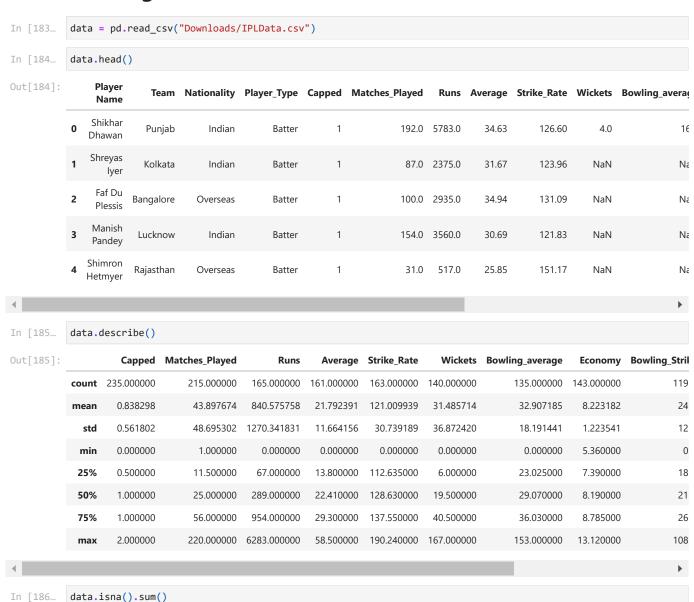
## **IPL Data Analysis**

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

## Loading the data



```
Out[186]: Player Name
          Team
          Nationality
                                    0
          Player_Type
                                   0
          Capped
                                    0
          Matches_Played
                                   20
          Runs
                                   70
          Average
                                   74
          Strike_Rate
                                   72
          Wickets
                                   95
          Bowling_average
                                   100
          Economy
                                   92
          Bowling_Strike_Rate
                                   116
          Catches
                                   208
                                   208
          Run_outs
          Stumps
                                   208
          dtype: int64
In [187...
          data.info()
          <class 'pandas.core.frame.DataFrame'>
          RangeIndex: 235 entries, 0 to 234
          Data columns (total 16 columns):
                                  Non-Null Count Dtype
           # Column
                                    235 non-null
               Player Name
                                                      object
                                    235 non-null
           1
               Team
                                                      object
           2 Nationality 235 non-null
3 Player_Type 235 non-null
4 Capped 235 non-null
5 Matches_Played 215 non-null
6 Runs 165 non-null
                                                      object
                                                      object
                                                      int64
                                                      float64
                                                      float64
           7 Average
                                   161 non-null float64
           8 Strike_Rate 163 non-null float64
9 Wickets 140 non-null float64
           10 Bowling_average 135 non-null float64
                                    143 non-null
                                                      float64
           11 Economy
           12 Bowling_Strike_Rate 119 non-null
                                                      float64
                           27 non-null
                                                      float64
           13 Catches
           14 Run outs
                                     27 non-null
                                                       float64
           15 Stumps
                                     27 non-null
                                                       float64
          dtypes: float64(11), int64(1), object(4)
          memory usage: 29.5+ KB
          cleaning data
In [188...
          # segregating data - capped batters
          batters = data.loc[(data["Player_Type"] == "Batter")]
          batters_new = batters.loc[(batters["Capped"] == 1)]
          Capped_Batters = batters_new[['Player Name',
                                         'Nationality',
                                         'Matches_Played',
                                         'Runs',
                                         'Average',
                                         'Strike_Rate']]
In [189...
         Capped_Batters.head()
```

localhost:8888/nbconvert/html/IPL Data Analysis.ipynb?download=false

```
Out[189]:
                  Player Name
                                   Team Nationality Matches_Played
                                                                        Runs Average Strike_Rate
                                                                192.0 5783.0
            0
                Shikhar Dhawan
                                   Punjab
                                               Indian
                                                                                 34.63
                                                                                            126.60
            1
                    Shreyas Iyer
                                  Kolkata
                                               Indian
                                                                 87.0 2375.0
                                                                                 31.67
                                                                                            123.96
                                                                      2935.0
            2
                  Faf Du Plessis
                                Bangalore
                                             Overseas
                                                                100.0
                                                                                 34.94
                                                                                            131.09
                                                                 154.0 3560.0
                 Manish Pandey
                                                                                 30.69
                                                                                            121.83
            3
                                 Lucknow
                                               Indian
            4 Shimron Hetmyer
                                                                 31.0
                                                                        517.0
                                                                                 25.85
                                                                                            151.17
                                Rajasthan
                                             Overseas
In [241...
            #segregating Data - capped bowlers
            Bowlers = data.loc[(data["Player_Type"] == "Bowler ")]
            Bowlers_new = Bowlers.loc[(Bowlers["Capped"] == 1)]
            Capped_Bowlers = Bowlers_new[['Player Name',
                                             'Nationality',
                                             'Matches_Played',
                                             'Wickets',
                                             'Bowling_average',
                                             'Economy',
                                             'Bowling_Strike_Rate']]
            Capped_Bowlers.head()
In [242...
Out[242]:
                     Player Name
                                             Nationality
                                                         Matches_Played Wickets
                                                                                   Bowling_average Economy Bowling_Strike_Rate
                                       Team
            36
                    Kagiso Rabada
                                      Punjab
                                                                     50.0
                                                                              76.0
                                                                                              20.53
                                                                                                         8.21
                                                                                                                             15.00
                                                Overseas
            37
                                   Rajasthan
                                                                     62.0
                                                                              76.0
                                                                                              26.09
                                                                                                         8.40
                                                                                                                             18.64
                       Trent Boult
                                                Overseas
                Mohammad Shami
                                                  Indian
                                                                     77.0
                                                                              79.0
                                                                                              30.41
                                                                                                         8.63
                                                                                                                             21.14
            38
                                     Gujarat
            39
                      T Natarajan Hyderabad
                                                  Indian
                                                                     24.0
                                                                              20.0
                                                                                                         8.24
                                                                                                                             25.05
                                                                                              34.40
            40
                   Deepak Chahar
                                                                     63.0
                                                                                                         7.80
                                     Chennai
                                                  Indian
                                                                              59.0
                                                                                              29.19
                                                                                                                             22.44
In [243...
            # segregating data - capped keepers
            Keepers = data.loc[(data["Player_Type"] == "Keeper")]
            Keepers_new = Keepers.loc[(Keepers["Capped"] == 1)]
            Capped_Keepers = Keepers_new[['Player Name',
                                              'Team',
                                             'Nationality',
                                             'Matches_Played',
                                             'Runs',
                                             'Average',
                                             'Strike_Rate',
                                             'Catches',
                                             'Run_outs',
                                             'Stumps']]
In [244...
            Capped_Keepers.head()
Out[244]:
                    Player Name
                                     Team
                                            Nationality Matches_Played
                                                                          Runs Average Strike_Rate Catches Run_outs Stumps
            105
                 Quinton De Kock
                                   Lucknow
                                               Overseas
                                                                   77.0
                                                                        2256.0
                                                                                    31.3
                                                                                               130.9
                                                                                                         53.0
                                                                                                                    0.0
                                                                                                                            14.0
            106
                                   Chennai
                                                 Indian
                                                                  175.0 3916.0
                                                                                    29.4
                                                                                               127.5
                                                                                                         58.0
                                                                                                                   12.0
                                                                                                                             2.0
                  Ambati Rayudu
                                                                   61.0 1452.0
            107
                                                 Indian
                                                                                    28.5
                                                                                               136.3
                                                                                                         19.0
                                                                                                                    1.0
                                                                                                                             2.0
                     Ishan Kishan
                                   Mumbai
            108
                   Jonny Bairstow
                                    Punjab
                                               Overseas
                                                                   28.0 1038.0
                                                                                    41.5
                                                                                               142.2
                                                                                                         18.0
                                                                                                                    1.0
                                                                                                                             4.0
            109
                                                                  213.0 4046.0
                                                                                    25.8
                                                                                               129.7
                                                                                                        123.0
                                                                                                                   14.0
                                                                                                                            32.0
                   Dinesh Karthik Bangalore
                                                 Indian
```

```
In [245...
           #segregating Data - Capped Allrounders
           Allrounders = data.loc[(data["Player_Type"] == "Allrounder")]
           Allrounders_new = Allrounders.loc[(Allrounders["Capped"] == 1)]
           Capped_Allrounders = Allrounders_new[['Player Name',
                                                    'Team',
                                                    'Nationality',
                                                    'Matches_Played',
                                                    'Runs',
                                                    'Average',
                                                    'Strike_Rate',
                                                    'Wickets',
                                                    'Bowling_average',
                                                    'Economy',
                                                    'Bowling_Strike_Rate']]
In [246...
           Capped_Allrounders.head()
Out[246]:
                      Player
                                Team Nationality Matches_Played
                                                                   Runs Average Strike_Rate Wickets Bowling_average Economy B
                      Name
                Ravichandran
           127
                             Rajasthan
                                           Indian
                                                           167.0
                                                                   456.0
                                                                            11.12
                                                                                      109.88
                                                                                                145.0
                                                                                                                27.80
                                                                                                                           6.91
                     Ashwin
           128
                Pat Cummins
                               Kolkata
                                         Overseas
                                                            37.0
                                                                  316.0
                                                                            19.75
                                                                                      140.44
                                                                                                38.0
                                                                                                                30.13
                                                                                                                           8.24
                     Dwayne
           129
                              Chennai
                                                            151.0 1537.0
                                                                            22.94
                                                                                      130.25
                                                                                                167.0
                                                                                                                24.32
                                                                                                                           8.36
                                         Overseas
                      Bravo
           130
                  Nitish Rana
                               Kolkata
                                           Indian
                                                            77.0 1820.0
                                                                            28.00
                                                                                      132.46
                                                                                                 7.0
                                                                                                                22.00
                                                                                                                           8.03
                                                                                                                           8.20
                Jason Holder
                             Lucknow
                                                            26.0
                                                                  189.0
                                                                            14.54
                                                                                      121.15
                                                                                                35.0
                                                                                                                22.46
           131
                                         Overseas
In [247...
           \# Cleaning the data by making the null or NAN values \theta
           Capped_Batters = Capped_Batters.fillna(0)
           Capped_Bowlers = Capped_Bowlers.fillna(0)
           Capped_Keepers = Capped_Keepers.fillna(0)
           Capped_Allrounders = Capped_Allrounders.fillna(0)
In [248...
           # Checking null values in the data
           print(Capped_Batters.isna().sum())
           print(Capped_Bowlers.isna().sum())
           print(Capped_Keepers.isna().sum())
           print(Capped_Allrounders.isna().sum())
```

```
Player Name
Team
Nationality
Matches_Played
Runs
Average
Strike_Rate
dtype: int64
                      0
Player Name
                      0
Team
                      0
Nationality
Matches_Played
Wickets
Bowling_average
Economy
Bowling_Strike_Rate
dtype: int64
Player Name
Nationality
Matches_Played
                 0
Runs
Average
                 0
Strike_Rate
Catches
                 0
Run_outs
                 0
Stumps
dtype: int64
Player Name
Team
Nationality
Matches_Played
Runs
Average
Strike Rate
Wickets
Bowling_average
                      0
Economy
Bowling_Strike_Rate
dtype: int64
```

## **Initial analysis**

```
In [249... #Analyzing the Batters data

top_batters = Capped_Batters.loc[(Capped_Batters["Average"] >= 32.0)]

#Sorting the data in descending order with respect to each parameter
top_batters_average = top_batters.sort_values('Average', ascending=False)
top_batters_strike_rate = top_batters.sort_values('Strike_Rate', ascending=False)
top_batters_runs = top_batters.sort_values('Runs', ascending=False)
top_batters_matches = top_batters.sort_values('Matches_Played')
In [250... top_batters_average
```

Out[250]:		Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate
	212	KL Rahul	Lucknow	Indian	94.0	3273.0	47.43	136.38
	231	Ruturaj Gaikwad	Chennai	Indian	22.0	839.0	46.61	132.13
	19	David Warner	Delhi	Overseas	150.0	5449.0	41.60	139.97
	207	Kane Williamson	Hyderabad	Overseas	63.0	1885.0	40.11	131.27
	208	Virat Kohli	Bangalore	Indian	207.0	6283.0	37.40	129.95
	2	Faf Du Plessis	Bangalore	Overseas	100.0	2935.0	34.94	131.09
	0	Shikhar Dhawan	Punjab	Indian	192.0	5783.0	34.63	126.60
	26	David Miller	Gujarat	Overseas	89.0	1974.0	32.90	136.51

In [251... top\_batters\_strike\_rate

Out[251]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate
19	David Warner	Delhi	Overseas	150.0	5449.0	41.60	139.97
26	David Miller	Gujarat	Overseas	89.0	1974.0	32.90	136.51
212	KL Rahul	Lucknow	Indian	94.0	3273.0	47.43	136.38
231	Ruturaj Gaikwad	Chennai	Indian	22.0	839.0	46.61	132.13
207	Kane Williamson	Hyderabad	Overseas	63.0	1885.0	40.11	131.27
2	Faf Du Plessis	Bangalore	Overseas	100.0	2935.0	34.94	131.09
208	Virat Kohli	Bangalore	Indian	207.0	6283.0	37.40	129.95
0	Shikhar Dhawan	Punjab	Indian	192.0	5783.0	34.63	126.60

In [252... to

top\_batters\_runs

Out[252]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate
208	Virat Kohli	Bangalore	Indian	207.0	6283.0	37.40	129.95
0	Shikhar Dhawan	Punjab	Indian	192.0	5783.0	34.63	126.60
19	David Warner	Delhi	Overseas	150.0	5449.0	41.60	139.97
212	KL Rahul	Lucknow	Indian	94.0	3273.0	47.43	136.38
2	Faf Du Plessis	Bangalore	Overseas	100.0	2935.0	34.94	131.09
26	David Miller	Gujarat	Overseas	89.0	1974.0	32.90	136.51
207	Kane Williamson	Hyderabad	Overseas	63.0	1885.0	40.11	131.27
231	Ruturaj Gaikwad	Chennai	Indian	22.0	839.0	46.61	132.13
2 26 207	Faf Du Plessis David Miller Kane Williamson	Bangalore Gujarat Hyderabad	Overseas Overseas	100.0 89.0 63.0	2935.0 1974.0 1885.0	34.94 32.90 40.11	131.09 136.51 131.27

In [253...

top\_batters\_matches

Out[253]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate
231	Ruturaj Gaikwad	Chennai	Indian	22.0	839.0	46.61	132.13
207	Kane Williamson	Hyderabad	Overseas	63.0	1885.0	40.11	131.27
26	David Miller	Gujarat	Overseas	89.0	1974.0	32.90	136.51
212	KL Rahul	Lucknow	Indian	94.0	3273.0	47.43	136.38
2	Faf Du Plessis	Bangalore	Overseas	100.0	2935.0	34.94	131.09
19	David Warner	Delhi	Overseas	150.0	5449.0	41.60	139.97
0 208	Shikhar Dhawan Punj		Indian	192.0	5783.0	34.63	126.60
	Virat Kohli	Bangalore	Indian	207.0	6283.0	37.40	129.95
0	Shikhar Dhawan	Punjab	Indian	192.0	5783.0	34.63	126.60

```
# Analyzing bowlers data
top_bowlers = Capped_Bowlers.loc[(Capped_Bowlers["Bowling_average"] <= 24.0)]

top_bowlers_average = top_bowlers.sort_values('Bowling_average')
top_bowlers_strike_rate = top_bowlers.sort_values('Bowling_Strike_Rate')
top_bowlers_wickets = top_bowlers.sort_values('Wickets', ascending = False)
top_bowlers_economy = top_bowlers.sort_values('Economy')
top_bowlers_matches = top_bowlers.sort_values('Matches_Played', ascending = False)</pre>
```

In [255... top\_bowlers\_average

Out[255]:

	Player Name	Team	Nationality	Matches_Played	Wickets	Bowling_average	Economy	Bowling_Strike_Rate
83	Rasikh Dar	Kolkata	Indian	1.0	1.0	0.00	10.50	0.00
87	Sean Abbott	Hyderabad	Overseas	2.0	0.0	0.00	11.40	0.00
101	Kuldip Yadav	Rajasthan	Indian	1.0	0.0	0.00	8.00	0.00
88	Alzarri Joseph	Gujarat	Overseas	3.0	6.0	14.50	10.04	8.67
93	Lungi Ngidi	Delhi	Overseas	14.0	25.0	17.92	8.30	12.96
36	Kagiso Rabada	Punjab	Overseas	50.0	76.0	20.53	8.21	15.00
234	Anrich Nortje	Delhi	Overseas	24.0	34.0	20.56	7.65	16.12
103	Nathan Coulter-Nile	Rajasthan	Overseas	38.0	48.0	21.92	7.52	17.48
50	Yuzvendra Chahal	Rajasthan	Indian	114.0	139.0	22.28	7.59	17.61
210	Arshdeep Singh	Punjab	Indian	23.0	30.0	22.30	8.78	15.23
215	Jasprit Bumrah	Mumbai	Indian	106.0	130.0	23.05	7.42	18.63
214	Varun Chakravarthy	Kolkata	Indian	31.0	36.0	23.31	6.82	20.50

In [256... top\_bowlers\_strike\_rate

Out[256]:

Player Name	Team	Nationality	Matches_Played	Wickets	Bowling_average	Economy	Bowling_Strike_Rate
Rasikh Dar	Kolkata	Indian	1.0	1.0	0.00	10.50	0.00
Sean Abbott	Hyderabad	Overseas	2.0	0.0	0.00	11.40	0.00
Kuldip Yadav	Rajasthan	Indian	1.0	0.0	0.00	8.00	0.00
Alzarri Joseph	Gujarat	Overseas	3.0	6.0	14.50	10.04	8.67
Lungi Ngidi	Delhi	Overseas	14.0	25.0	17.92	8.30	12.96
Kagiso Rabada	Punjab	Overseas	50.0	76.0	20.53	8.21	15.00
Arshdeep Singh	Punjab	Indian	23.0	30.0	22.30	8.78	15.23
Anrich Nortje	Delhi	Overseas	24.0	34.0	20.56	7.65	16.12
athan Coulter-Nile	Rajasthan	Overseas	38.0	48.0	21.92	7.52	17.48
Yuzvendra Chahal	Rajasthan	Indian	114.0	139.0	22.28	7.59	17.61
Jasprit Bumrah	Mumbai	Indian	106.0	130.0	23.05	7.42	18.63
arun Chakravarthy	Kolkata	Indian	31.0	36.0	23.31	6.82	20.50
	Sean Abbott Kuldip Yadav Alzarri Joseph Lungi Ngidi Kagiso Rabada Arshdeep Singh Anrich Nortje than Coulter-Nile /uzvendra Chahal Jasprit Bumrah	Sean Abbott Hyderabad Kuldip Yadav Rajasthan Alzarri Joseph Gujarat Lungi Ngidi Delhi Kagiso Rabada Punjab Arshdeep Singh Punjab Anrich Nortje Delhi than Coulter-Nile Rajasthan Auzvendra Chahal Rajasthan Jasprit Bumrah Mumbai	Sean Abbott Hyderabad Overseas Kuldip Yadav Rajasthan Indian Alzarri Joseph Gujarat Overseas Lungi Ngidi Delhi Overseas Kagiso Rabada Punjab Overseas Arshdeep Singh Punjab Indian Anrich Nortje Delhi Overseas than Coulter-Nile Rajasthan Overseas Auzvendra Chahal Rajasthan Indian Jasprit Bumrah Mumbai Indian	Sean Abbott Hyderabad Overseas 2.0  Kuldip Yadav Rajasthan Indian 1.0  Alzarri Joseph Gujarat Overseas 3.0  Lungi Ngidi Delhi Overseas 14.0  Kagiso Rabada Punjab Overseas 50.0  Arshdeep Singh Punjab Indian 23.0  Anrich Nortje Delhi Overseas 24.0  than Coulter-Nile Rajasthan Overseas 38.0  //uzvendra Chahal Rajasthan Indian 114.0  Jasprit Bumrah Mumbai Indian 106.0	Sean AbbottHyderabadOverseas2.00.0Kuldip YadavRajasthanIndian1.00.0Alzarri JosephGujaratOverseas3.06.0Lungi NgidiDelhiOverseas14.025.0Kagiso RabadaPunjabOverseas50.076.0Arshdeep SinghPunjabIndian23.030.0Anrich NortjeDelhiOverseas24.034.0than Coulter-NileRajasthanOverseas38.048.0Zuzvendra ChahalRajasthanIndian114.0139.0Jasprit BumrahMumbaiIndian106.0130.0	Sean AbbottHyderabadOverseas2.00.00.00Kuldip YadavRajasthanIndian1.00.00.00Alzarri JosephGujaratOverseas3.06.014.50Lungi NgidiDelhiOverseas14.025.017.92Kagiso RabadaPunjabOverseas50.076.020.53Arshdeep SinghPunjabIndian23.030.022.30Anrich NortjeDelhiOverseas24.034.020.56than Coulter-NileRajasthanOverseas38.048.021.92Zuzvendra ChahalRajasthanIndian114.0139.022.28Jasprit BumrahMumbaiIndian106.0130.023.05	Sean Abbott         Hyderabad         Overseas         2.0         0.0         0.00         11.40           Kuldip Yadav         Rajasthan         Indian         1.0         0.0         0.00         8.00           Alzarri Joseph         Gujarat         Overseas         3.0         6.0         14.50         10.04           Lungi Ngidi         Delhi         Overseas         14.0         25.0         17.92         8.30           Kagiso Rabada         Punjab         Overseas         50.0         76.0         20.53         8.21           Arshdeep Singh         Punjab         Indian         23.0         30.0         22.30         8.78           Anrich Nortje         Delhi         Overseas         24.0         34.0         20.56         7.65           than Coulter-Nile         Rajasthan         Overseas         38.0         48.0         21.92         7.52           Zuzvendra Chahal         Rajasthan         Indian         114.0         139.0         22.28         7.59           Jasprit Bumrah         Mumbai         Indian         106.0         130.0         23.05         7.42

In [257... top\_bowlers\_wickets

Out[257]:		Player Name	Team	Nationality	Matches_Played	Wickets	Bowling_average	Economy	Bowling_Strike_Rate
	50	Yuzvendra Chahal	Rajasthan	Indian	114.0	139.0	22.28	7.59	17.61
	215	Jasprit Bumrah	Mumbai	Indian	106.0	130.0	23.05	7.42	18.63
	36	Kagiso Rabada	Punjab	Overseas	50.0	76.0	20.53	8.21	15.00
	103	Nathan Coulter-Nile	Rajasthan	Overseas	38.0	48.0	21.92	7.52	17.48
	214	Varun Chakravarthy	Kolkata	Indian	31.0	36.0	23.31	6.82	20.50
	234	Anrich Nortje	Delhi	Overseas	24.0	34.0	20.56	7.65	16.12
	210	Arshdeep Singh	Punjab	Indian	23.0	30.0	22.30	8.78	15.23
	93	Lungi Ngidi	Delhi	Overseas	14.0	25.0	17.92	8.30	12.96
	88	Alzarri Joseph	Gujarat	Overseas	3.0	6.0	14.50	10.04	8.67
	83	Rasikh Dar	Kolkata	Indian	1.0	1.0	0.00	10.50	0.00
	87	Sean Abbott	Hyderabad	Overseas	2.0	0.0	0.00	11.40	0.00
	101	Kuldip Yadav	Rajasthan	Indian	1.0	0.0	0.00	8.00	0.00
In [258	top_	_bowlers_economy							
Out[258]:		Player Name	Team	Nationality	Matches_Played	Wickets	Bowling_average	Economy	Bowling_Strike_Rate
	214	Varun Chakravarthy	Kolkata	Indian	31.0	36.0	23.31	6.82	20.50

	Player Name	Team	Nationality	Matches_Played	Wickets	Bowling_average	Economy	Bowling_Strike_Rate
214	Varun Chakravarthy	Kolkata	Indian	31.0	36.0	23.31	6.82	20.50
215	Jasprit Bumrah	Mumbai	Indian	106.0	130.0	23.05	7.42	18.63
103	Nathan Coulter-Nile	Rajasthan	Overseas	38.0	48.0	21.92	7.52	17.48
50	Yuzvendra Chahal	Rajasthan	Indian	114.0	139.0	22.28	7.59	17.61
234	Anrich Nortje	Delhi	Overseas	24.0	34.0	20.56	7.65	16.12
101	Kuldip Yadav	Rajasthan	Indian	1.0	0.0	0.00	8.00	0.00
36	Kagiso Rabada	Punjab	Overseas	50.0	76.0	20.53	8.21	15.00
93	Lungi Ngidi	Delhi	Overseas	14.0	25.0	17.92	8.30	12.96
210	Arshdeep Singh	Punjab	Indian	23.0	30.0	22.30	8.78	15.23
88	Alzarri Joseph	Gujarat	Overseas	3.0	6.0	14.50	10.04	8.67
83	Rasikh Dar	Kolkata	Indian	1.0	1.0	0.00	10.50	0.00
87	Sean Abbott	Hyderabad	Overseas	2.0	0.0	0.00	11.40	0.00

In [259...

top\_bowlers\_matches

Out[259]

:	Player Name	Team	Nationality	Matches_Played	Wickets	Bowling_average	Economy	Bowling_Strike_Rate
50	Yuzvendra Chahal	Rajasthan	Indian	114.0	139.0	22.28	7.59	17.61
215	Jasprit Bumrah	Mumbai	Indian	106.0	130.0	23.05	7.42	18.63
36	Kagiso Rabada	Punjab	Overseas	50.0	76.0	20.53	8.21	15.00
103	Nathan Coulter-Nile	Rajasthan	Overseas	38.0	48.0	21.92	7.52	17.48
214	Varun Chakravarthy	Kolkata	Indian	31.0	36.0	23.31	6.82	20.50
234	Anrich Nortje	Delhi	Overseas	24.0	34.0	20.56	7.65	16.12
210	Arshdeep Singh	Punjab	Indian	23.0	30.0	22.30	8.78	15.23
93	Lungi Ngidi	Delhi	Overseas	14.0	25.0	17.92	8.30	12.96
88	Alzarri Joseph	Gujarat	Overseas	3.0	6.0	14.50	10.04	8.67
87	Sean Abbott	Hyderabad	Overseas	2.0	0.0	0.00	11.40	0.00
83	Rasikh Dar	Kolkata	Indian	1.0	1.0	0.00	10.50	0.00
101	Kuldip Yadav	Rajasthan	Indian	1.0	0.0	0.00	8.00	0.00

In [260... # Analyzing Allrounders data

top\_allrounders = Capped\_Allrounders.loc[(Capped\_Allrounders["Strike\_Rate"] >= 140.0)]

top\_allrounders\_average = top\_allrounders.sort\_values('Average', ascending=False)
 top\_allrounders\_strike\_rate = top\_allrounders.sort\_values('Strike\_Rate', ascending = False)
 top\_allrounders\_runs = top\_allrounders.sort\_values('Runs', ascending= False)
 top\_allrounders\_matches = top\_allrounders.sort\_values('Matches\_Played', ascending= False)
 top\_allrounders\_bowling\_average = top\_allrounders.sort\_values('Bowling\_average')
 top\_allrounders\_bowling\_strike\_rate = top\_allrounders.sort\_values('Bowling\_Strike\_Rate')
 top\_allrounders\_wickets = top\_allrounders.sort\_values('Wickets', ascending= False)
 top\_allrounders\_economy = top\_allrounders.sort\_values('Economy')

In [261... top\_allrounders\_average

Out[261]:

:	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Во
233	Kieron Pollard	Mumbai	Overseas	178.0	3268.0	29.98	149.77	65.0	31.62	8.78	
204	Andre Russell	Kolkata	Overseas	84.0	1700.0	29.31	178.57	72.0	26.40	9.05	
211	Hardik Pandya	Gujarat	Indian	92.0	1476.0	27.33	153.91	42.0	31.26	9.07	
218	Glen Maxwell	Bangalore	Overseas	97.0	2018.0	25.23	151.84	22.0	41.59	8.55	
223	Moeen Ali	Chennai	Overseas	34.0	666.0	22.97	146.37	16.0	29.31	6.85	
128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.13	8.24	
232	Sunil Narine	Kolkata	Overseas	134.0	954.0	15.64	161.69	143.0	24.53	6.74	
165	Jofra Archer	Mumbai	Overseas	35.0	195.0	15.00	157.26	46.0	21.33	7.13	
195	Mohammad Nabi	Kolkata	Overseas	17.0	180.0	15.00	151.26	13.0	31.38	7.14	
154	K Gowtham	Lucknow	Indian	24.0	186.0	14.31	169.09	13.0	43.23	8.26	
202	Aman Khan	Kolkata	Indian	5.0	40.0	13.30	148.10	0.0	0.00	7.00	

In [262... top\_allrounders\_strike\_rate

Out[262]:

:	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Во
204	Andre Russell	Kolkata	Overseas	84.0	1700.0	29.31	178.57	72.0	26.40	9.05	
154	K Gowtham	Lucknow	Indian	24.0	186.0	14.31	169.09	13.0	43.23	8.26	
232	Sunil Narine	Kolkata	Overseas	134.0	954.0	15.64	161.69	143.0	24.53	6.74	
165	Jofra Archer	Mumbai	Overseas	35.0	195.0	15.00	157.26	46.0	21.33	7.13	
211	Hardik Pandya	Gujarat	Indian	92.0	1476.0	27.33	153.91	42.0	31.26	9.07	
218	Glen Maxwell	Bangalore	Overseas	97.0	2018.0	25.23	151.84	22.0	41.59	8.55	
195	Mohammad Nabi	Kolkata	Overseas	17.0	180.0	15.00	151.26	13.0	31.38	7.14	
233	Kieron Pollard	Mumbai	Overseas	178.0	3268.0	29.98	149.77	65.0	31.62	8.78	
202	Aman Khan	Kolkata	Indian	5.0	40.0	13.30	148.10	0.0	0.00	7.00	
223	Moeen Ali	Chennai	Overseas	34.0	666.0	22.97	146.37	16.0	29.31	6.85	
128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.13	8.24	

In [263...

top\_allrounders\_runs

Out[263]:

:		Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Во
:	233	Kieron Pollard	Mumbai	Overseas	178.0	3268.0	29.98	149.77	65.0	31.62	8.78	
:	218	Glen Maxwell	Bangalore	Overseas	97.0	2018.0	25.23	151.84	22.0	41.59	8.55	
:	204	Andre Russell	Kolkata	Overseas	84.0	1700.0	29.31	178.57	72.0	26.40	9.05	
:	211	Hardik Pandya	Gujarat	Indian	92.0	1476.0	27.33	153.91	42.0	31.26	9.07	
:	232	Sunil Narine	Kolkata	Overseas	134.0	954.0	15.64	161.69	143.0	24.53	6.74	
:	223	Moeen Ali	Chennai	Overseas	34.0	666.0	22.97	146.37	16.0	29.31	6.85	
	128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.13	8.24	
	165	Jofra Archer	Mumbai	Overseas	35.0	195.0	15.00	157.26	46.0	21.33	7.13	
	154	K Gowtham	Lucknow	Indian	24.0	186.0	14.31	169.09	13.0	43.23	8.26	
	195	Mohammad Nabi	Kolkata	Overseas	17.0	180.0	15.00	151.26	13.0	31.38	7.14	
:	202	Aman Khan	Kolkata	Indian	5.0	40.0	13.30	148.10	0.0	0.00	7.00	
												•

In [264...

top\_allrounders\_matches

Out[264]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Во
233	Kieron Pollard	Mumbai	Overseas	178.0	3268.0	29.98	149.77	65.0	31.62	8.78	
232	Sunil Narine	Kolkata	Overseas	134.0	954.0	15.64	161.69	143.0	24.53	6.74	
218	Glen Maxwell	Bangalore	Overseas	97.0	2018.0	25.23	151.84	22.0	41.59	8.55	
211	Hardik Pandya	Gujarat	Indian	92.0	1476.0	27.33	153.91	42.0	31.26	9.07	
204	Andre Russell	Kolkata	Overseas	84.0	1700.0	29.31	178.57	72.0	26.40	9.05	
128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.13	8.24	
165	Jofra Archer	Mumbai	Overseas	35.0	195.0	15.00	157.26	46.0	21.33	7.13	
223	Moeen Ali	Chennai	Overseas	34.0	666.0	22.97	146.37	16.0	29.31	6.85	
154	K Gowtham	Lucknow	Indian	24.0	186.0	14.31	169.09	13.0	43.23	8.26	
195	Mohammad Nabi	Kolkata	Overseas	17.0	180.0	15.00	151.26	13.0	31.38	7.14	
202	Aman Khan	Kolkata	Indian	5.0	40.0	13.30	148.10	0.0	0.00	7.00	

In [265...

top\_allrounders\_bowling\_average

Out[265]:

:		Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Во
7	202	Aman Khan	Kolkata	Indian	5.0	40.0	13.30	148.10	0.0	0.00	7.00	
	165	Jofra Archer	Mumbai	Overseas	35.0	195.0	15.00	157.26	46.0	21.33	7.13	
:	232	Sunil Narine	Kolkata	Overseas	134.0	954.0	15.64	161.69	143.0	24.53	6.74	
2	204	Andre Russell	Kolkata	Overseas	84.0	1700.0	29.31	178.57	72.0	26.40	9.05	
:	223	Moeen Ali	Chennai	Overseas	34.0	666.0	22.97	146.37	16.0	29.31	6.85	
	128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.13	8.24	
7	211	Hardik Pandya	Gujarat	Indian	92.0	1476.0	27.33	153.91	42.0	31.26	9.07	
	195	Mohammad Nabi	Kolkata	Overseas	17.0	180.0	15.00	151.26	13.0	31.38	7.14	
	233	Kieron Pollard	Mumbai	Overseas	178.0	3268.0	29.98	149.77	65.0	31.62	8.78	
2	218	Glen Maxwell	Bangalore	Overseas	97.0	2018.0	25.23	151.84	22.0	41.59	8.55	
	154	K Gowtham	Lucknow	Indian	24.0	186.0	14.31	169.09	13.0	43.23	8.26	
												•

In [266...

top\_allrounders\_bowling\_strike\_rate

$\cap \dots +$	$\Gamma \supset C \subset I$	
Out	200	

•	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Во
202	Aman Khan	Kolkata	Indian	5.0	40.0	13.30	148.10	0.0	0.00	7.00	
204	Andre Russell	Kolkata	Overseas	84.0	1700.0	29.31	178.57	72.0	26.40	9.05	
165	Jofra Archer	Mumbai	Overseas	35.0	195.0	15.00	157.26	46.0	21.33	7.13	
211	Hardik Pandya	Gujarat	Indian	92.0	1476.0	27.33	153.91	42.0	31.26	9.07	
233	Kieron Pollard	Mumbai	Overseas	178.0	3268.0	29.98	149.77	65.0	31.62	8.78	
232	Sunil Narine	Kolkata	Overseas	134.0	954.0	15.64	161.69	143.0	24.53	6.74	
128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.13	8.24	
223	Moeen Ali	Chennai	Overseas	34.0	666.0	22.97	146.37	16.0	29.31	6.85	
195	Mohammad Nabi	Kolkata	Overseas	17.0	180.0	15.00	151.26	13.0	31.38	7.14	
218	Glen Maxwell	Bangalore	Overseas	97.0	2018.0	25.23	151.84	22.0	41.59	8.55	
154	K Gowtham	Lucknow	Indian	24.0	186.0	14.31	169.09	13.0	43.23	8.26	

Tn [267

top\_allrounders\_wickets

Out[267]:

:		Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Во
	232	Sunil Narine	Kolkata	Overseas	134.0	954.0	15.64	161.69	143.0	24.53	6.74	
	204	Andre Russell	Kolkata	Overseas	84.0	1700.0	29.31	178.57	72.0	26.40	9.05	
	233	Kieron Pollard	Mumbai	Overseas	178.0	3268.0	29.98	149.77	65.0	31.62	8.78	
	165	Jofra Archer	Mumbai	Overseas	35.0	195.0	15.00	157.26	46.0	21.33	7.13	
	211	Hardik Pandya	Gujarat	Indian	92.0	1476.0	27.33	153.91	42.0	31.26	9.07	
	128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.13	8.24	
	218	Glen Maxwell	Bangalore	Overseas	97.0	2018.0	25.23	151.84	22.0	41.59	8.55	
	223	Moeen Ali	Chennai	Overseas	34.0	666.0	22.97	146.37	16.0	29.31	6.85	
	154	K Gowtham	Lucknow	Indian	24.0	186.0	14.31	169.09	13.0	43.23	8.26	
	195	Mohammad Nabi	Kolkata	Overseas	17.0	180.0	15.00	151.26	13.0	31.38	7.14	
	202	Aman Khan	Kolkata	Indian	5.0	40.0	13.30	148.10	0.0	0.00	7.00	
												•

In [268...

top\_allrounders\_economy

		Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	ge Econo
	232	Sunil Narine	Kolkata	Overseas	134.0	954.0	15.64	161.69	143.0	24.	53
- 1	223	Moeen Ali	Chennai	Overseas	34.0	666.0	22.97	146.37	16.0	29.3	31
7	202	Aman Khan	Kolkata	Indian	5.0	40.0	13.30	148.10	0.0	0.0	00
•	165	Jofra Archer	Mumbai	Overseas	35.0	195.0	15.00	157.26	46.0	21.3	33
	195	Mohammad Nabi	Kolkata	Overseas	17.0	180.0	15.00	151.26	13.0	31.3	38
	128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.	13
	154	K Gowtham	Lucknow	Indian	24.0	186.0	14.31	169.09	13.0	43.2	23
2	218	Glen Maxwell	Bangalore	Overseas	97.0	2018.0	25.23	151.84	22.0	41.	59
7	233	Kieron Pollard	Mumbai	Overseas	178.0	3268.0	29.98	149.77	65.0	31.0	62
7	204	Andre Russell	Kolkata	Overseas	84.0	1700.0	29.31	178.57	72.0	26.4	40
7	211	Hardik Pandya	Gujarat	Indian	92.0	1476.0	27.33	153.91	42.0	31.2	26
t t t	top_ top_ top_ top_	keepers_str keepers_run keepers_mat keepers_cat	rage = to ike_rate s = top_k ches = to ches = to	p_keepers.: = top_keepe eepers.sort p_keepers.: p_keepers.:	sort_values('Avers.sort_values :_values('Runs sort_values('Masort_values('Casort_values('Avers_values('Avers_values('Avers_values('Avers_values('Avers_values('Avers_values('Avers_values('Avers_values('Avers_values('Avers_values('Casort_values('Ca	verage', s('Strik ', ascer atches_F atches',	ke_Rate', nding=Fal Played', , ascendi	ing=False) , ascending lse) ascending ing=False)	= False)		
t t t	top_ top_ top_ top_ top_	keepers_str keepers_run keepers_mat keepers_cat keepers_run	rage = to ike_rate s = top_k ches = to ches = to outs = to	p_keepers.: = top_keepe eepers.sori p_keepers.: p_keepers.:	sort_values('Avers.sort_values c_values('Runs cort_values('Mares	verage', s('Strik ', ascer atches_F atches', un_outs	, ascendi ke_Rate', nding=Fal Played', , ascendi ', ascend	ing=False) , ascending lse) ascending ing=False) ding=False)	= False)		
t t t t	top_ top_ top_ top_ top_ top_	keepers_str keepers_run keepers_mat keepers_cat keepers_run	rage = to ike_rate s = top_k ches = to outs = to mps = top	p_keepers.: = top_keepe eepers.sori p_keepers.: p_keepers.:	sort_values('Avers.sort_values'values('Runs.sort_values('Masort_values('Casort_values('Russort_values('Russort_values('Russort_values('Russort_values('Russort_values('Russort_values('Russort_values('Russort_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values')'.	verage', s('Strik ', ascer atches_F atches', un_outs	, ascendi ke_Rate', nding=Fal Played', , ascendi ', ascend	ing=False) , ascending lse) ascending ing=False) ding=False)	= False)		
t t t t t t t t	top_ top_ top_ top_ top_ top_	keepers_str keepers_run keepers_mat keepers_cat keepers_run keepers_stu	rage = to ike_rate s = top_k ches = to outs = to mps = top	p_keepers.s = top_keepe eepers.sort p_keepers.s p_keepers.s p_keepers.s	sort_values('Avers.sort_values'values('Runs.sort_values('Masort_values('Casort_values('Russort_values('Russort_values('Russort_values('Russort_values('Russort_values('Russort_values('Russort_values('Russort_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values('Rusport_values')'.	verage', s('Strik ', ascer atches_f atches', un_outs'	, ascendi ke_Rate', nding=Fal Played', , ascendi ', ascend	ing=False) , ascending lse) ascending ing=False) ding=False) g=False)	= False)	nes Run_outs	Stumps
t t t t t t t t t t t t t t t t t t t	top_ top_ top_ top_ top_ top_	keepers_str keepers_run keepers_mat keepers_cat keepers_run keepers_stu	rage = to ike_rate s = top_k ches = to ches = to outs = to mps = top rage me Te	p_keepers.setepers.se	cort_values('Avers.sort_values'.cvalues('Runs.sort_values('Mas.sort_values('Mas.sort_values('Cort_values('Strot_va	verage', s('Strik ', ascer atches_f atches', un_outs'	, ascendi ke_Rate', nding=Fal Played', , ascendi ', ascending	ing=False) , ascending lse) ascending ing=False) ding=False) g=False)	= False)		Stumps 4.0
t t t t t t t t t t t t t t t t t t t	top_ top_ top_ top_ top_	keepers_str keepers_run keepers_mat keepers_cat keepers_run keepers_stu keepers_ave	rage = to ike_rate s = top_k ches = to ches = to outs = to mps = top  rage  me Te tow Pur	p_keepers.setepp	cort_values('Avers.sort_values', values('Runs.sort_values('Muns.sort_values('Mus.sort_values('Cort_values('Strot_v	verage'; s('Stril ', ascer atches_[ atches'; un_outs umps', a	, ascendike_Rate', nding=Fal Played', , ascendi ', ascendi ascending uns Avera	ing=False) , ascending lse) ascending ing=False) ding=False) g=False)  age Strike_R	= False)  Late Catcl	nes Run_outs	
t t t t t t t t t t t t t t t t t t t	top_ top_ top_ top_ top_ top_	keepers_str keepers_run keepers_mat keepers_cat keepers_run keepers_stu keepers_ave Player Na	rage = to ike_rate s = top_k ches = to outs = to outs = to prage  me	p_keepers.s = top_keepers.sori p_keepers.s p_keepers.s _keepers.s am Nationa	cort_values('Aners.sort_values', values('Runs.sort_values('Muns.sort_values('Mus.sort_values('Cort_values('Strott_values('Stro	verage', s('Stril ', ascer atches_f atches', un_outs' umps', a	, ascendike_Rate', nding=Fal Played', ascending', ascending	ing=False) , ascending lse) ascending ing=False) ding=False) g=False)  age Strike_R 1.50 142	= False)  Eate Catcl 2.20 1 5.80 12	nes Run_outs 8.0 1.0	4.0
t t t t t t t t t t t t t t t t t t t	top_ top_ top_ top_ top_ top_	keepers_str keepers_run keepers_mat keepers_cat keepers_run keepers_stu keepers_ave Player Na Jonny Bairs	rage = to ike_rate s = top_k ches = to outs = to mps = top  rage  me    Te tow    Pur oni    Cher arat    D	p_keepers.setepp_keepers.setepp_keepers.setepp	cort_values('Avers.sort_values('Runs.sort_values('Runs.sort_values('Mosort_values('Cort_values('Stro	verage', s('Stril ', ascer atches_f atches', un_outs' umps', a	, ascendike_Rate', nding=Fal Played', , ascending ', ascending ascending ascending 41 46.0 39	ing=False) , ascending lse) ascending ing=False) ding=False) g=False)  age Strike_R 1.50 142 0.50 133	= False)  Late Catcl 2.20 1 5.80 12	Run_outs 8.0 1.0 6.0 21.0	4.0
t t t t t t t t t t t t t t t t t t t	top_ top_ top_ top_ top_ top_ 108 213	keepers_str keepers_run keepers_mat keepers_cat keepers_run keepers_stu keepers_ave Player Na Jonny Bairs MS Dh	rage = to ike_rate s = top_k ches = to outs = to mps = top  rage  me    Te tow    Pur oni    Cher arat    D	p_keepers.setepp	cort_values('Avers.sort_values', 'Runs', values('Runs', values('Mus', values', 'Mus', values('Cort_values('Strott_values('Strott_values('Strott_values', values', val	verage', s('Strik', ascer atches_t atches', un_outs umps', a 28.0 103 20.0 474	, ascendike_Rate', nding=Fal Played', , ascendi ', ascendi ascending  41 46.0 39 41.0 38	ing=False) , ascending lse) ascending ing=False) ding=False) g=False)  age Strike_R 1.50 142 3.50 135 3.20 122 5.18 147	= False)  Late Catcl 2.20 1 5.80 12 2.40 7.46 5	Run_outs 8.0 1.0 6.0 21.0 4.0 0.0	4.0 39.0 1.0
t t t t t t t t t t t t t t t t t t t	top_ top_ top_ top_ top_ top_ top_ 108 213 111	keepers_str keepers_run keepers_mat keepers_cat keepers_run keepers_stu keepers_ave Player Na Jonny Bairs MS Dh KS Bha	rage = to ike_rate s = top_k ches = to outs = to mps = top  rage  me Te tow Pur oni Cher arat D tant D tler Rajast	p_keepers.setepp	sort_values('Avers.sort_values('Runs.sort_values('Runs.sort_values('Massort_values('Casort_values('Strot_values('S	verage', s('Stril ', ascer atches_F atches', un_outs' umps', a  28.0 103 20.0 474 8.0 19 344.0 249	, ascendike_Rate', nding=Fal Played', , ascending ', ascending ascending ascending 38.0 41 46.0 39 11.0 38	ing=False) , ascending lse) ascending ing=False) ding=False) g=False)  age Strike_R 0.50 142 0.50 135 0.51 142 0.51 142	= False)  Eate Catcl 2.20 1 5.80 12 2.40 7.46 5 0.00 3	Run_outs 8.0 1.0 6.0 21.0 4.0 0.0 6.0 5.0	4.0 39.0 1.0 14.0
t t t t t t t t t t t t t t t t t t t	top_ top_ top_ top_ top_ top_ top_ 108 213 111 206 219	keepers_str keepers_run keepers_mat keepers_cat keepers_stu keepers_stu keepers_ave Player Na Jonny Bairs MS Dh KS Bha Rishabh F	rage = to ike_rate s = top_k ches = to outs = to mps = top  rage  me	p_keepers.setepp	cort_values('Arers.sort_values ers.sort_values ers.sort_values ers.sort_values ers.sort_values('Matches') ers.sort_values('Creaters ers.sort_values('Streaters ers.sort_values('Streaters ers.sort_values('Streaters ers.sort_values('Streaters ers.sort_values('Streaters ers.sort_values('Streaters ers.sort_values('Streaters ers.sort_values('Streaters ers.sort_values('Streaters ers.sort_values('Arers ers.sort_values	verage'; s('Stril', asceratches_tatches'; un_outs' umps', a 28.0 103 20.0 474 8.0 19 834.0 249	, ascendike_Rate', nding=Fal Played', , ascending ', ascending ascending 38.0 41 46.0 39 91.0 38 98.0 35 68.0 35	ing=False) , ascending lse) ascending ing=False) ding=False) g=False)  age Strike_R 1.50 142 3.20 122 5.18 147 5.14 150 1.30 130	= False)  Late Catcl 2.20 1 5.80 12 2.40 7.46 5 0.00 3 0.90 5	Run_outs 8.0 1.0 6.0 21.0 4.0 0.0 6.0 5.0 4.0 3.0 3.0 0.0 8.0 12.0	4.0 39.0 1.0 14.0
0 t	top_ top_ top_ top_ top_ top_ 108 213 111 206 219	keepers_str keepers_run keepers_mat keepers_cat keepers_run keepers_stu keepers_ave Player Na Jonny Bairs MS Dh KS Bha Rishabh F Jos Bu	rage = to ike_rate s = top_k ches = to outs = to mps = top  rage  me    Te tow    Pur oni    Cher arat    D tant    D tant    D tant    Cher udu    Cher	p_keepers.se top_keepers.sorip	cort_values('Arers.sort_values('Runs.sort_values('Runs.sort_values('Misort_values('Misort_values('Cort_values('Strott_values('	verage', s('Strik', ascer atches_tatches', un_outs umps', a  28.0 103 20.0 474 8.0 19 84.0 249 655.0 196 77.0 225	, ascendike_Rate', nding=Fal Played', , ascending ', ascending ascending B8.0 41 B8.0 35 B8.0 35 B8.0 35 B8.0 35	ing=False) , ascending lse) ascending ing=False) ding=False)  age Strike_R 1.50 142 3.50 135 3.20 122 5.18 147 5.14 150 1.30 130 3.40 127	= False)  Late Catcl 2.20 1 5.80 12 2.40 7.46 5 0.00 3 0.90 5 7.50 5	Run_outs 8.0 1.0 6.0 21.0 4.0 0.0 6.0 5.0 4.0 3.0 3.0 0.0	4.0 39.0 1.0 14.0 1.0
t t t t t t t t t t t t t t t t t t t	top_ top_ top_ top_ top_ top_ 108 213 111 206 219 105	keepers_str keepers_run keepers_mat keepers_cat keepers_run keepers_stu keepers_ave Player Na Jonny Bairs MS Dh KS Bha Rishabh F Jos Bu Quinton De K	rage = to ike_rate s = top_k ches = to outs = to mps = top  rage  me	p_keepers.setepp_keepers.sort p_keepers.sort p_keepers.sp_keepers.setepp_keepers.setep	cort_values('Arers.sort_values crs.sort_values cryalues('Runs cort_values('Matches') cort_values('Creaters') cort_values('Streaters') cort_values('Arers') cor	verage', s('Stril', asceratches_fatches', un_outs', as week Ru 28.0 103 20.0 474 8.0 19 34.0 249 355.0 196577.0 225 75.0 391	, ascendike_Rate', nding=Fal Played', ascending ', ascending ascending ascending 38.0 41 36.0 39 36.0 35 36.0 35 36.0 35	ing=False) , ascending lse) ascending ing=False) ding=False) g=False)  age Strike_R 1.50 142 0.50 139 5.18 147 5.14 150 1.30 130 0.40 127	= False)  Eate Catcl 2.20 1 5.80 12 2.40 7.46 5 0.00 3 0.90 5 7.50 5 4.21 5	Run_outs 8.0 1.0 6.0 21.0 4.0 0.0 6.0 5.0 4.0 3.0 3.0 0.0 8.0 12.0	4.0 39.0 1.0 14.0 1.0 14.0 2.0

Out[271]:		Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Catches	Run_outs	Stumps
	219	Jos Butler	Rajasthan	Overseas	65.0	1968.0	35.14	150.00	34.0	3.0	1.0
	206	Rishabh Pant	Delhi	Indian	84.0	2498.0	35.18	147.46	56.0	5.0	14.0
	108	Jonny Bairstow	Punjab	Overseas	28.0	1038.0	41.50	142.20	18.0	1.0	4.0
	107	Ishan Kishan	Mumbai	Indian	61.0	1452.0	28.50	136.30	19.0	1.0	2.0
	213	MS Dhoni	Chennai	Indian	220.0	4746.0	39.50	135.80	126.0	21.0	39.0
	209	Sanju Samson	Rajasthan	Indian	121.0	3068.0	29.22	134.21	59.0	8.0	10.0
	105	Quinton De Kock	Lucknow	Overseas	77.0	2256.0	31.30	130.90	53.0	0.0	14.0
	109	Dinesh Karthik	Bangalore	Indian	213.0	4046.0	25.80	129.70	123.0	14.0	32.0
	106	Ambati Rayudu	Chennai	Indian	175.0	3916.0	29.40	127.50	58.0	12.0	2.0
	111	KS Bharat	Delhi	Indian	8.0	191.0	38.20	122.40	4.0	0.0	1.0
In [272	top_keepers_runs										
Out[272]:		Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Catches	Run_outs	Stumps
	213	MS Dhoni	Chennai	Indian	220.0	4746.0	39.50	135.80	126.0	21.0	39.0
	109	Dinesh Karthik	Bangalore	Indian	213.0	4046.0	25.80	129.70	123.0	14.0	32.0
	106	Ambati Rayudu	Chennai	Indian	175.0	3916.0	29.40	127.50	58.0	12.0	2.0
	209	Sanju Samson	Rajasthan	Indian	121.0	3068.0	29.22	134.21	59.0	8.0	10.0
	206	Rishabh Pant	Delhi	Indian	84.0	2498.0	35.18	147.46	56.0	5.0	14.0
	105	Quinton De Kock	Lucknow	Overseas	77.0	2256.0	31.30	130.90	53.0	0.0	14.0
	219	Jos Butler	Rajasthan	Overseas	65.0	1968.0	35.14	150.00	34.0	3.0	1.0
	107	Ishan Kishan	Mumbai	Indian	61.0	1452.0	28.50	136.30	19.0	1.0	2.0
	108	Jonny Bairstow	Punjab	Overseas	28.0	1038.0	41.50	142.20	18.0	1.0	4.0
	111	KS Bharat	Delhi	Indian	8.0	191.0	38.20	122.40	4.0	0.0	1.0
In [273	top_	_keepers_matche	S								
Out[273]:		Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Catches	Run_outs	Stumps
	213	MS Dhoni	Chennai	Indian	220.0	4746.0	39.50	135.80	126.0	21.0	39.0
	109	Dinesh Karthik	Bangalore	Indian	213.0	4046.0	25.80	129.70	123.0	14.0	32.0
	106	Ambati Rayudu	Chennai	Indian	175.0	3916.0	29.40	127.50	58.0	12.0	2.0
	209	Sanju Samson	Rajasthan	Indian	121.0	3068.0	29.22	134.21	59.0	8.0	10.0
	206	Rishabh Pant	Delhi	Indian	84.0	2498.0	35.18	147.46	56.0	5.0	14.0
	105	Quinton De Kock	Lucknow	Overseas	77.0	2256.0	31.30	130.90	53.0	0.0	14.0
	219	Jos Butler	Rajasthan	Overseas	65.0	1968.0	35.14	150.00	34.0	3.0	1.0
		Ishan Kishan	Mumbai	Indian	61.0	1452.0	28.50	136.30	19.0	1.0	2.0
	107										
	107 108	Jonny Bairstow	Punjab	Overseas	28.0	1038.0	41.50	142.20	18.0	1.0	4.0

top\_keepers\_catches

In [274...

Out[274]:		Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Catches	Run_outs	Stumps
	213	MS Dhoni	Chennai	Indian	220.0	4746.0	39.50	135.80	126.0	21.0	39.0
	109	Dinesh Karthik	Bangalore	Indian	213.0	4046.0	25.80	129.70	123.0	14.0	32.0
	209	Sanju Samson	Rajasthan	Indian	121.0	3068.0	29.22	134.21	59.0	8.0	10.0
	106	Ambati Rayudu	Chennai	Indian	175.0	3916.0	29.40	127.50	58.0	12.0	2.0
	206	Rishabh Pant	Delhi	Indian	84.0	2498.0	35.18	147.46	56.0	5.0	14.0
	105	Quinton De Kock	Lucknow	Overseas	77.0	2256.0	31.30	130.90	53.0	0.0	14.0
	219	Jos Butler	Rajasthan	Overseas	65.0	1968.0	35.14	150.00	34.0	3.0	1.0
	107	Ishan Kishan	Mumbai	Indian	61.0	1452.0	28.50	136.30	19.0	1.0	2.0
	108	Jonny Bairstow	Punjab	Overseas	28.0	1038.0	41.50	142.20	18.0	1.0	4.0
	111	KS Bharat	Delhi	Indian	8.0	191.0	38.20	122.40	4.0	0.0	1.0

In [275... top\_keepers\_runouts

Out[275]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Catches	Run_outs	Stumps
213	MS Dhoni	Chennai	Indian	220.0	4746.0	39.50	135.80	126.0	21.0	39.0
109	Dinesh Karthik	Bangalore	Indian	213.0	4046.0	25.80	129.70	123.0	14.0	32.0
106	Ambati Rayudu	Chennai	Indian	175.0	3916.0	29.40	127.50	58.0	12.0	2.0
209	Sanju Samson	Rajasthan	Indian	121.0	3068.0	29.22	134.21	59.0	8.0	10.0
206	Rishabh Pant	Delhi	Indian	84.0	2498.0	35.18	147.46	56.0	5.0	14.0
219	Jos Butler	Rajasthan	Overseas	65.0	1968.0	35.14	150.00	34.0	3.0	1.0
107	Ishan Kishan	Mumbai	Indian	61.0	1452.0	28.50	136.30	19.0	1.0	2.0
108	Jonny Bairstow	Punjab	Overseas	28.0	1038.0	41.50	142.20	18.0	1.0	4.0
<b>105</b> Q	Quinton De Kock	Lucknow	Overseas	77.0	2256.0	31.30	130.90	53.0	0.0	14.0
111	KS Bharat	Delhi	Indian	8.0	191.0	38.20	122.40	4.0	0.0	1.0

In [276... top\_keepers\_stumps

Out[276]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Catches	Run_outs	Stumps
213	MS Dhoni	Chennai	Indian	220.0	4746.0	39.50	135.80	126.0	21.0	39.0
109	Dinesh Karthik	Bangalore	Indian	213.0	4046.0	25.80	129.70	123.0	14.0	32.0
105	Quinton De Kock	Lucknow	Overseas	77.0	2256.0	31.30	130.90	53.0	0.0	14.0
206	Rishabh Pant	Delhi	Indian	84.0	2498.0	35.18	147.46	56.0	5.0	14.0
209	Sanju Samson	Rajasthan	Indian	121.0	3068.0	29.22	134.21	59.0	8.0	10.0
108	Jonny Bairstow	Punjab	Overseas	28.0	1038.0	41.50	142.20	18.0	1.0	4.0
106	Ambati Rayudu	Chennai	Indian	175.0	3916.0	29.40	127.50	58.0	12.0	2.0
107	Ishan Kishan	Mumbai	Indian	61.0	1452.0	28.50	136.30	19.0	1.0	2.0
111	KS Bharat	Delhi	Indian	8.0	191.0	38.20	122.40	4.0	0.0	1.0
219	Jos Butler	Rajasthan	Overseas	65.0	1968.0	35.14	150.00	34.0	3.0	1.0

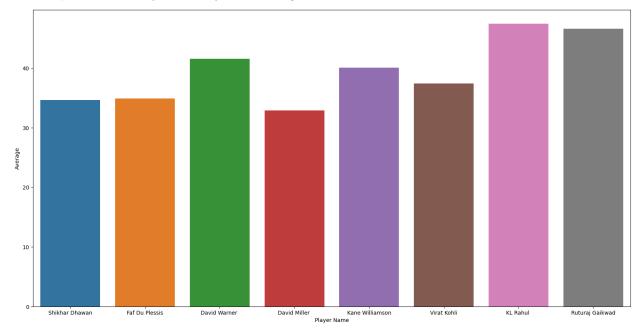
if we rank the keepers in order of 1-10 on the above parameters. The top 3 Keepers will be

- 1. MS Dhoni
- 2. Dinesh Kartik
- 3. Rishabh Pant

## **Visualisation for Enhanced Analysis**

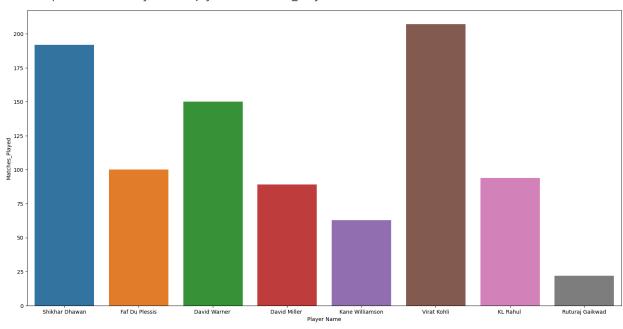
```
In [277...
            #Visualisation of Batters Data
            #the plot shows top batters strike rate
            plt.figure(figsize=(20, 10))
            sns.barplot(x ='Player Name', y='Strike_Rate', data= top_batters)
Out[277]: <AxesSubplot:xlabel='Player Name', ylabel='Strike_Rate'>
              120
                                                                           Kane Williamson
Player Name
                                                                                                   Virat Kohli
                    Shikhar Dhawan
                                    Faf Du Plessis
                                                   David Warner
                                                                   David Miller
                                                                                                                   KL Rahul
                                                                                                                                 Ruturaj Gaikwad
            #this plot shows top batters runs
In [278...
            plt.figure(figsize=(20,10))
            sns.barplot(x='Player Name', y='Runs', data=top_batters)
Out[278]: <AxesSubplot:xlabel='Player Name', ylabel='Runs'>
              6000
              3000
              1000
                                                                           Kane Williamson
Player Name
                                     Faf Du Plessis
                                                    David Warner
                                                                    David Miller
                                                                                                   Virat Kohli
                                                                                                                   KL Rahul
                                                                                                                                 Ruturaj Gaikwad
In [279...
            #this plot shows top batters average
            plt.figure(figsize=(20,10))
            sns.barplot(x='Player Name', y='Average', data=top_batters)
```

Out[279]: <AxesSubplot:xlabel='Player Name', ylabel='Average'>



```
In [280... #this plot shows top batters matches played
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Matches_Played', data=top_batters)
```

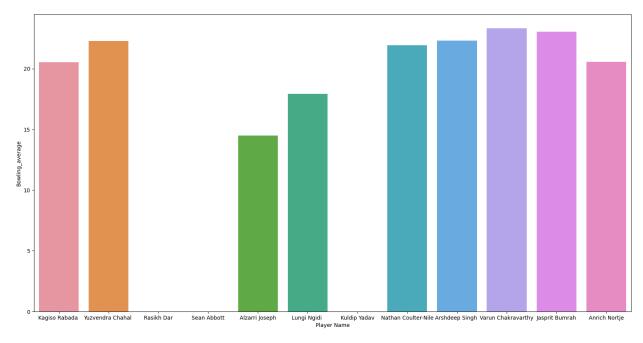
Out[280]: <AxesSubplot:xlabel='Player Name', ylabel='Matches\_Played'>



```
#Visualising the bowlers data
#this plot shows the bowling average of top bowlers

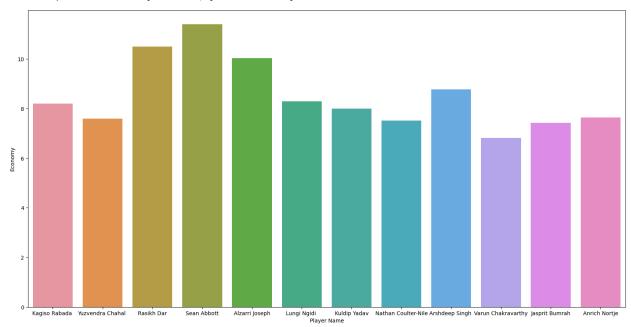
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Bowling_average', data=top_bowlers)
```

Out[281]: <AxesSubplot:xlabel='Player Name', ylabel='Bowling\_average'>



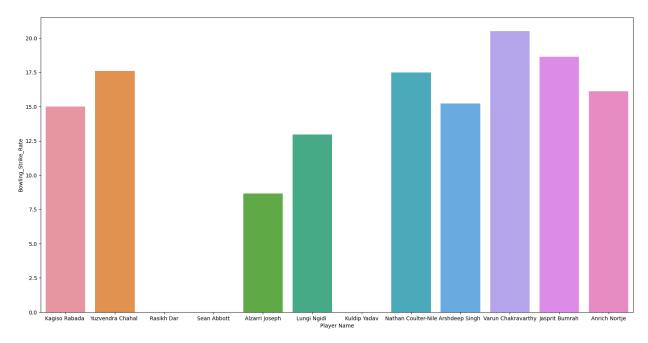
```
In [282... #this plot shows the economy of each of the bowlers
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Economy', data=top_bowlers)
```

Out[282]: <AxesSubplot:xlabel='Player Name', ylabel='Economy'>



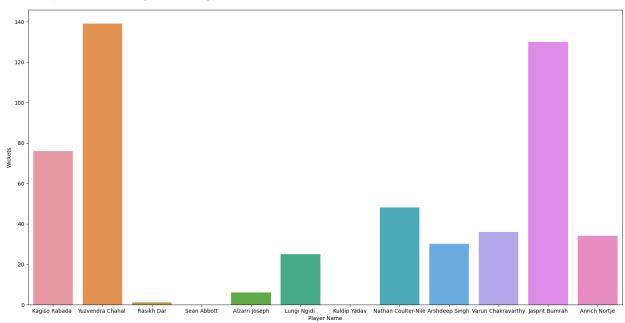
```
In [284... #this plot shows the bowling strike rate of the bowlers
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Bowling_Strike_Rate', data=top_bowlers)
```

Out[284]: <AxesSubplot:xlabel='Player Name', ylabel='Bowling\_Strike\_Rate'>



```
In [285... #this plot shows the wickets taken by each of bowlers
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Wickets', data=top_bowlers)
```

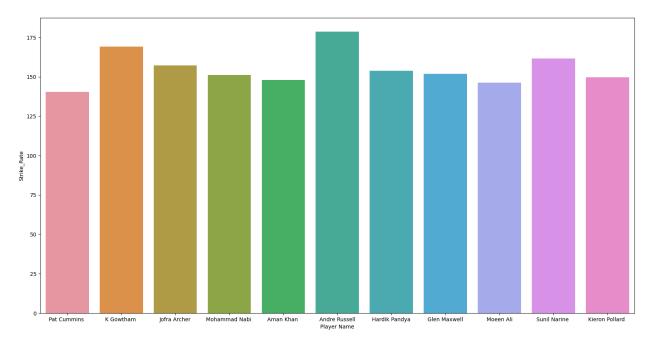
Out[285]: <AxesSubplot:xlabel='Player Name', ylabel='Wickets'>



```
#Visualising allrounders data
#this plot shows the strike rate of top allrounders

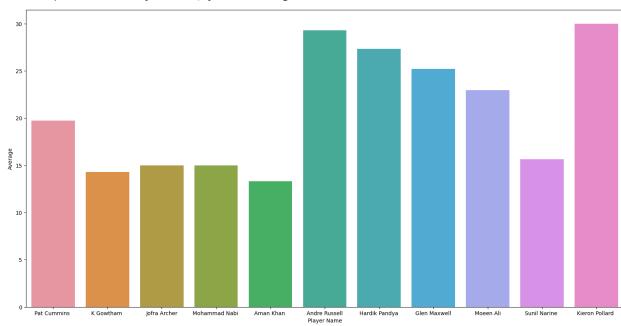
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Strike_Rate', data=top_allrounders)
```

Out[286]: <AxesSubplot:xlabel='Player Name', ylabel='Strike\_Rate'>



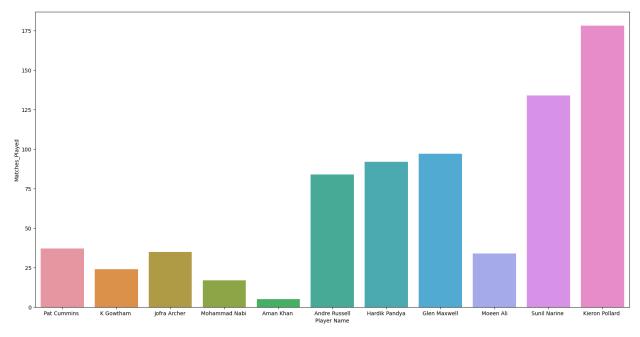
```
In [287... plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Average', data=top_allrounders)
```

Out[287]: <AxesSubplot:xlabel='Player Name', ylabel='Average'>



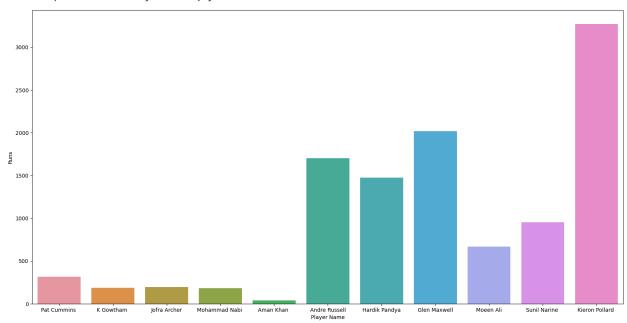
```
In [288... plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Matches_Played', data=top_allrounders)
```

Out[288]: <AxesSubplot:xlabel='Player Name', ylabel='Matches\_Played'>



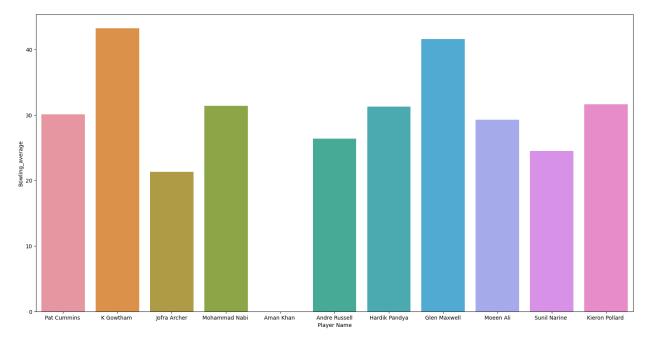
```
In [289... #this plot shows the top allrounder's runs
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Runs', data=top_allrounders)
```

Out[289]: <AxesSubplot:xlabel='Player Name', ylabel='Runs'>



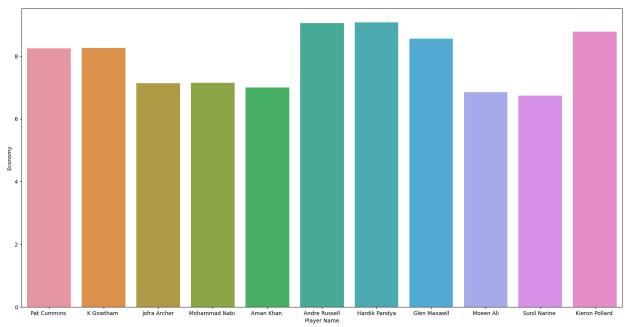
```
In [290... plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Bowling_average', data=top_allrounders)
```

Out[290]: <AxesSubplot:xlabel='Player Name', ylabel='Bowling\_average'>



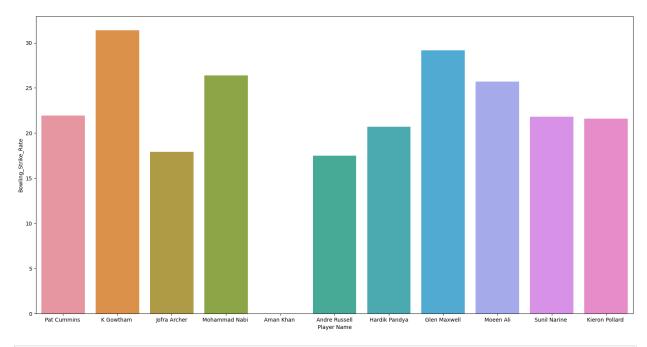
```
In [291... plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Economy', data=top_allrounders)
```

Out[291]: <AxesSubplot:xlabel='Player Name', ylabel='Economy'>



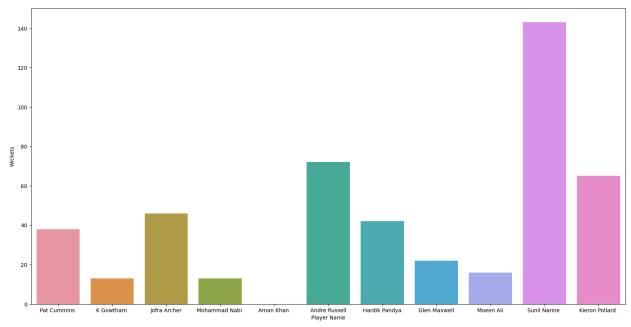
```
In [292...
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Bowling_Strike_Rate', data=top_allrounders)
```

Out[292]: <AxesSubplot:xlabel='Player Name', ylabel='Bowling\_Strike\_Rate'>



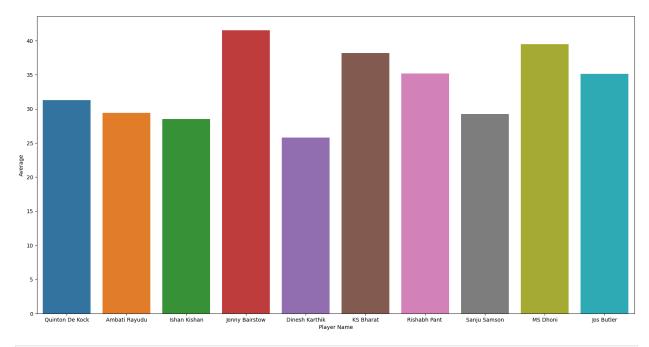
```
In [293... plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Wickets', data=top_allrounders)
```

Out[293]: <AxesSubplot:xlabel='Player Name', ylabel='Wickets'>



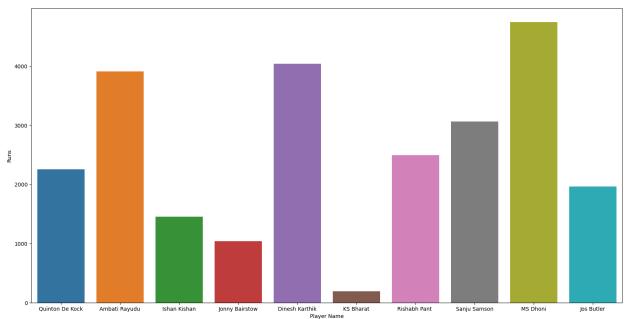
```
In [294... #Visualising Keepers data
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Average', data=top_keepers)
```

Out[294]: <AxesSubplot:xlabel='Player Name', ylabel='Average'>



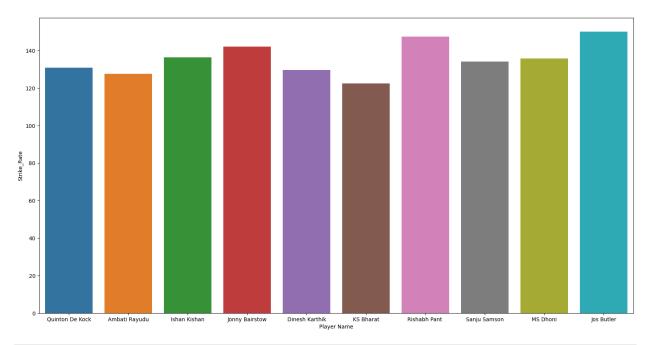
```
In [295... plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Runs', data=top_keepers)
```

Out[295]: <AxesSubplot:xlabel='Player Name', ylabel='Runs'>



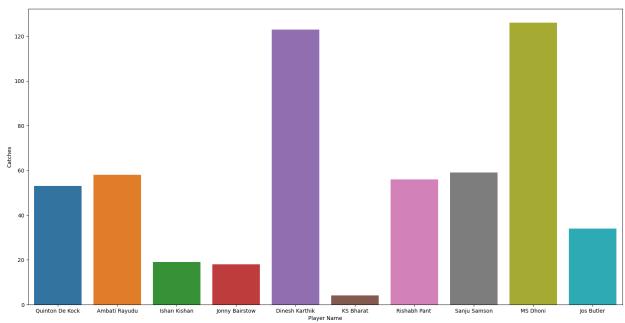
```
In [296...
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Strike_Rate', data=top_keepers)
```

Out[296]: <AxesSubplot:xlabel='Player Name', ylabel='Strike\_Rate'>



```
In [297... plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Catches', data=top_keepers)
```

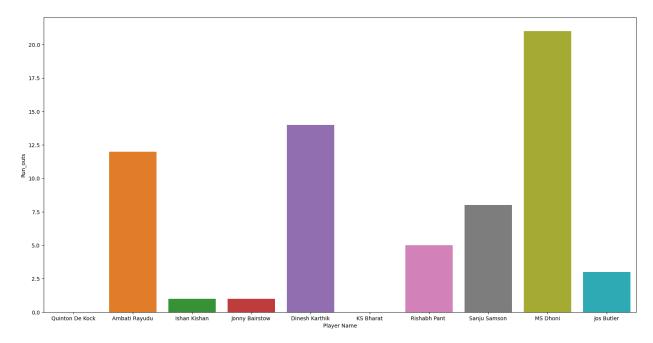
Out[297]: <AxesSubplot:xlabel='Player Name', ylabel='Catches'>



```
In [298... plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Run_outs', data=top_keepers)
```

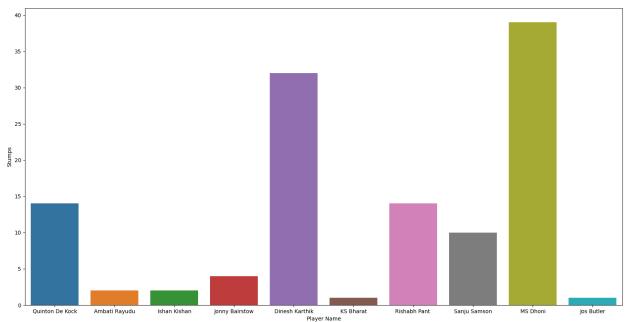
Out[298]: <AxesSubplot:xlabel='Player Name', ylabel='Run\_outs'>

8/4/24, 11:30 PM IPL Data Analysis



```
In [299... plt.figure(figsize=(20,10))
sns.barplot(x = 'Player Name', y='Stumps', data=top_keepers)
```

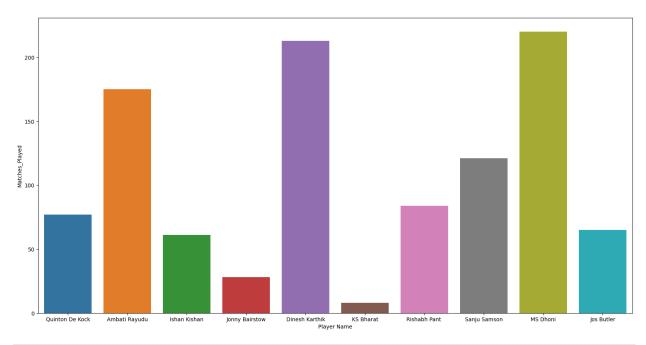
Out[299]: <AxesSubplot:xlabel='Player Name', ylabel='Stumps'>



```
In [300... plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Matches_Played', data=top_keepers)
```

Out[300]: <AxesSubplot:xlabel='Player Name', ylabel='Matches\_Played'>

8/4/24, 11:30 PM IPL Data Analysis



In [ ]