

# SOURCE CODE :>>

Overview of pycharm file:::

```
"""
Project on : Data Science
Dataset Used: Ipl Data Analysis
Project Contributed by : Saurabh Aher & Tushar Ambekar
Date: 4/sep/2022
Time: 11:25 AM

Guided By: Teachnook Mentors
"""
import math
import matplotlib
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import time

# Reading csv file....
df = pd.read_csv('C:/Users/Administrator/PycharmProjects/pythonProject/IPL
Dataset.csv')

matchesdf =
pd.read_csv('C:/Users/Administrator/PycharmProjects/pythonProject/matches.c
sv')
deliveriespdf =
pd.read_csv('C:/Users/Administrator/PycharmProjects/pythonProject/deliverie
s.csv')

def check_ipl_datasets_attributes():
    print("1: See Top 5 Contents of file")
    print("2: See columns,count and datatypes")
    user_input1 = int(input("Choose 1/2:>> "))
    if(user_input == 1):
        print(df.head())
    elif(user_input== 2):
        print(df.info())

def General_Analysis_of_IPL_Matches():
    print("1: List of Sessions")
    print("2: First Ball of IPL history")
    print("3: Season Wise IPL Matches")
    print("4: Most IPL matches played in a venue")
    print("5: IPL matches played by each team")
    user_input = int(input("Choose:> "))
    if(user_input==1):
        print(df['season'].unique())
    elif (user_input == 2):
        print(df.iloc[0])
    elif(user_input==3):
        plt.figure(figsize=(10, 8))
        data = df.groupby(['match_id',
'season']).count().index.droplevel(level=0).value_counts().sort_index()
        sns.barplot(y=data.index, x=data, orient='h')
        plt.xlabel('Matches Played')
```

```

        plt.ylabel('Season')
        plt.show()
    elif(user_input==4):
        print(df.groupby(['venue',
'match_id']).count().droplevel(level=1).index.value_counts())
    elif(user_input==5):
        plt.figure(figsize=(10, 8))
        data =
df['bowling_team'].value_counts().sort_values(ascending=False)
        sns.barplot(y=data.index, x=data, orient='h')
        plt.xlabel('Matches Played')
        plt.ylabel('Team')
        plt.show()

def IPL_batting_analysis():
    print("1: Most Run Scored By IPL Teams")
    print("2: Most IPL runs by a batsman")
    print("3: Avg Run by Teams in Powerplay")
    print("4: Most IPL Century by a Player")
    print("5: Most IPL Fifty by Player")
    print("6: Orange Cap Holder Each Season")
    print("7: Most Sixes in an IPL Inning")
    print("8: Most Boundary (4s) hit by a Batsman")
    print("9: Most runs in an IPL season by Player")
    print("10: No. of Sixes in IPL Seasons")
    print("11: Highest Total by IPL Teams")
    print("12: Most IPL Sixes Hit by a batsman")
    print("13: Highest Individual IPL Score")

    user_input = int(input("Choose:> "))
    if(user_input==1):
print(df.groupby(['batting_team'])['run'].sum().sort_values(ascending=False
))
        elif(user_input==2):
            plt.figure(figsize=(10,8))

data=df.groupby(['striker'])['runs_off_bat'].sum().sort_values(ascending=Fa
lse)[:10]
            sns.barplot(y=data.index,x=data,orient='h')
            plt.xlabel('Batsman')
            plt.ylabel('Runs')
            plt.show()
        elif(user_input==3):
            print(df[df['over'] < 6].groupby(['match_id',
'batting_team']).sum()['run'].groupby('batting_team').mean().sort_values(as
cending=False)[2:])

            elif(user_input==4):
                runs = df.groupby(['striker', 'match_id'])['runs_off_bat'].sum()
                print(runs[runs >=
100].droplevel(level=1).groupby('striker').count().sort_values(ascending=Fa
lse)[:10])

            elif(user_input==5):
                plt.figure(figsize=(10, 8))
                runs = df.groupby(['striker', 'start_date'])['runs_off_bat'].sum()
                data = runs[runs >=
50].droplevel(level=1).groupby('striker').count().sort_values(ascending=Fal
se)[:10]

```

```

sns.barplot(y=data.index, x=data, orient='h')
plt.xlabel('Half-Centuries')
plt.ylabel('Batsman')
plt.show()

elif(user_input==6):
    data = df.groupby(['season',
'striker'])['runs_off_bat'].sum().groupby('season').max()
    temp_df = pd.DataFrame(df.groupby(['season',
'striker'])['runs_off_bat'].sum())
    print("{0:10}{1:20}{2:30}".format("Season", "Player", "Runs"))
    for season, run in data.items():
        player =
temp_df.loc[season][temp_df.loc[season]['runs_off_bat'] == run].index[0]
        print(season, '\t ', player, '\t\t', run)

elif(user_input==7):
    print(df[df['runs_off_bat'] == 6].groupby(['start_date',
'striker']).count()['season'].sort_values(ascending=False).droplevel(level=
0)[:10])

elif(user_input==8):
    plt.figure(figsize=(10, 8))
    data = df[df['runs_off_bat'] == 4]['striker'].value_counts()[:10]
    sns.barplot(y=data.index, x=data, orient='h')
    plt.xlabel('Fours')
    plt.ylabel('Batsman')
    plt.show()

elif(user_input==9):
    print(df.groupby(['striker',
'season'])['runs_off_bat'].sum().sort_values(ascending=False)[:10])

elif(user_input==10):
    plt.figure(figsize=(10, 8))
    data = df[df['runs_off_bat'] ==
6].groupby('season').count()['match_id'].sort_values(ascending=False)
    sns.barplot(y=data.index, x=data, orient='h')
    plt.xlabel('Sixes')
    plt.ylabel('Season')
    plt.show()

elif(user_input==11):
    print(df.groupby(['start_date',
'batting_team']).sum()['run'].droplevel(level=0).sort_values(ascending=Fals
e)[:10])

elif(user_input==12):
    plt.figure(figsize=(10, 8))
    data = df[df['runs_off_bat'] == 6]['striker'].value_counts()[:10]
    sns.barplot(y=data.index, x=data, orient='h')
    plt.xlabel('Sixes')
    plt.ylabel('Batsman')
    plt.show()

elif(user_input==13):
    print(df.groupby(['striker',
'start_date'])['runs_off_bat'].sum().sort_values(ascending=False)[:10])

def Balling_statistics():

```

```

print("1: Most run conceded by a bowler in an inning")
print("2: Purple Cap Holders")
print("3: Most IPL Wickets by a Bowler")
print("4: Most Dot Ball by a Bowler")
print("5: Most Maiden over by a Bowler")
print("6: Most Wickets by an IPL Team")
print("7: Most No Balls by an IPL team")
print("8: Most No Balls by an IPL Bowler")
print("9: Most run given by a team in Extras")
print("10: Most Wides Conceded by an IPL team")

user_input = int(input("Choose:> "))
if(user_input==1):
    print(df.groupby(['bowler',
'start_date'])['run'].sum().droplevel(level=1).sort_values(ascending=False)
[:10])
elif(user_input==2):
    lst = 'caught,bowled,lbw,stumped,caught and bowled,hit wicket'
    data = df[df['wicket_type'].apply(lambda x: True if x in lst and x
!= ' ' else False)].groupby(
    ['season', 'bowler']).count()['ball']
    data = data.sort_values(ascending=False)[:30].sort_index(level=0)
    val = 0
    lst = []
    print("{0:10}{1:20}{2:30}".format("Season", "Player", "Runs"))
    for (season, bowler), wicket in data.items():
        if season == val:
            lst.append(wicket)
        else:
            print(season, '\t ', bowler, '\t\t', wicket)
            val = season
            lst = []
elif(user_input==3):
    lst = 'caught,bowled,lbw,stumped,caught and bowled,hit wicket'
    print(df[df['wicket_type'].apply(lambda x: True if x in lst and x
!= ' ' else False)]['bowler'].value_counts()[:10])

elif(user_input==4):
    plt.figure(figsize=(10, 8))
    data = df[df['run'] ==
0].groupby('bowler').count()['match_id'].sort_values(ascending=False)[:10]
    sns.barplot(y=data.index, x=data, orient='h')
    plt.xlabel('Dot Balls')
    plt.ylabel('bowler')
    plt.show()

elif(user_input==5):
    data = df.groupby(['start_date', 'bowler', 'over'])['run'].sum()
    data = data[data.values == 0].droplevel(level=[0, 2])
    print(data.index.value_counts()[:10])

elif(user_input==6):
    plt.figure(figsize=(10, 8))
    lst = 'caught,bowled,lbw,stumped,caught and bowled,hit wicket'
    data = df[df['wicket_type'].apply(lambda x: True if x in lst and x
!= ' ' else False)][
    'bowling_team'].value_counts()

df.groupby(['batting_team'])['extras'].agg('sum').sort_values(ascending=False)
se)

```

```

        sns.barplot(y=data.index, x=data, orient='h')
        plt.xlabel('Wickets')
        plt.ylabel('Teams')
        plt.show()

    elif(user_input==7):

print(df.groupby(['batting_team'])['noballs'].agg('sum').sort_values(ascending=False))

        elif(user_input==8):
            print(df[df['noballs'] != 0]['bowler'].value_counts()[:10])

        elif(user_input==9):
            plt.figure(figsize=(10, 8))
            data =
df.groupby(['batting_team'])['extras'].agg('sum').sort_values(ascending=False)
            sns.barplot(y=data.index, x=data, orient='h')
            plt.xlabel('Runs')
            plt.ylabel('Teams')
            plt.show()

        elif(user_input==10):

print(df.groupby(['batting_team'])['wides'].agg('sum').sort_values(ascending=False))

def Toss_winner():
    winner = matchesdf[matchesdf['toss_winner'] == matchesdf['winner']]
    labels = ['YES', 'NO']
    # pie plot
    plt.pie([len(winner), (577 - len(winner))], labels=labels,
autopct='%1.1f%%', shadow=True, startangle=90)
    plot = plt.gcf()
    plt.title("DJ2 TOSS WINNER PERCENTAGE OF SUCCES")
    plt.show()

while (1):
    print("1: Check IPL Datasets Attributes.....")
    print("2: General Analysis of IPL Matches")
    print("3: IPL Batting Analysis")
    print("4: Balling Statistics")
    print("5: Toss Winner's Percentage of Success")
    print("6: Exit")
    user_input = int(input("Press:> "))
    if(user_input == 1):
        check_ipl_datasets_attributes()

    elif(user_input==2):
        print("Let's See General Analysis of IPL Matches")
        # call to "2: General Analysis of IPL Matches"
        General_Analysis_of_IPL_Matches()

    elif(user_input==3):
        print("let's see IPL Batting Analysis")
        IPL_batting_analysis()

    elif(user_input==4):
        Balling_statistics()

```

```

elif(user_input==5):
    print("Let's See Toss Winners Percentage")
    Toss_winner()

elif(user_input==6):
    print("THANKYOU.....")
    time.sleep(3)
    exit()

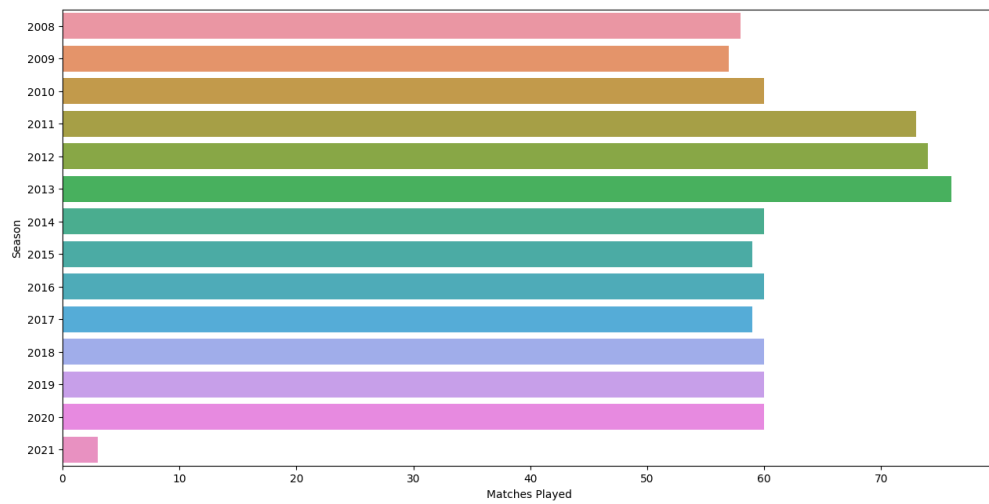
```

## SCREEN SHOT OF CODES ::>>>

```

1  """
2  Project on : Data Science
3  Dataset Used: Ipl Data Analysis
4  Project Contributed by : Saurabh Aher & Tushar Ambekar
5  Date: 4/sep/2022
6  Time: 11:25 AM
7
8  Guided By: Teachnook Mentors
9  """
10
11 import math
12 import matplotlib
13 import numpy as np
14 import pandas as pd
15 import seaborn as sns
16 import matplotlib.pyplot as plt
17
18 # Reading csv file....
19 df = pd.read_csv('C:/Users/Administrator/PycharmProjects/pythonProject/IPL Dataset.csv')
20
21 matchesdf = pd.read_csv('C:/Users/Administrator/PycharmProjects/pythonProject/matches.csv')
22 deliveriesdf = pd.read_csv('C:/Users/Administrator/PycharmProjects/pythonProject/deliveries.csv')
23
24
25 def check_ipl_datasets_attributes():
26     print("1: See Top 5 Contents of file")
27     print("2: See columns,count and datatypes")
28     user_input1 = int(input("Choose 1/2:>> "))
29     if(user_input == 1):
30         print(df.head())

```

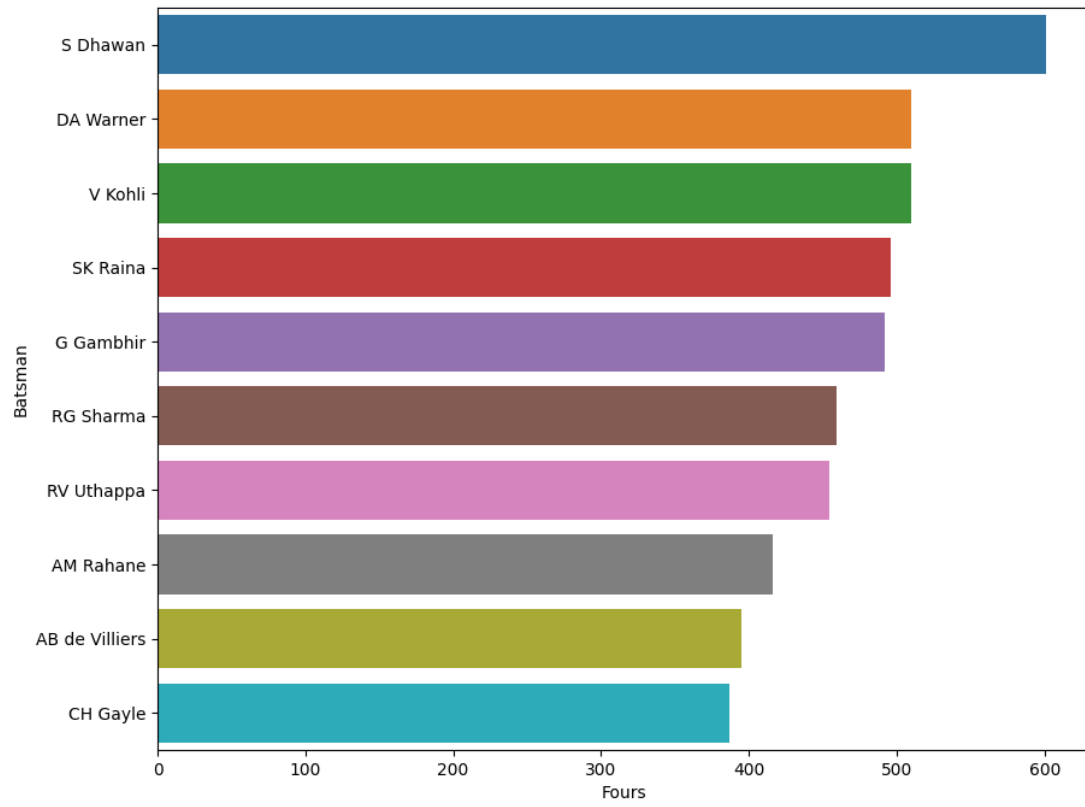


```

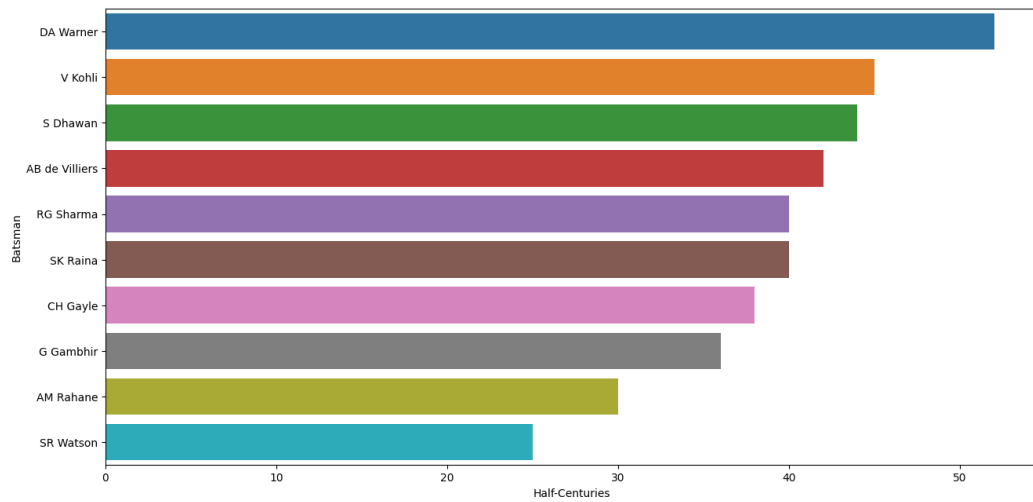
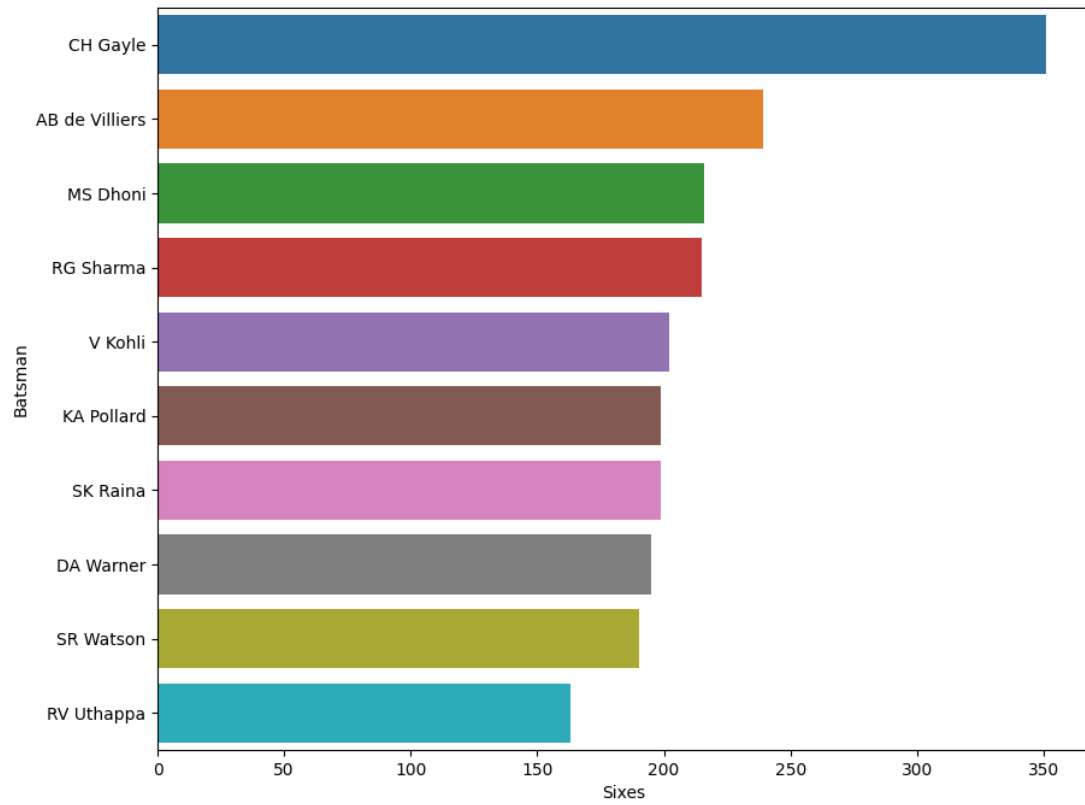
pythonProject DMW 3.py
File Edit View Navigate Code Refactor Run Tools VCS Window Help pythonProject DMW 3.py - Administrator
pythonProject DMW 3.py
q1.py x DMW MINIPROJECT.py x DMW 2.py x trial.py x Jarvis ok.py x python handbook.py x St Management.py x DMW 3.py x DMW 4.py x
Run DMW 3
C:\Users\Administrator\AppData\Local\Programs\Python\Python310\python.exe "C:/Users/Administrator/PycharmProjects/pythonProject/DMW 3.py"
1: Check IPL Datasets Attributes.....
2: General Analysis of IPL Matches
3: IPL Batting Analysis
4: Balling Statistics
5: Toss Winner's Percentage of Success
6: Exit
Press:>
1: See Top 5 Contents of file
2: See columns,count and datatypes
Choose 1/2:>>
match_id season start_date ... player_dismissed run over
0 335982 2008 2008-04-18 ... 1 0
1 335982 2008 2008-04-18 ... 0 0
2 335982 2008 2008-04-18 ... 1 0
3 335982 2008 2008-04-18 ... 0 0
4 335982 2008 2008-04-18 ... 0 0

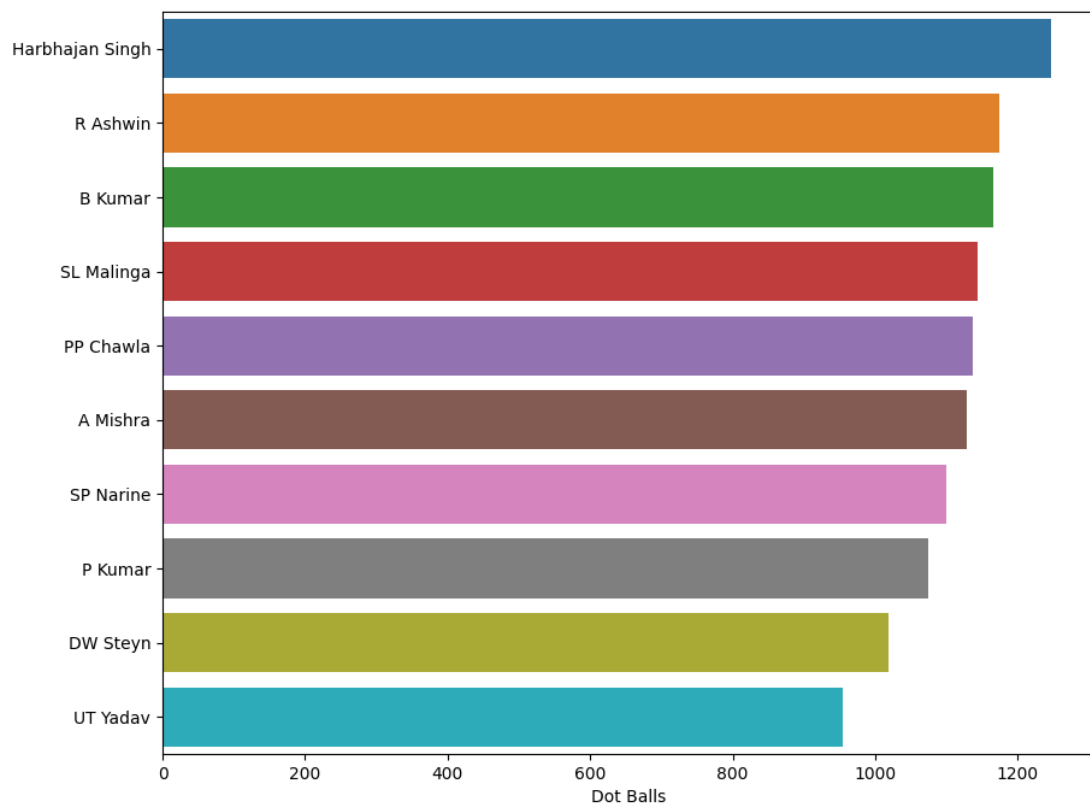
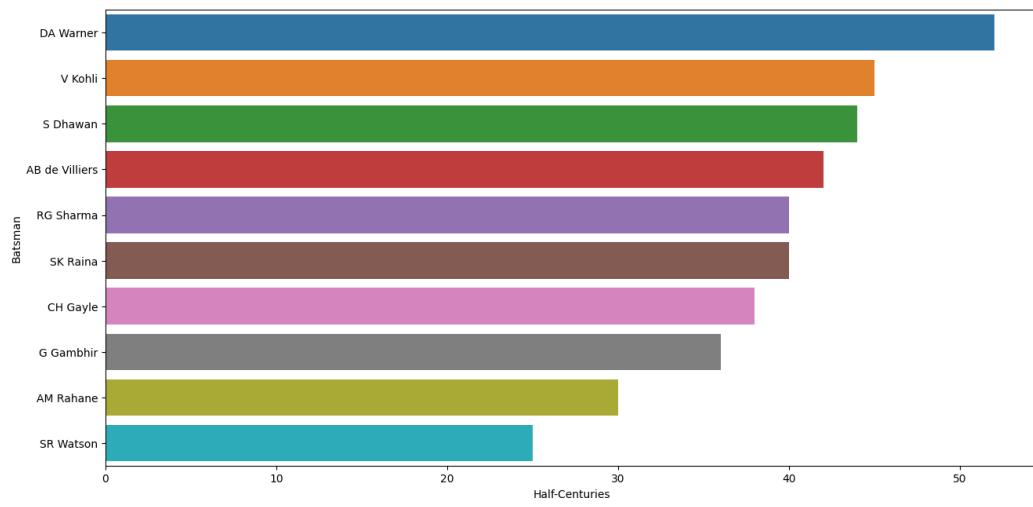
[5 rows x 21 columns]
1: Check IPL Datasets Attributes.....
2: General Analysis of IPL Matches
3: IPL Batting Analysis
4: Balling Statistics
5: Toss Winner's Percentage of Success
6: Exit
Press:>
1: See Top 5 Contents of file
2: See columns,count and datatypes
Choose 1/2:>>
match_id season start_date ... player_dismissed run over

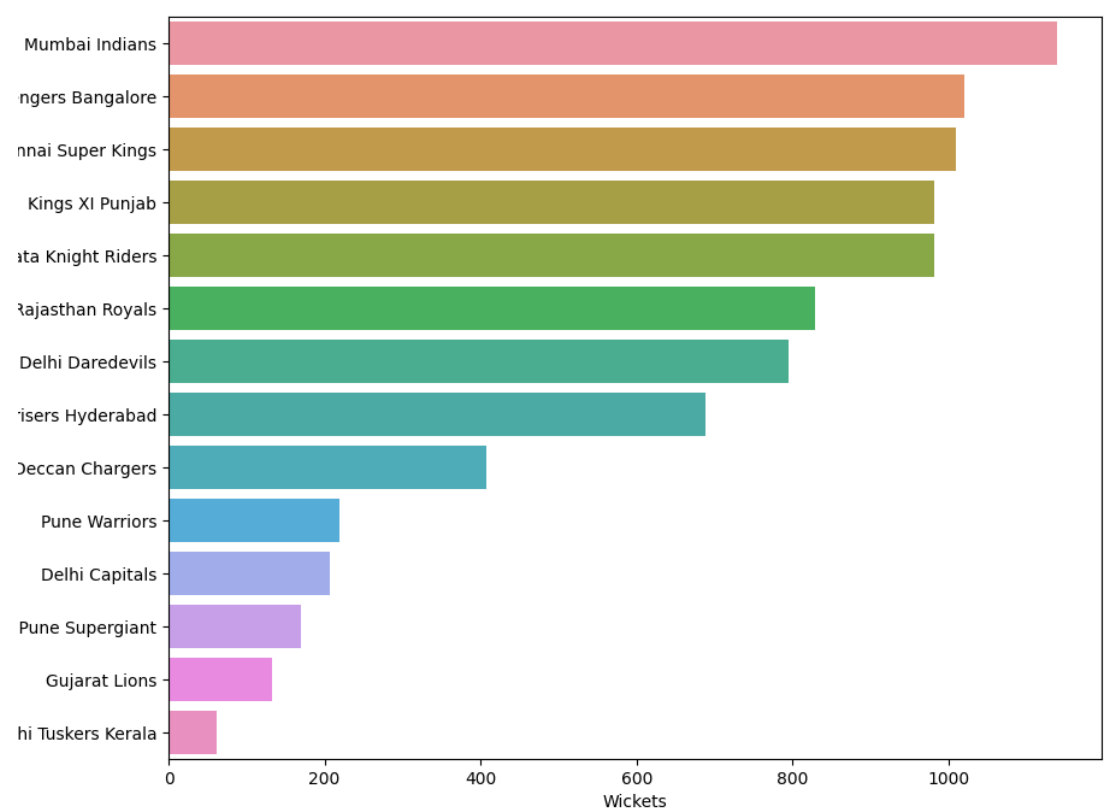
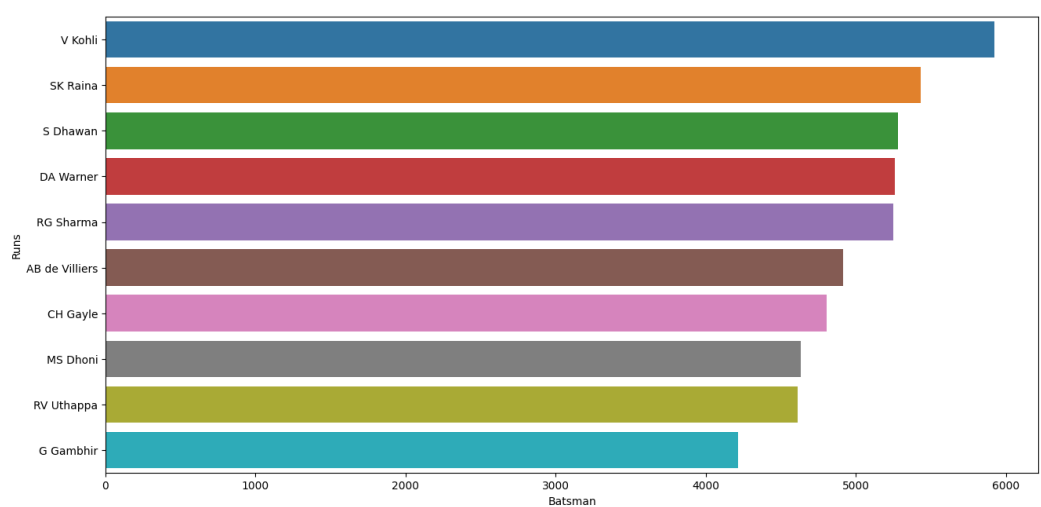
```



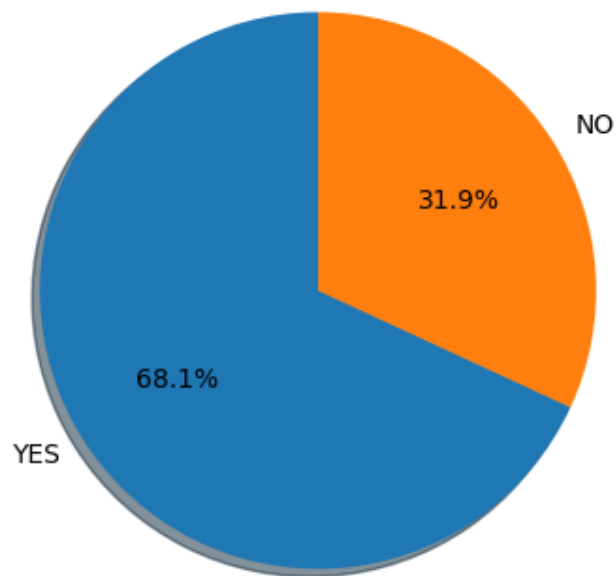








## DJ2 TOSS WINNER PERCENTAGE OF SUCCES



---

Thanks and Regards:

At last but not least we would like to tank-you "TEACHNOOK TEAM" for giving this golden opportunity to learn data science.

---

Your's Sincerely

Aher Saurabh & Ambekar Tushar