")  # Use the defined variables
plt.show()
```

```
<ipython-input-34-7b4da1497ab9>:23: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend

  sns.barplot(x=teams, y=wins, palette="magma")  # Use the defined variables
```



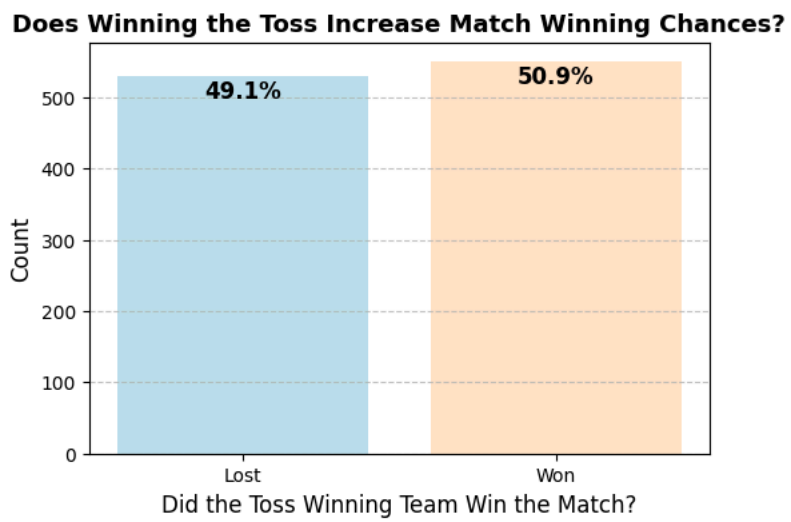

```
import matplotlib.pyplot as plt
import seaborn as sns

players = ["AB de Villiers", "CH Gayle", "RG Sharma", "DA Warner", "V Kohli",
           "MS Dhoni", "SR Watson", "YK Pathan", "RA Jadeja", "AD Russell"]
awards = [25, 22, 19, 18, 18, 17, 16, 16, 16, 15]

plt.figure(figsize=(12, 6))
colors = sns.color_palette("magma", len(players))  # Premium color theme
```

```
plt.bar(players, awards, color=colors)

plt.title("Top 10 Players with Most 'Player of the Match' Awards", fontsize=16, fontweight='bold')
plt.xlabel("Players", fontsize=14, fontweight="bold")
plt.ylabel("Awards", fontsize=14, fontweight="bold")

plt.xticks(rotation=45, ha='right', fontsize=12)  # Rotate for better readability
plt.yticks(fontsize=12)
plt.grid(axis='y', linestyle='--', alpha=0.5)

# Remove unnecessary borders
plt.gca().spines['top'].set_visible(False)
plt.gca().spines['right'].set_visible(False)

plt.show()
```

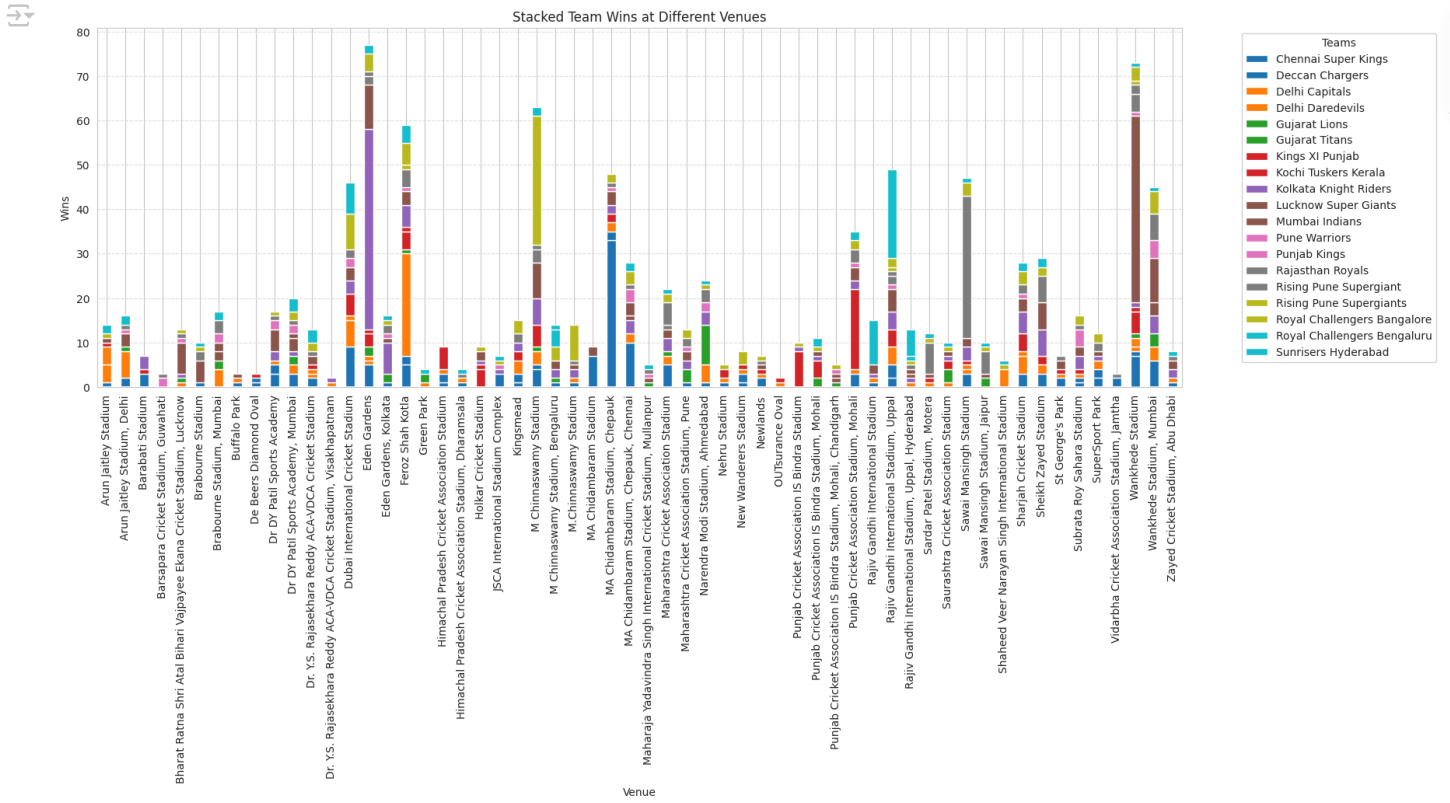



```
import matplotlib.pyplot as plt
import seaborn as sns

# Sample Data
toss_results = ["Lost", "Won"]
match_counts = [530, 550]  # Example counts

# Calculate percentages
total_matches = sum(match_counts)
percentages = [(count / total_matches) * 100 for count in match_counts]

# Plot
plt.figure(figsize=(6, 4))
colors = ["lightblue", "peachpuff"]
bars = plt.bar(toss_results, match_counts, color=colors, alpha=0.8)

# Add Percentage Labels on Bars
for bar, percent in zip(bars, percentages):
    plt.text(bar.get_x() + bar.get_width()/2, bar.get_height() - 30, f"{percent:.1f}%",
             ha='center', fontsize=12, fontweight='bold', color="black")

# Improved Labels & Title
plt.xlabel("Did the Toss Winning Team Win the Match?", fontsize=12)
```

```
plt.ylabel("Count", fontsize=12)
plt.title("Does Winning the Toss Increase Match Winning Chances?", fontsize=13, fontweight='bold')

# Grid Lines
plt.grid(axis='y', linestyle="--", alpha=0.7)

# Show Plot
plt.show()
```



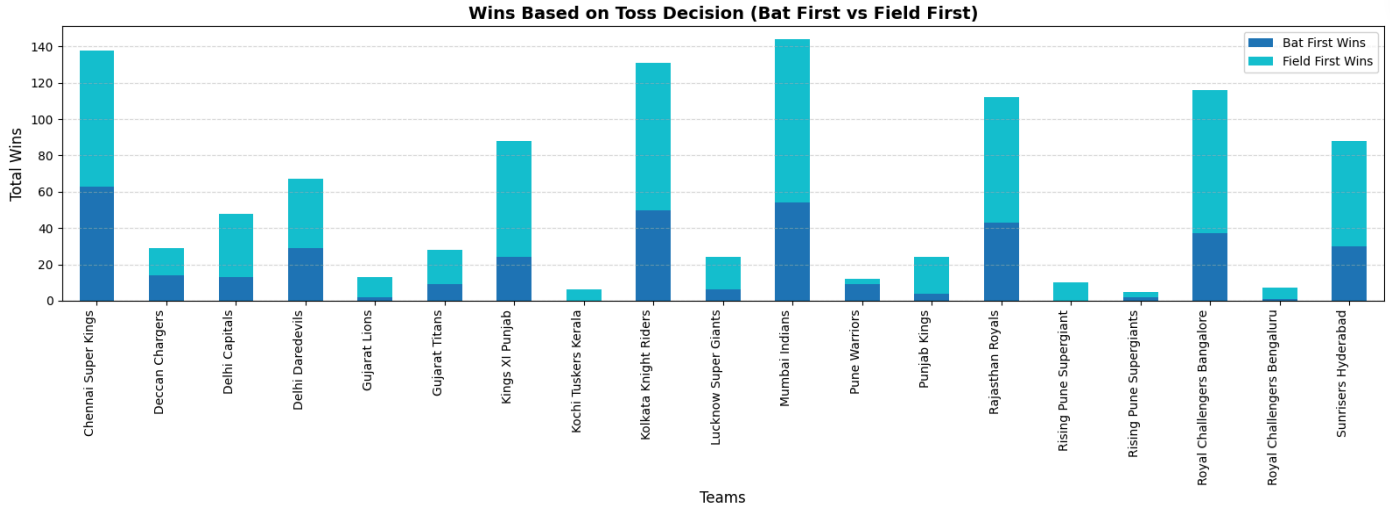

```
import pandas as pd
import matplotlib.pyplot as plt

# Grouping data by 'venue' and 'winner', and counting wins
df_grouped = df.groupby(["venue", "winner"]).size().reset_index(name="Wins")

# Pivot table banana (venue vs team-wise wins)
df_pivot = df_grouped.pivot(index="venue", columns="winner", values="Wins").fillna(0)

# Stacked bar chart plot karna
df_pivot.plot(kind="bar", stacked=True, figsize=(18, 6), colormap="tab10")
plt.ylabel("Wins")
plt.xlabel("Venue")
plt.title("Stacked Team Wins at Different Venues")
plt.xticks(rotation=90)
plt.legend(title="Teams", bbox_to_anchor=(1.05, 1), loc="upper left")
plt.grid(axis="y", linestyle="--", alpha=0.5)
plt.show()
```




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

# Sample Data (Replace with actual dataset)
data = {
    "Winner": [
        "Chennai Super Kings", "Deccan Chargers", "Delhi Capitals",
        "Delhi Daredevils", "Gujarat Lions", "Gujarat Titans",
        "Kings XI Punjab", "Kochi Tuskers Kerala", "Kolkata Knight Riders",
        "Lucknow Super Giants", "Mumbai Indians", "Pune Warriors",
        "Punjab Kings", "Rajasthan Royals", "Rising Pune Supergiant",
        "Rising Pune Supergiants", "Royal Challengers Bangalore",
        "Royal Challengers Bengaluru", "Sunrisers Hyderabad"
    ],
    "Bat First Wins": [63, 14, 13, 29, 2, 9, 24, 0, 50, 6, 54, 9, 4, 43, 0, 2, 37, 1, 30],
    "Field First Wins": [75, 15, 35, 38, 11, 19, 64, 6, 81, 18, 90, 3, 20, 69, 10, 3, 79, 6, 58]
}

df = pd.DataFrame(data)

# Plot setup
plt.figure(figsize=(16, 6))
df.set_index("Winner").plot(kind="bar", stacked=True, figsize=(16, 6), colormap="tab10")

# Formatting
plt.xlabel("Teams", fontsize=12)
plt.ylabel("Total Wins", fontsize=12)
plt.title("Wins Based on Toss Decision (Bat First vs Field First)", fontsize=14, fontweight="bold")
plt.xticks(rotation=90, ha="right")
plt.legend(["Bat First Wins", "Field First Wins"])
plt.grid(axis="y", linestyle="--", alpha=0.5)

# Adjust layout
plt.tight_layout()
```

```
plt.show()
```

```
<Figure size 1600x600 with 0 Axes>
```




```
print(df.columns)  # Yeh check karega ki kaunse columns hai
```

```
Index(['Winner', 'Bat First Wins', 'Field First Wins'], dtype='object')
```

```
df.columns = df.columns.str.strip().str.lower()  # Extra spaces hatao & lowercase me convert karo
print(df.columns)  # Dobara check karo
```

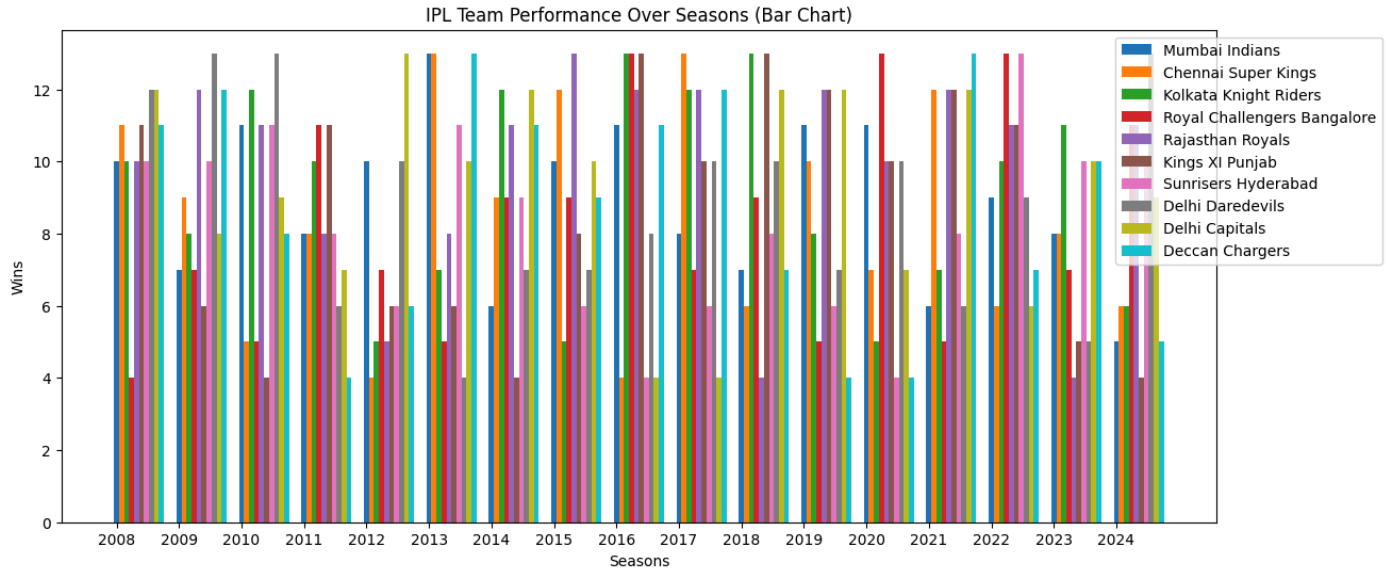
```
Index(['winner', 'bat first wins', 'field first wins'], dtype='object')
```

```
print(df.head())  # Pehle 5 rows dekho
```

```
                winner  bat first wins  field first wins
0  Chennai Super Kings              63               75
1      Deccan Chargers              14               15
2       Delhi Capitals              13               35
3     Delhi Daredevils              29               38
4        Gujarat Lions               2               11
```

```
import matplotlib.pyplot as plt
import numpy as np

# IPL Teams
teams = ["Mumbai Indians", "Chennai Super Kings", "Kolkata Knight Riders",
         "Royal Challengers Bangalore", "Rajasthan Royals", "Kings XI Punjab",
         "Sunrisers Hyderabad", "Delhi Daredevils", "Delhi Capitals", "Deccan Chargers"]

# IPL Seasons
seasons = np.arange(2008, 2025)

# Random wins data (Replace this with actual data)
np.random.seed(42)
wins = np.random.randint(4, 14, size=(10, len(seasons)))

# Plot Bar Chart
fig, ax = plt.subplots(figsize=(14, 6))
width = 0.08  # Width of each bar

for i, team in enumerate(teams):
    ax.bar(seasons + i * width, wins[i], width=width, label=team)
```

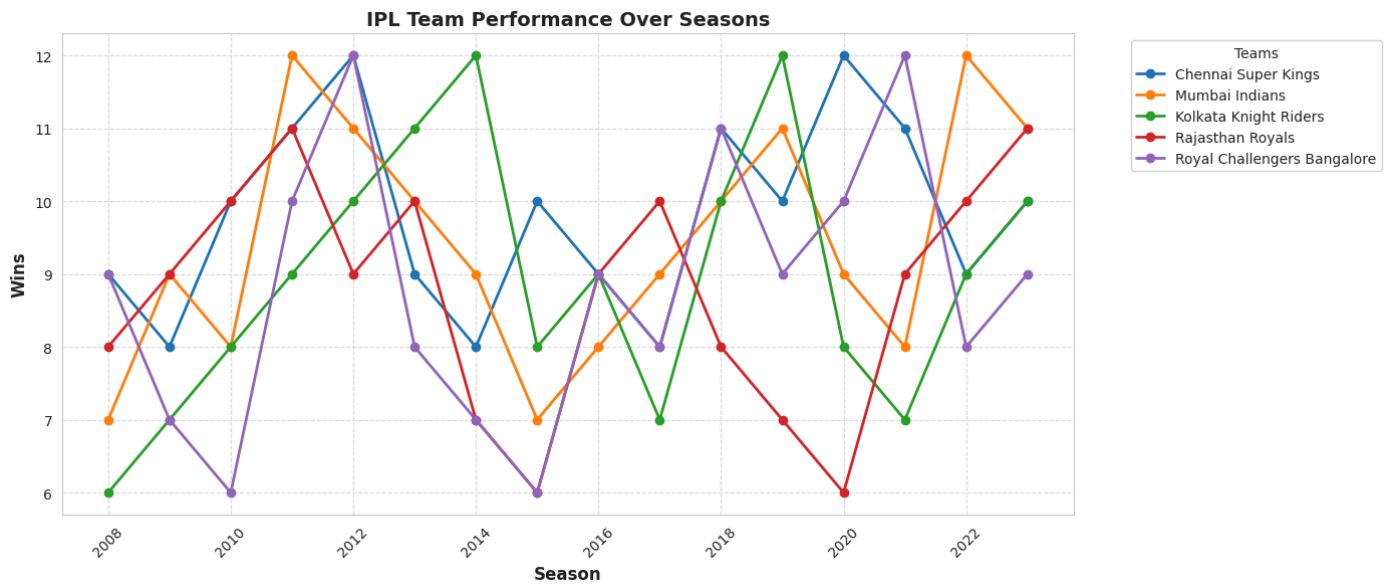
```
# Graph Labels and Title
ax.set_xlabel("Seasons")
ax.set_ylabel("Wins")
ax.set_title("IPL Team Performance Over Seasons (Bar Chart)")
ax.set_xticks(seasons)
ax.legend(loc="upper right", bbox_to_anchor=(1.15, 1))

# Show the chart
plt.show()
```




```
df.rename(columns={'Team': 'team'}, inplace=True)  # Jo bhi actual naam hai, usko 'team' se replace karo
```

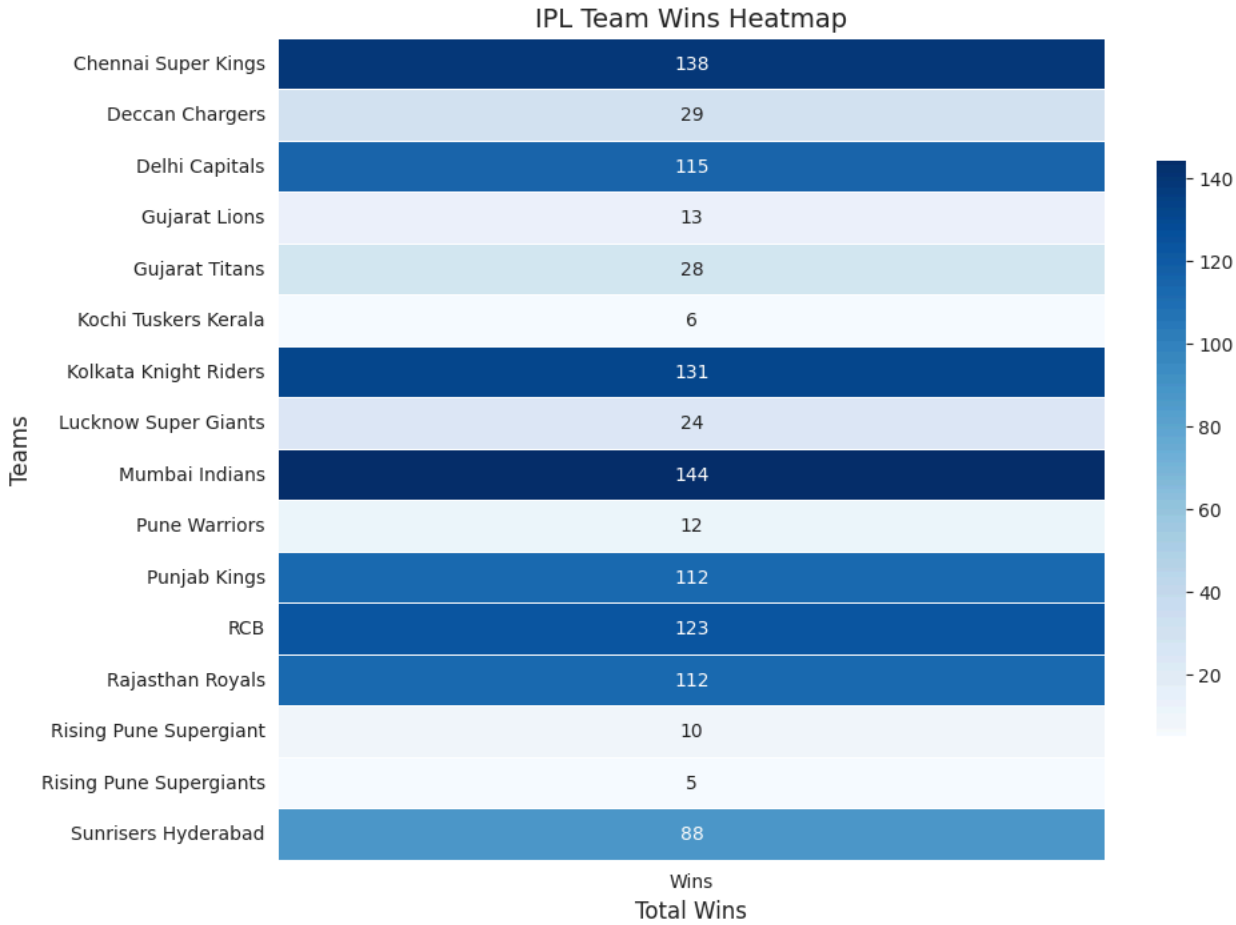
```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# Sample DataFrame (Replace this with actual data)
data = {
    "Season": [2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023],
    "Chennai Super Kings": [9, 8, 10, 11, 12, 9, 8, 10, 9, 8, 11, 10, 12, 11, 9, 10],
    "Mumbai Indians": [7, 9, 8, 12, 11, 10, 9, 7, 8, 9, 10, 11, 9, 8, 12, 11],
    "Kolkata Knight Riders": [6, 7, 8, 9, 10, 11, 12, 8, 9, 7, 10, 12, 8, 7, 9, 10],
    "Rajasthan Royals": [8, 9, 10, 11, 9, 10, 7, 6, 9, 10, 8, 7, 6, 9, 10, 11],
    "Royal Challengers Bangalore": [9, 7, 6, 10, 12, 8, 7, 6, 9, 8, 11, 9, 10, 12, 8, 9]
}

df = pd.DataFrame(data)

# Set Plot Style
plt.figure(figsize=(14, 6))
sns.set_style("whitegrid")

# Plot each team's performance
for team in df.columns[1:]:
    plt.plot(df["Season"], df[team], marker="o", linestyle="-", linewidth=2, label=team)

# Formatting the Plot
plt.xlabel("Season", fontsize=12, fontweight="bold")
plt.ylabel("Wins", fontsize=12, fontweight="bold")
plt.title("IPL Team Performance Over Seasons", fontsize=14, fontweight="bold")
plt.xticks(rotation=45)
```

```
plt.legend(title="Teams", bbox_to_anchor=(1.05, 1), loc="upper left", frameon=True)
plt.grid(True, linestyle="--", alpha=0.6)

# Show Plot
plt.tight_layout()
plt.show()
```



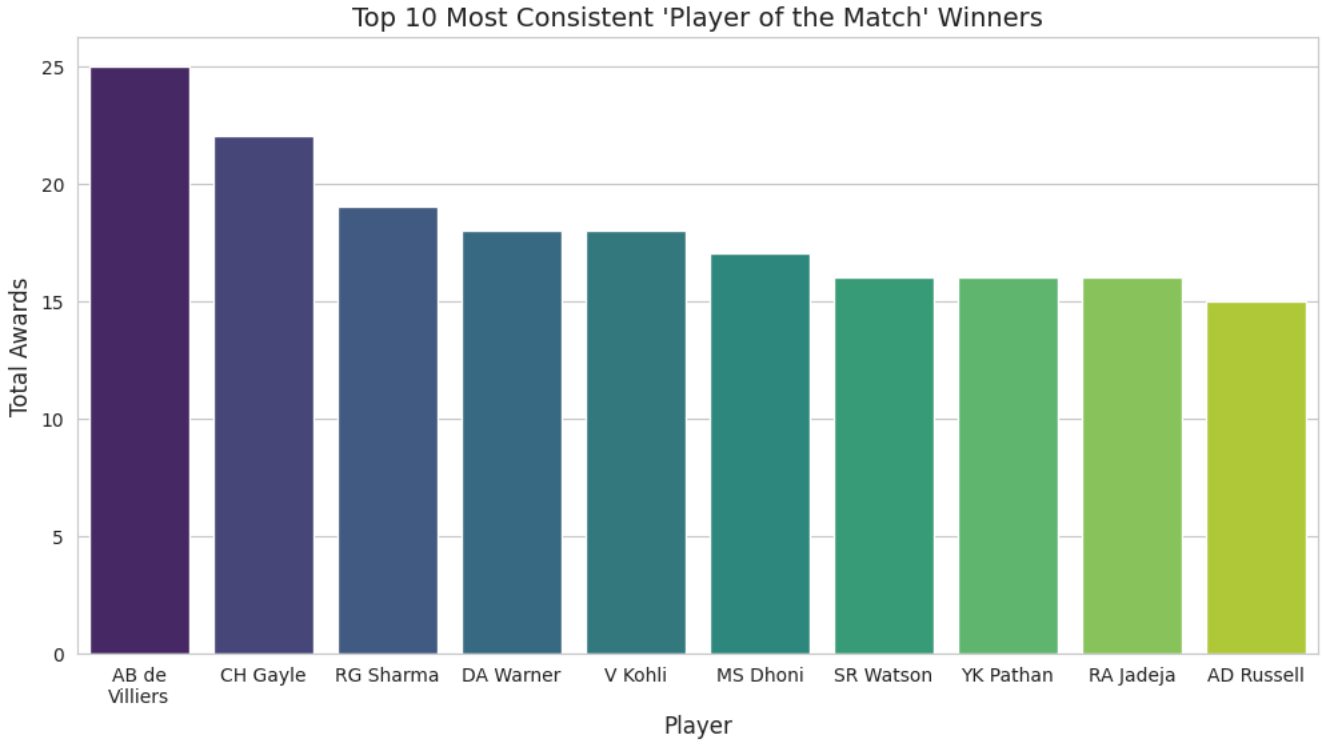

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load dataset (update with correct file path)
df = pd.read_csv("/content/drive/MyDrive/matches.csv")

# Standardizing team names (fixing duplicates)
df["winner"] = df["winner"].replace({
    "Royal Challengers Bangalore": "RCB",
    "Royal Challengers Bengaluru": "RCB",
    "Kings XI Punjab": "Punjab Kings",
    "Delhi Daredevils": "Delhi Capitals"
})

# Count number of wins
team_wins = df["winner"].value_counts().reset_index()
team_wins.columns = ["Team", "Wins"]

# Sort by Wins (Descending)
team_wins = team_wins.sort_values(by="Wins", ascending=False)

# Create pivot table for heatmap
heatmap_data = team_wins.pivot_table(values="Wins", index="Team", aggfunc="sum")

# Set figure size
plt.figure(figsize=(10, 8))

# Plot heatmap with better color and formatting
sns.heatmap(heatmap_data, annot=True, fmt="d", cmap="Blues", linewidths=0.5, cbar_kws={"shrink": 0.7})

# Fix axis labels
plt.title("IPL Team Wins Heatmap", fontsize=14)
plt.ylabel("Teams", fontsize=12)
plt.xlabel("Total Wins", fontsize=12)

# Improve readability
```

```
plt.xticks(rotation=0)
plt.yticks(rotation=0)

# Show plot
plt.show()
```




```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import textwrap

# Load dataset
df = pd.read_csv("/content/drive/MyDrive/matches.csv")

# Count most 'Player of the Match' awards
player_awards = df["player_of_match"].value_counts().head(10)

# Set figure size
plt.figure(figsize=(12, 6))

# Create a barplot
sns.barplot(x=player_awards.index, y=player_awards.values, palette="viridis")

# Improve labels and title
plt.title("Top 10 Most Consistent 'Player of the Match' Winners", fontsize=14)
plt.xlabel("Player", fontsize=12)
plt.ylabel("Total Awards", fontsize=12)

# Apply text wrapping
plt.xticks(rotation=0)
plt.gca().set_xticklabels([textwrap.fill(label, 10) for label in player_awards.index])

# Show plot
plt.show()
```

```
<ipython-input-15-b5c1629fe95c>:16: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend

  sns.barplot(x=player_awards.index, y=player_awards.values, palette="viridis")
<ipython-input-15-b5c1629fe95c>:25: UserWarning: set_ticklabels() should only be used with a fixed number of ticks, i.e. after set_ticks
  plt.gca().set_xticklabels([textwrap.fill(label, 10) for label in player_awards.index])
```




```
import pandas as pd

# Sample DataFrame
venue_wins = pd.DataFrame({
    'Venue': ['Stadium A', 'Stadium B', 'Stadium C', 'Stadium D', 'Stadium E'],
    'Wins': [10, 15, 8, 20, 12]
})

# Print first few rows
print(venue_wins.head())

# Print data types of columns
print(venue_wins.dtypes)
```

```
      Venue  Wins
0  Stadium A    10
1  Stadium B    15
2  Stadium C     8
3  Stadium D    20
4  Stadium E    12
Venue    object
Wins      int64
dtype: object
```

```
import pandas as pd

# Load data from CSV
venue_wins = pd.read_csv('/content/drive/MyDrive/matches.csv')

# Print first few rows
print(venue_wins.head())

# Print data types
```

```
print(venue_wins.dtypes)
```

```
       id   season        city        date match_type player_of_match  \
0  335982  2007/08   Bangalore  2008-04-18     League     BB McCullum
1  335983  2007/08  Chandigarh  2008-04-19     League      MEK Hussey
2  335984  2007/08       Delhi  2008-04-19     League     MF Maharoof
3  335985  2007/08      Mumbai  2008-04-20     League      MV Boucher
4  335986  2007/08     Kolkata  2008-04-20     League       DJ Hussey

                                    venue                       team1  \
0                   M Chinnaswamy Stadium  Royal Challengers Bangalore
1  Punjab Cricket Association Stadium, Mohali              Kings XI Punjab
2                          Feroz Shah Kotla             Delhi Daredevils
3                          Wankhede Stadium               Mumbai Indians
4                              Eden Gardens        Kolkata Knight Riders

                        team2                 toss_winner toss_decision  \
0       Kolkata Knight Riders  Royal Challengers Bangalore         field
1         Chennai Super Kings          Chennai Super Kings           bat
2            Rajasthan Royals             Rajasthan Royals           bat
3  Royal Challengers Bangalore               Mumbai Indians           bat
4              Deccan Chargers              Deccan Chargers           bat

                       winner   result  result_margin  target_runs  \
0       Kolkata Knight Riders     runs          140.0        223.0
1         Chennai Super Kings     runs           33.0        241.0
2            Delhi Daredevils  wickets            9.0        130.0
3  Royal Challengers Bangalore  wickets            5.0        166.0
4       Kolkata Knight Riders  wickets            5.0        111.0

   target_overs super_over method    umpire1         umpire2
0          20.0          N    NaN  Asad Rauf     RE Koertzen
1          20.0          N    NaN  MR Benson      SL Shastri
2          20.0          N    NaN  Aleem Dar  GA Pratapkumar
3          20.0          N    NaN   SJ Davis       DJ Harper
4          20.0          N    NaN  BF Bowden     K Hariharan
id                 int64
season            object
city              object
date              object
match_type        object
player_of_match   object
venue             object
team1             object
team2             object
toss_winner       object
toss_decision     object
winner            object
result            object
result_margin    float64
target_runs      float64
target_overs     float64
super_over        object
method            object
umpire1           object
umpire2           object
dtype: object
```

```
import pandas as pd

# Sample DataFrame
venue_wins = pd.DataFrame({
    'Venue': ['Stadium A', 'Stadium B', 'Stadium C', 'Stadium D', 'Stadium E'],
    'Wins': [10, 15, 8, 20, 12]
})

# Print first few rows
print(venue_wins.head())

# Print data types of columns
print(venue_wins.dtypes)
```

```
      Venue  Wins
0  Stadium A    10
1  Stadium B    15
2  Stadium C     8
3  Stadium D    20
```

```
4  Stadium E    12
Venue    object
Wins      int64
dtype: object
```

```
import pandas as pd

# Load data from CSV file
venue_wins = pd.read_csv('/content/drive/MyDrive/matches.csv')  # Apni file ka sahi naam dalen

# Print first few rows
print(venue_wins.head())

# Print data types
print(venue_wins.dtypes)
```

```
        id   season        city        date match_type player_of_match  \
0  335982  2007/08   Bangalore  2008-04-18     League     BB McCullum
1  335983  2007/08  Chandigarh  2008-04-19     League      MEK Hussey
2  335984  2007/08       Delhi  2008-04-19     League     MF Maharoof
3  335985  2007/08      Mumbai  2008-04-20     League      MV Boucher
4  335986  2007/08     Kolkata  2008-04-20     League       DJ Hussey

                                    venue                       team1  \
0                   M Chinnaswamy Stadium  Royal Challengers Bangalore
1  Punjab Cricket Association Stadium, Mohali              Kings XI Punjab
2                        Feroz Shah Kotla             Delhi Daredevils
3                        Wankhede Stadium               Mumbai Indians
4                            Eden Gardens        Kolkata Knight Riders

                      team2                 toss_winner toss_decision  \
0       Kolkata Knight Riders  Royal Challengers Bangalore         field
1         Chennai Super Kings          Chennai Super Kings           bat
2            Rajasthan Royals             Rajasthan Royals           bat
3  Royal Challengers Bangalore               Mumbai Indians           bat
4             Deccan Chargers              Deccan Chargers           bat

                     winner   result  result_margin  target_runs  \
0       Kolkata Knight Riders     runs          140.0        223.0
1         Chennai Super Kings     runs           33.0        241.0
2            Delhi Daredevils  wickets            9.0        130.0
3  Royal Challengers Bangalore  wickets            5.0        166.0
4       Kolkata Knight Riders  wickets            5.0        111.0

   target_overs super_over method    umpire1         umpire2
0          20.0          N    NaN  Asad Rauf     RE Koertzen
1          20.0          N    NaN  MR Benson      SL Shastri
2          20.0          N    NaN  Aleem Dar  GA Pratapkumar
3          20.0          N    NaN   SJ Davis       DJ Harper
4          20.0          N    NaN  BF Bowden     K Hariharan
id                 int64
season            object
city              object
date              object
match_type        object
player_of_match   object
venue             object
team1             object
team2             object
toss_winner       object
toss_decision     object
winner            object
result            object
result_margin    float64
target_runs      float64
target_overs     float64
super_over        object
method            object
umpire1           object
umpire2           object
dtype: object
```

```
df = pd.read_csv("/content/drive/MyDrive/matches.csv")  # Apni file ka naam yahan likho
print(df.columns)  # Check karo ki "winner", "venue", aur "wins" columns correctly hai ya nahi
```

```
Index(['id', 'season', 'city', 'date', 'match_type', 'player_of_match',
       'venue', 'team1', 'team2', 'toss_winner', 'toss_decision', 'winner',
       'result', 'result_margin', 'target_runs', 'target_overs', 'super_over',
```

```
            'method', 'umpire1', 'umpire2'],
           dtype='object')
```

```
df.rename(columns={"Wins": "wins"}, inplace=True)
```

```
print(df.columns)  # Yeh batayega ki actual column names kya hain
```

```
Index(['id', 'season', 'city', 'date', 'match_type', 'player_of_match',
       'venue', 'team1', 'team2', 'toss_winner', 'toss_decision', 'winner',
       'result', 'result_margin', 'target_runs', 'target_overs', 'super_over',
       'method', 'umpire1', 'umpire2'],
      dtype='object')
```

```
df.rename(columns={"Wins": "wins"}, inplace=True)  # Agar 'Wins' capital me hai
df.rename(columns={"Total Wins": "wins"}, inplace=True)  # Agar alag naam ho
```

```
['id', 'season', 'city', 'date', 'match_type', 'player_of_match',
 'venue', 'team1', 'team2', 'toss_winner', 'toss_decision', 'winner',
 'result', 'result_margin', 'target_runs', 'target_overs', 'super_over',
 'method', 'umpire1', 'umpire2']
```

```
['id',
 'season',
 'city',
 'date',
 'match_type',
 'player_of_match',
 'venue',
 'team1',
 'team2',
 'toss_winner',
 'toss_decision',
 'winner',
 'result',
 'result_margin',
 'target_runs',
 'target_overs',
 'super_over',
 'method',
 'umpire1',
 'umpire2']
```

```
import pandas as pd

# Venue-wise wins count
pivot_df = df.pivot_table(values="id", index="venue", aggfunc="count", fill_value=0)

# Rename column
pivot_df.rename(columns={"id": "Total"}, inplace=True)

# Sorting
pivot_df = pivot_df.sort_values(by="Total", ascending=False)

# Print output
print(pivot_df)
```

```
                                              Total
venue
Eden Gardens                                     77
Wankhede Stadium                                 73
M Chinnaswamy Stadium                            65
Feroz Shah Kotla                                 60
Rajiv Gandhi International Stadium, Uppal        49
MA Chidambaram Stadium, Chepauk                  48
Sawai Mansingh Stadium                           47
Dubai International Cricket Stadium              46
Wankhede Stadium, Mumbai                         45
Punjab Cricket Association Stadium, Mohali       35
Sheikh Zayed Stadium                             29
Sharjah Cricket Stadium                          28
MA Chidambaram Stadium, Chepauk, Chennai         28
```

```
Narendra Modi Stadium, Ahmedabad                    24
Maharashtra Cricket Association Stadium             22
Dr DY Patil Sports Academy, Mumbai                  20
Dr DY Patil Sports Academy                          17
Brabourne Stadium, Mumbai                           17
Arun Jaitley Stadium, Delhi                         16
Eden Gardens, Kolkata                               16
Subrata Roy Sahara Stadium                          16
M.Chinnaswamy Stadium                               15
Kingsmead                                           15
Rajiv Gandhi International Stadium                  15
Arun Jaitley Stadium                                14
M Chinnaswamy Stadium, Bengaluru                    14
Bharat Ratna Shri Atal Bihari Vajpayee Ekana Cr...  14
Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Sta...  13
Maharashtra Cricket Association Stadium, Pune       13
Rajiv Gandhi International Stadium, Uppal, Hyde...  13
SuperSport Park                                     12
Sardar Patel Stadium, Motera                        12
Punjab Cricket Association IS Bindra Stadium, M...  11
Sawai Mansingh Stadium, Jaipur                      10
Brabourne Stadium                                   10
Saurashtra Cricket Association Stadium              10
Punjab Cricket Association IS Bindra Stadium        10
MA Chidambaram Stadium                               9
Himachal Pradesh Cricket Association Stadium         9
Holkar Cricket Stadium                               9
Zayed Cricket Stadium, Abu Dhabi                     8
New Wanderers Stadium                                8
Newlands                                             7
Barabati Stadium                                     7
JSCA International Stadium Complex                   7
St George's Park                                     7
Shaheed Veer Narayan Singh International Stadium     6
Punjab Cricket Association IS Bindra Stadium, M...   5
Maharaja Yadavindra Singh International Cricket...   5
Nehru Stadium                                        5
Himachal Pradesh Cricket Association Stadium, D...   4
Green Park                                           4
De Beers Diamond Oval                                3
Buffalo Park                                         3
Vidarbha Cricket Association Stadium, Jamtha         3
Barsapara Cricket Stadium, Guwahati                  3
```

```
colormap = plt.cm.get_cmap("tab20", len(pivot_df.columns))
pivot_df.plot(kind="barh", stacked=True, figsize=(15, 12), colormap=colormap, edgecolor="black")
```

```
<ipython-input-28-3c562f0b63aa>:1: MatplotlibDeprecationWarning: The get_cmap function was deprecated in Matplotlib 3.7 and will be remo
  colormap = plt.cm.get_cmap("tab20", len(pivot_df.columns))
```