IPL Data Analysis

In [182…
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Loading the data

In [183…
```
data = pd.read_csv("Downloads/IPLData.csv")
```

In [184…
```
data.head()
```

Out[184]:

	Player Name	Team	Nationality	Player_Type	Capped	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_averag
0	Shikhar Dhawan	Punjab	Indian	Batter	1	192.0	5783.0	34.63	126.60	4.0	16
1	Shreyas Iyer	Kolkata	Indian	Batter	1	87.0	2375.0	31.67	123.96	NaN	Na
2	Faf Du Plessis	Bangalore	Overseas	Batter	1	100.0	2935.0	34.94	131.09	NaN	Na
3	Manish Pandey	Lucknow	Indian	Batter	1	154.0	3560.0	30.69	121.83	NaN	Na
4	Shimron Hetmyer	Rajasthan	Overseas	Batter	1	31.0	517.0	25.85	151.17	NaN	Na

In [185…
```
data.describe()
```

Out[185]:

	Capped	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Bowling_Strik
count	235.000000	215.000000	165.000000	161.000000	163.000000	140.000000	135.000000	143.000000	119
mean	0.838298	43.897674	840.575758	21.792391	121.009939	31.485714	32.907185	8.223182	24
std	0.561802	48.695302	1270.341831	11.664156	30.739189	36.872420	18.191441	1.223541	12
min	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	5.360000	0
25%	0.500000	11.500000	67.000000	13.800000	112.635000	6.000000	23.025000	7.390000	18
50%	1.000000	25.000000	289.000000	22.410000	128.630000	19.500000	29.070000	8.190000	21
75%	1.000000	56.000000	954.000000	29.300000	137.550000	40.500000	36.030000	8.785000	26
max	2.000000	220.000000	6283.000000	58.500000	190.240000	167.000000	153.000000	13.120000	108

In [186…
```
data.isna().sum()
```

Out[186]:
```
Player Name              0
Team                     0
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
Run_outs               208
Stumps                 208
dtype: int64
```

In [187…
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 235 entries, 0 to 234
Data columns (total 16 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   Player Name          235 non-null    object 
 1   Team                 235 non-null    object 
 2   Nationality          235 non-null    object 
 3   Player_Type          235 non-null    object 
 4   Capped               235 non-null    int64  
 5   Matches_Played       215 non-null    float64
 6   Runs                 165 non-null    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
 11  Economy              143 non-null    float64
 12  Bowling_Strike_Rate  119 non-null    float64
 13  Catches              27 non-null     float64
 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',
                              'Team',
                              'Nationality',
                              'Matches_Played',
                              'Runs',
                              'Average',
                              'Strike_Rate']]
```

In [189…
```
Capped_Batters.head()
```

Out[189]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate
0	Shikhar Dhawan	Punjab	Indian	192.0	5783.0	34.63	126.60
1	Shreyas Iyer	Kolkata	Indian	87.0	2375.0	31.67	123.96
2	Faf Du Plessis	Bangalore	Overseas	100.0	2935.0	34.94	131.09
3	Manish Pandey	Lucknow	Indian	154.0	3560.0	30.69	121.83
4	Shimron Hetmyer	Rajasthan	Overseas	31.0	517.0	25.85	151.17

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',
                              'Team',
                              'Nationality',
                              'Matches_Played',
                              'Wickets',
                              'Bowling_average',
                              'Economy',
                              'Bowling_Strike_Rate']]
```

In [242…

```
Capped_Bowlers.head()
```

Out[242]:

	Player Name	Team	Nationality	Matches_Played	Wickets	Bowling_average	Economy	Bowling_Strike_Rate
36	Kagiso Rabada	Punjab	Overseas	50.0	76.0	20.53	8.21	15.00
37	Trent Boult	Rajasthan	Overseas	62.0	76.0	26.09	8.40	18.64
38	Mohammad Shami	Gujarat	Indian	77.0	79.0	30.41	8.63	21.14
39	T Natarajan	Hyderabad	Indian	24.0	20.0	34.40	8.24	25.05
40	Deepak Chahar	Chennai	Indian	63.0	59.0	29.19	7.80	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	Ambati Rayudu	Chennai	Indian	175.0	3916.0	29.4	127.5	58.0	12.0	2.0
107	Ishan Kishan	Mumbai	Indian	61.0	1452.0	28.5	136.3	19.0	1.0	2.0
108	Jonny Bairstow	Punjab	Overseas	28.0	1038.0	41.5	142.2	18.0	1.0	4.0
109	Dinesh Karthik	Bangalore	Indian	213.0	4046.0	25.8	129.7	123.0	14.0	32.0

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 Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	B
127	Ravichandran Ashwin	Rajasthan	Indian	167.0	456.0	11.12	109.88	145.0	27.80	6.91	
128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.13	8.24	
129	Dwayne Bravo	Chennai	Overseas	151.0	1537.0	22.94	130.25	167.0	24.32	8.36	
130	Nitish Rana	Kolkata	Indian	77.0	1820.0	28.00	132.46	7.0	22.00	8.03	
131	Jason Holder	Lucknow	Overseas	26.0	189.0	14.54	121.15	35.0	22.46	8.20	

In [247…
```
# Cleaning the data by making the null or NAN values 0

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

```
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

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	Shikhar Dhawan	Punjab	Indian	192.0	5783.0	34.63	126.60
208	Virat Kohli	Bangalore	Indian	207.0	6283.0	37.40	129.95

In [254…
```
# 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)
```

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
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
210	Arshdeep Singh	Punjab	Indian	23.0	30.0	22.30	8.78	15.23
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
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 [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
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	Bo
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	Bo
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	Bo
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	Bo
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	Bo
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	
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	
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	
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
```

Out[266]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Bo
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	

In [267…

```
top_allrounders_wickets
```

Out[267]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Bo
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
```

Out[268]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Wickets	Bowling_average	Economy	Bo
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	
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	
195	Mohammad Nabi	Kolkata	Overseas	17.0	180.0	15.00	151.26	13.0	31.38	7.14	
128	Pat Cummins	Kolkata	Overseas	37.0	316.0	19.75	140.44	38.0	30.13	8.24	
154	K Gowtham	Lucknow	Indian	24.0	186.0	14.31	169.09	13.0	43.23	8.26	
218	Glen Maxwell	Bangalore	Overseas	97.0	2018.0	25.23	151.84	22.0	41.59	8.55	
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	

In [269…

```
# Analysing the keepers data
top_keepers = Capped_Keepers.loc[(Capped_Keepers["Average"] >= 25.0)]

top_keepers_average = top_keepers.sort_values('Average', ascending=False)
top_keepers_strike_rate = top_keepers.sort_values('Strike_Rate', ascending=False)
top_keepers_runs = top_keepers.sort_values('Runs', ascending=False)
top_keepers_matches = top_keepers.sort_values('Matches_Played', ascending = False)
top_keepers_catches = top_keepers.sort_values('Catches', ascending=False)
top_keepers_runouts = top_keepers.sort_values('Run_outs', ascending=False)
top_keepers_stumps = top_keepers.sort_values('Stumps', ascending=False)
```

In [270…

```
top_keepers_average
```

Out[270]:

	Player Name	Team	Nationality	Matches_Played	Runs	Average	Strike_Rate	Catches	Run_outs	Stumps
108	Jonny Bairstow	Punjab	Overseas	28.0	1038.0	41.50	142.20	18.0	1.0	4.0
213	MS Dhoni	Chennai	Indian	220.0	4746.0	39.50	135.80	126.0	21.0	39.0
111	KS Bharat	Delhi	Indian	8.0	191.0	38.20	122.40	4.0	0.0	1.0
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
105	Quinton De Kock	Lucknow	Overseas	77.0	2256.0	31.30	130.90	53.0	0.0	14.0
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
107	Ishan Kishan	Mumbai	Indian	61.0	1452.0	28.50	136.30	19.0	1.0	2.0
109	Dinesh Karthik	Bangalore	Indian	213.0	4046.0	25.80	129.70	123.0	14.0	32.0

In [271…

```
top_keepers_strike_rate
```

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_matches
```

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
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 [274…

```
top_keepers_catches
```

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
105	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

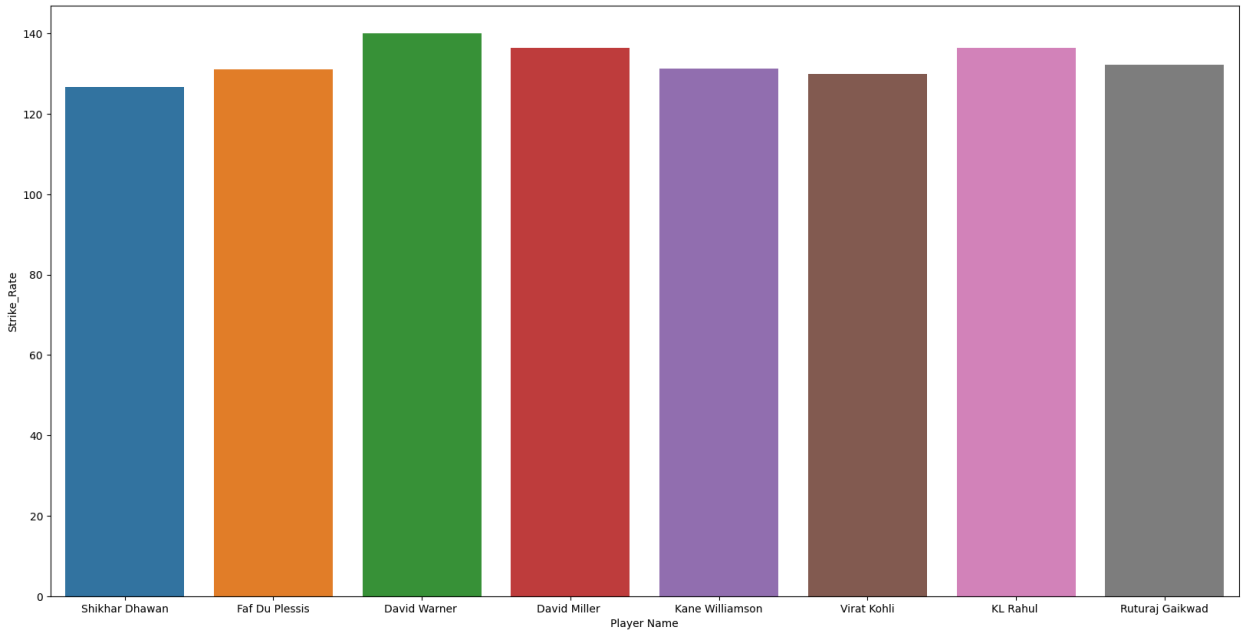
if we rank the keepers in order of 1-10 on the above parameters. The top 3 Keepers will be

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

Visualisation for Enhanced Analysis

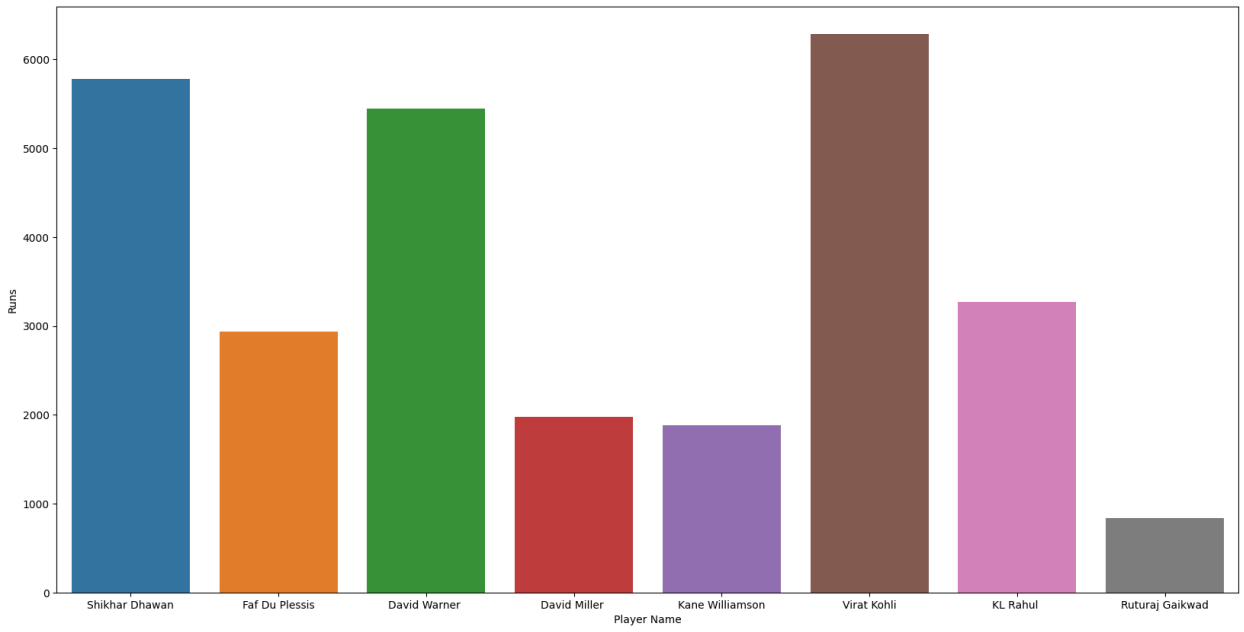
In [277…
```
#Visualisation of Batters Data
#the plot shows top batters strike rate

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'>



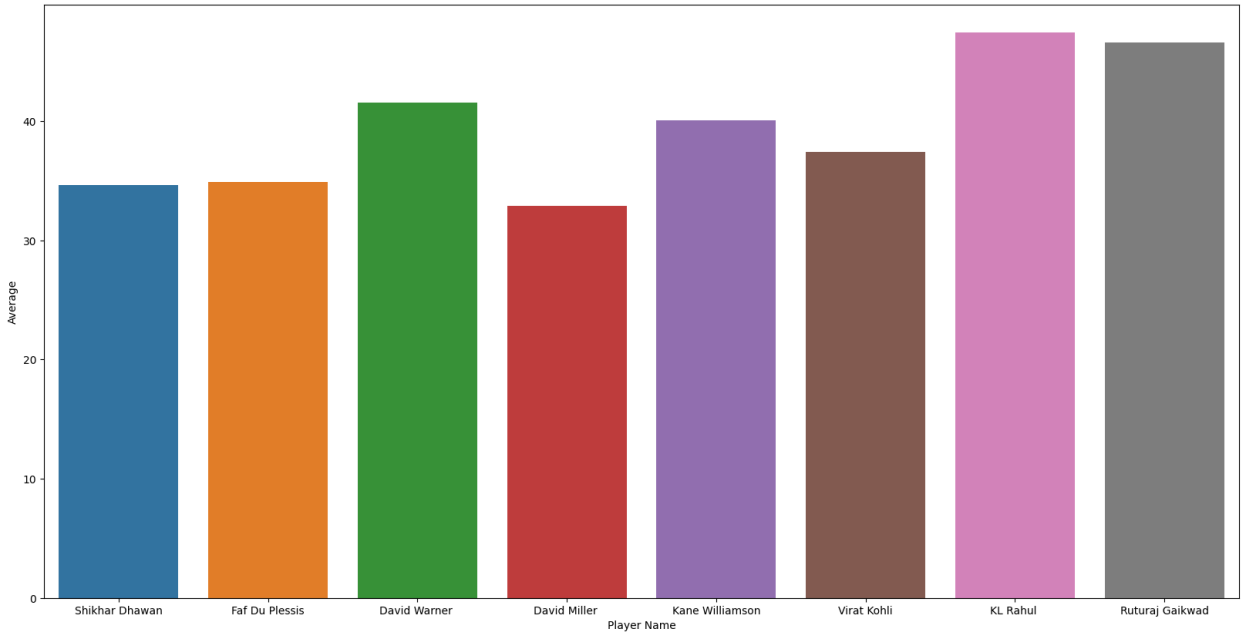
In [278…
```
#this plot shows top batters runs
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Runs', data=top_batters)
```

Out[278]: <AxesSubplot:xlabel='Player Name', ylabel='Runs'>



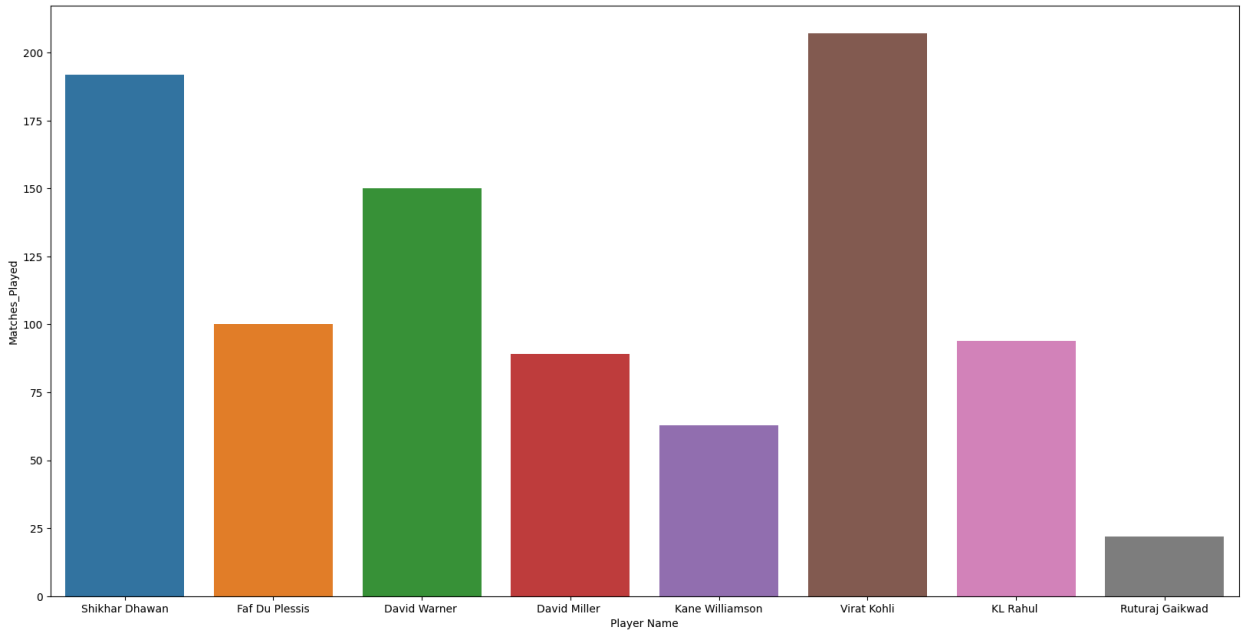
In [279…
```
#this plot shows top batters average
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Average', data=top_batters)
```

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



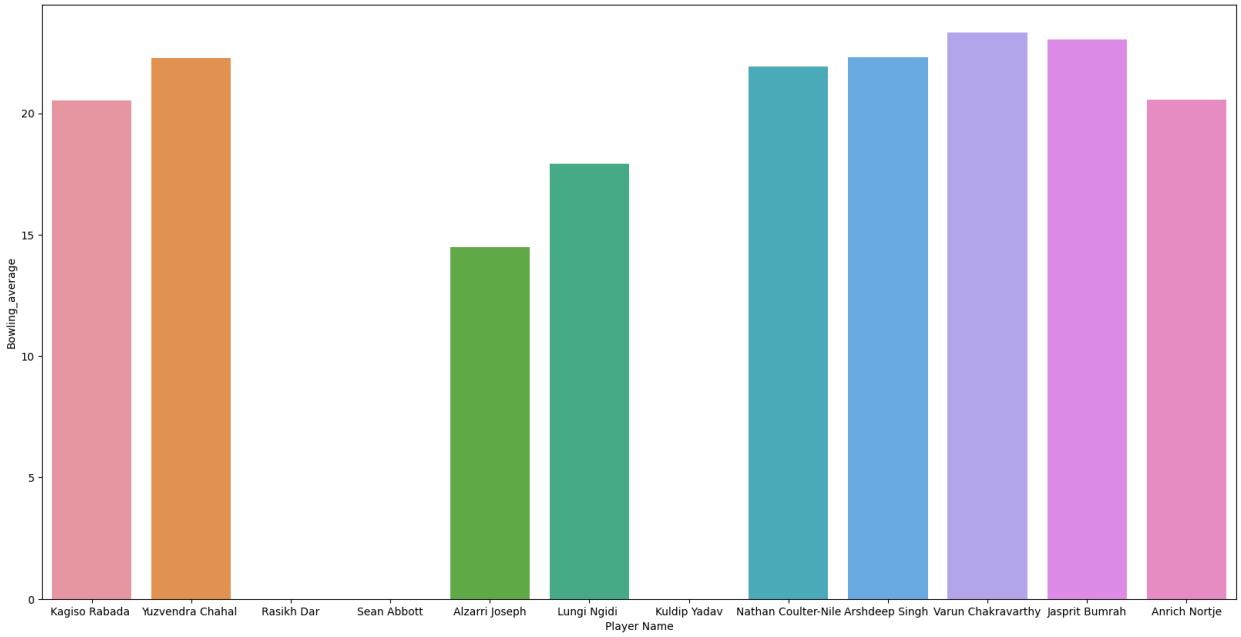
In [280…
```
#this plot shows top batters matches played
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Matches_Played', data=top_batters)
```

Out[280]: <AxesSubplot:xlabel='Player Name', ylabel='Matches_Played'>



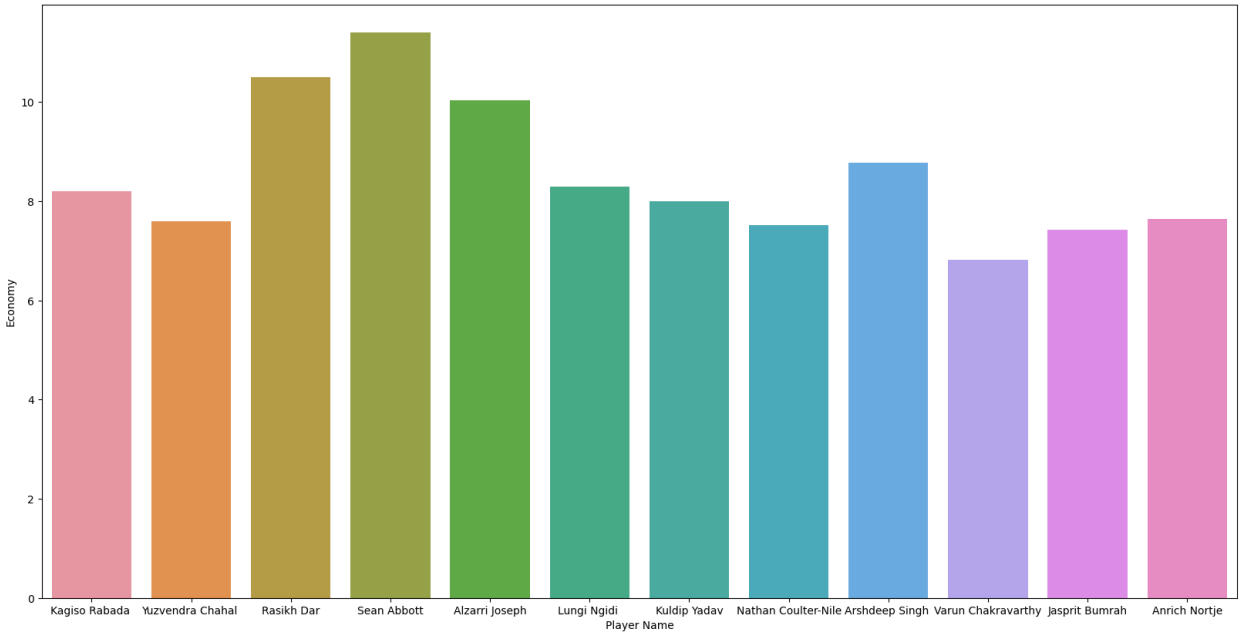
In [281…
```
#Visualising the bowlers data
#this plot shows the bowling average of top bowlers

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'>



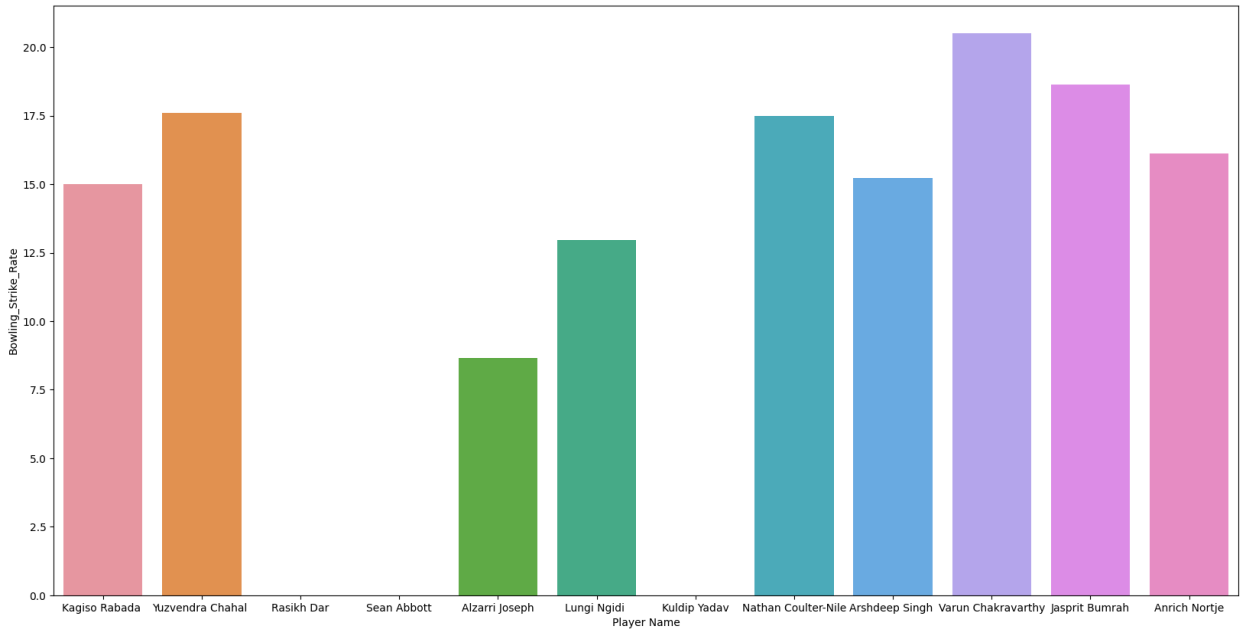
In [282…
```
#this plot shows the economy of each of the bowlers
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Economy', data=top_bowlers)
```

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



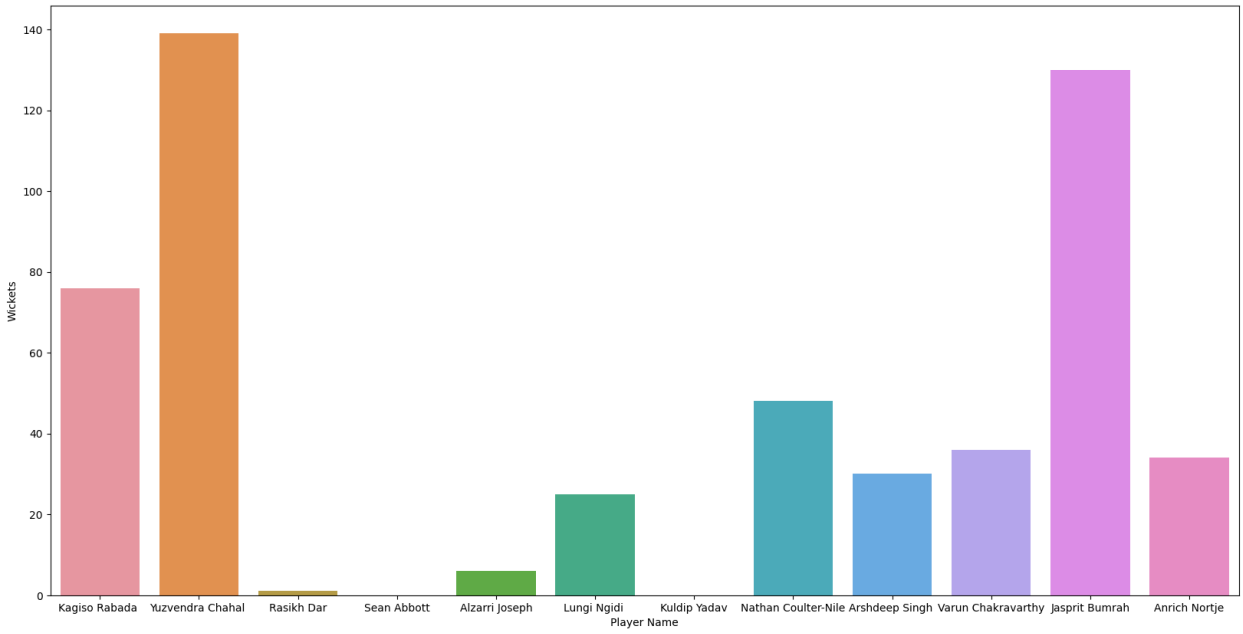
In [284…
```
#this plot shows the bowling strike rate of the bowlers
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Bowling_Strike_Rate', data=top_bowlers)
```

Out[284]: <AxesSubplot:xlabel='Player Name', ylabel='Bowling_Strike_Rate'>



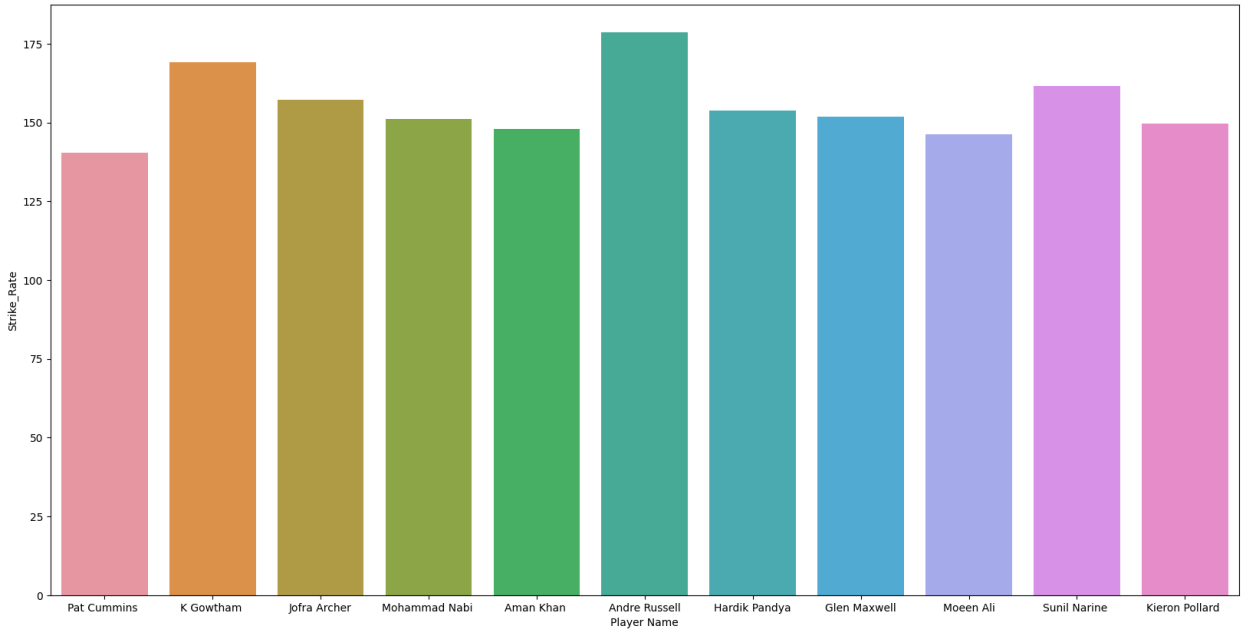
In [285…
```
#this plot shows the wickets taken by each of bowlers
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Wickets', data=top_bowlers)
```

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



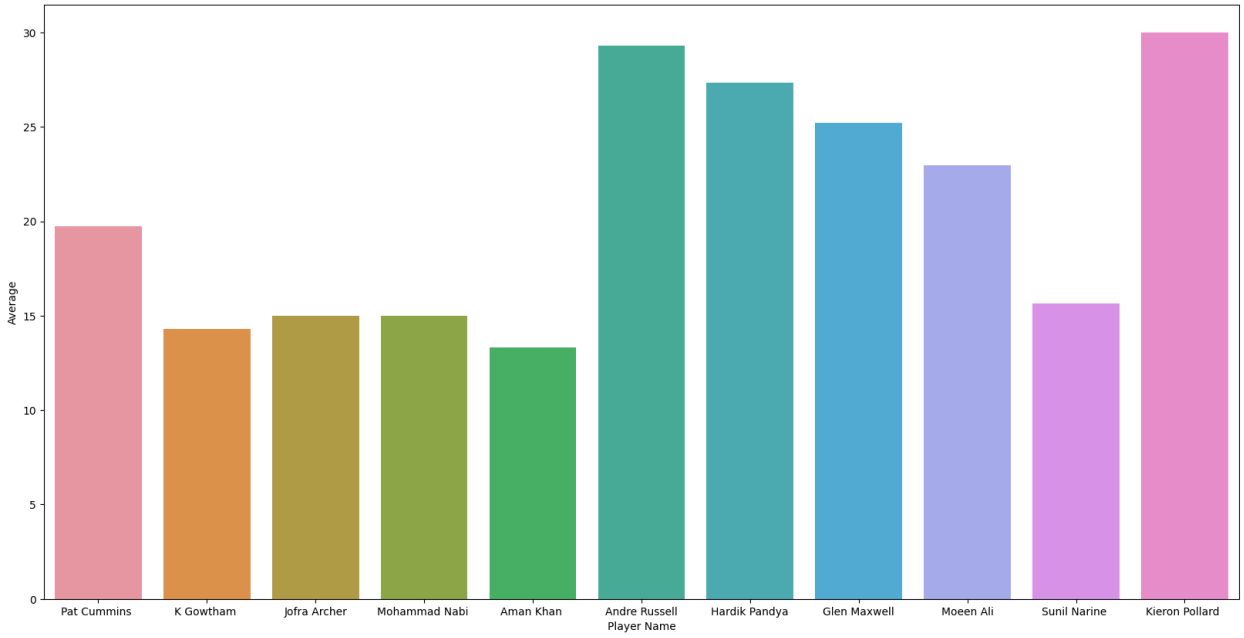
In [286…
```
#Visualising allrounders data
#this plot shows the strike rate of top allrounders

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'>



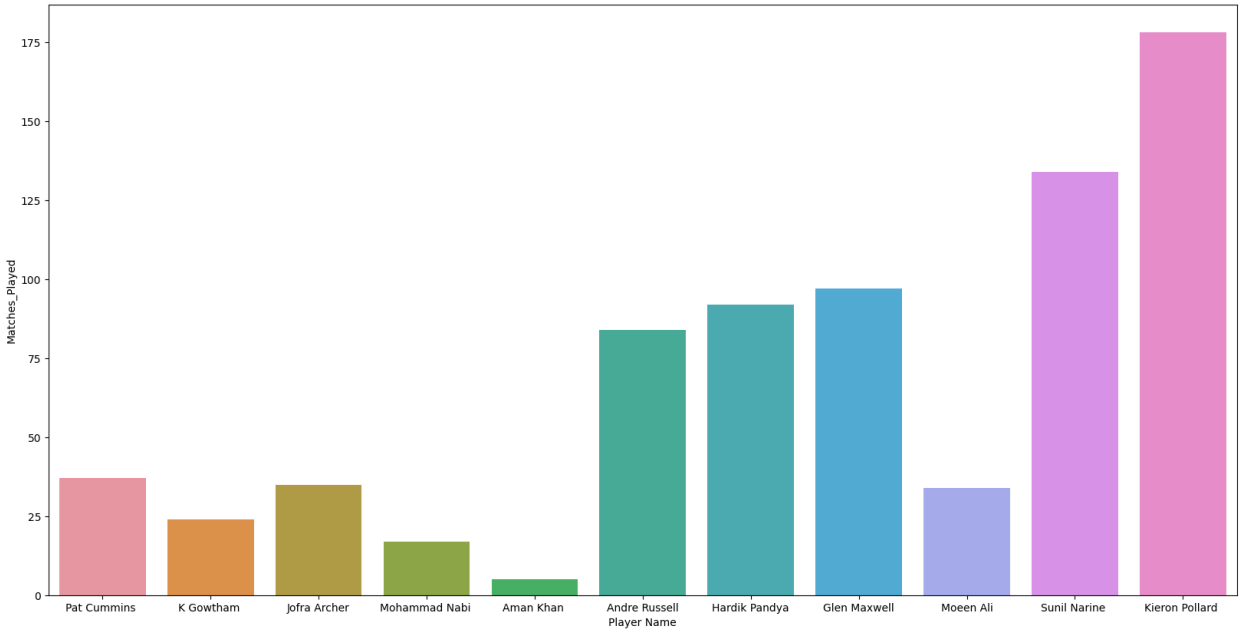
In [287…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Average', data=top_allrounders)
```

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



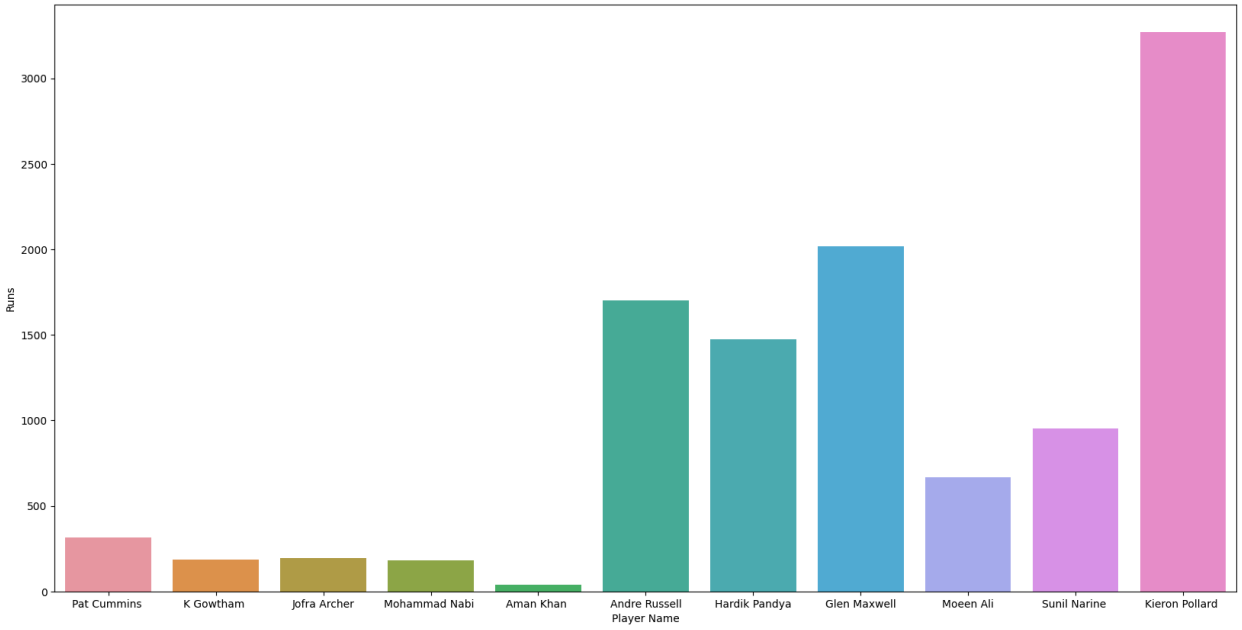
In [288…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Matches_Played', data=top_allrounders)
```

Out[288]: <AxesSubplot:xlabel='Player Name', ylabel='Matches_Played'>



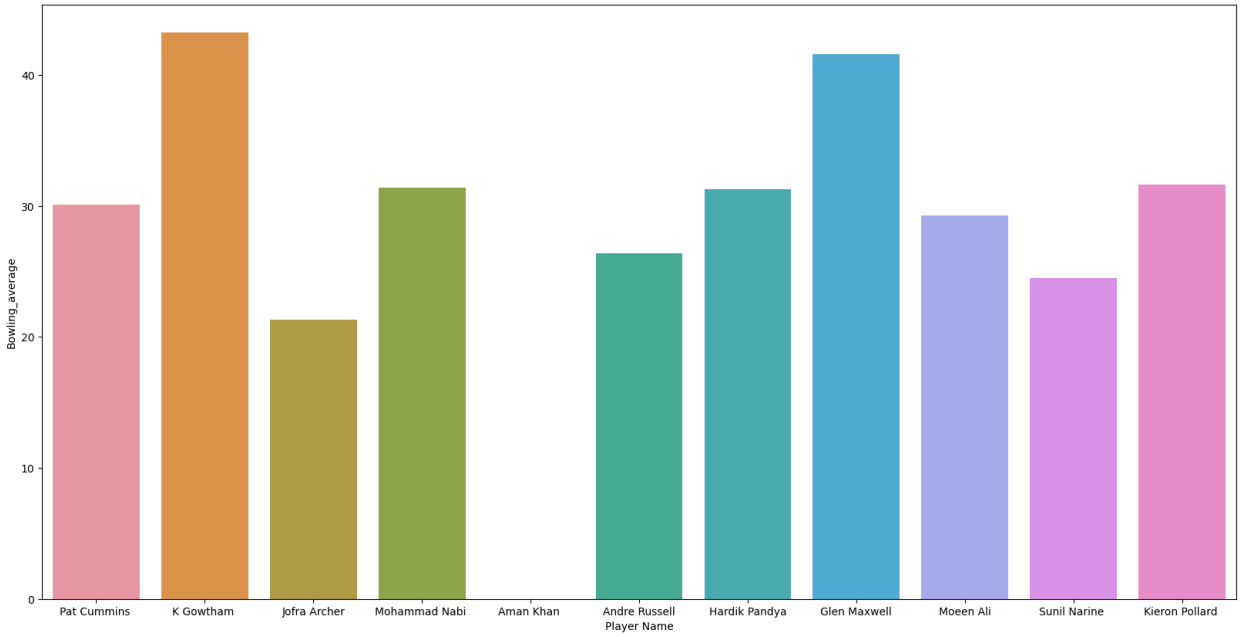
In [289…
```
#this plot shows the top allrounder's runs
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Runs', data=top_allrounders)
```

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



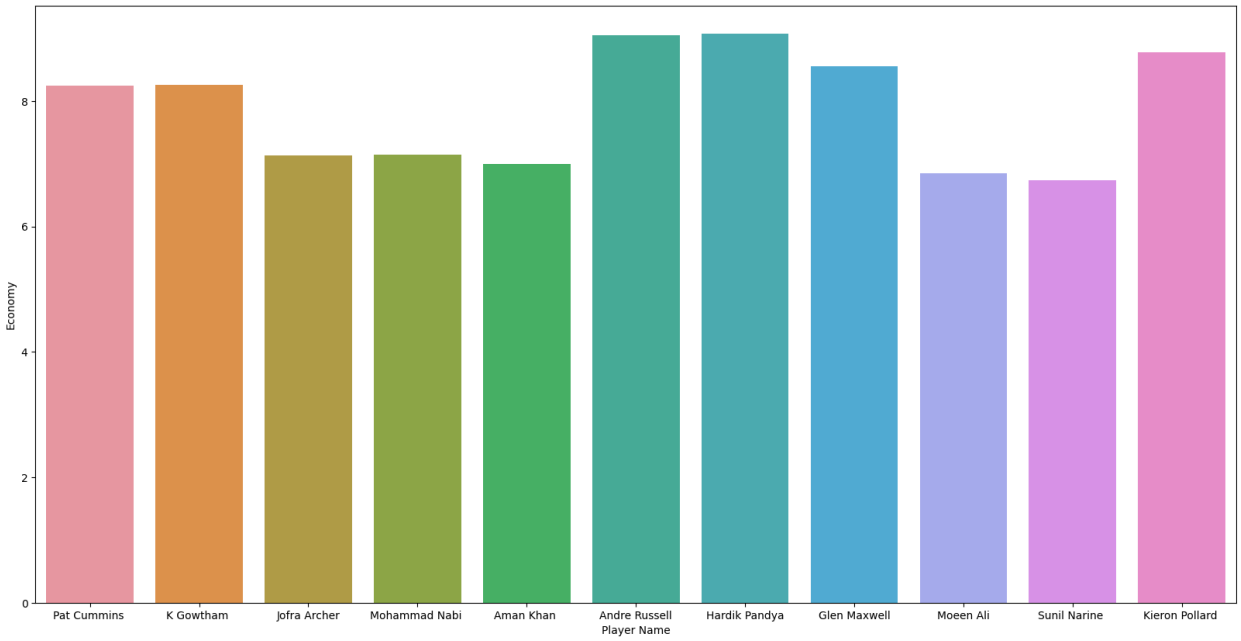
In [290…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Bowling_average', data=top_allrounders)
```

Out[290]: <AxesSubplot:xlabel='Player Name', ylabel='Bowling_average'>



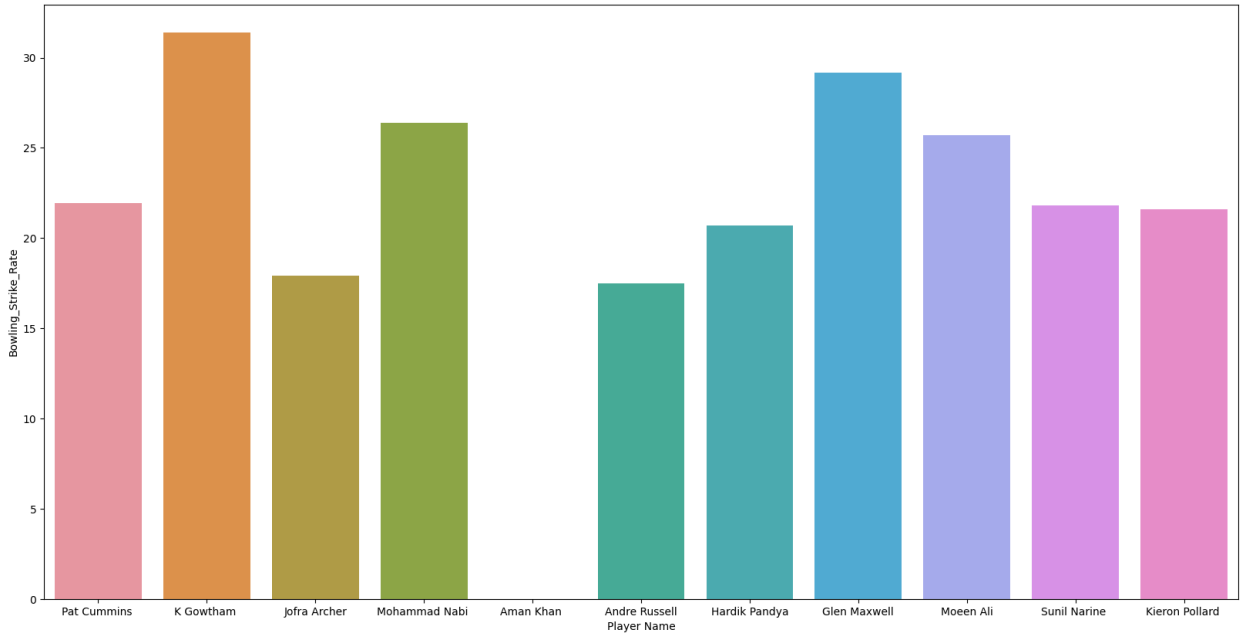
In [291…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Economy', data=top_allrounders)
```

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



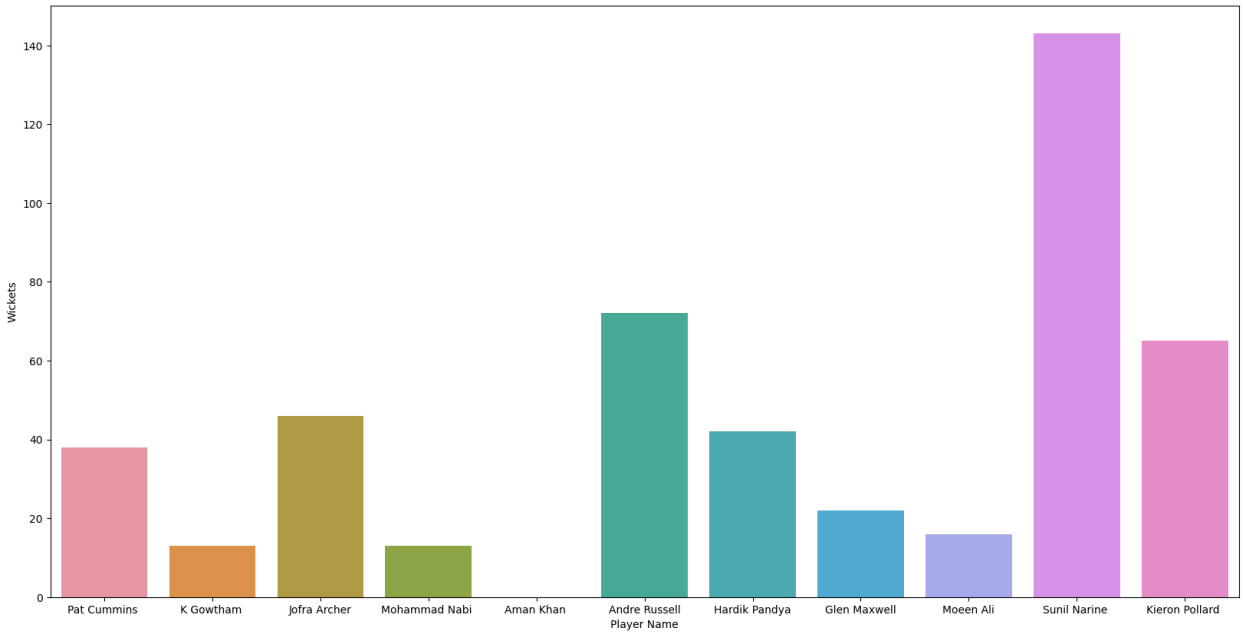
In [292…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Bowling_Strike_Rate', data=top_allrounders)
```

Out[292]: <AxesSubplot:xlabel='Player Name', ylabel='Bowling_Strike_Rate'>



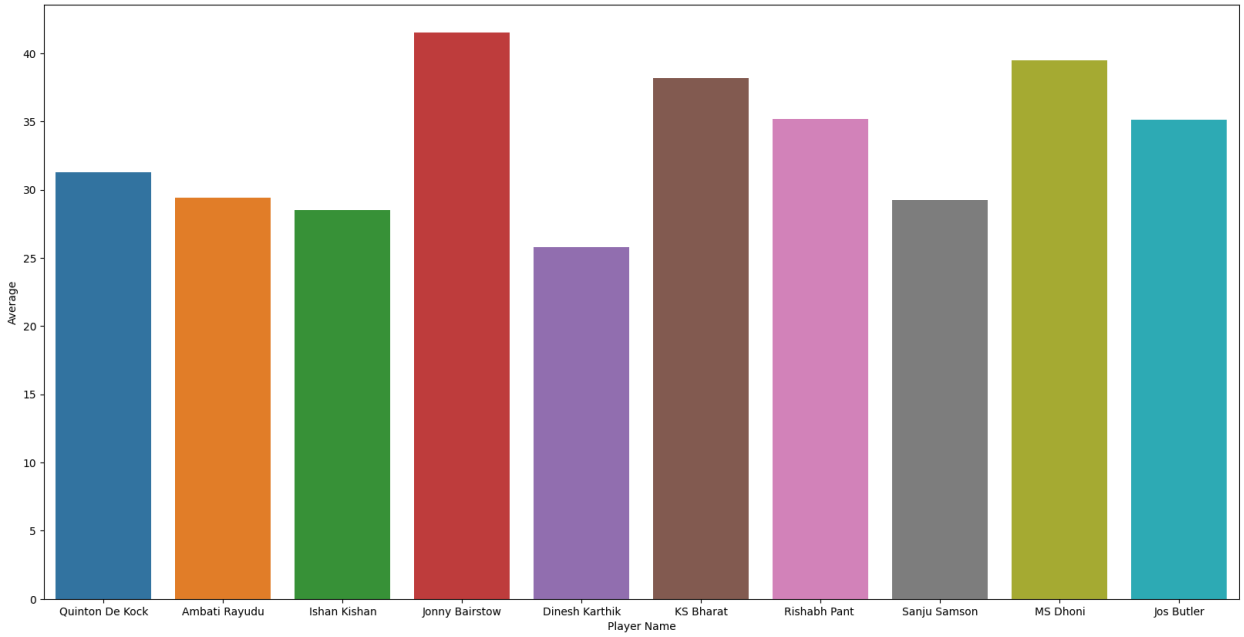
In [293…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Wickets', data=top_allrounders)
```

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



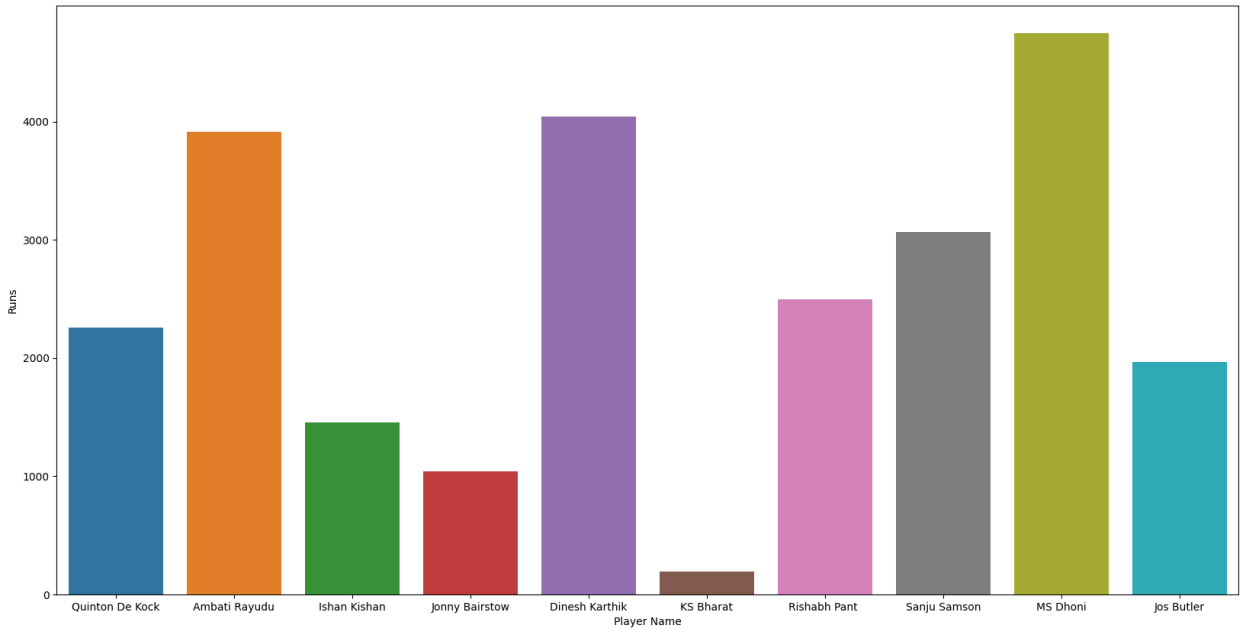
In [294…
```
#Visualising Keepers data
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Average', data=top_keepers)
```

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



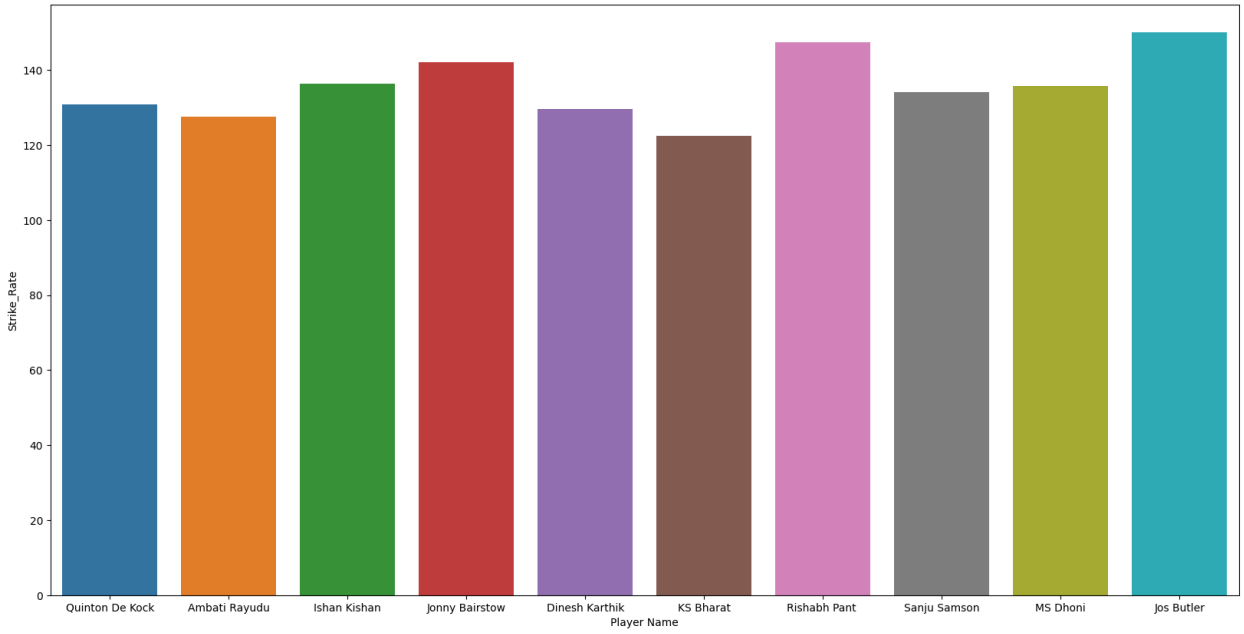
In [295…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Runs', data=top_keepers)
```

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



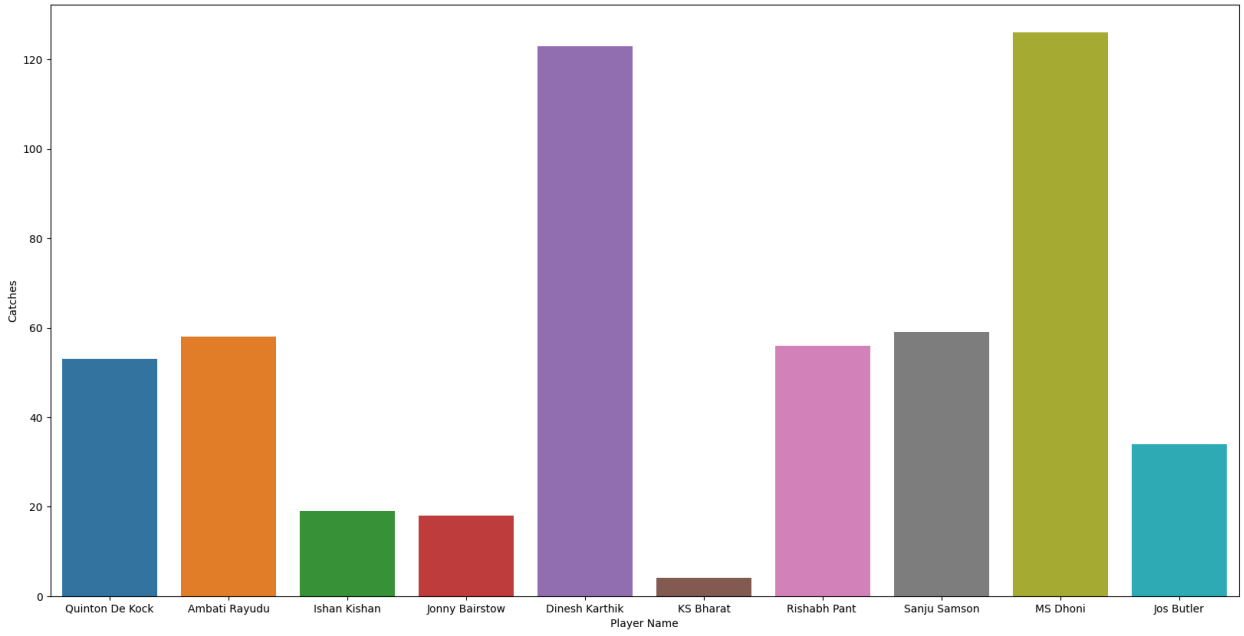
In [296…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Strike_Rate', data=top_keepers)
```

Out[296]: <AxesSubplot:xlabel='Player Name', ylabel='Strike_Rate'>



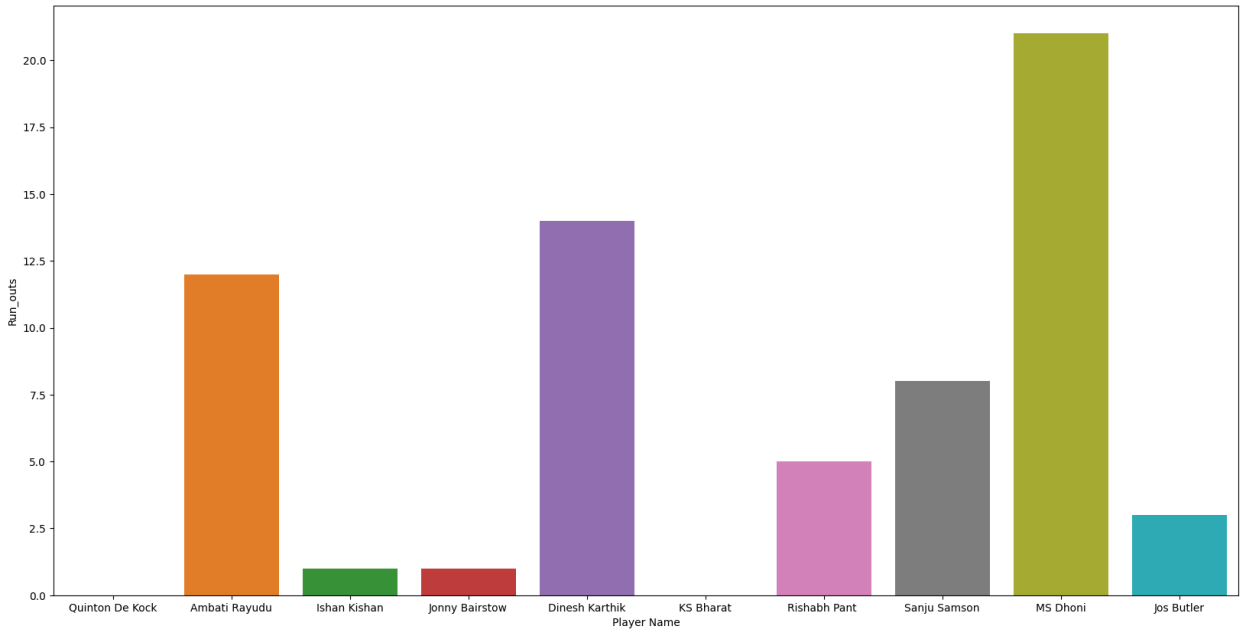
In [297…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Catches', data=top_keepers)
```

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



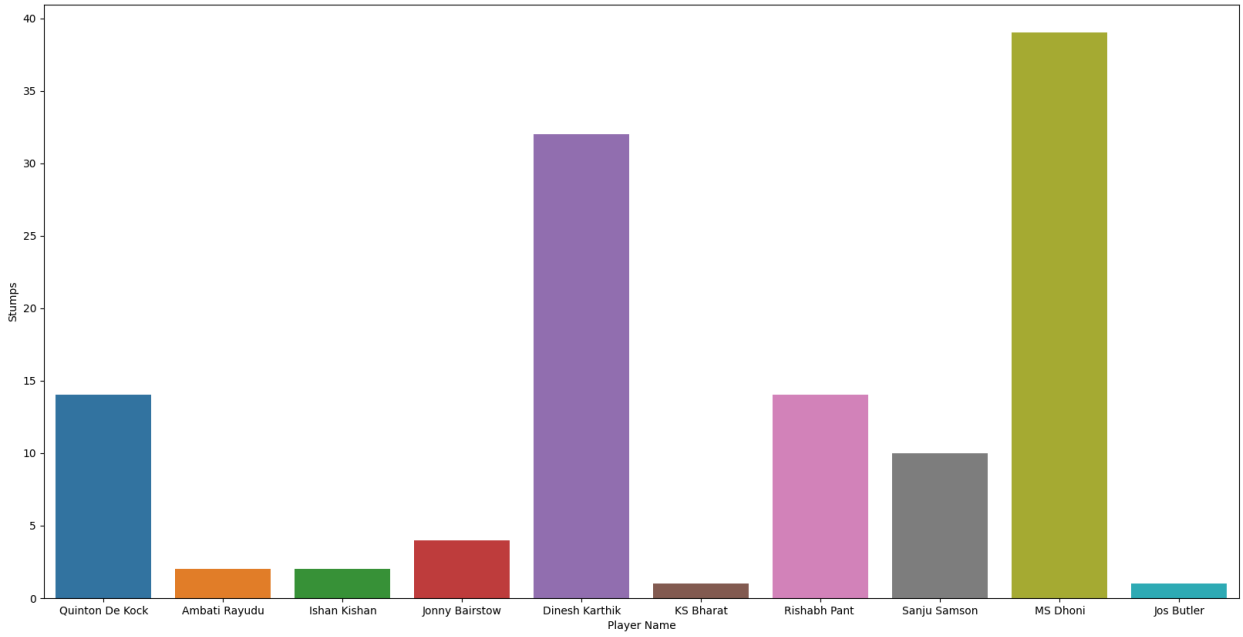
In [298…
```
plt.figure(figsize=(20,10))
sns.barplot(x='Player Name', y='Run_outs', data=top_keepers)
```

Out[298]: <AxesSubplot:xlabel='Player Name', ylabel='Run_outs'>



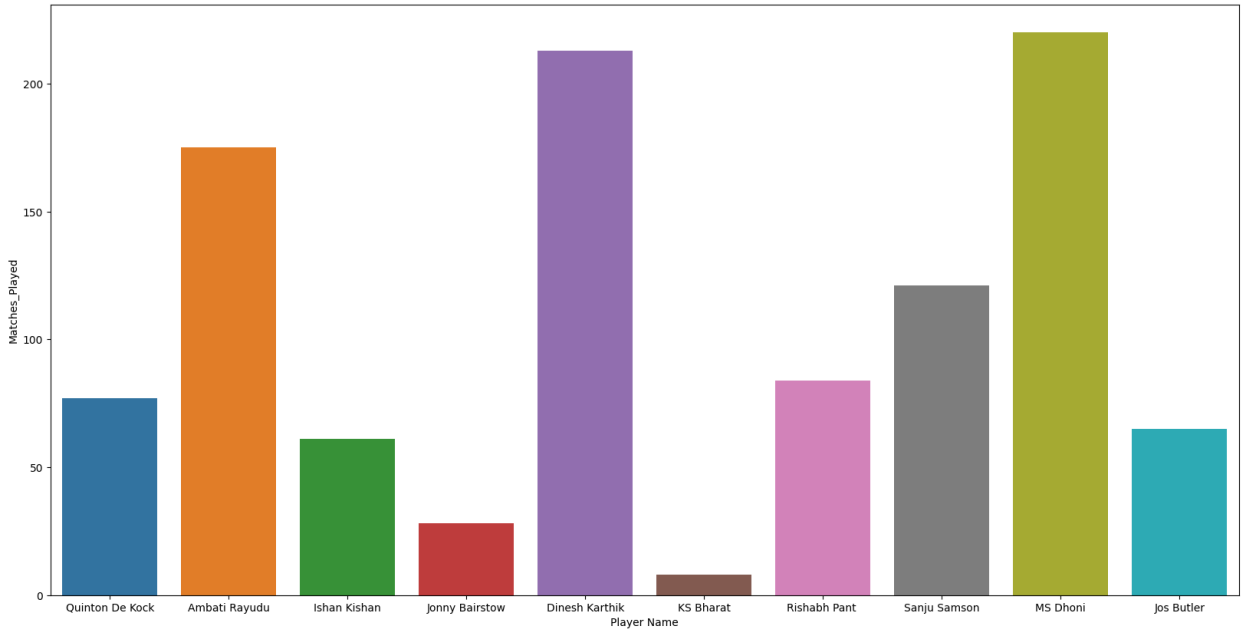
In [299…
```
plt.figure(figsize=(20,10))
sns.barplot(x ='Player Name', y='Stumps', data=top_keepers)
```

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'>


Matches_Played
0
50
100
150
200
Quinton De Kock
Ambati Rayudu
Ishan Kishan
Jonny Bairstow
Dinesh Karthik
KS Bharat
Rishabh Pant
Sanju Samson
MS Dhoni
Jos Butler
Player Name

In [ ]: