

FIFA Player Performance

Analyzing



Project Participants

Name	Enrollment No.
<i>Pareshkumar Sharma</i>	185223693104

Synopsis

FIFA is an international association football competition contested by the senior men's national teams of the members of the Federation International de Football Association (FIFA), the sport's global governing body.

For this project I have selected FIFA 19 complete player dataset that contains all the player's performance data. After detailed analysis this project, the extracted information will help the football team managers to improve their team performance. This performance is also takes important role to select any team or club or player for purpose of awarding cup or prizes.

This project contains this analysis modules:-

- Top 10 players of world
- What would be the best team
- Top 10 teams
- 10 poorest player of world
- Most paid Players
- Left and right footed players count, dribbling, ball control abilities



Description

This database contains 88 columns and 18207 rows that contains all player's performance like nationality, position wise performance, his club, strength etc.

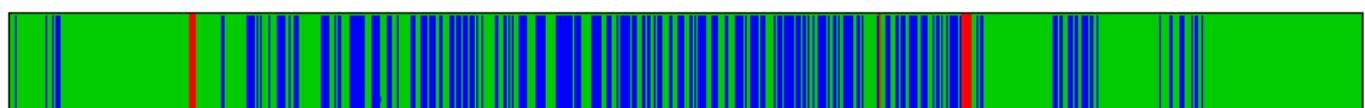
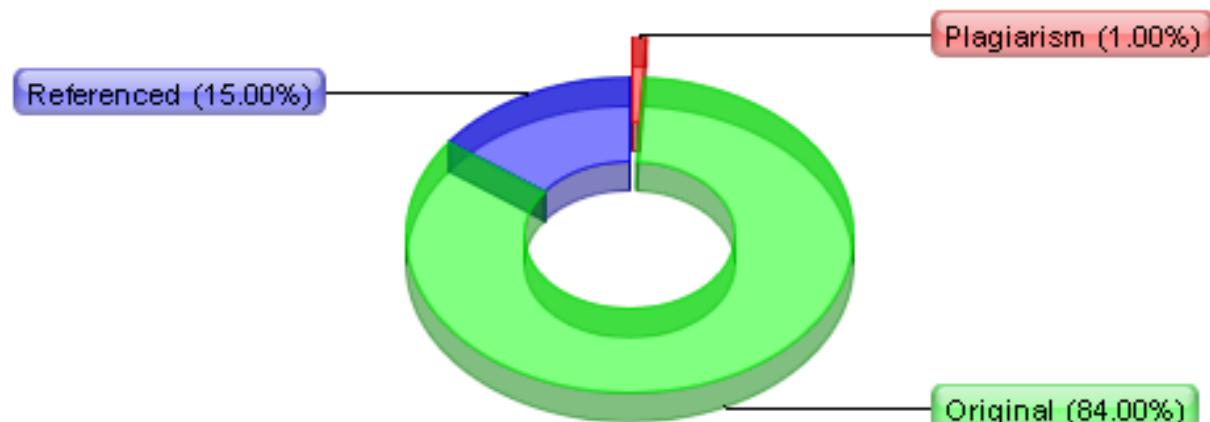
This dataset is use full for football manager and any award giving organization that want to know the top players, top football clubs, and top nations.

FIFA is an international association football competition contested by the senior men's national teams of the members of the Federation International de Football Association (FIFA), the sport's global governing body. The championship has been awarded every four years since the inaugural tournament in 1930, except in 1942 and 1946 when it was not held because of the Second World War. The current champion is France, which won its second title at the 2018 tournament in Russia. All football clubs of world are participating in this FIFA tournament.

For this project I have selected FIFA 19 complete player dataset that contains all the player's performance data. After detailed analysis this project, the extracted information will help the football team managers to improve their team performance. This performance is also takes important role to select any team or club or player for awarding cup or prizes.

Plagiarism report

Relation chart:



Comparison Preset: Word-to-Word. Detected language: English

Top sources of plagiarism:

	% 5	wrds: 388	https://en.wikipedia.org/wiki/List_of_men%27s_national_association_football_teams
	% 1	wrds: 88	https://en.wikipedia.org/wiki/FIFA
	% 1	wrds: 102	https://en.wikipedia.org/wiki/France_national_football_team

[Show other Sources:]

Processed resources details:

470 - Ok / 41 - Failed

[Show other Sources:]

Important notes:

Wikipedia:



Google Books:



Ghostwriting services:



Anti-cheating:



Wiki Detected!

[not detected]

[not detected]

[not detected]

Understanding Data

Column Description and Basic Questions

Analysis on **Nationality** of Players, how many players are coming from each nation and how many nations are from best performance players are coming

Analysis on **Potential** of player, who is the player who has max performance. Top 10 players, Top 10 clubs, Top 10 Nations

Analysis Worlds **football club**, total players in each club, total wage of all players of a club, total clubs from each nation

Analysis on value and **wage** of player, who has maximum wage, wage by nationality, wage by club is came just previous analysis

Analysis **Preferred foot** of player, how many players are playing from left foot, and how many are righty, nation wise player's foot, and maximum left or right footed players.

International **reputation** of player, total players on each category

Weak foot of player, player's histogram

Skill moves of the player

How many players have **real face?**

Position of player

Jersey Number of player

Joined date of player on to the club

Loaned From other club

Contact of players.

Height of players

Weight of players

Performance on different **position** of ground like LS (Left Striker), ST (Striker), RS (Right Striker), LW (Left Wing), LF (Left Forward), CF (Center Forward), RF (Right Forward), RW (Right Wing), LAM (Left Attacking Midfield), CAM (Center Attacking Midfield), RAM (Right Attacking Midfield), LM (Left Midfield), LCM (Left Center Midfield), CM (Center Midfield), RCM (Right Center Midfield), RM (Right Midfield), LWB (Left Wing Back), LDM (Left Defensive Midfield),

CDM (Center Defensive Midfield), RDM(Right Defensive Midfield), RWB(Right Wing Back), LB(Left Back), LCB(Left Center Back), CB(Center Back), RCB(Right Center Back), RB(Right Back)

Performance on different **task** like Crossing, Finishing, Heading Accuracy, Short Passing, Volleys, Dribbling, Curve, Free Kick Accuracy, Long Passing, Ball Control, Acceleration, Sprint Speed, Agility, Reactions, Balance, Shot Power, Jumping, Stamina, Strength, Longshot, Aggression, Interceptions, Positioning, Vision, Penalties, Composure, Marking, Standing Tackle, Sliding Tackle, GK Diving, GK Handling, GK Kicking, GK Positioning, GK Reflexes.

Big Questions: -

- Which nation is paying maximum wage and getting the best performance from players.
- Which are the players who can be best in world and have all abilities as any best footballer should have.
- Which club has best amount of player and which are the club that should be awarded by their players performance.
- Which jersey numbers are famous, what is number of player in each jersey number? Any club has two players with one number of jersey?
- Which player are good in specific action of football? Like they are in range of almost up of 85 or 90 in performance of particular position or action.
- How many players are loaned?
- Total how much wage is invested in all players?

Data Processing and Cleaning

Data is processed using data frames in all languages, but all languages have different way and methods to process the data, in this project I am using both Python and R technologies because both have their own advantages and disadvantages.

For processing I'm using module pandasql in Python. I am storing my processed data using two format one is txt (in some analysis) and second is excel file (in some analysis) using python's xlwt module.

For analysis on accurate data I have removed the 2085 players only when I performed operation on these columns 'RS', 'LW', 'LF', 'CF', 'RF', 'RW', 'LAM', 'CAM', 'RAM', 'LM', 'LCM', 'CM', 'RCM', 'RM', 'LWB', 'LDM', 'CDM', 'RDM', 'RWB', 'LB', 'LCB', 'CB', 'RCB', I removed it because these have null values. After doing it 16122 players are in dataset for analysis.

Apart from this processing i can't find any cleaning process for data till pre visualization, if needed in post visualization I will point it at top of any code for that query.

Minimum Hardware Requirement

Ram	2 GB
Processor	Dual Core 1.60 GHz or Higher
Resolution	1024 X 768 or higher

Software Used

- Python 3.7 (32 bit)
- MS Excel 2016

Module Used in Python

pandas:

It is the module used to read and manipulate data in excel or csv file using dataframe objects. It provides function and classes for manipulate the data in dataframes such as `read_excel()`, `concat()`, `max()`, `count()`, `sum()`, `groupby()` etc.

pandasql:

Pandasql is needed if we want to get data or filter data in a form of sql language. It is providing `sqldf()` in which we can fetch data and filter it using select statements.

pylab:

In Python pylab is provided by matplotlib module that contains number of methods for draw and save graphs or visualize the data in graphical representation. We can draw simple dot plot, line plot, bar graph, histograms and pie charts using this module.

seaborn:

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. It provide almost everything for data visualization with very good colors and all eye catching representation. We can draw countplot, swarmplot, factorplot, heatmap graphs using this module.

Algorithm

Linear Regression

It is the most well-known and popular algorithm in machine learning and statistics. This model will assume a linear relationship between the input and the output variable. It is represented in the form of linear equation which has a set of inputs and a predictive output. Then it will estimate the values of coefficient used in the representation.

Classification Algorithm

In this data is classified in different categories and further analysis done on each category for particular result. For example we can classify players of Fifa 19 in good players or weak players etc.

Collaborative Algorithm

This is used for predicting some output or information based on different inputs are came, for example in University maximum student learning masters in Mathematics from 8 years it's may happen that next year also be max students from mathematics.

Decision Tree:

Decision trees are used for both classification and regression problems, Decision trees often mimic the human level thinking so it's so simple to understand the data and make some good interpretations.

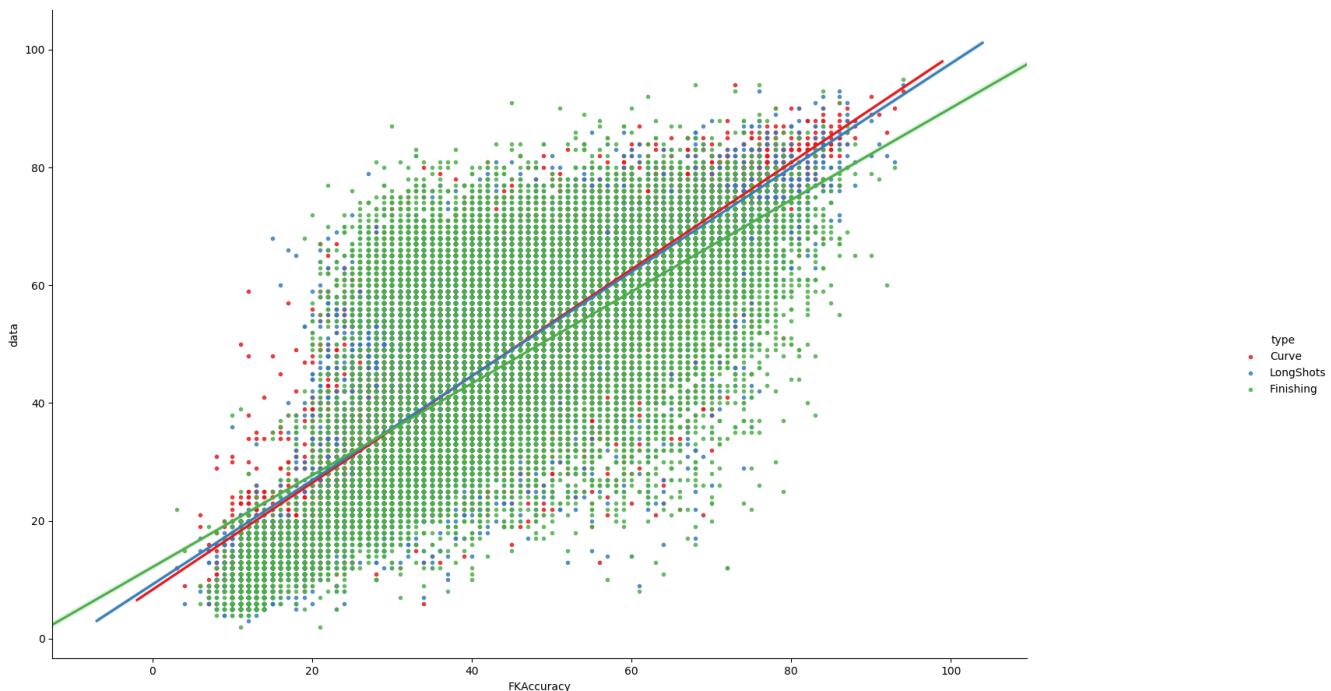
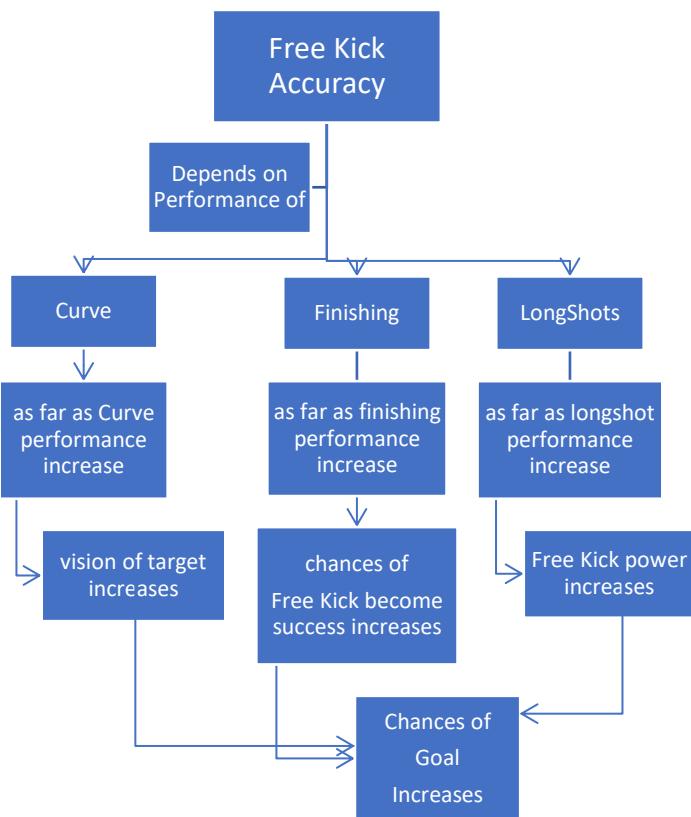
Decision Tree –

Making decision about Players Free Kick Accuracy

As this graph represents we can take a decision of Free Kick Accuracy of Player on the performance of player's Curve, Long Shots and Finishing.

Python Code

```
Import seaborn as sns  
Imprt matplotlib.pyplot as plt  
reg = p.read_excel(r"Football Data New  
Excel Format.xlsx", "Reggression")  
  
sns.lmplot("FKAccuracy","data",hue="type",data=reg)  
plt.show()
```



Conclusion:

The **Red** dots representing the **vision clarity**, **ORANGE** dots representing the **Power** and **GREEN** dots representing **chances of success** of players while free kick is concerned. The green dots representing players who are equal in these 3 things, in graph players with green dots and having more than 60 FK Accuracy are eligible for taking free kick.

As the lines of Curve, Long Shots and Finishing represents FK Accuracy highly depend on these 3 things.

Pre Visualization

The dataset named Fifa 19 Complete player dataset is downloaded from Kaggle.com.

In this dataset there are many numeric columns, I need to check for Null values, and different values available on that columns like

- Age
- Preferred Foot
- Total Players
- Clubs
- Etc.

For that I performed these following visualizations you can see it from next page.

Player Visualization

- **Total players in world : 18207**

```
#players is data frame taken from pandas.read_excel () function
```

```
>>> players["ID"].count()
```

- **Player with left preferred foot: 4211**

```
>>> players["Preferred Foot"][players["Preferred Foot"]=="Left"].count()
```

- **Player with right preferred foot: 13948**

```
>>> players["Preferred Foot"][players["Preferred Foot"]=="Right"].count()
```

- **Player with unknown preferred foot: 48**

```
•
```

```
>>> players["Preferred Foot"][players["Preferred Foot"].isna()].count()
```

```
>>> x
```

```
['Left Foot', 'Right Foot', 'Unknown']
```

```
>>> y
```

```
[4211, 13948, 48]
```

```
>>> py.bar(x,y)
```

```
<BarContainer object of 3 artists>
```

```
>>> py.title("Prefered Foot")
```

```
Text(0.5, 1.0, 'Prefered Foot')
```

```
>>> py.show()
```

```
>>> g.pie([4211,13948],labels=["Left","Right"],autopct=".2f%%")
```

```
([<matplotlib.patches.Wedge object at 0x145731D0>, <matplotlib.patches.Wedge object at 0x145735F0>], [Text(0.8207745153900478, 0.7323449971743047, 'Left'), Text(-0.8207745496735903, -0.732344958751076, 'Right')], [Text(0.4476951902127533, 0.39946090754962066, '23.19%'), Text(-0.4476952089128674, -0.3994608865914959, '76.81%')]))
```

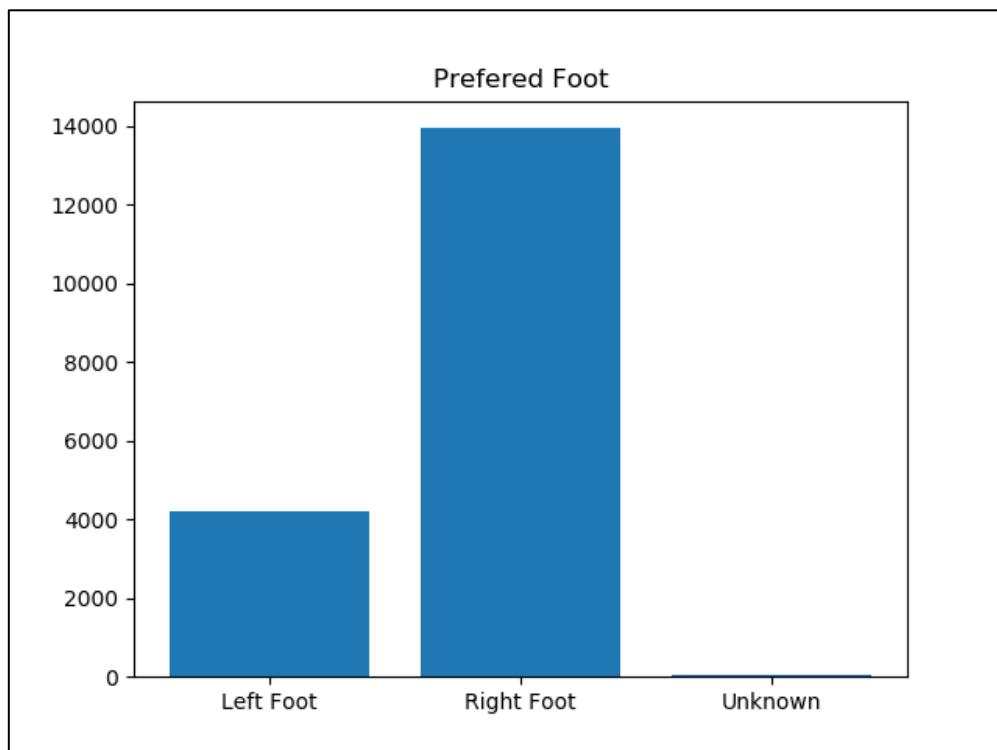
```
>>> g.title("Preferred Foot of players")
```

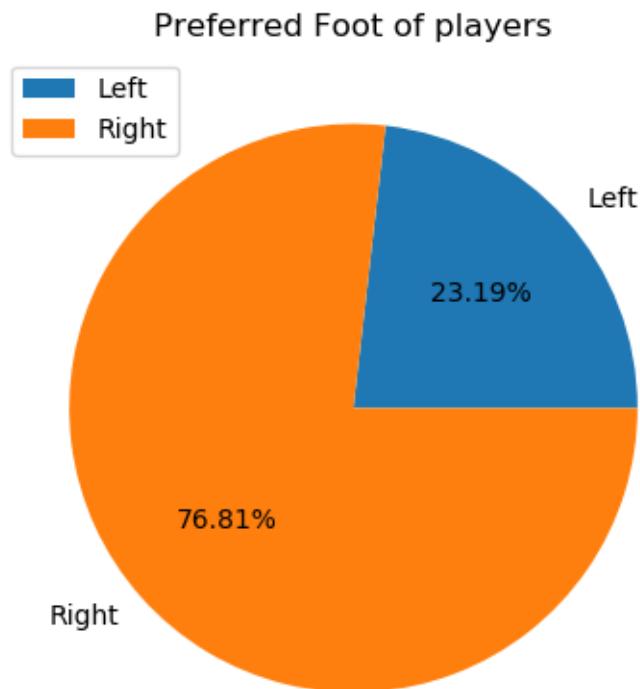
```
Text(0.5, 1.0, 'Preferred Foot of players')
```

```
>>> g.legend()
```

```
<matplotlib.legend.Legend object at 0x14567B70>
```

```
>>> g.show()
```





Conclusion

- ✓ As per the analysis more than 75% of players of the world are having Right as preferred foot

Club Player Visualization

- Total players which are not part of any club are: 241

```
>>> len(players["Club"][players["Club"].isna()])
```

- Total Players have joined club: 17966

```
>>> players["Club"].count()
```

- Total Clubs in world : 651

```
>>> len(players["Club"].unique())
```

#it gives 652 but one subtracted which is “nan”

```
>>> x
```

['Joined Club', "Don't joined Club"]

```
>>> y
```

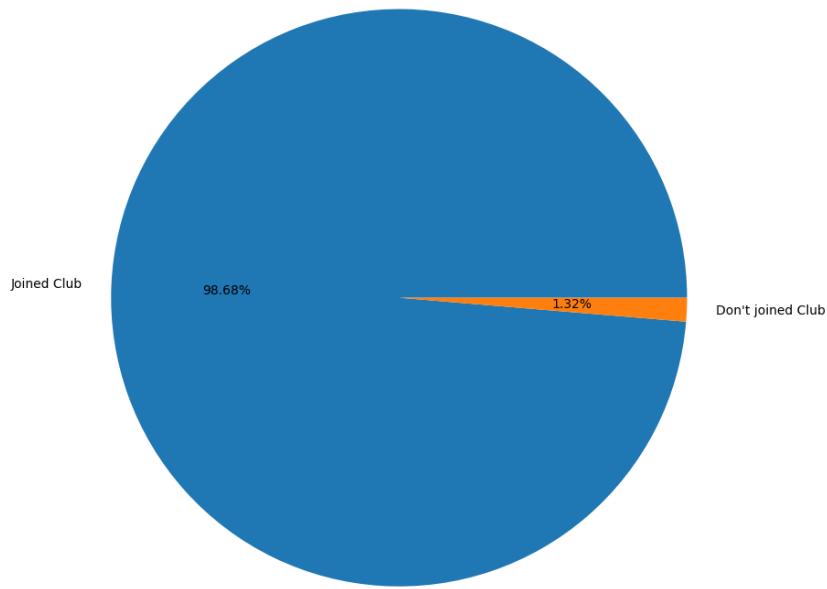
[17966, 241]

```
>>> py.pie(y, autopct="%.2f%%", labels=x)
```

```
>>> py.title("Player participation in club")
```

```
>>> py.show()
```

- Graph



- Conclusion

Very less amount of player are who didn't joined any club. It 98% players joined clubs.

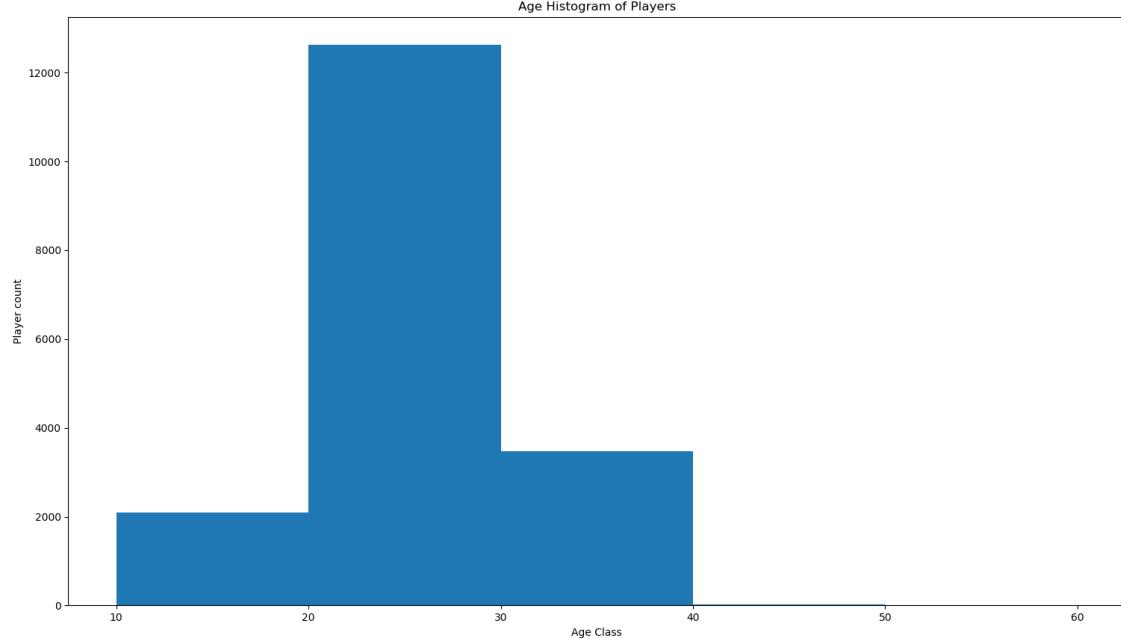
Age Visualization

- Histogram Graph of Player age

```
>>> ages = list(players["Age"])
>>> len(ages)
18207
>>> g.hist(ages,[10,20,30,40,50,60])
(array([ 2087., 12621., 3477., 22., 0.]), array([10, 20, 30, 40, 50, 60]), <a list of 5 Patch
objects>
>>> g.title("Age Histogram of Players")
Text(0.5, 1.0, 'Age Histogram of Players')
>>> g.xlabel("Age Class")Text(0.5, 0, 'Age Class')
>>> g.ylabel("Player count")
Text(0, 0.5, 'Player count')
>>> g.show()

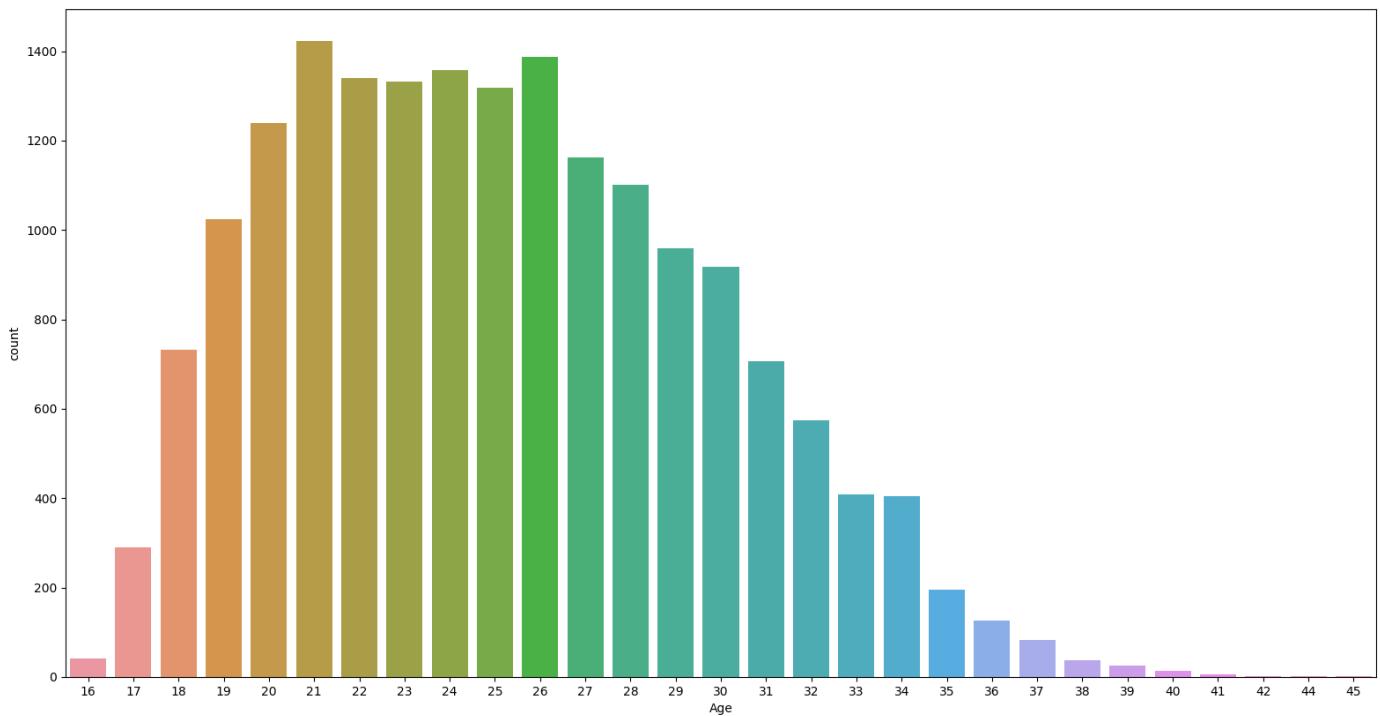
#the following is age data
>>> age_count = {}
>>> for x in range(16,46):
    count = 0
    for xx in ages:
        if(x==xx):
            count+=1
    age_count.update({x:count})
>>> age_count
{16: 42, 17: 289, 18: 732, 19: 1024, 20: 1240, 21: 1423, 22: 1340, 23: 1332, 24: 1358, 25: 1319,
26: 1387, 27: 1162, 28: 1101, 29: 959, 30: 917, 31: 707, 32: 574, 33: 408, 34: 404, 35: 196, 36:
127, 37: 82, 38: 37, 39: 25, 40: 13, 41: 5, 42: 1, 43: 0, 44: 2, 45: 1}
>>> max(age_count.values())
1423
```

>>>



min(age_count.values())

0



- Conclusion

most of players are with age class, between 20 – 30, max age is 45, min age is 16, max amount of players are with 21 age which are 1423

Post- Visualization

- Nationality

Analysis on Nationality of Players, how many players are coming from each nation and how many nations are from best performance players are coming

Q.1 Total nations have participated in Fifa 19.

Python

```
import pandas as p
import pandasql as sql
import pylab as g

# read excel file
nation_data = p.read_excel(r"..\\Football Data New Excel Format.xlsx","Nation")

# total records
# output 18207
print(nation_data.count())

# no null value data found
# output 0
print(nation_data["Nationality"][nation_data["Nationality"].isna()].count())

#unique nation
un_na = set(nation_data["Nationality"])
count = len(un_na)
#output 164
```

Conclusion

This year 18207 player are taking participation in Fifa 19 from 164 different nations

Q.2 How many players from each nation?

Python

```
import pandas as p
import pandasql as sql
import pylab as g

# read excel file
nation_data = p.read_excel(r"..\Football Data New Excel Format.xlsx","Nation")

# total records
# output 18207
print(nation_data.count())

# no null value data found
# output 0
nation = nation_data["Nationality"]
print(nation_data["Nationality"][nation.isna()].count())

#unique nation
#output 164
un_na = set(nation)
count = len(un_na)

pl_dict = {}
for u_n in un_na:
    count = 0
    for n in nation:
        if n == u_n:
            count+=1
    pl_dict.update({u_n:count})

# x is nation name
x = list(pl_dict.keys())
# y is player count
y = list(pl_dict.values())
g.bar(list(range(1,165)),y)
##count = 1
##while count < 165 :
##    g.bar(count,y[count-1])
##    count += 1
##g.legend(x)
g.title("Total player from each Nation")
g.savefig(".\\Total_Payers Each Nation.png")
#g.show()
g.close()
```

```

g.barh(list(range(1,165)),y)
g.title("Total player from each Nation")
g.savefig("./\\Total_Payers H Bar.png")
#g.show()
g.close()

# sorting dictionary in python

def srt(d,base="k|v",reversed1=False):
    assert type(d)==dict,"Dictionary argument is required"
    assert base.upper()=="K|V" or base.upper()=="K" or base.upper()=="V", "enter valid \"base\" argument"

    sorted_dict = {}

    if base.upper()=="K|V" or base.upper()=="K":
        for x in sorted(d.keys(),reverse=reversed1):
            sorted_dict.update({x:d[x]})
            print(x)
    else:
        empty_dict = {}

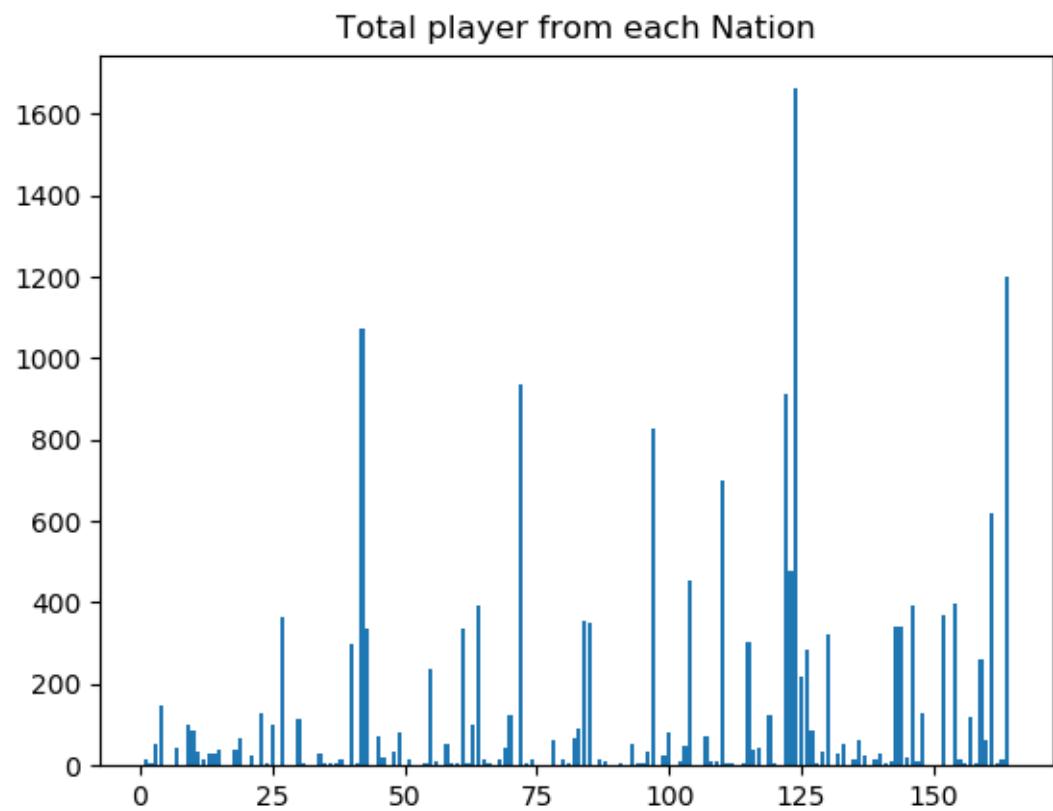
        for x in d:
            empty_dict.update({d[x]:x})

        for x in sorted(empty_dict,reverse=reversed1):
            sorted_dict.update({empty_dict[x]:x})

    return sorted_dict

sorted_nation = srt(pl_dict,"v",False)
file = open("nation_playerAscending.txt","w")
for x in sorted_nation:
    file.write(str(x)+","+str(sorted_nation[x])+"\n")
file.close()
sorted_nation = srt(pl_dict,"v",True)
file = open("nation_playerDescending.txt","w")
for x in sorted_nation:
    file.write(str(x)+","+str(sorted_nation[x])+"\n")
file.close()

```



Q3. Total wage pay by each nation.

Python

```
wage_data = p.read_excel("../Football Data New Excel Format.xlsx","Wage")

# checking for null values
wage_data["Wage(€)"][wage_data["Wage(€)"].isna()].count()
# output is 0

#
# cleaning data
#

# renaming column
wage_data = wage_data.rename(columns={'Wage(€)':'Wage'})

# removing the k from data and making it numeric

temp_list = []

# because only first element is 0 means numeric other are of string type so it is like that...

wage = list(wage_data["Wage"])

for x in wage:
    if type(x) == str:
        temp_list.append(int(x[:-1]))
    elif type(x) == int:
        temp_list.append(x)
    else:
        temp_list.append(x)

wage = temp_list

# merging both data nation and wage
#
# data_merge = p.concat([nation_data,wage_data],axis=1)
#
# 

nation_wage = {}

count=0

while count<len(nation):
```

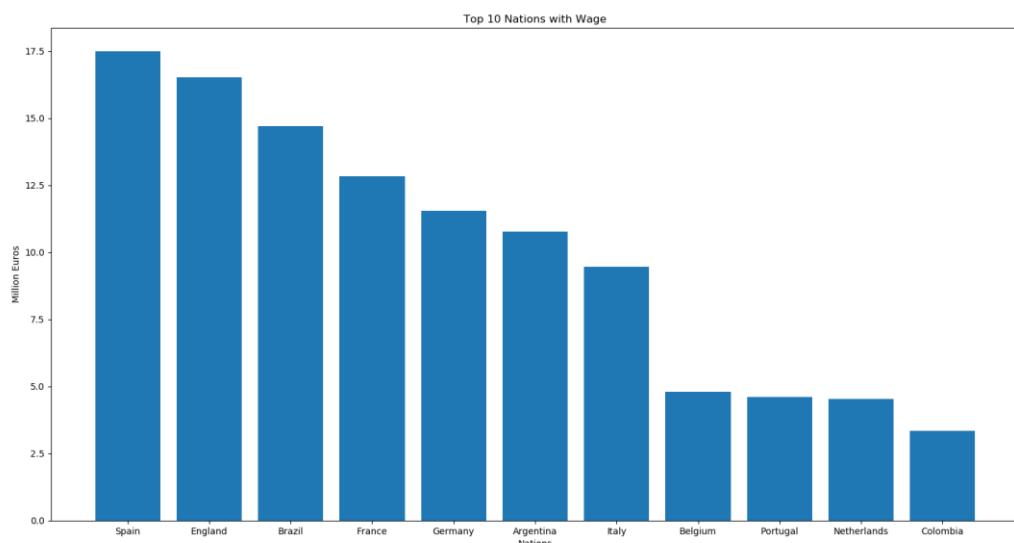
```

try:
    wage_am = nation_wage[nation[count]] + wage[count]
    nation_wage.update({nation[count]:wage_am})
except:
    nation_wage.update({nation[count]:wage[count]})
```

count+=1

```

file = open("Nation_Wage_Ascending.txt","w")
for x in srt(nation_wage,"V"):
    file.write(str(x)+","+str(nation_wage[x])+"\n")
file.close()
```



Conclusion:

Here are top 10 nations with 1st is Spain then England, then Brazil and so on.

Q.4 How many clubs are from each nation.

```
import pandas as p
import pandasql as sql
import pylab as g
import os

print(os.getcwd())

nation = p.read_excel(r"..\\Football Data New Excel Format.xlsx","Nation")

club = p.read_excel(r"..\\Football Data New Excel Format.xlsx","Club")

club_dict = {}

df_merged = p.concat([nation,club],axis=1)

club_count = sql.sql("Select Nationality,count(*) \"Clubs\" from df_merged group by Nationality order by Clubs desc")

count = 0

# dictionary for analysis two variable nation and clubs
dict_na_clubs = {}

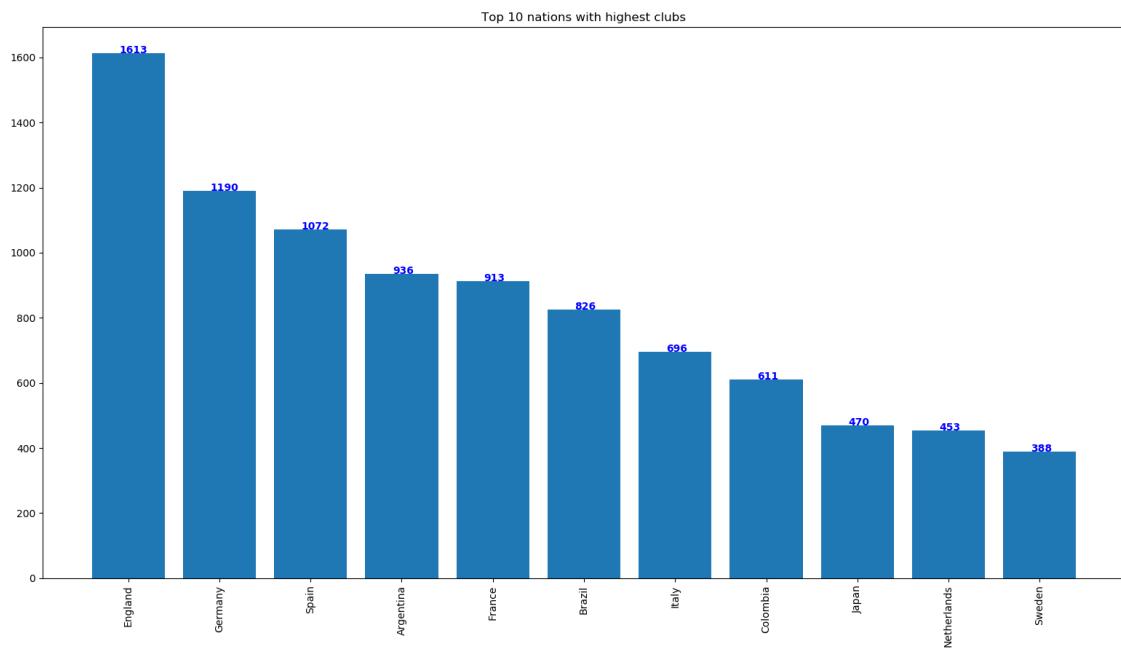
# total nations are 164
file = open("Nation_Clubs.txt","w")

while count<164:
    dict_na_clubs.update({club_count["Nationality"][count]:club_count["Clubs"][count]})
    file.write(str(club_count["Nationality"][count])+", "+str(club_count["Clubs"][count])+"\n")
    count+=1

file.close()
value = list(dict_na_clubs.values())
key = list(dict_na_clubs.keys())
g.bar(key[:11],value[:11])

for x in range(0,11):
    g.text(x-0.25,value[x]+3,str(value[x]),color="blue",fontweight="bold")

g.xticks(rotation=90)
g.title("Top 10 nations with highest clubs")
g.show()
```



Conclusion:

As maximum player are from England so max clubs are also from there.

Q.5 & 6 Nation wise best player using overall column.

& Maximum Overall performance by nation

```
# Q5 and Q6
```

```
import pandas as p
```

```
import pandasql as sql
```

```
import pylab as g
```

```
import os
```

```
nation = p.read_excel(r"..\Football Data New Excel Format.xlsx","Nation")
```

```
name = p.read_excel(r"..\Football Data New Excel Format.xlsx","Name")
```

```
Overall = p.read_excel(r"..\Football Data New Excel Format.xlsx","Overall")
```

```
data = p.concat([nation,name,Overall],axis=1)
```

```
# best player of nation
```

```
result = sql.sqldf("Select a.Nationality,a.Name,a.Overall from data a where Overall = (Select Max(Overall) from data b where a.Nationality=b.Nationality) order by Nationality")
```

```
uni_nation = list(set(result["Nationality"]))
```

```
uni_nation.sort()
```

```
dict_players_of_nation = {}
```

```
dict_players_overall = {}
```

```
file = open("Nation_Best_Player.txt","wb")
```

```
file1 = open("Nation_Overall_Max.txt","wb")
```

```
for x in uni_nation:
```

```
list1 = []
```

```
for y in list(result["Name"])[result["Nationality"]==x]:
```

```
    list1.append(y)
```

```
dict_players_of_nation.update({x:list1})
```

```
dict_players_overall.update ({x:list(result["Overall"])[result["Nationality"]==x])[0]})
```

```
file.write((str(x)+","+str(list1)+"\n").encode())
```

```
file1.write((str(x)+","+str(list(result["Overall"])[result["Nationality"]==x])[0])+"\n").encode())
```

```
file.close()
```

```
file1.close()
```

```
import SortDict as sd
```

```
dict_players_overall = sd.srt(dict_players_overall,"V",True)
```

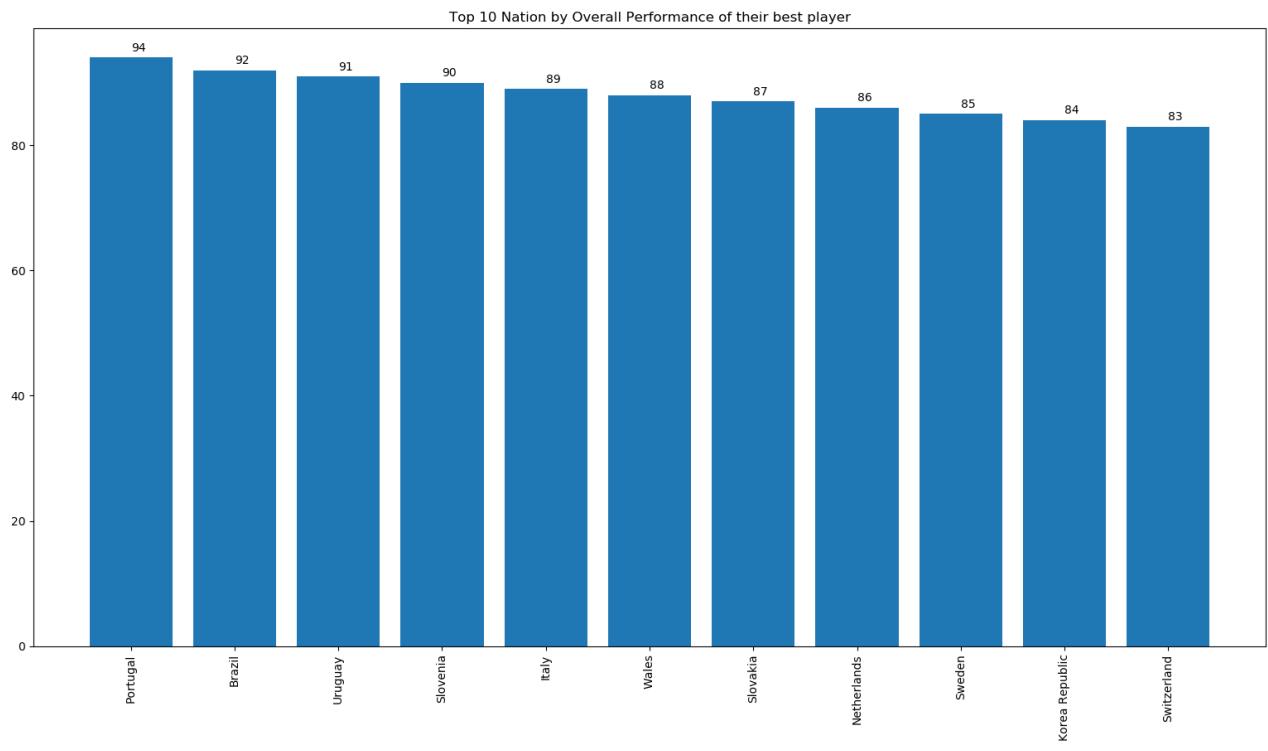
```
x = list(dict_players_overall.keys())
```

```
y = list(dict_players_overall.values())
```

```

g.bar(x[:11],y[:11])
g.xticks(rotation=90)
count = 0
while count<11:
    g.text(count,y[count]+1,str(y[count]))
    count+=1
g.title("Top 10 Nation by Overall Performance of their best player")

```



```
g.show()
```

Conclusion:

We found many players nation wise best as his performance you can see it in file **nation_best_player.txt** and here are nations on which Portugal who reached 94, and followed by 92 of Brazil and so on.

- **Club**

Club is a football team a football team basically includes only 11 players but at many different location different numbers of players like Australia it rule to have 18 players in club

Q.1 Total clubs in this dataset.

```
import pandas as p
import pandasql as sql
import pylab as g

# reading data
club_data = p.read_excel(r"..\\..\\Football Data New Excel Format.xlsx","Club")

# Clubs
clubs = list(club_data["Club"])

# Total
print(len(set(clubs)))
# 651 clubs in all of Fifa 19
```

Conclusion:

Total 651 different clubs are in Fifa 19.

Q.2 Players in each club.

```
import pandas as p
import pandasql as sql
import pylab as g

# reading data
club_data = p.read_excel(r"..\..\Football Data New Excel Format.xlsx","Club")

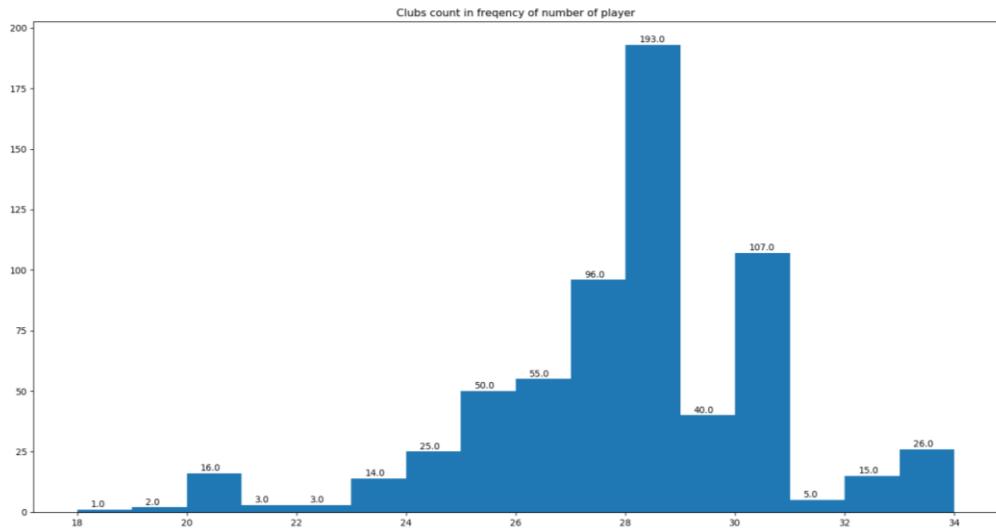
player_count = sql.sqldf("Select club, count(club)\\"Players\" from club_data group by club order by
Players desc")

print(player_count)

count_list = list(range(18,35))
hist_freq = list(player_count["Players"])
x,y,z=g.hist(hist_freq,count_list)
for i in range(0,len(count_list)-1):
    g.text(count_list[i]+0.25,x[i]+1,str(x[i]))

g.title("Clubs count in frequency of number of player")

g.show()
```



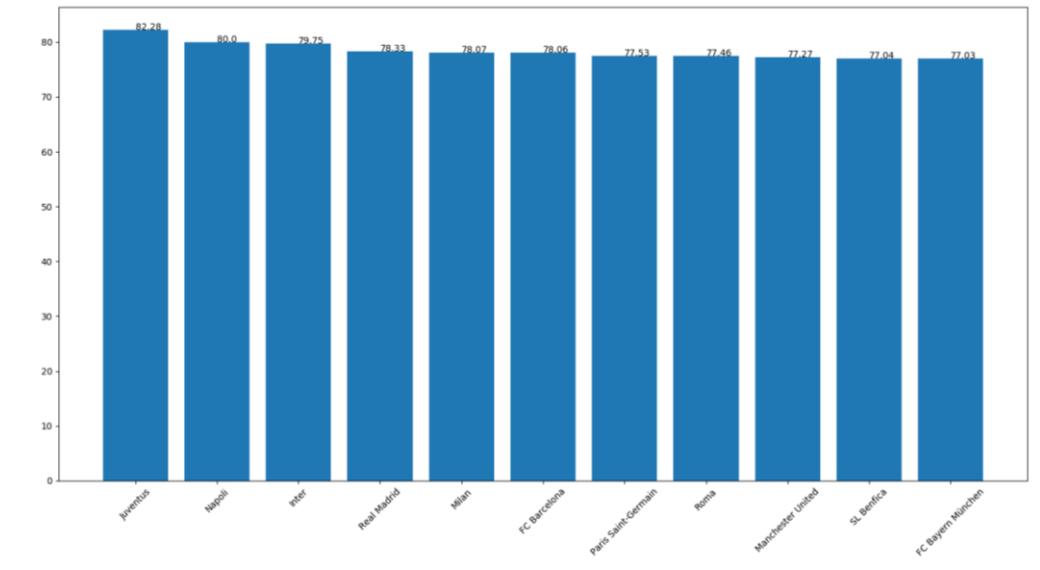
Conclusions:

Here are some clubs with 40 player and maximum 193 clubs are with 28 players.

Q.3 Average Overall of each club

```
import pandas as p
import pandasql as sql
import pylab as g

club_d = p.read_excel(r"Football Data New Excel Format.xlsx","Club")
overall_d = p.read_excel(r"Football Data New Excel Format.xlsx","Overall")
merge = p.concat([club_d,overall_d],axis=1)
avg = merge.groupby("Club").mean()
avg = avg.sort_values("Overall",ascending=False)
club1 = list(avg.index)
overall1 = list(avg["Overall"])
g.bar(club1[:11],overall1[:11])
g.xticks(rotation=45)
count = 0
while count<11:
    g.text(count,overall1[count],str(round(overall1[count],2)))
    count+=1
g.show()
```



Conclusion:

In performance Juventure club is at top with 82.28 of its all players overall performance.

Q.4 Total investment on each club

```
import pandas as p
import pandasql as sql
import pylab as g

club_d = p.read_excel(r"Football Data New Excel Format.xlsx","Club")
wage_d = p.read_excel(r"Football Data New Excel Format.xlsx","Wage")

tmp = []

tmp = p.concat([club_d,wage_d],axis=1)

tmp1 = tmp.groupby("Club").sum()

tmp2 = tmp1.sort_values(ascending=False,by="Wage")

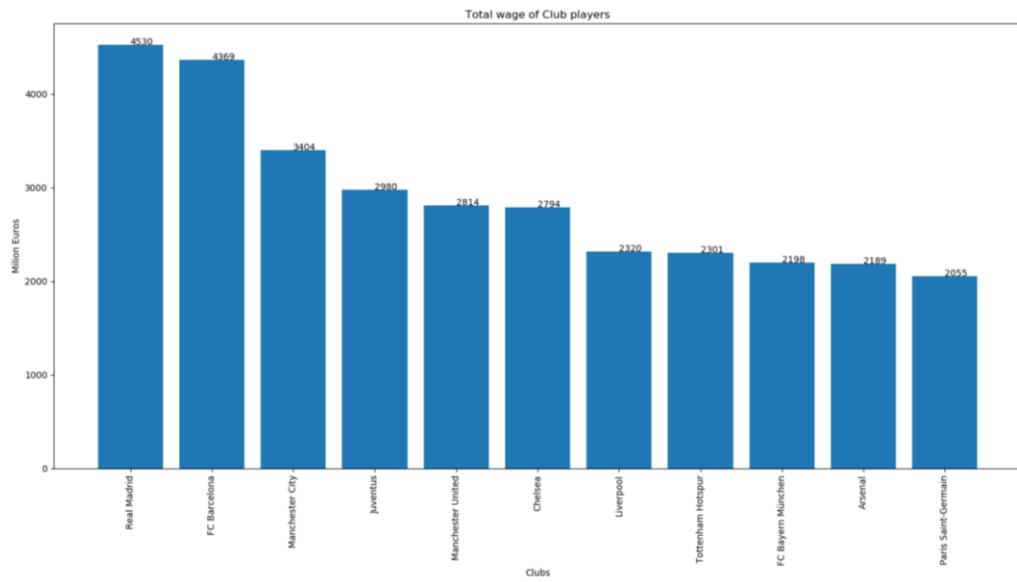
x = list(tmp2.index)
y = list(tmp2["Wage"])

tmp = []
for t in y:
    tmp.append(t/1000)

g.xticks(rotation=90)

g.ylabel("Milion Euros")
g.xlabel("Clubs")
g.title("Total wage of Club players")
c=0
while c<11:
    g.text(c,y[c],str(y[c]))
    c+=1
g.bar(x[:11],y[:11])

g.show()
```



Conclusion:

Talking about wage, it is very complicated because it is the thing where money is going, so Real Madrid is spending lot of money than other clubs it spends 4.5 million Euros on their player and of course next is FC Barcelona 4.3 M Euros.

Q.5 Investment vs performance of clubs

```
import pandas as p
import pandasql as sql
import pylab as g

club_d = p.read_excel(r"C:\Users\Hp\OneDrive\MCA\Sem 4\Data Science Project\data\Football Data New Excel Format.xlsx","Club")
wage_d = p.read_excel(r"C:\Users\Hp\OneDrive\MCA\Sem 4\Data Science Project\data\Football Data New Excel Format.xlsx","Wage")
overall_d = p.read_excel(r"C:\Users\Hp\OneDrive\MCA\Sem 4\Data Science Project\data\Football Data New Excel Format.xlsx","Overall")

df = p.concat([club_d,wage_d,overall_d],axis=1)

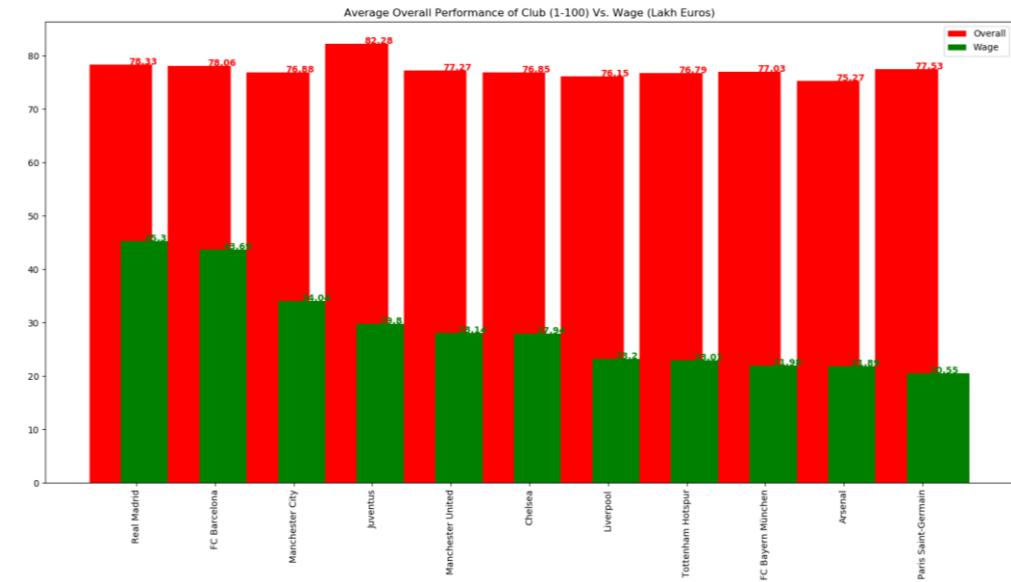
conclusion = sqlssql("Select Club,avg(Overall)\\"Overall\\",(sum(Wage)/100.00)\\\"All_Wage\\\" from df group by club order by All_Wage desc")

conclusion = conclusion.round(2)

x = list(conclusion["Club"])
y = list(conclusion["Overall"])
z = list(conclusion["All_Wage"])

g.xticks(range(11),rotation=90,labels=x[:11])

for i in range(0,11):
    g.bar(i-0.20,y[i],color="red")
    g.text(i-0.10,y[i],str(y[i]),color="Red",weight="bold")
    g.bar(i+0.20,z[i],color="green")
    g.text(i+0.10,z[i],str(z[i]),color="green",weight="bold")
g.title("Average Overall Performance of Club (1-100) Vs. Wage (Lakh Euros)")
g.legend(["Overall","Wage"])
g.show()
```

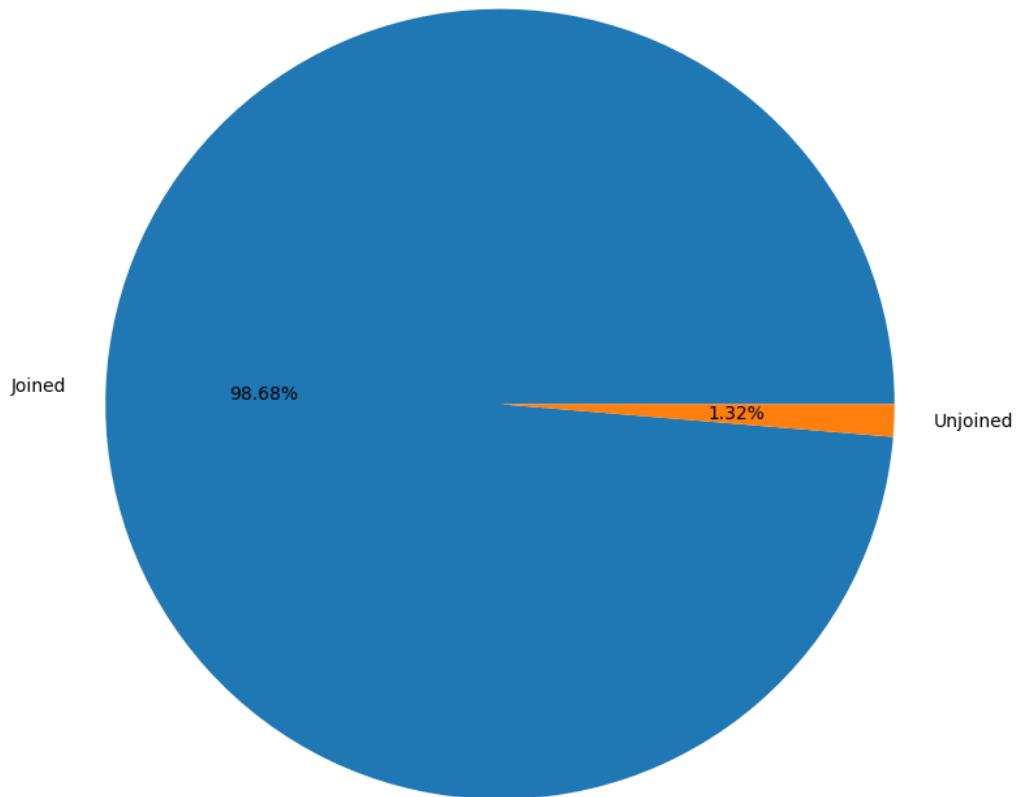


Conclusion:

Here the graph comparison of Wages of club with overall performance of club, with graph we can say Real Madrid is a Richest in all Fifa 19 clubs with both money and skills and followed by FC Barcelona and Manchester City the maximum Wage is 45.3 Lakh Euro and performance is 82.28 of Juventus.

Q.6 How many players have participated in club.

```
import pandas as p
import pandasql as sql
import pylab as g
club_d = p.read_excel(r"Football Data New Excel Format.xlsx","Club")
name_d = p.read_excel(r"Football Data New Excel Format.xlsx","Name")
df = p.concat([club_d,name_d],axis=1)
# players didn't joined any club yet
unjoined = df["Name"][df["Club"].isna()].count()
# output 241
# player joined clubs
joined = df["Name"][df["Club"].isna()==False].count()
#output 17966
g.pie([joined,unjoined],labels=["Joined","Unjoined"],autopct=".2f%%")
g.show()
```



Conclusion:

241 player are haven't joined yet to any football club they are have only their team.

- **Wage**

Q.1 Total wage invested in fifa players

```
import pandas as p
import pandasql as sql
import pylab as g

wage_d = p.read_excel(r"Football Data New Excel Format.xlsx","Wage")

#total wage

total_wage = float(wage_d.sum())

total_wage = total_wage/1000

print("Total wage of all fifa players is ",total_wage," Million Euros")

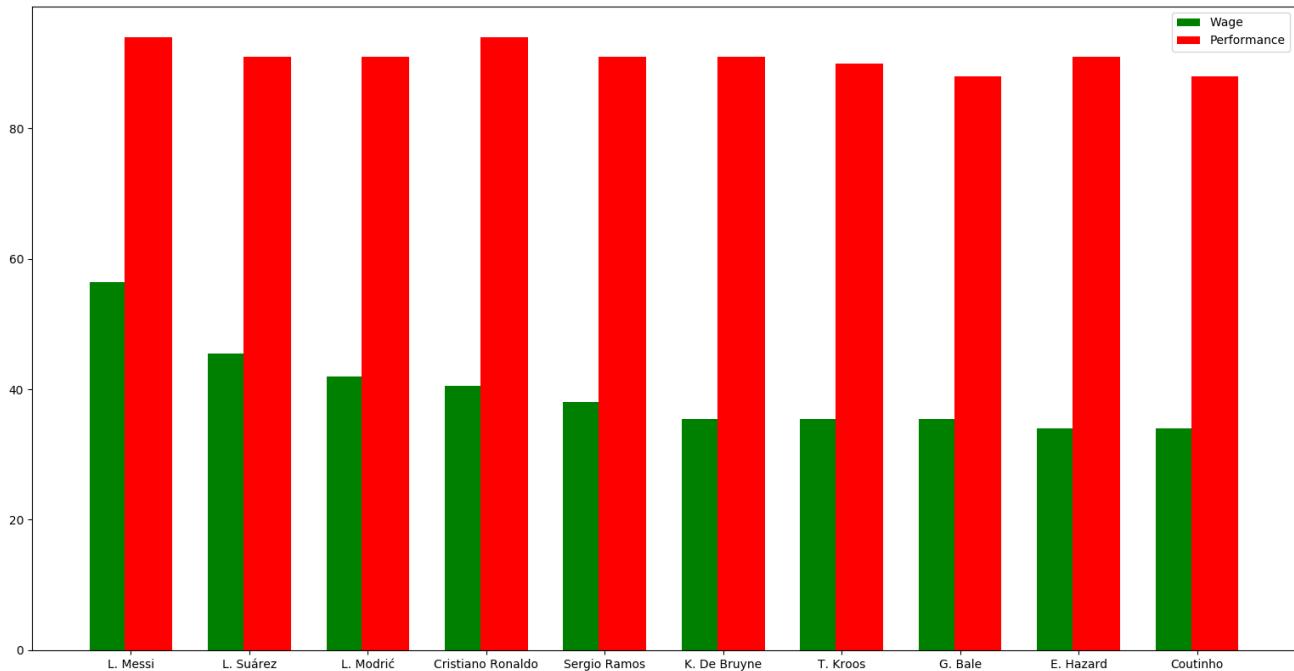
#output is 177.178 Million Euros
```

Conclusion:

Total wage given to all fifa players is 177.178 Million Euros.

Q.2 Wage vs player performance

```
#  
# Data cleaned for Records where wage is not available  
#  
  
import pandas as p  
import pandasql as sql  
import pylab as g  
  
wage_d = p.read_excel(r"C:\Users\Hp\OneDrive\MCA\Sem 4\Data Science Project\data\Football Data  
New Excel Format.xlsx","Wage")  
overall_d = p.read_excel(r"C:\Users\Hp\OneDrive\MCA\Sem 4\Data Science Project\data\Football Data  
New Excel Format.xlsx","Overall")  
name_d = p.read_excel(r"C:\Users\Hp\OneDrive\MCA\Sem 4\Data Science Project\data\Football Data  
New Excel Format.xlsx","Name")  
  
df = p.concat([wage_d,overall_d,name_d],axis=1)  
  
wage_v_overall = sqlssql("Select (Wage/10.00)\\"Wage\\", Overall, Name from df order by Wage desc")  
  
c = 0  
c1 = 0  
list_x = []  
  
while c<10:  
    tmp = c-0.6+c1  
    g.bar(c-0.6+c1,wage_v_overall["Wage"][c],color="green")  
    g.bar(c+c1,wage_v_overall["Overall"][c],color="red")  
    c+=1  
    c1+=1  
    list_x.append(tmp)  
  
g.xticks(list_x,labels=wage_v_overall["Name"][0:10])  
g.legend(["Wage","Performance"])  
g.show()
```



Conclusion:

L. Messi has highest wage but performance is similar to Cristiano Ronaldo and E. Hazard. In this graph wage is wage $\times 10^4$ Euros.

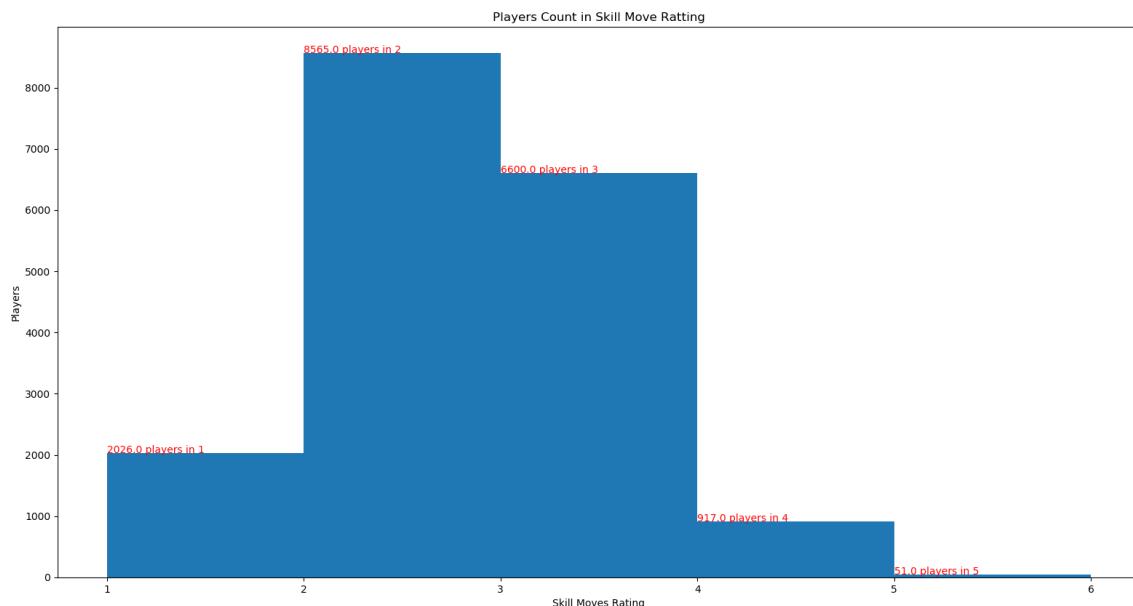
- Skill Moves

Q.1 Most skilled players in world top 10

```
import pylab as g
import pandasql as s
import pandas as p

skill_d = p.read_excel(r"Football Data New Excel Format.xlsx","Skill")

y,x,c = g.hist(list(skill_d.Skill),[1,2,3,4,5,6])
i = 0
while i<len(y):
    g.text(x[i],y[i],str(y[i])+" players in "+str(x[i]),color="red")
    i+=1
g.xlabel("Skill Moves Rating")
g.ylabel("Players")
g.title("Players Count in Skill Move Rating")
g.show()
```



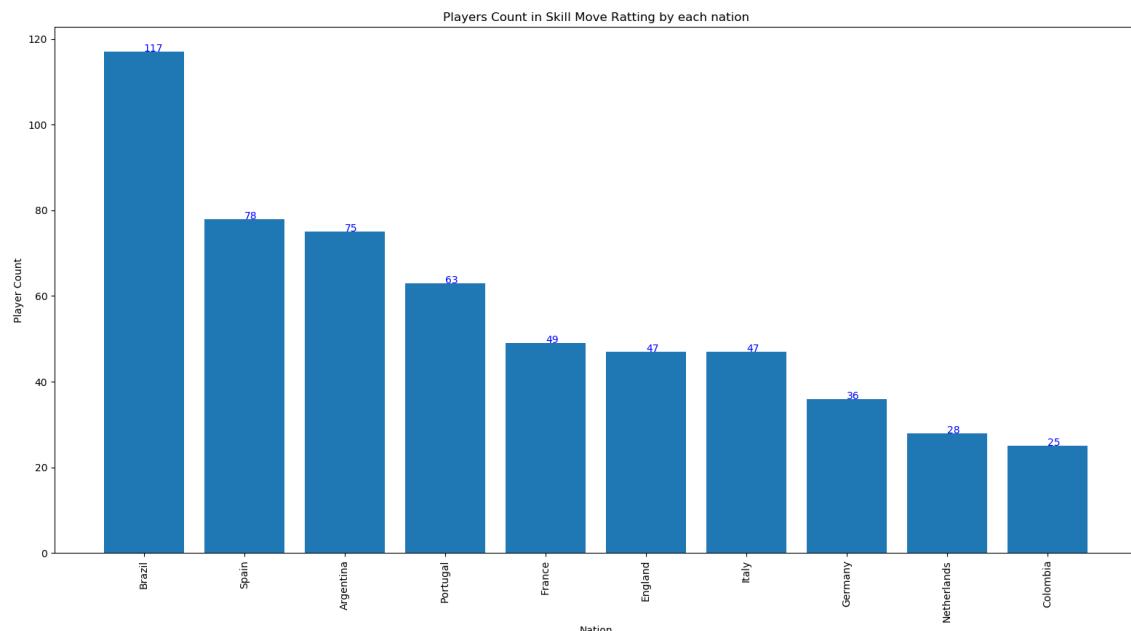
Conclusion:

Skill Moves, maximum players having rating 2 and rating 3

Q.2 Nationality of max skilled players

```
import pylab as g
import pandasql as s
import pandas as p

skill_d = p.read_excel(r"C:\Users\Hp\Documents\Data Science Project\data\Football Data New Excel Format.xlsx","Skill")
nation = p.read_excel(r"C:\Users\Hp\Documents\Data Science Project\data\Football Data New Excel Format.xlsx","Nation")
df = p.concat([skill_d,nation],axis=1)
mostskille = s.sql("Select Nationality,count(*)'Player' from df where skill>=4 group by Nationality Order by Player desc limit 10;")
g.bar(mostskille.Nationality,mostskille.Player)
g.xticks(rotation=90)
plyer = list(mostskille.Player)
nati = list(mostskille.Nationality)
cnt=0
while cnt<len(plyer):
    g.text(cnt,plyer[cnt],str(plyer[cnt]),color="blue")
    cnt+=1
g.xlabel("Nation")
g.ylabel("Player Count")
g.title("Players Count in Skill Move Rating by each nation")
g.show()
```



Conclusion:

177 skilled player who has skill move 4 or more from 5 are from Brazil and next is Spain with 78 and then Argentina with 75 players.

Q.3 Most skilled clubs

```
import pylab as g
import pandasql as s
import pandas as p

skill_d = p.read_excel(r"Football Data New Excel Format.xlsx","Skill")
club = p.read_excel(r"Football Data New Excel Format.xlsx","Club")
name = p.read_excel(r"Football Data New Excel Format.xlsx","Name")

df = p.concat([skill_d,club,name],axis=1)

mostskille = s.sql("Select Club,count(*)'Player' from df where skill>=4 Group by club Order by Player desc limit 10;")

g.bar(mostskille.Club,mostskille.Player)
g.xticks(rotation=45)
plyer = list(mostskille.Player)
club = list(mostskille.Club)

cnt=0

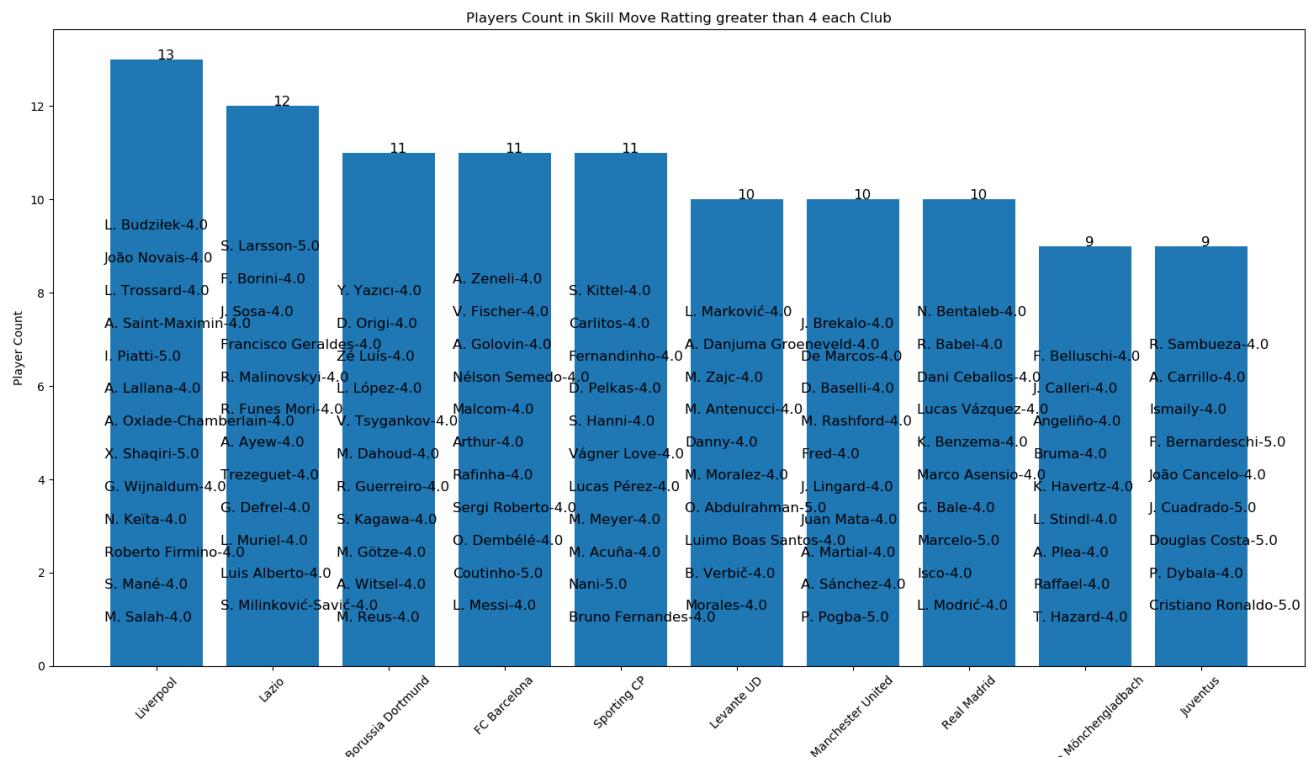
while cnt<len(plyer):
    c = 0
    y = 0
    name = s.sql("select Name,Skill from df where skill>=4 and club='"+club[cnt]+"'")
    names = list(name.Name)
    skils = list(name.Skill)
    while c<len(names):
        col = "black"
        if(cnt%2==0):
            col = "black"
            if(c==0):
                y+=0.25
        else:
            col = "black"
            if(c==0):
                y+=0.50
        y+=0.70
        g.text(cnt-0.45,y,str(names[c])+"-"+str(skils[c]),color=col,fontsize="large")
        c+=1
    g.text(cnt,plyer[cnt],str(plyer[cnt]),color=col,size="large")
    cnt+=1

g.xlabel("Club")
g.ylabel("Player Count")
```

```

g.title("Players Count in Skill Move Rating greater than 4 each Club")
g.show()

```



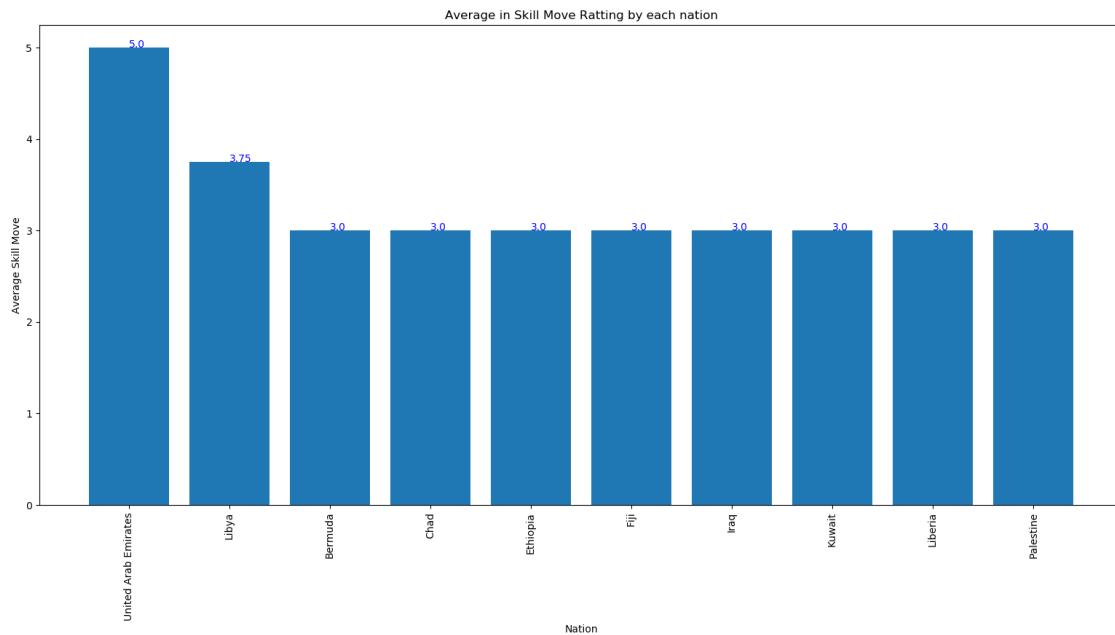
Conclusion:

Most skilled player in club wise Liverpool then Lazio and Borussia Dortmund.

Q.4 Most skilled Nations

```
import pylab as g
import pandasql as s
import pandas as p

skill_d = p.read_excel(r"C:\Users\Hp\Documents\Data Science Project\data\Football Data New Excel Format.xlsx","Skill")
nation = p.read_excel(r"C:\Users\Hp\Documents\Data Science Project\data\Football Data New Excel Format.xlsx","Nation")
df = p.concat([skill_d,nation],axis=1)
mostskille = s.sql("Select Nationality,avg(skill)'avg_skill' from df group by Nationality Order by avg_skill desc limit 10;")
g.bar(mostskille.Nationality,mostskille.avg_skill)
g.xticks(rotation=90)
avg1 = list(mostskille.avg_skill)
nati = list(mostskille.Nationality)
cnt=0
while cnt<len(avg1):
    g.text(cnt,avg1[cnt],str(avg1[cnt]),color="blue")
    cnt+=1
g.xlabel("Nation")
g.ylabel("Average Skill Move")
g.title("Average in Skill Move Rating by each nation")
g.show()
```



Conclusion:

By this analysis this graph shows average players coming from United Arab Emirates are with maximum skilled and after if we go on 2nd number it is Libya.

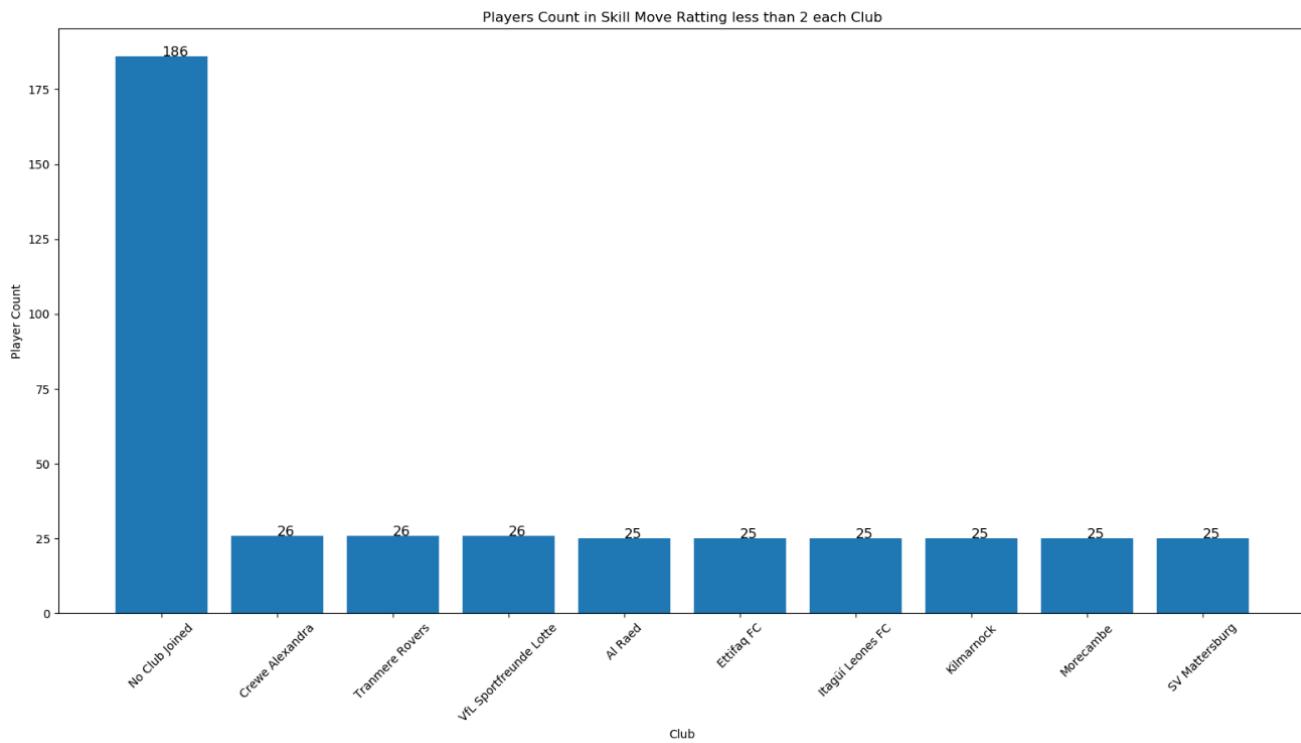
Q.5 Most poor player skills in club and Nation

```
import pylab as g
import pandasql as s
import pandas as p

skill_d = p.read_excel(r"Football Data New Excel Format.xlsx","Skill")
club = p.read_excel(r"Football Data New Excel Format.xlsx","Club")
nation = p.read_excel(r"Football Data New Excel Format.xlsx","Nation")
name = p.read_excel(r"Football Data New Excel Format.xlsx","Name")

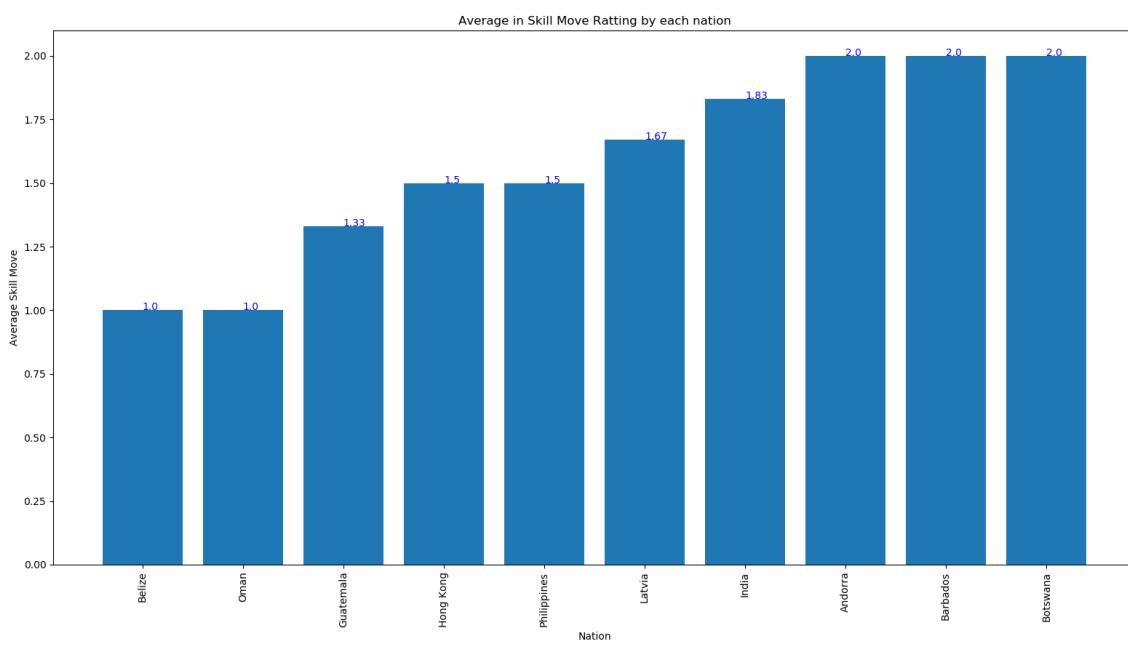
df = p.concat([skill_d,nation,club,name],axis=1)

mostskille1 = s.sql("Select Club,count(*)'Player' from df where skill<=2 Group by club Order by Player desc limit 10;")
mostskille1=mostskille1.fillna("No Club Joined")
g.bar(mostskille1.Club,mostskille1.Player)
g.xticks(rotation=45)
plyer = list(mostskille1.Player)
club = list(mostskille1.Club)
cnt=0
while cnt<len(plyer):
    c = 0; y = 0
    name = s.sql("select Name,Skill from df where skill<=2 and club='"+club[cnt]+"'")
    names = list(name.Name)
    skils = list(name.Skill)
    while c<len(names):
        col = "black"
        if(cnt%2==0):
            col = "black"
            if(c==0):
                y+=0.25
        else:
            col = "black"
            if(c==0):
                y+=0.50
        y+=0.70
        g.text(cnt-0.45,y,str(names[c])+"-"+str(skils[c]),color=col,fontsize="large")
        c+=1
    g.text(cnt,plyer[cnt],str(plyer[cnt]),color=col,size="large")
    cnt+=1
g.xlabel("Club");g.ylabel("Player Count")
g.title("Players Count in Skill Move Rating greater than 4 each Club")
g.show()
```



Conclusion:

Most poor nation in skilled player is Belize then Oman and then Guatemala.



Conclusion

Mostly poor player are from No Club Joined players and then the club Crewe Alexandra has 26 poor player and so on.

Q.6 Histogram of all player skill

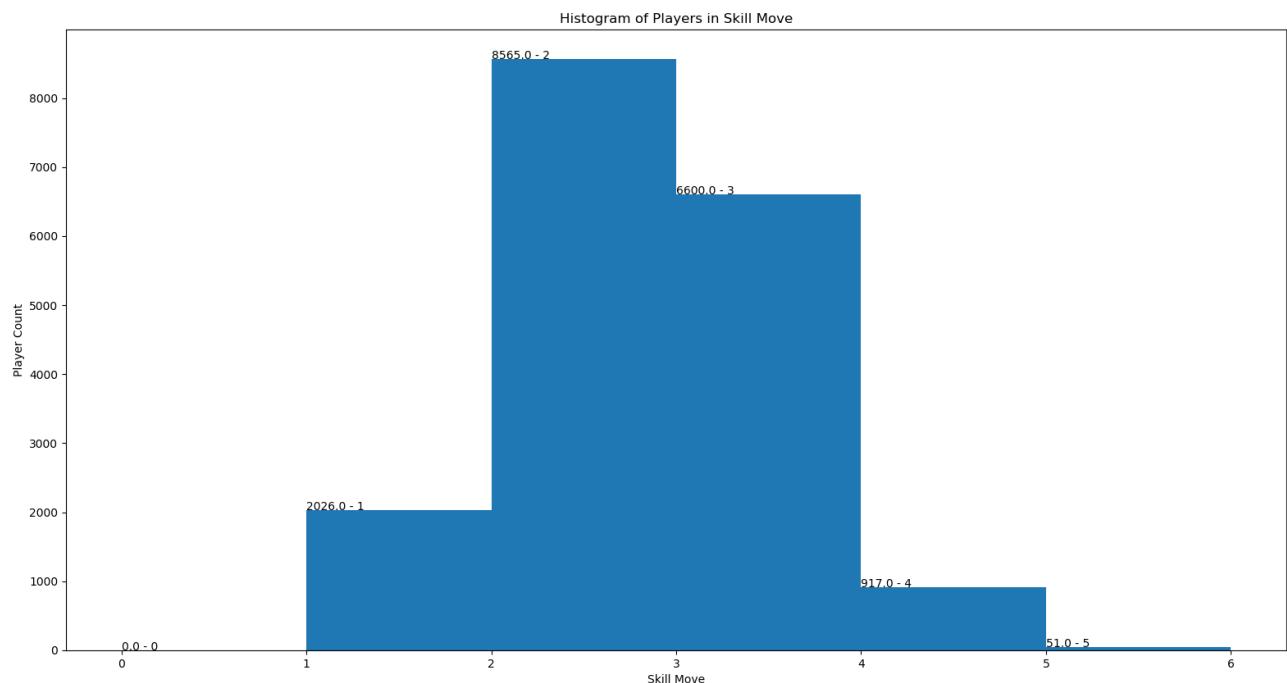
```
import pylab as g
import pandasql as s
import pandas as p

skill_d = p.read_excel(r"Football Data New Excel Format.xlsx","Skill")

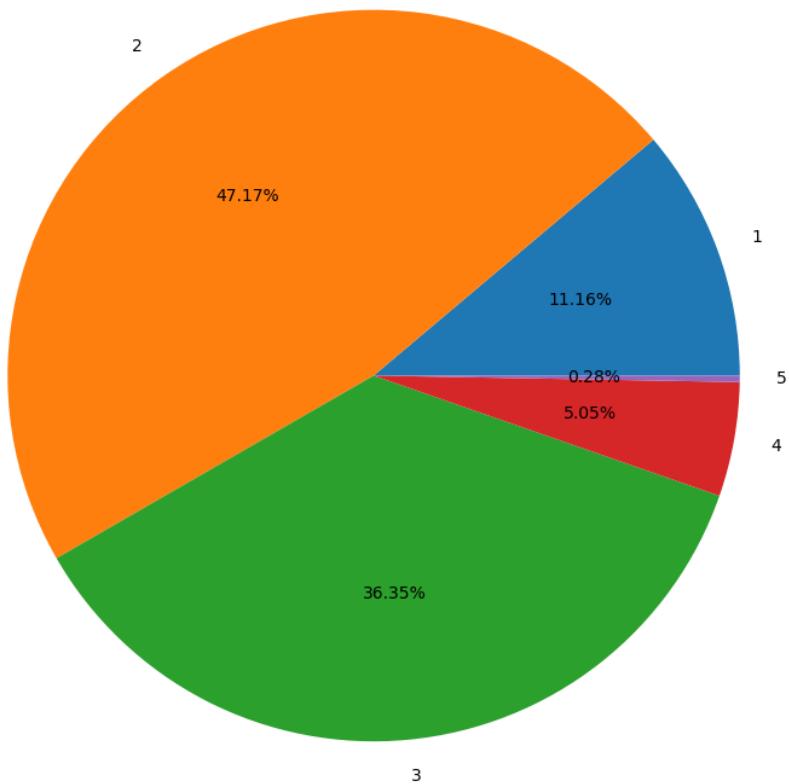
a,b,c=g.hist(skill_d.Skill,[0,1,2,3,4,5,6])

cnt = 0
while cnt<len(a):
    g.text(b[cnt],a[cnt],str(a[cnt])+" - "+str(b[cnt]))
    cnt+=1

g.title("Histogram of Players in Skill Move")
g.xlabel("Skill Move")
g.ylabel("Player Count")
g.show()
```



Players Percentage in Skill Moves



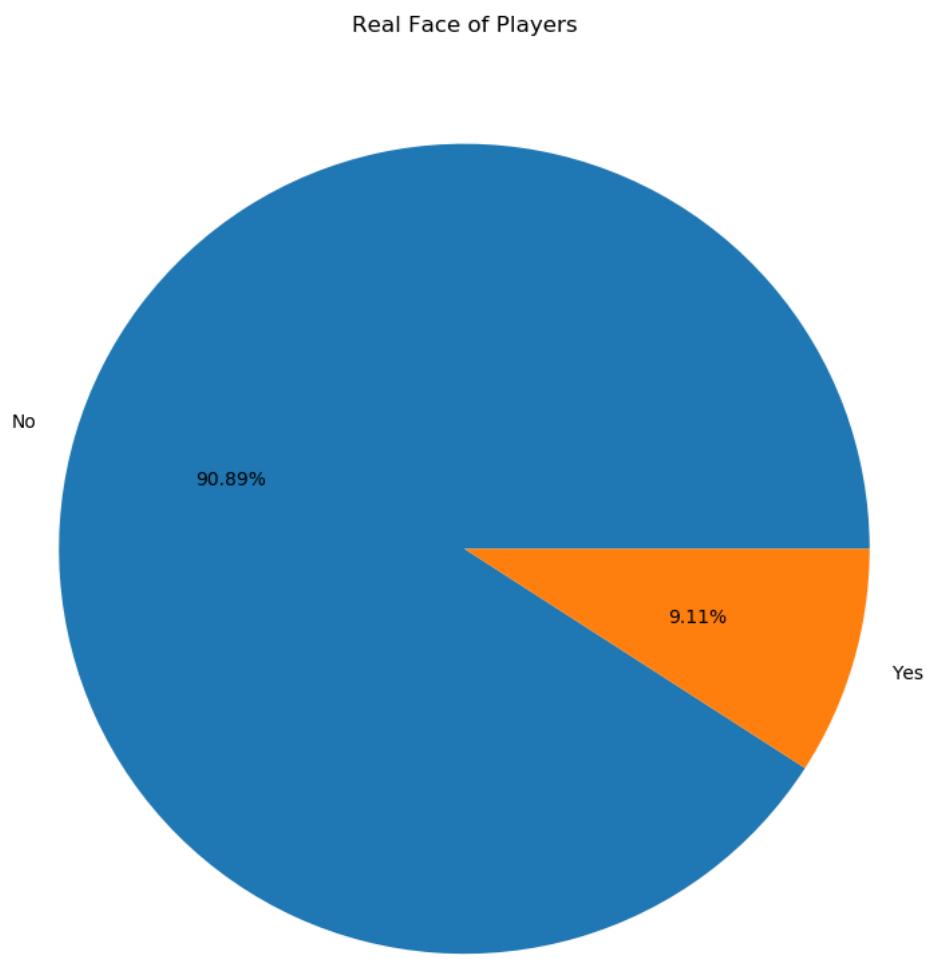
Conclusion

Maximum players are with skill move 2 they need to improvement.

- Face

Q.1 Total Face types

```
face_d = pd.read_excel(r"Football Data New Excel Format.xlsx","RealFace")  
  
df = pd.read_sql("select RealFace,count(*)\\"Total\" from face_d group by RealFace;")  
  
g.title("Real Face of Players")  
g.pie(df.Total,labels=df.RealFace,autopct=".2f%%")  
g.show()
```



Conclusion:

Only 9% players have real face.

• Body Type

Q.1 Total types of body and player count

```
body_d = pd.read_excel(r"Football Data New Excel Format.xlsx","BodyType")
```

```
a,b,c = g.hist(list(body_d.BodyType))
```

```
cnt = 0
```

```
while cnt<len(a):
```

```
    g.text(b.item(cnt),a.item(cnt),str(a.item(cnt)),size="large")
```

```
    cnt+=1
```

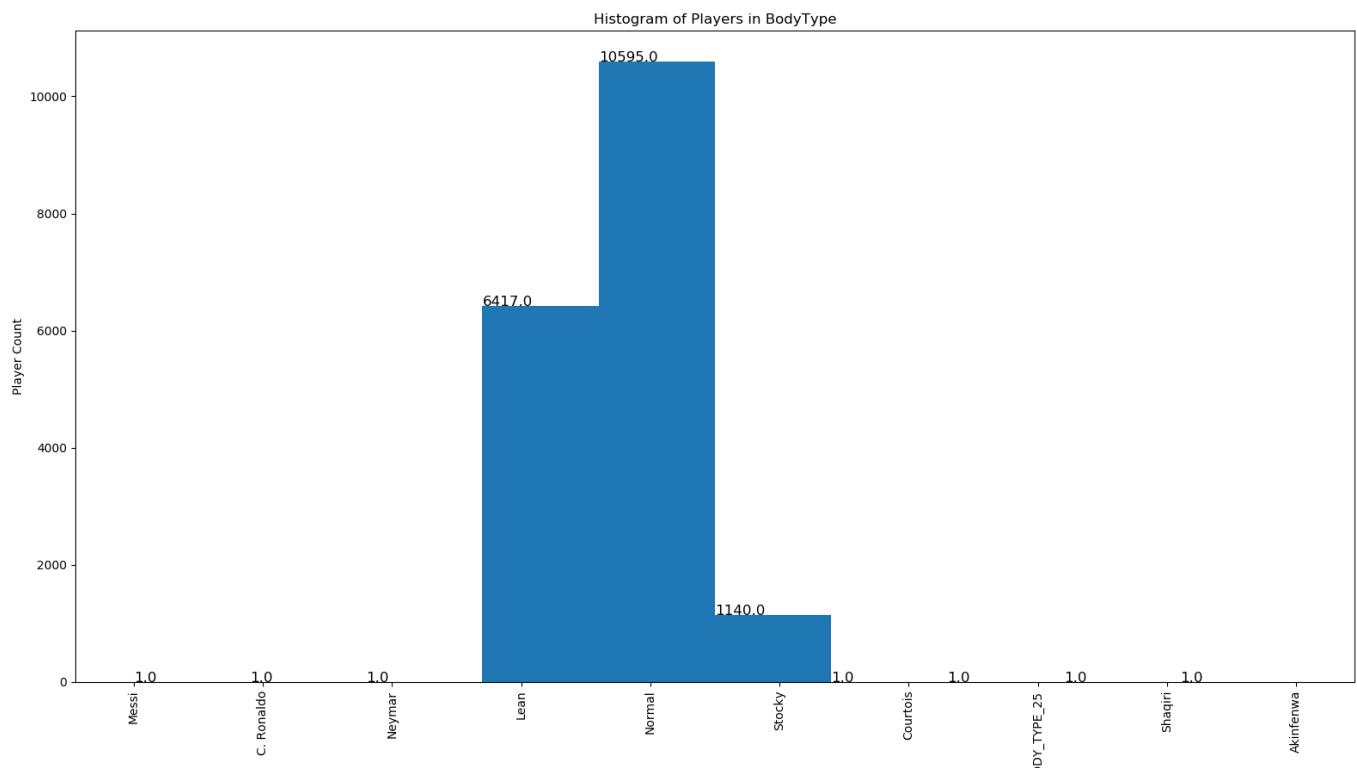
```
g.xticks(rotation=90)
```

```
g.title("Histogram of Players in BodyType")
```

```
g.xlabel("Body Type")
```

```
g.ylabel("Player Count")
```

```
g.show()
```



Conclusion

Here maximum player having normal body after that Lean and then very less have Stocky

● Position

```
position_d = pd.read_excel(r"Football Data New Excel Format.xlsx","Position")
```

Q.1 Total positions

```
print("Positions : ",len(set(position_d.Position))," Positions: \n",set(position_d.Position))
```

Q.2 Player count at particular position

```
count = sqldf("select position,count(*)\\"Player\" from position_d group by position;")
```

```
g.bar(count.Position,count.Player)
```

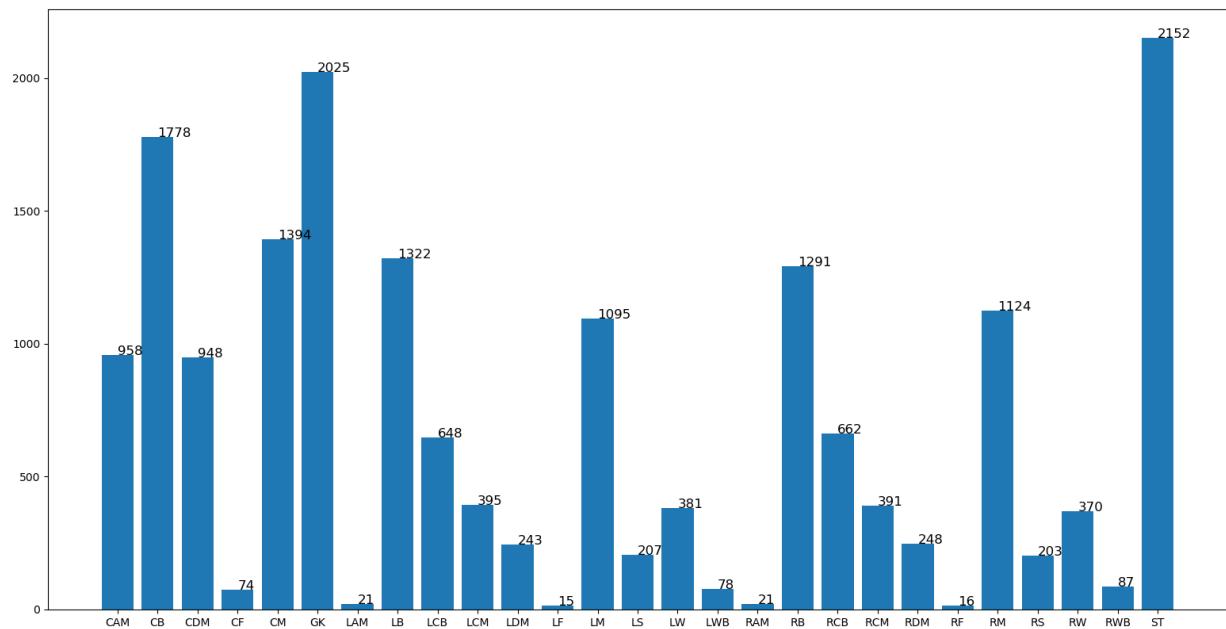
```
cnt = 0
```

```
while cnt<len(count.Position):
```

```
    g.text(count.Position[cnt],count.Player[cnt],str(count.Player[cnt]),size="large")
```

```
    cnt+=1
```

```
g.show()
```



Conclusion:

Maximum player are at ST (Striker) position which are 2152 players then after it is GK (Goal Keeper) with 2025 and then 1778 CB (Corner Back).

Q.3 Total wage invested in each position

```
position_d = p.read_excel(r"Football Data New Excel Format.xlsx","Position")
wage_d = p.read_excel(r"Football Data New Excel Format.xlsx","Wage")

df = p.concat([position_d,wage_d],axis=1)

count = s.sql("select position,sum(wage)\\"Wage\\" from df group by position;")

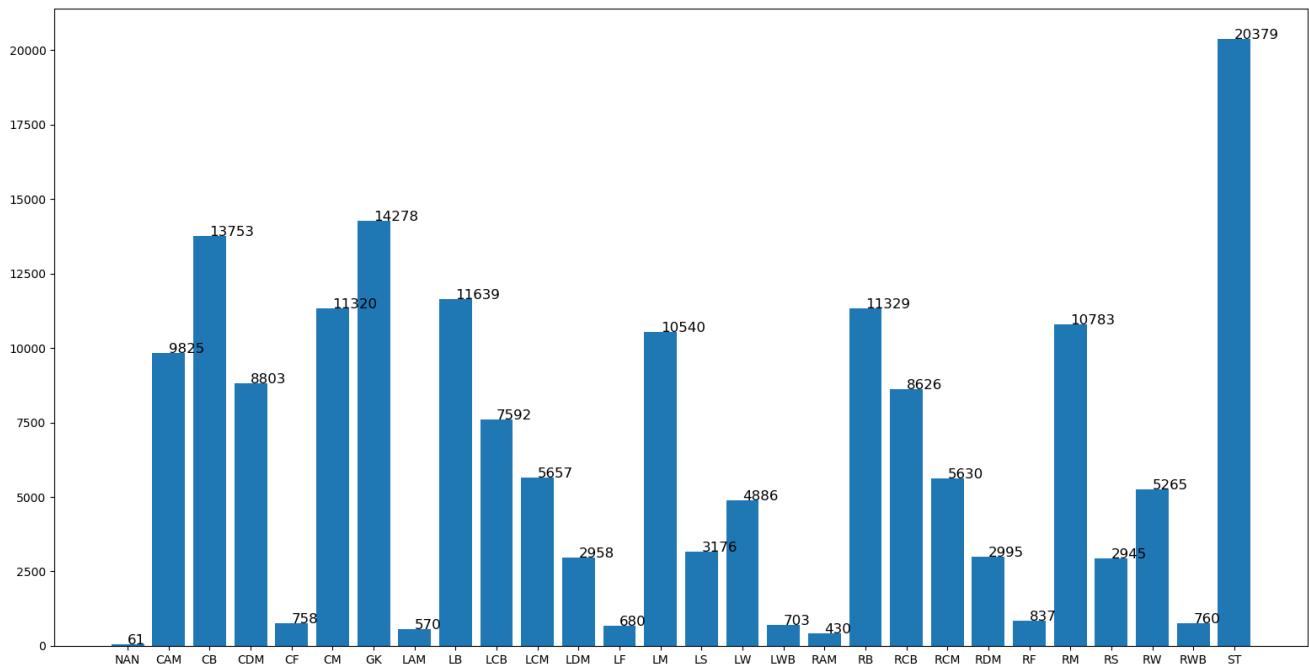
count= count.fillna("NAN")

g.bar(count.Position,count.Wage)

cnt = 0

while cnt<len(count.Position):
    g.text(count.Position[cnt],count.Wage[cnt],str(count.Wage[cnt]),size="large")
    cnt+=1

g.show()
```



Conclusion:

Maximum wage is being paid to players which are play at ST (striker) which is 20M Euros

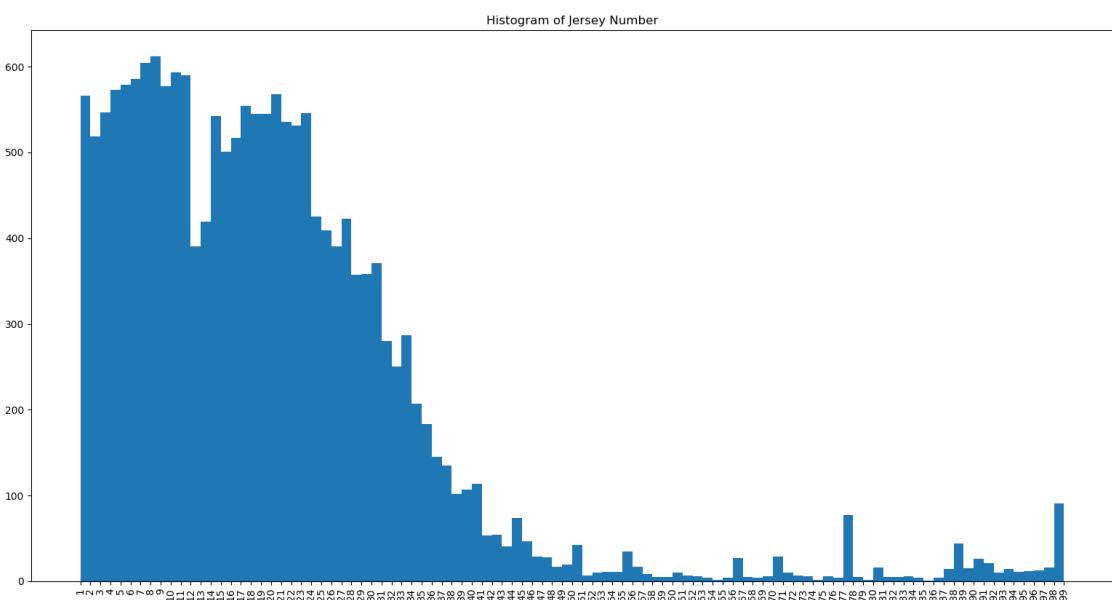
- Jersey Number

Q.1 Total count of different Jersey number of player

```
jersey_d = pd.read_excel(r"Football Data New Excel Format.xlsx","Jersey")
x = set(jersey_d.Jersey)
print(x)
# 1 to 99
```

Q.2 Player count in each number

```
g.hist(jersey_d.Jersey,list(x))
g.xticks(list(range(1,100)),labels=list(range(1,100)),rotation=90)
g.title("Histogram of Jersey Number")
g.show()
```



Conclusion:

Maximum player in Fifa are with jersey number 8, or overall we can say 1 to 40.

- **Loan**

Q.1 Total Loaned players in world

```
loan_d = pd.read_excel(r"Football Data New Excel Format.xlsx","Loan")
```

```
loan_d=loan_d.dropna()
```

```
#Total players
```

```
len(loan_d)
```

```
#1264
```

```
#Total different Clubs
```

```
len(set(loan_d["Loaned"]))
```

```
#341
```

Q.2 Club wise max loaned

```
df = pd.read_sql("Select  
count(*)\"Player\", \"Loaned\" from loan_d group by \"Loaned\" order by Player desc limit 10;")
```

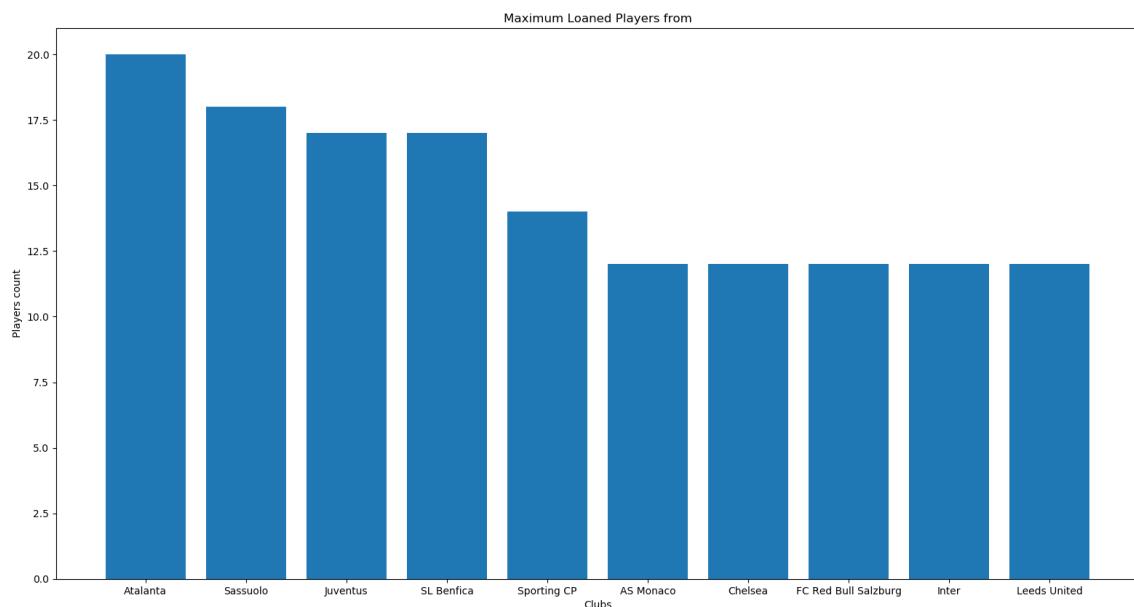
```
g.title("Maximum Loaned Players from")
```

```
g.xlabel("Clubs")
```

```
g.ylabel("Players count")
```

```
g.bar(df.Loaned,df.Player)
```

```
g.show()
```



Conclusion

Atalanta club has loaned maximum players then after Sassuolo and then Juventus.

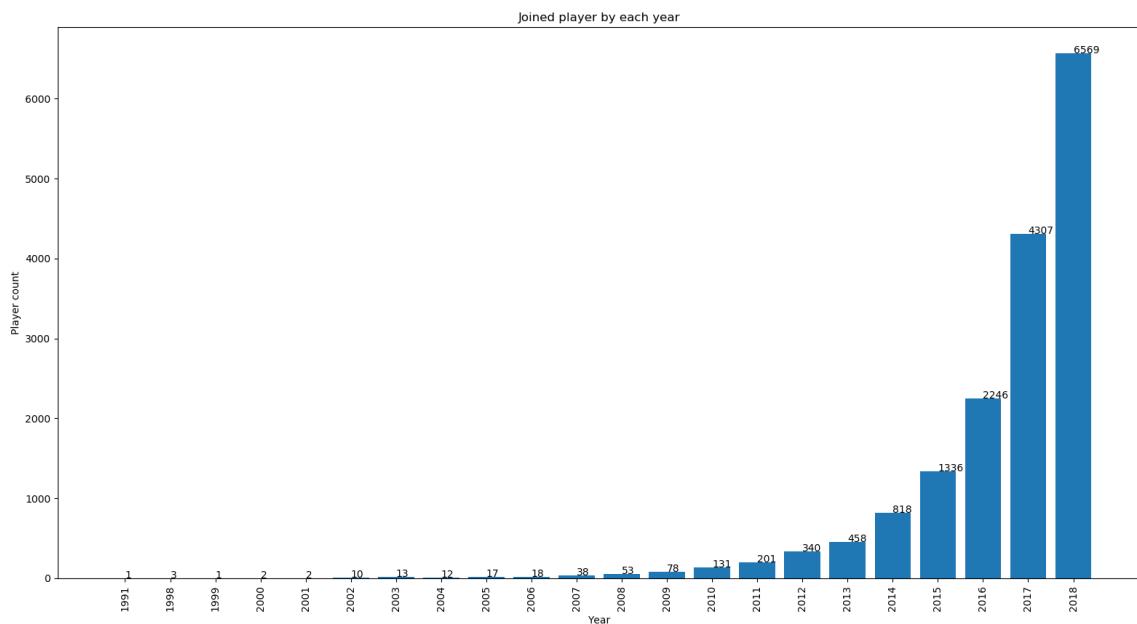
- **Joined From**

Q.1 Joined player by each year

```
join_d = pd.read_excel(r"Football Data New Excel Format.xlsx","Join")

df = sqldf("select
strftime('%Y',Joined) 'Year',count(*) 'Player' from join_d group by year order by year")

g.bar(df.Year,df.Player)
g.title("Joined player by each year")
g.xlabel("Year")
g.ylabel("Player count")
cnt = 0
while cnt<len(df.Year):
    g.text(df.Year[cnt],df.Player[cnt],str(df.Player[cnt]))
    cnt+=1
g.xticks(rotation=90)
g.show()
```



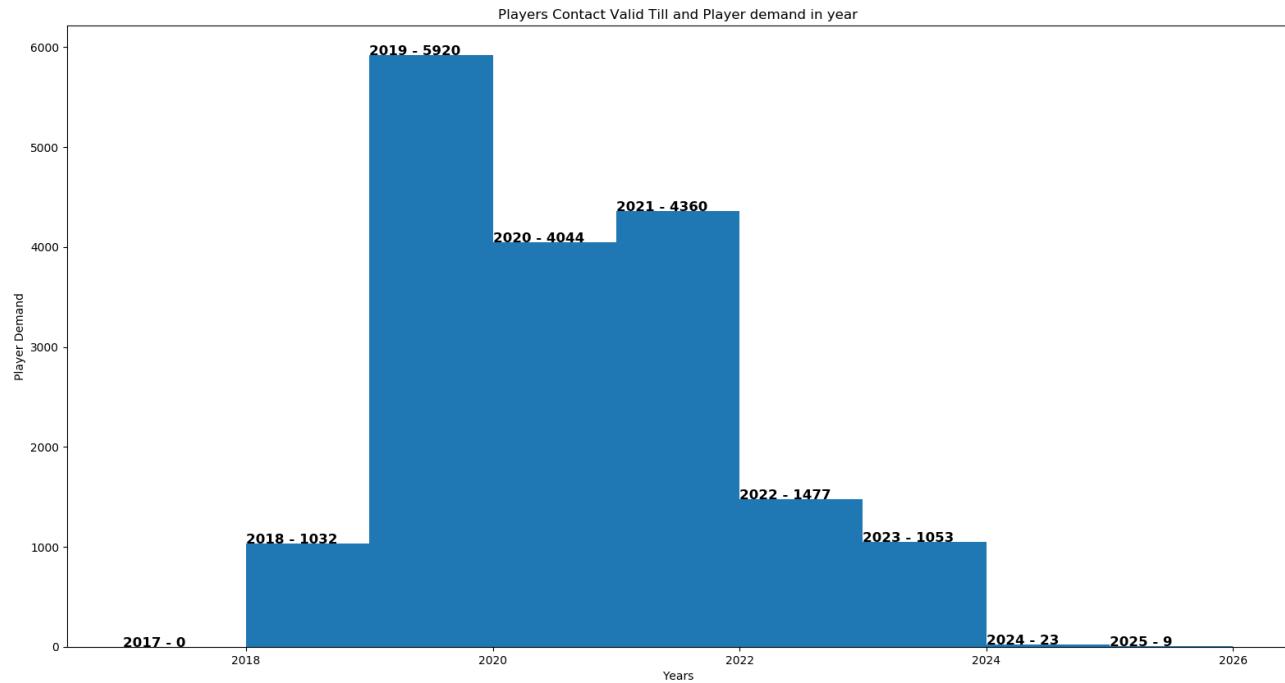
Conclusion:

Maximum player joined in 2018 and the ratio starts growing very increasingly from year 2013. It means the speedy rise of awareness of fifa is started from 2013.

- **Contact Valid Till**

Q.1 Find out the demand of players in different years.

```
##  
# Data is cleaned in some places date was in year only and In some places it was as dd-mm-yyyy.  
##  
import pandas as p  
import pandasql as s  
import pylab as g  
  
validtill_d = p.read_excel(r"Football Data New Excel Format.xlsx","ContactValidTill")  
  
list_yr=[]  
  
from datetime import datetime as dt  
  
a = dt(2018,4,14)  
  
for x in validtill_d.ValidTill:  
    if type(x) == type(a):  
        list_yr.append(x.year)  
    else:  
        list_yr.append(x)  
  
a,b,c = g.hist(list_yr,list(range(2017,2027)))  
  
cnt = 0  
  
while cnt<len(a):  
    g.text(b.item(cnt),a.item(cnt)+0.10,str(int(b.item(cnt)))+"  
"+str(int(a.item(cnt))),fontweight="bold",size="large")  
    cnt+=1  
  
g.title("Players Contact Valid Till and Player demand in year")  
g.xlabel("Years")  
g.ylabel("Player Demand")  
g.show()
```



Conclusion:

This graph shows the highest demand of players will be in 2019, and in previous year 2018 there was very less demand of players.

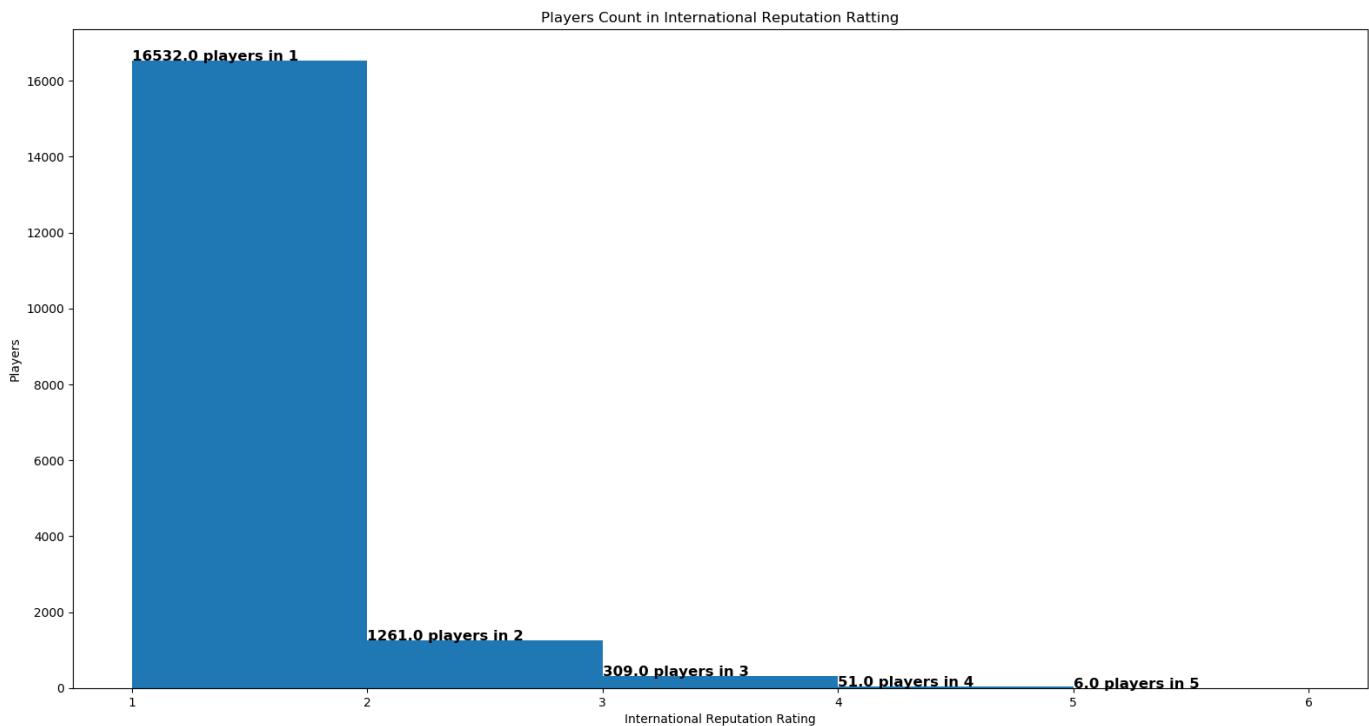
- **International Reputation**

Q.1 Histogram

```
import pylab as g
import pandasql as s
import pandas as p

reputation_d = p.read_excel(r"Football Data New Excel Format.xlsx","Reputation")

y,x,c = g.hist(list(reputation_d.Reputation),[1,2,3,4,5,6])
i = 0
while i<len(y):
    g.text(x[i],y[i],str(y[i])+" players in "+str(x[i]),size="large",fontweight="bold")
    i+=1
g.xlabel("International Reputation Rating")
g.ylabel("Players")
g.title("Players Count in International Reputation Rating")
g.show()
```



Conclusion:

This graph shows the maximum players have very less international reputation.

- **Weak Foot**

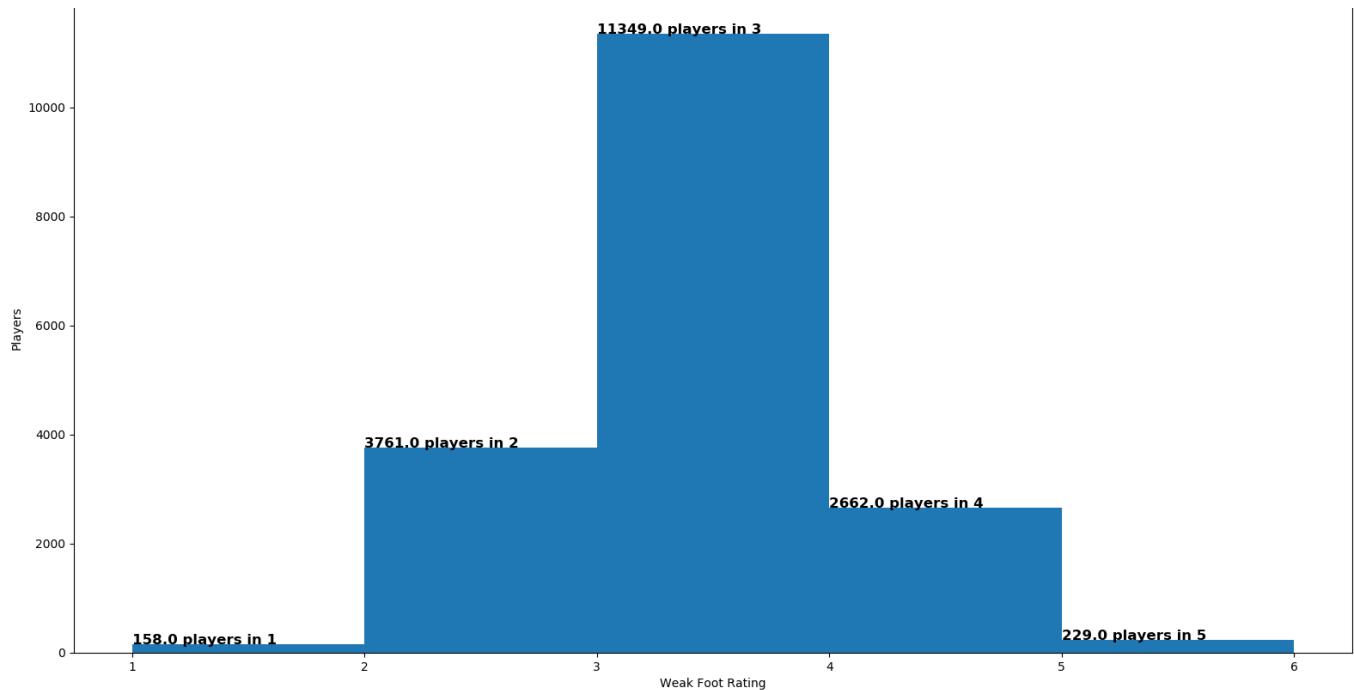
Q.1 Histogram

```
import pylab as g
import pandasql as s
import pandas as p

wfoot_d = p.read_excel(r"Football Data New Excel Format.xlsx","WeakFoot")

y,x,c = g.hist(list(wfoot_d.WeakFoot),[1,2,3,4,5,6])
i = 0
while i<len(y):
    g.text(x[i],y[i],str(y[i])+" players in "+str(x[i]),fontweight="bold",size="large")
    i+=1
g.xlabel("Weak Foot Rating")

g.ylabel("Players")
g.title("Players Count in Weak Foot Rating")
g.show()
```



Conclusion:

This graph shows the very large number of players are intermediate in weak foot.

- **Age**

Q.1 Min & Max Height

```
age_d = pd.read_excel(r"C:\Users\Hp\Documents\Data Science Project\data\Football Data New Excel Format.xlsx","Age")
```

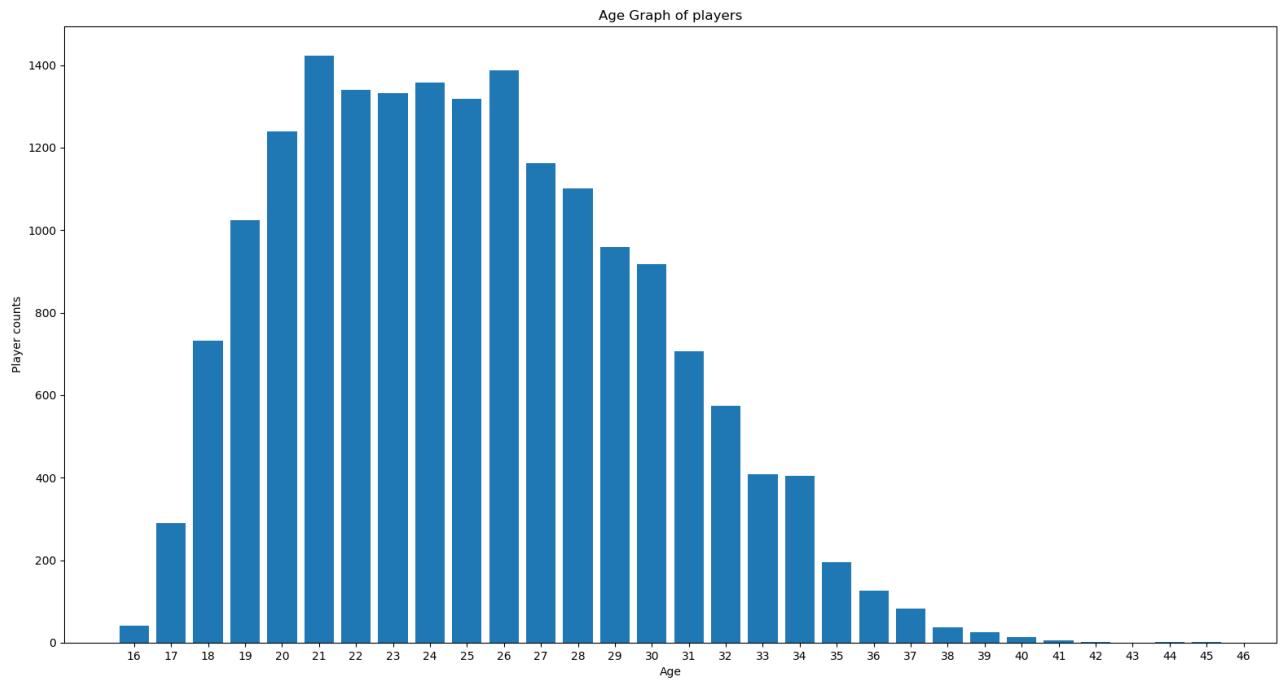
```
age_d.Age.max()  
# 46
```

```
age_d.Age.min()  
# 16
```

Q.2 Histogram

```
df = pd.read_sql("Select Age,count(Age)'Player' from age_d group by Age;")
```

```
g.title("Age Graph of players")  
g.xlabel("Age")  
g.ylabel("Player counts")  
g.bar(df.Age,df.Player)  
g.xticks(ticks=list(range(16,47)),labels=list(range(16,47)))  
g.show()
```



Conclusion:

Maximum players age 20 – 26. Very less players of age 16 and range of 33-46.

- **Height**

Q.1 Min & Max Height

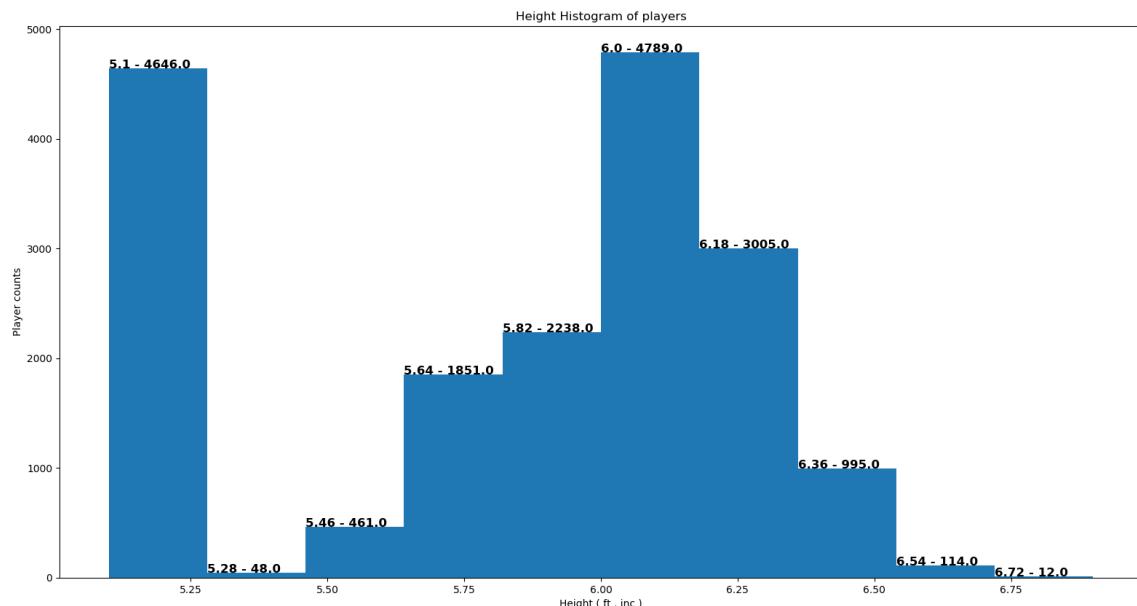
```
height_d = pd.read_excel(r"Football Data New Excel Format.xlsx","Height")
```

```
height_d.Height.max()  
# 6.9 ft. in.
```

```
height_d.Height.min()  
# 5.1 ft. in.
```

Q.2 Histogram

```
g.title("Height Histogram of players")  
g.xlabel("Height ( ft . inc )")  
g.ylabel("Player counts")  
a,b,c = g.hist(height_d.Height)  
cnt = 0  
while cnt<len(a):  
    g.text(b.item(cnt),a.item(cnt),str(round(b.item(cnt),2))+"  
"+str(a.item(cnt)),size="large",fontweight="bold")  
    cnt+=1  
g.show()
```



Conclusion:

Maximum player's Height is between 5.1 to 5.28 and 6.0 to 6.8.

- **Weight**

Q.1 Min & Max Weight

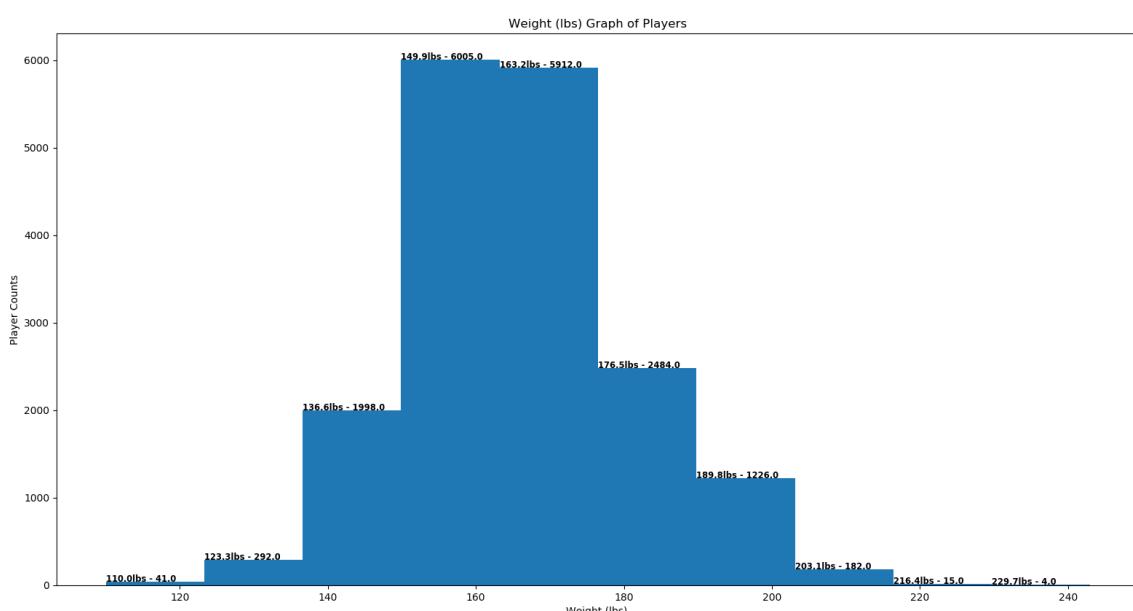
```
weight_d = p.read_excel(r"Football Data New Excel Format.xlsx","Weight")
weight_d.Weight.max()
# 243lbs
weight_d.Weight.min()
# 110lbs
```

Q.2 Histogram

```
df = s.sql("Select Weight,count(Weight)'Player' from weight_d group by Weight;")
g.title("Weight (lbs) Graph of Players")
g.xlabel("Weight (lbs) ")
g.ylabel("Player Counts")
a,b,c = g.hist(weight_d.Weight)
cnt = 0
while cnt<len(a):
    g.text(b.item(cnt),a.item(cnt),str(round(b.item(cnt),2))+"lbs
"+str(round(a.item(cnt),2)),size="small",fontweight="bold")
    cnt+=1
g.show()
```

Conclusion:

Maximum player are having 149.9lbs



- Task

Q.1 Total tasks

```
task_d = pd.read_excel(r" Football Data New Excel Format.xlsx","TaskPerformance")
task_d.columns
```

```
Index(['Crossing', 'Finishing', 'HeadingAccuracy', 'ShortPassing', 'Volleys',
       'Dribbling', 'Curve', 'FKAccuracy', 'LongPassing', 'BallControl',
       'Acceleration', 'SprintSpeed', 'Agility', 'Reactions', 'Balance',
       'ShotPower', 'Jumping', 'Stamina', 'Strength', 'LongShots',
       'Aggression', 'Interceptions', 'Positioning', 'Vision', 'Penalties',
       'Composure', 'Marking', 'StandingTackle', 'SlidingTackle', 'GKDiving',
       'GKHandling', 'GKKicking', 'GKPositioning', 'GKReflexes'],
      dtype='object')
```

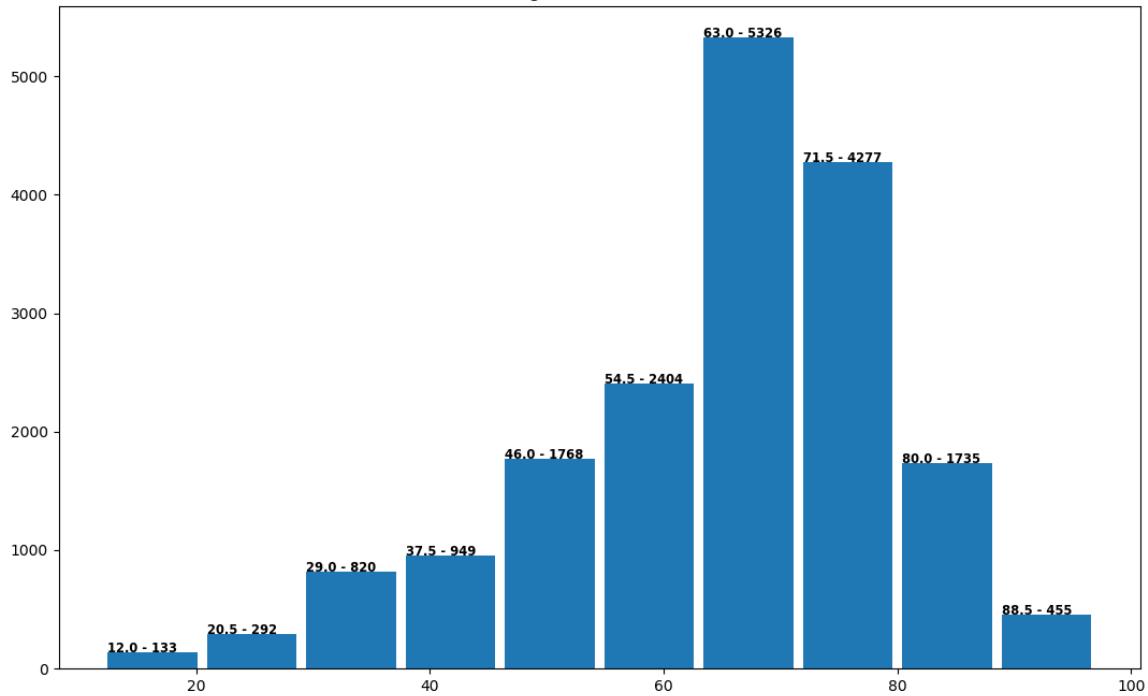
```
len(task_d.columns)
```

```
# 34
```

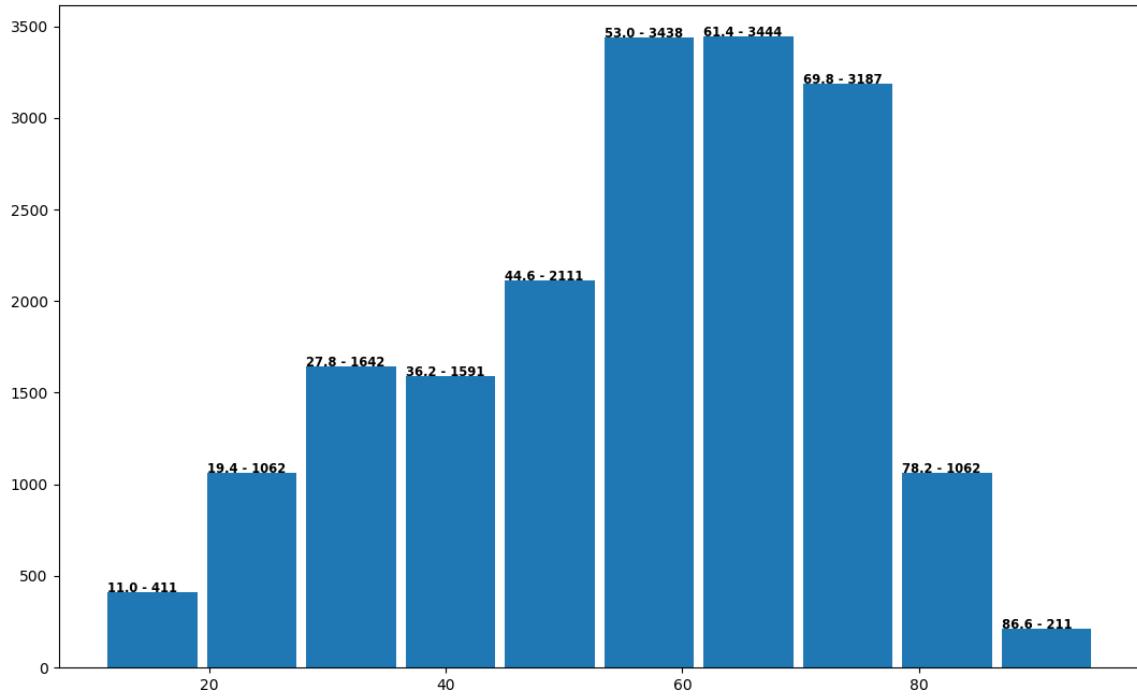
Q.2 Total players in each task

```
for x in task_d.columns:
    g.figure(figsize=(13,8))
    g.title("Histogram of "+x)
    a,b,c = g.hist(task_d[x])
    cnt=0
    while cnt<len(a):
        g.text(b.item(cnt),a.item(cnt),str(round(b.item(cnt),2)))+
        "+str(int(a.item(cnt))),fontweight="bold",size="small")
        cnt+=1
    g.savefig("fig-"+x+".png",figsize=(13,8))
    g.close()
```

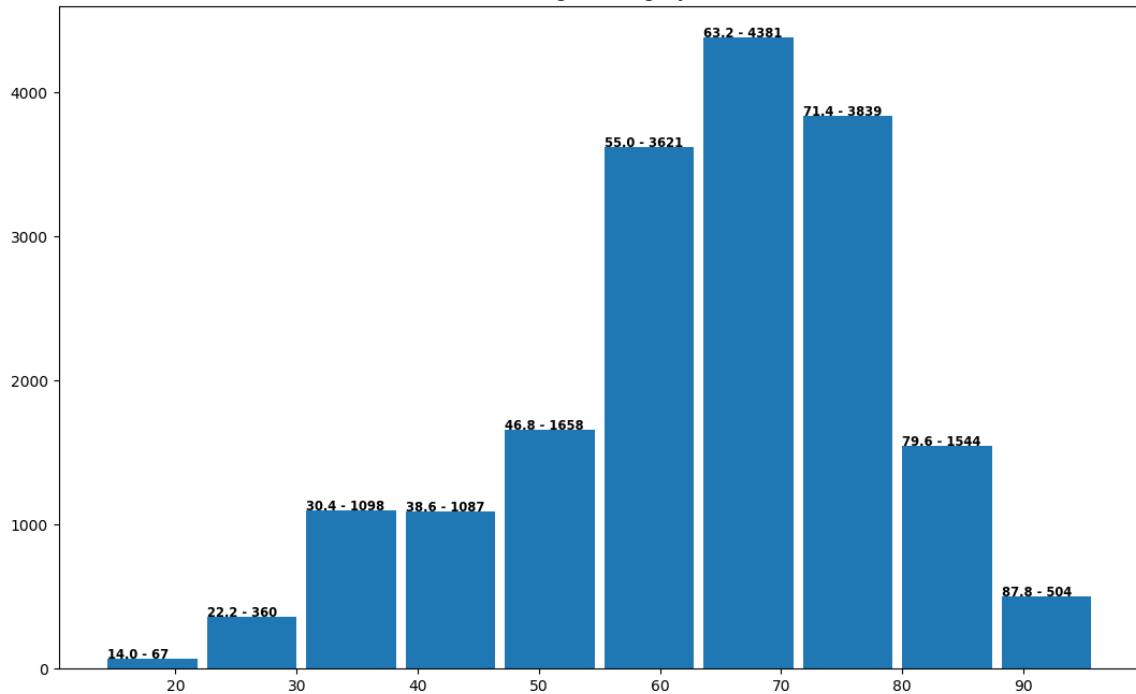
Histogram of Acceleration



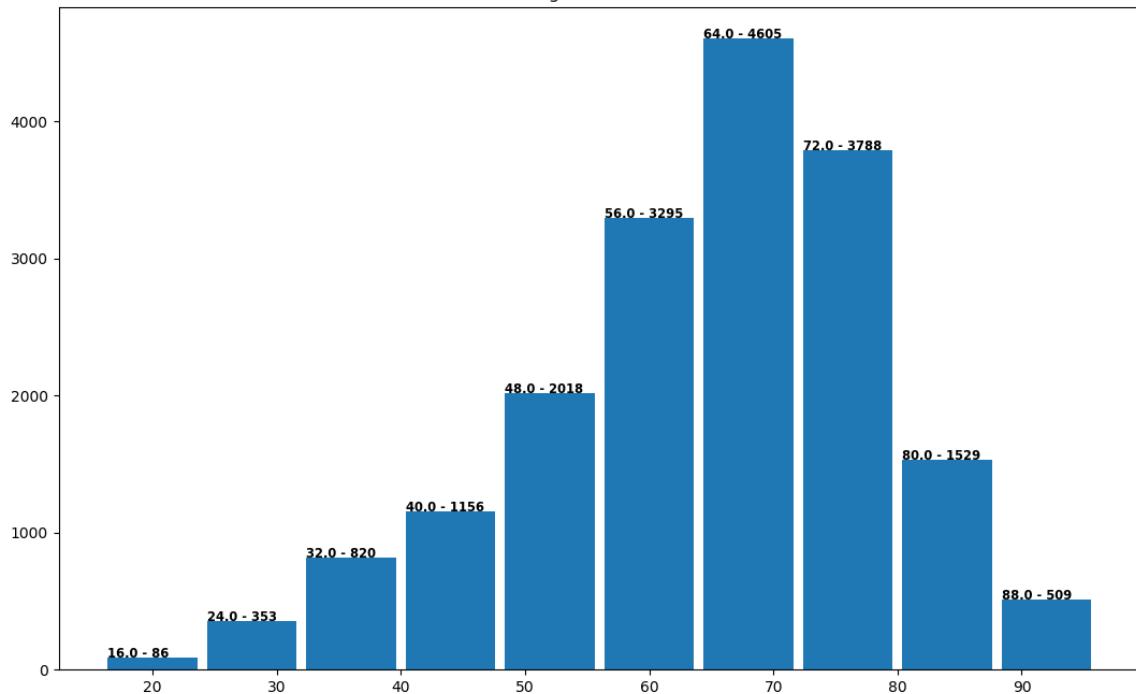
Histogram of Aggression



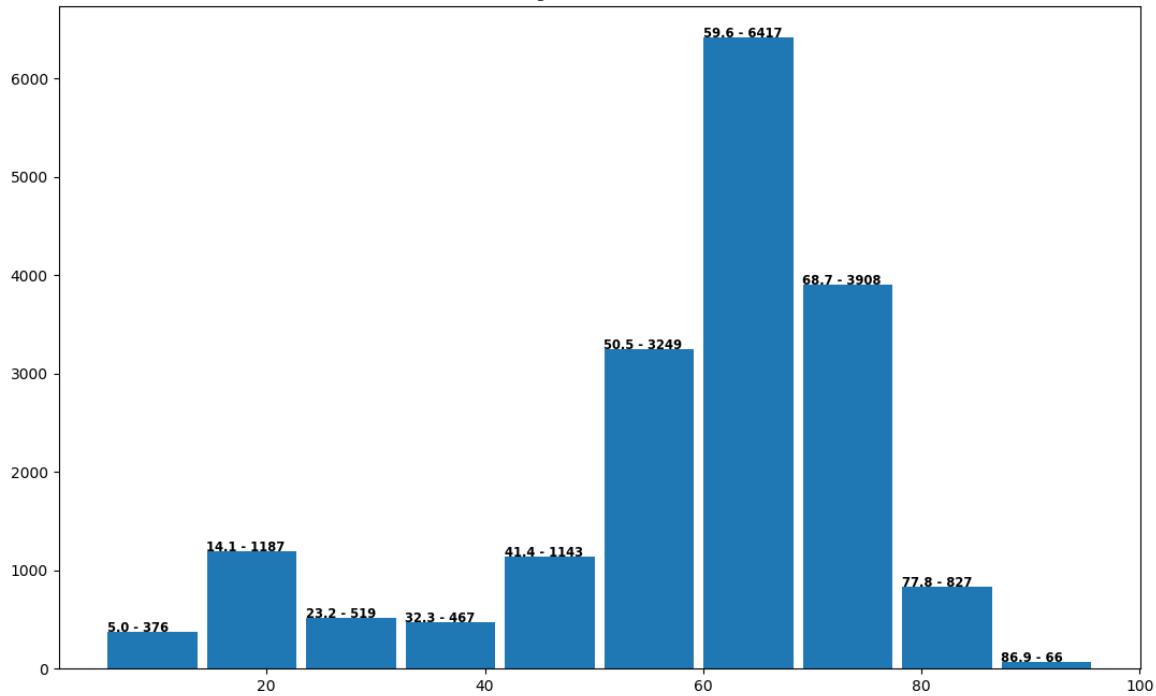
Histogram of Agility



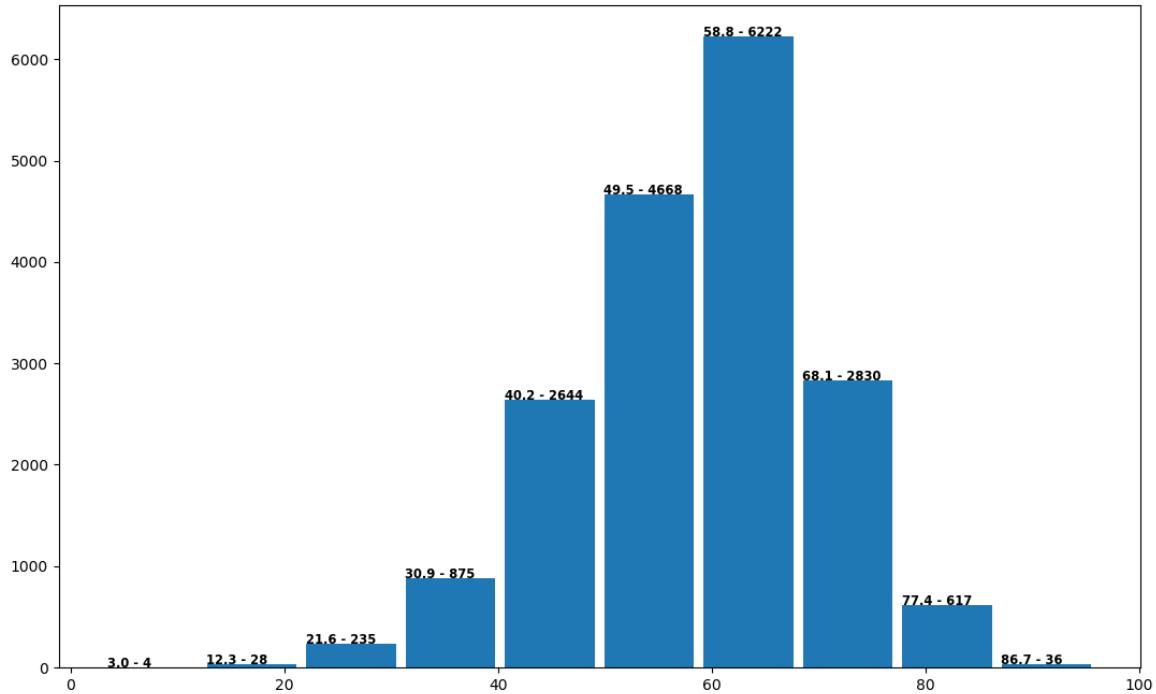
Histogram of Balance



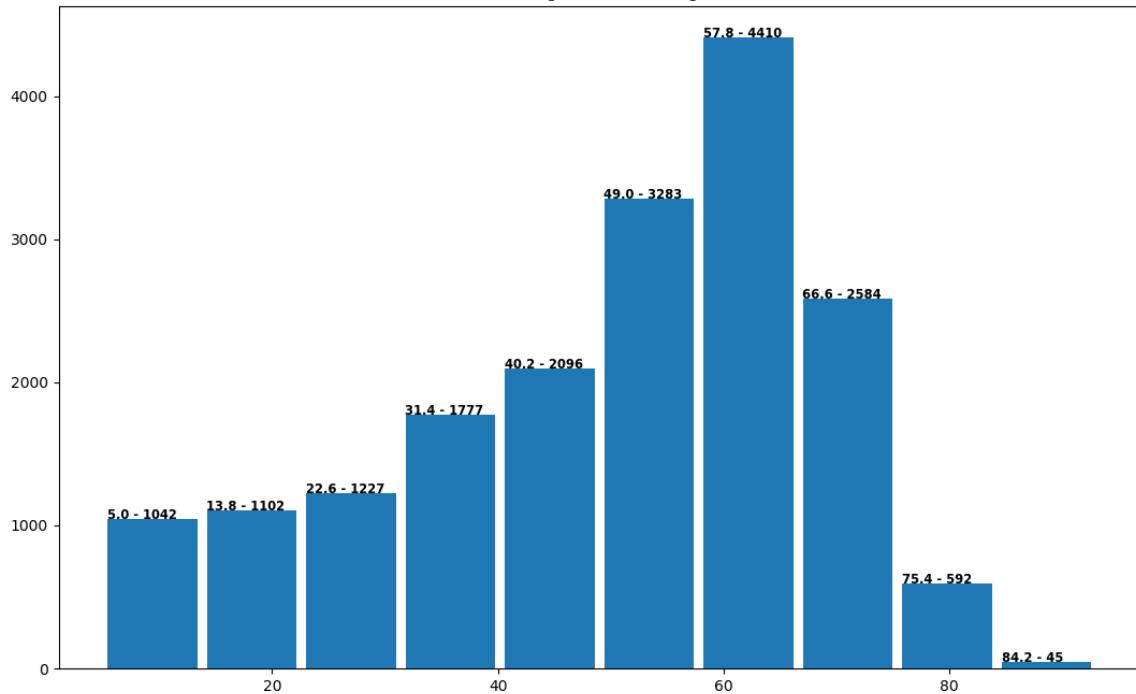
Histogram of BallControl



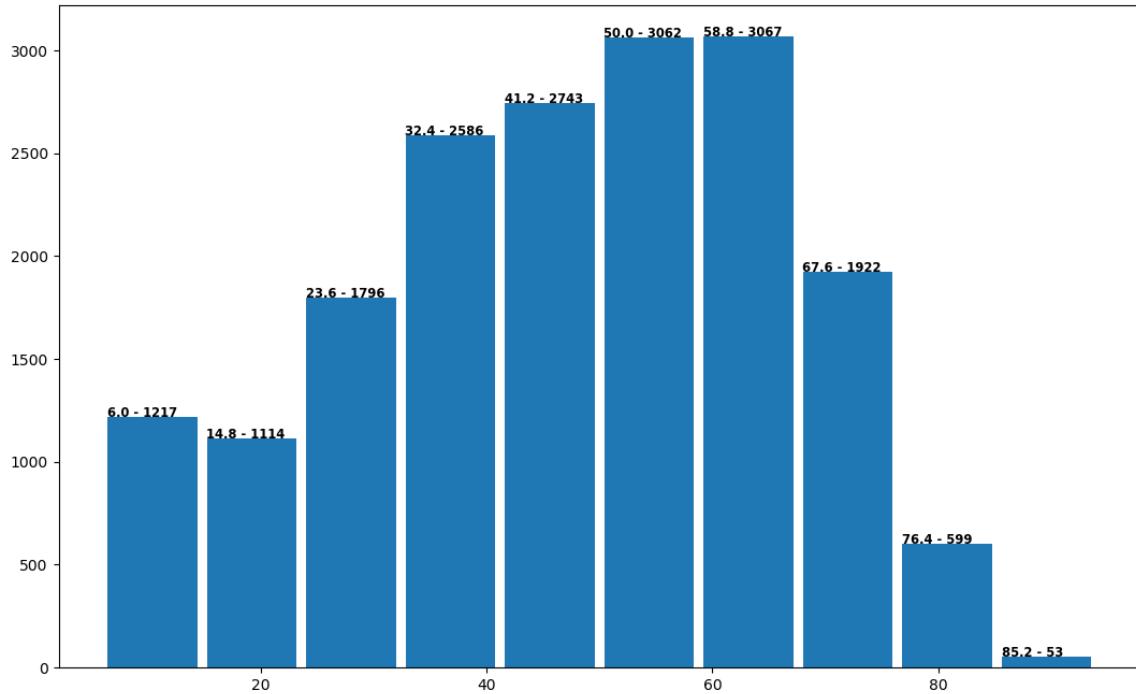
Histogram of Composure



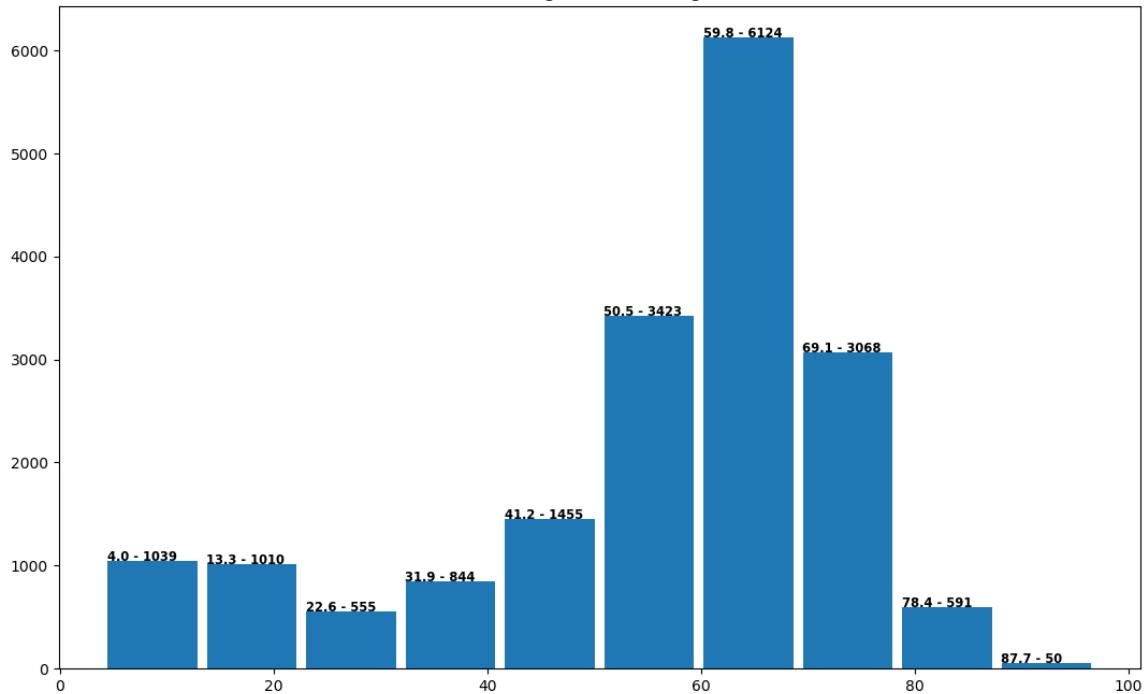
Histogram of Crossing



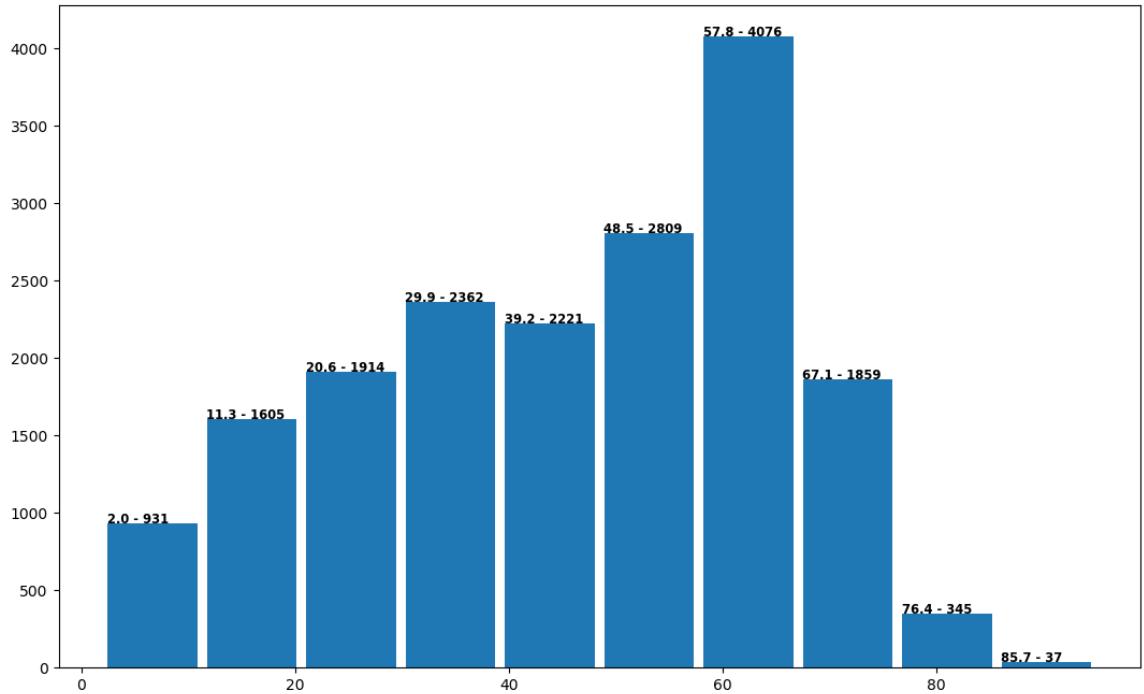
Histogram of Curve



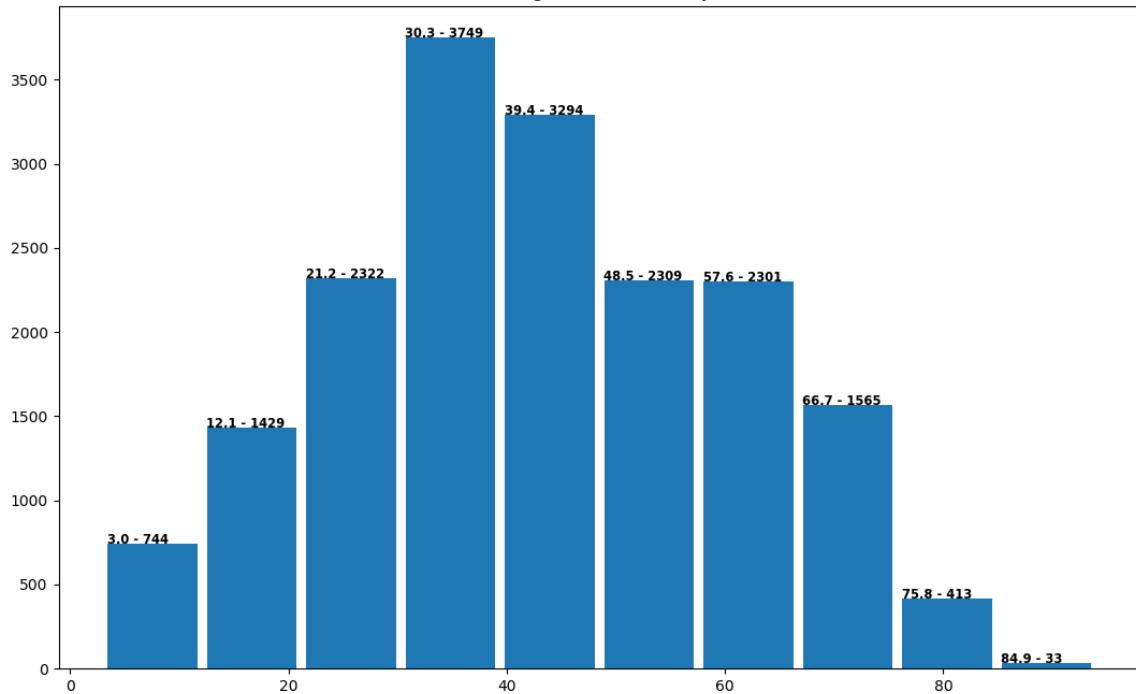
Histogram of Dribbling



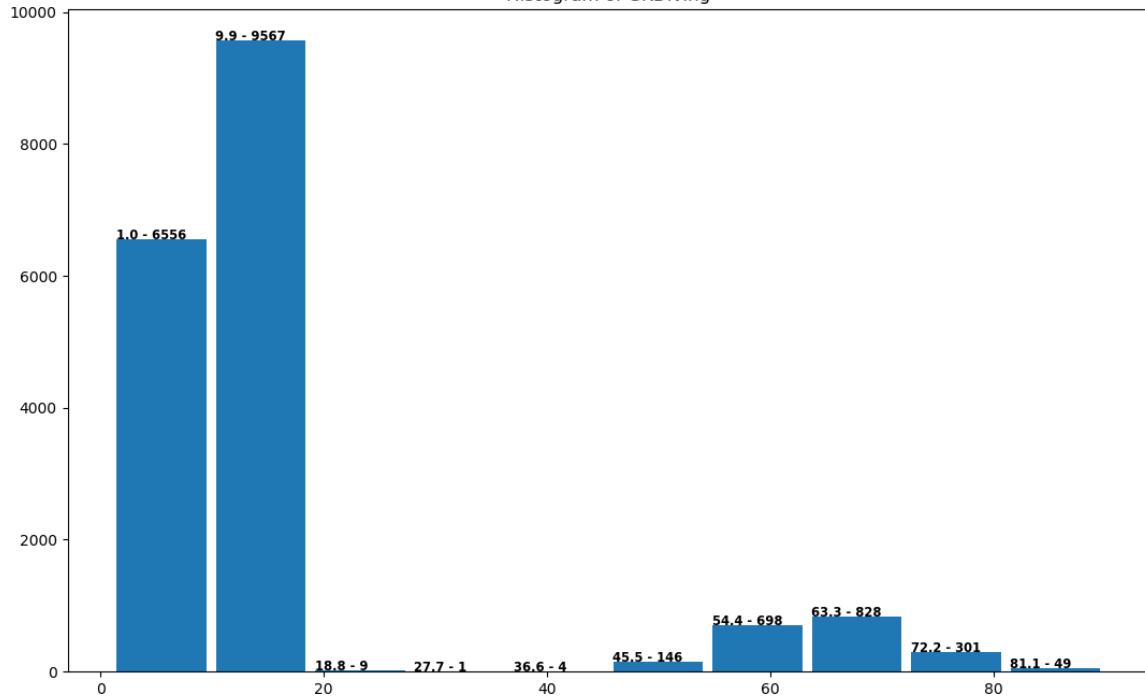
Histogram of Finishing



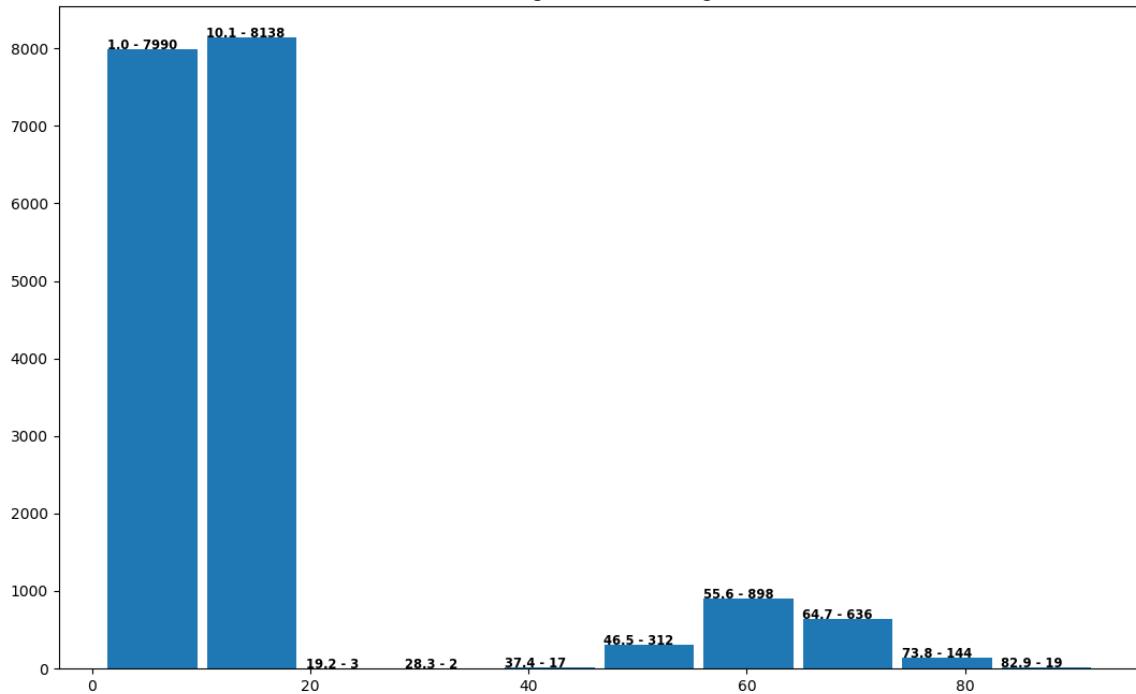
Histogram of FKAccuracy



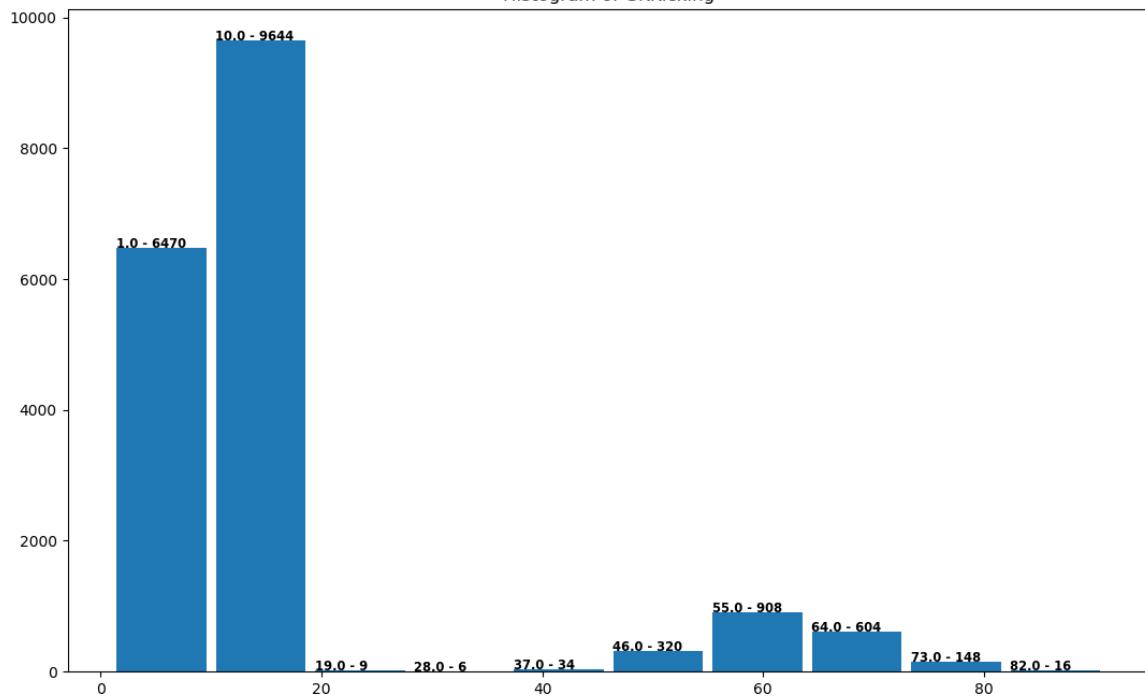
Histogram of GKDiving



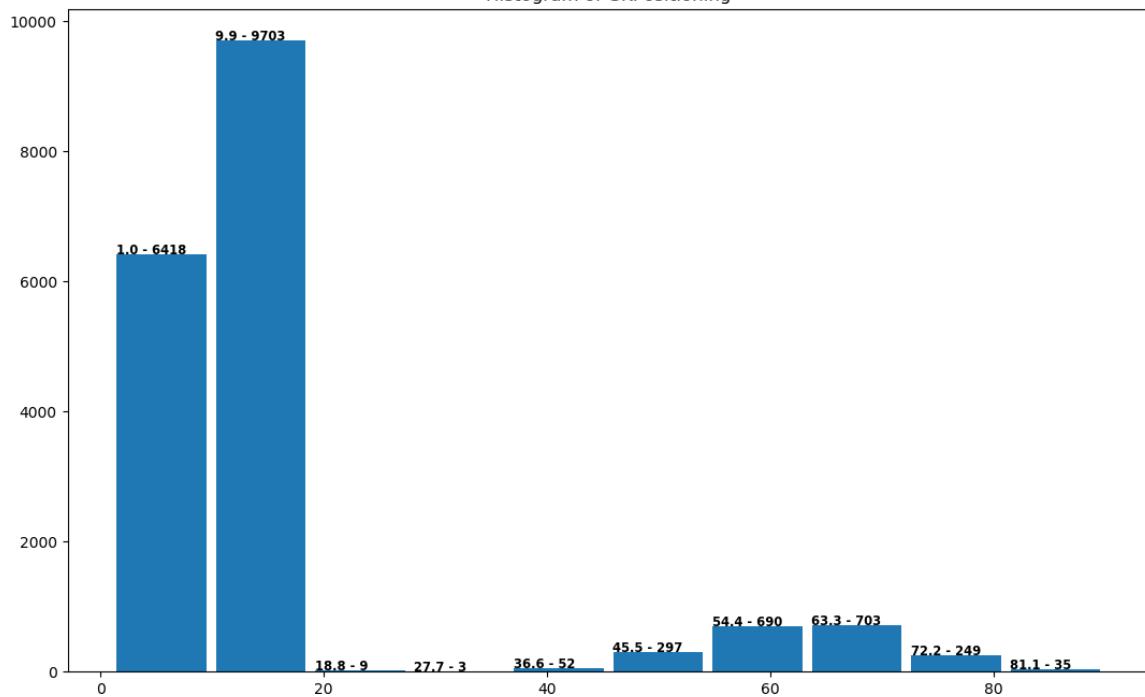
Histogram of GKHandling



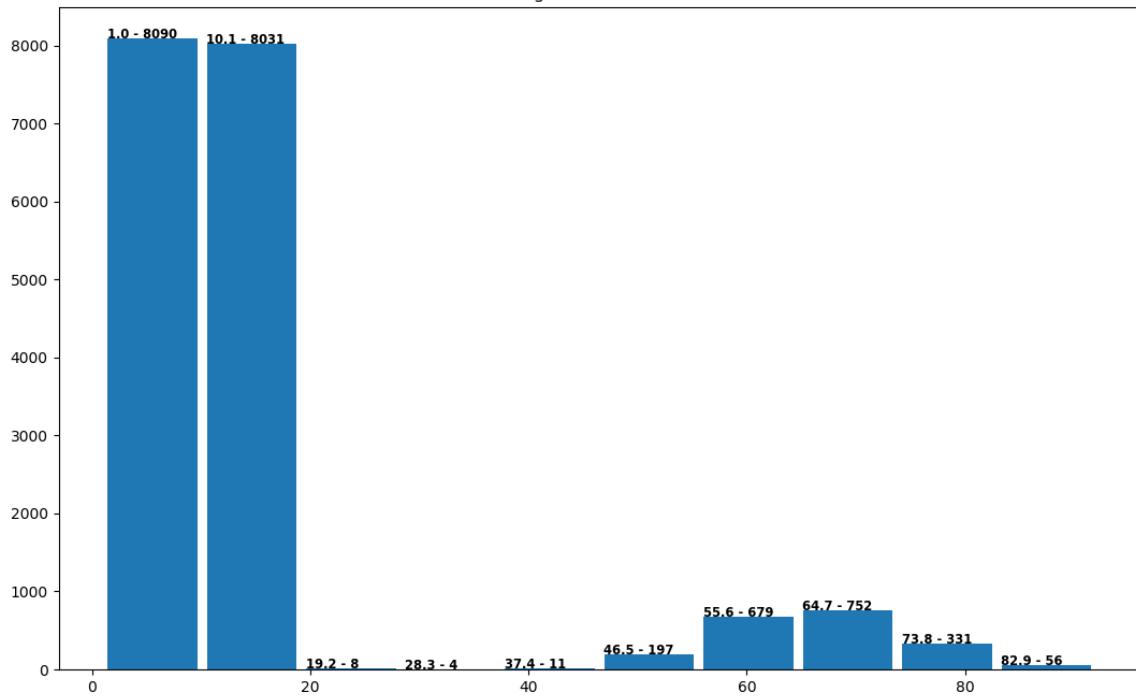
Histogram of GKKicking



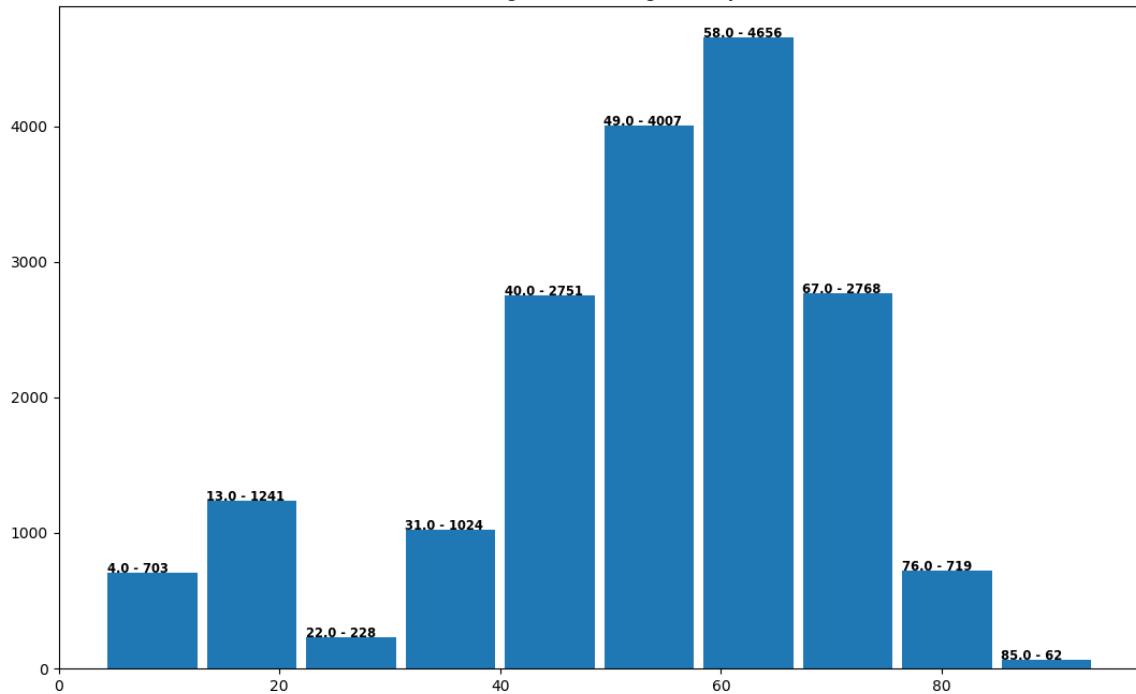
Histogram of GKPositioning



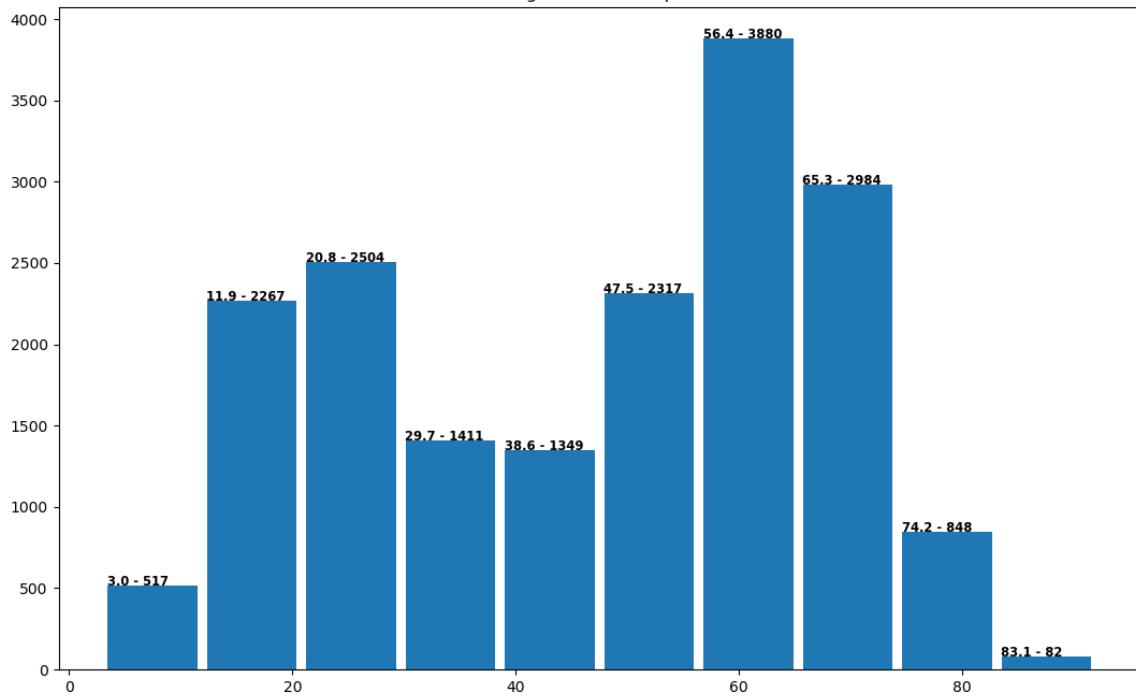
Histogram of GKReflexes



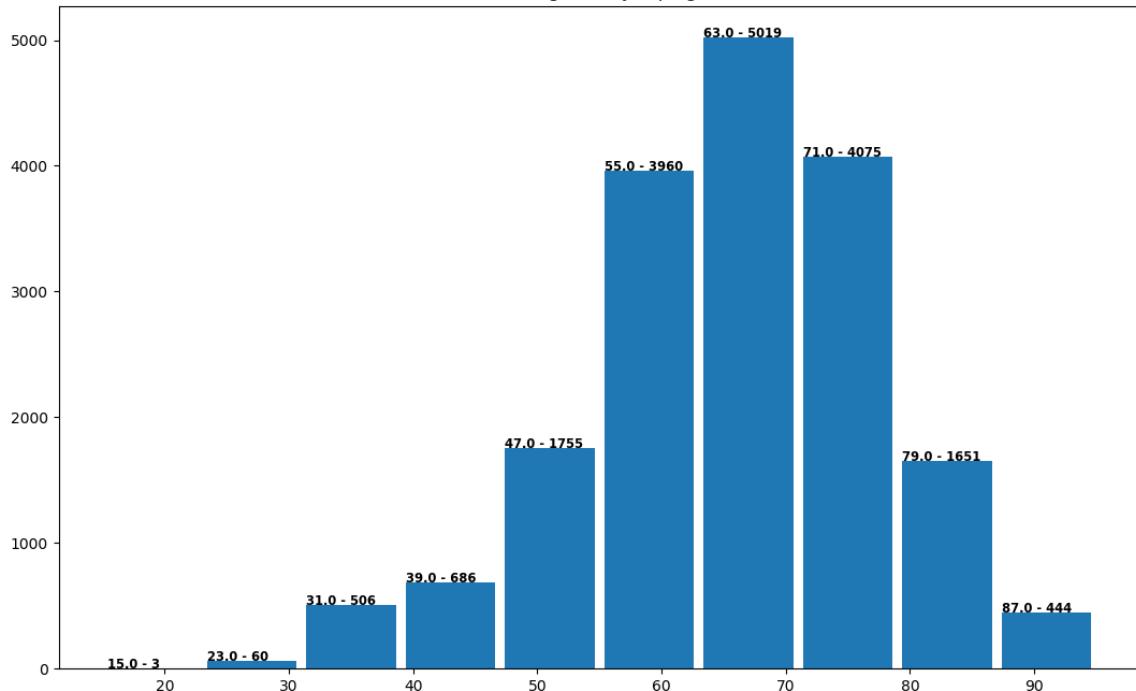
Histogram of HeadingAccuracy



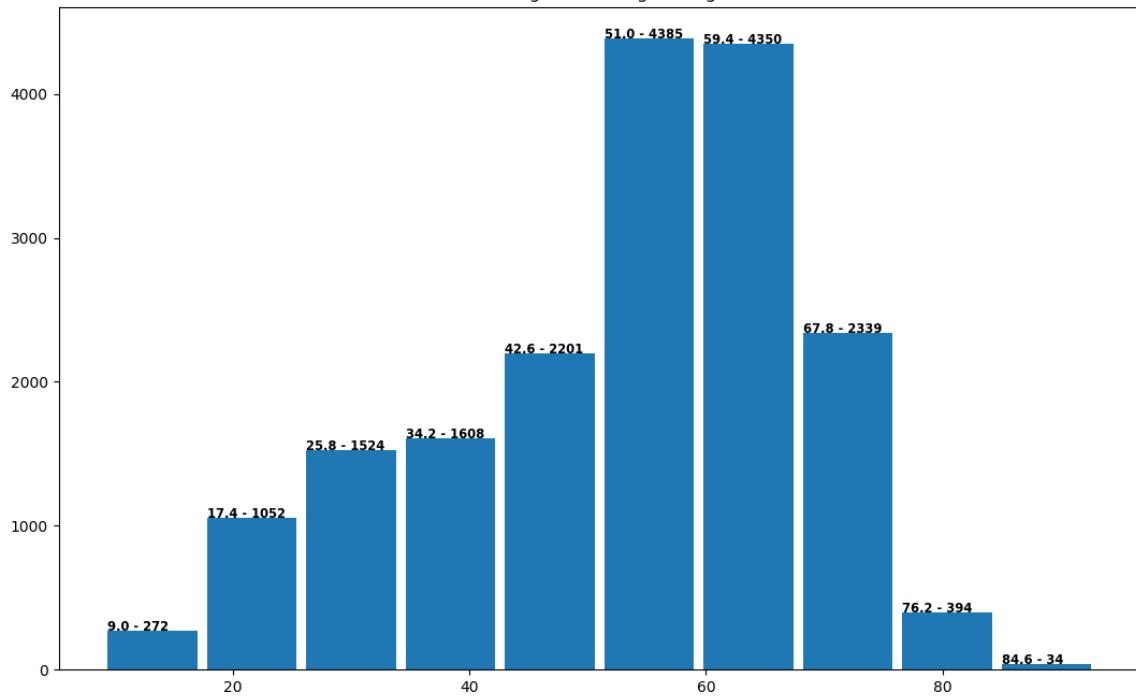
Histogram of Interceptions



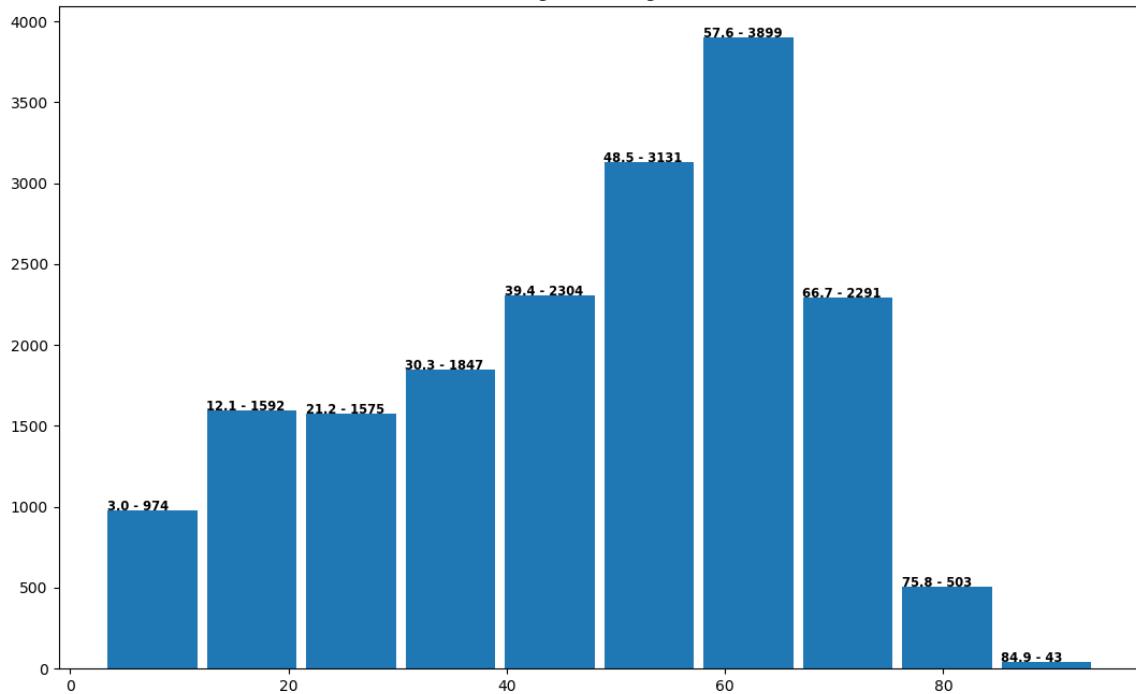
Histogram of Jumping



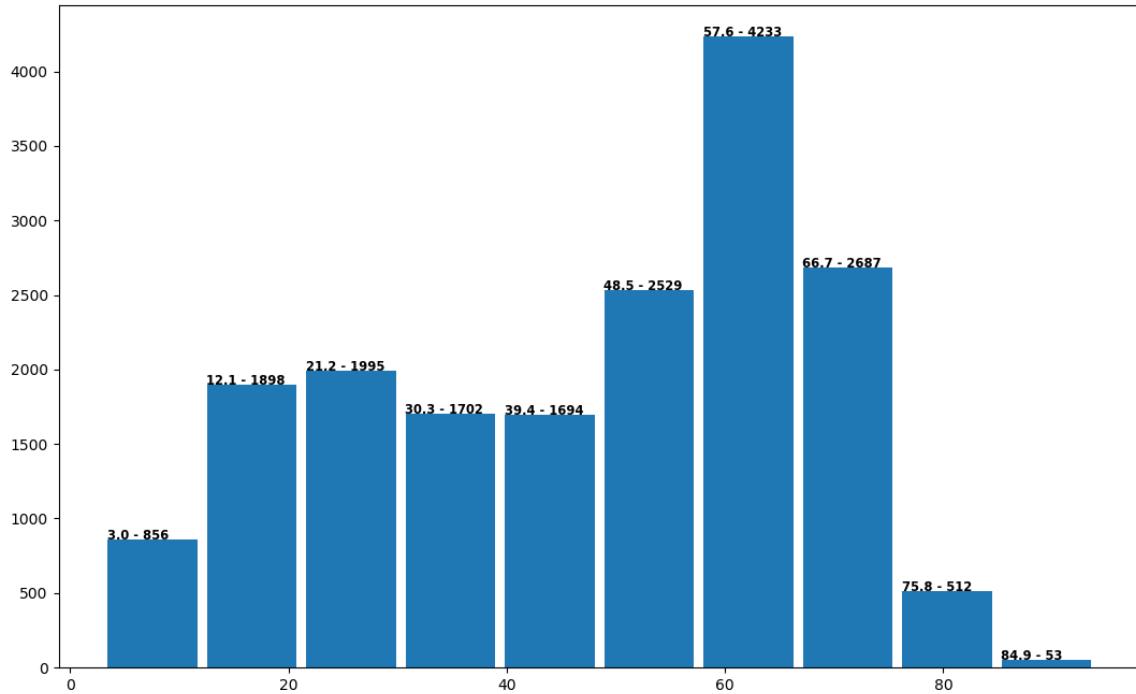
Histogram of LongPassing



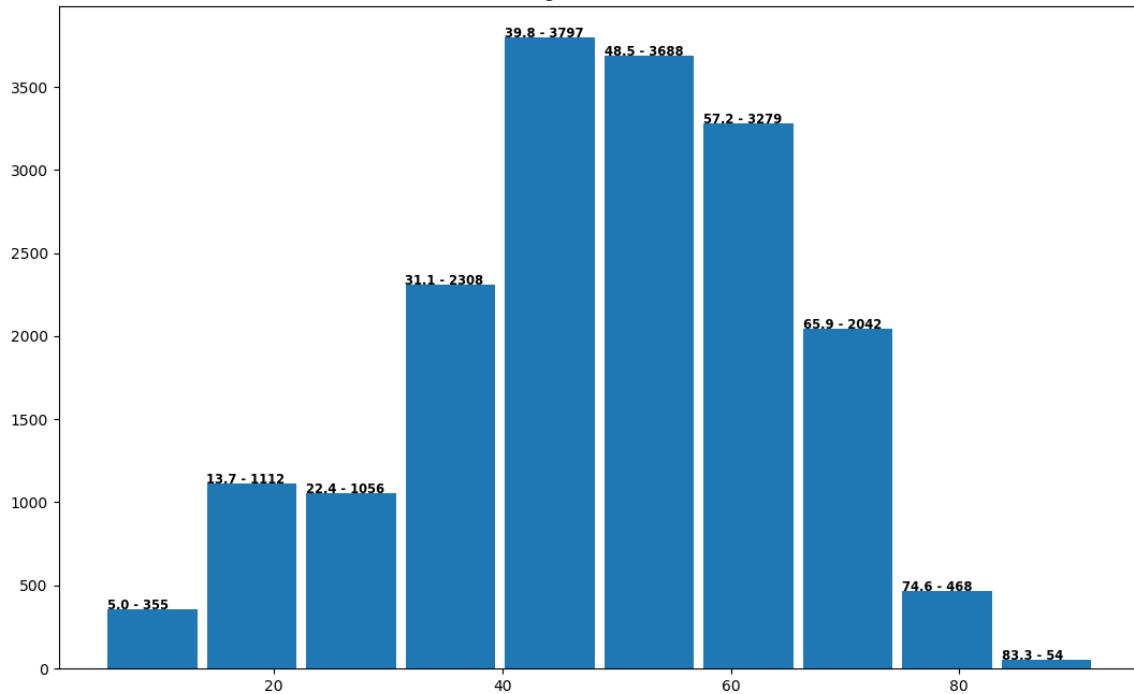
Histogram of LongShots



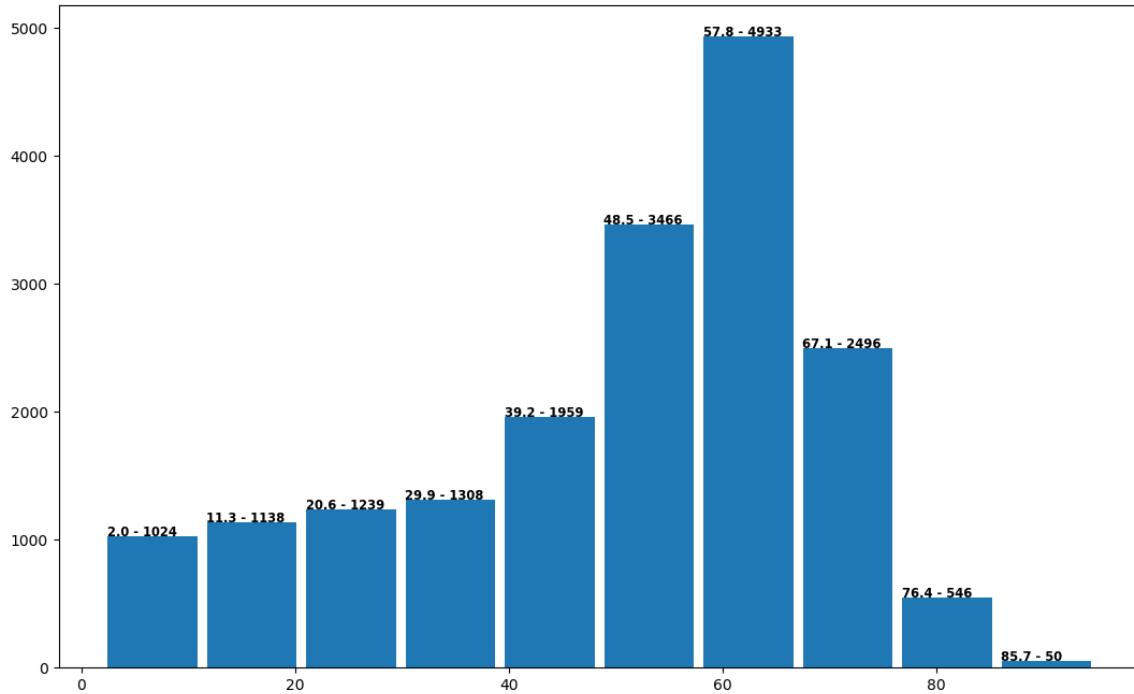
Histogram of Marking



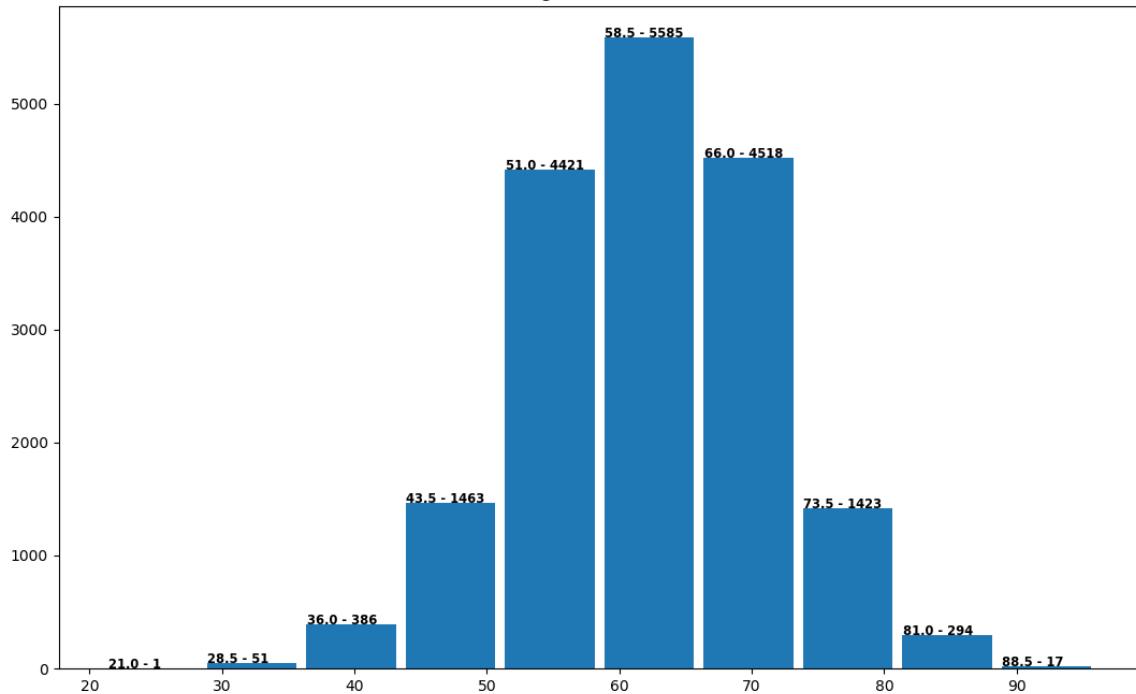
Histogram of Penalties



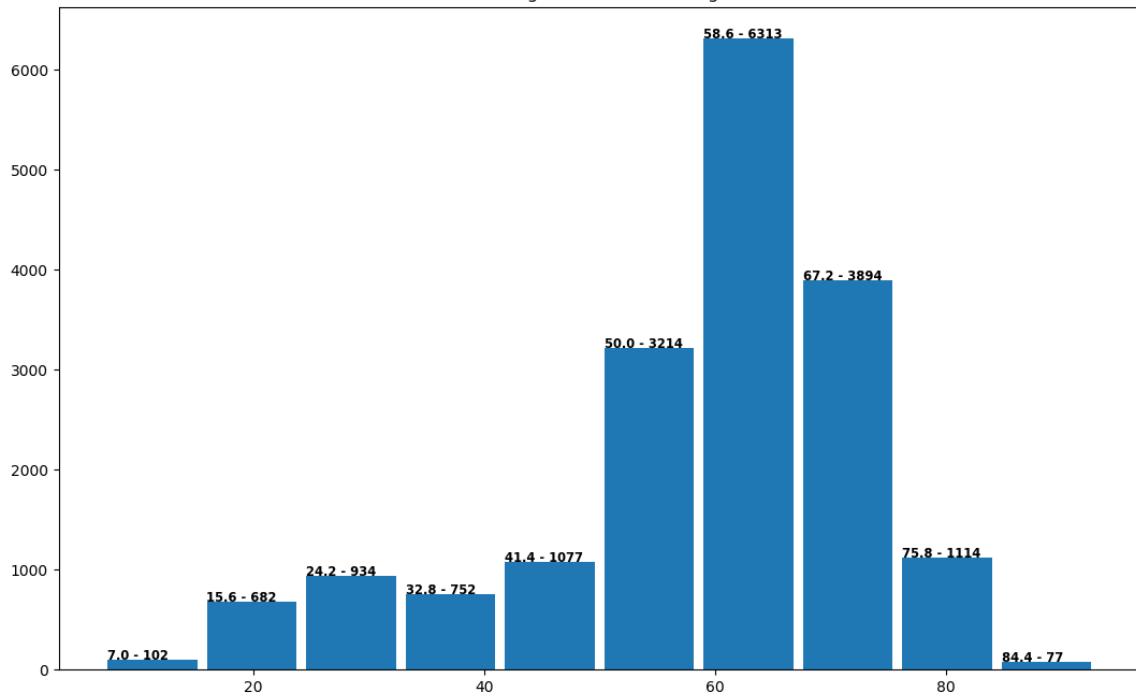
Histogram of Positioning

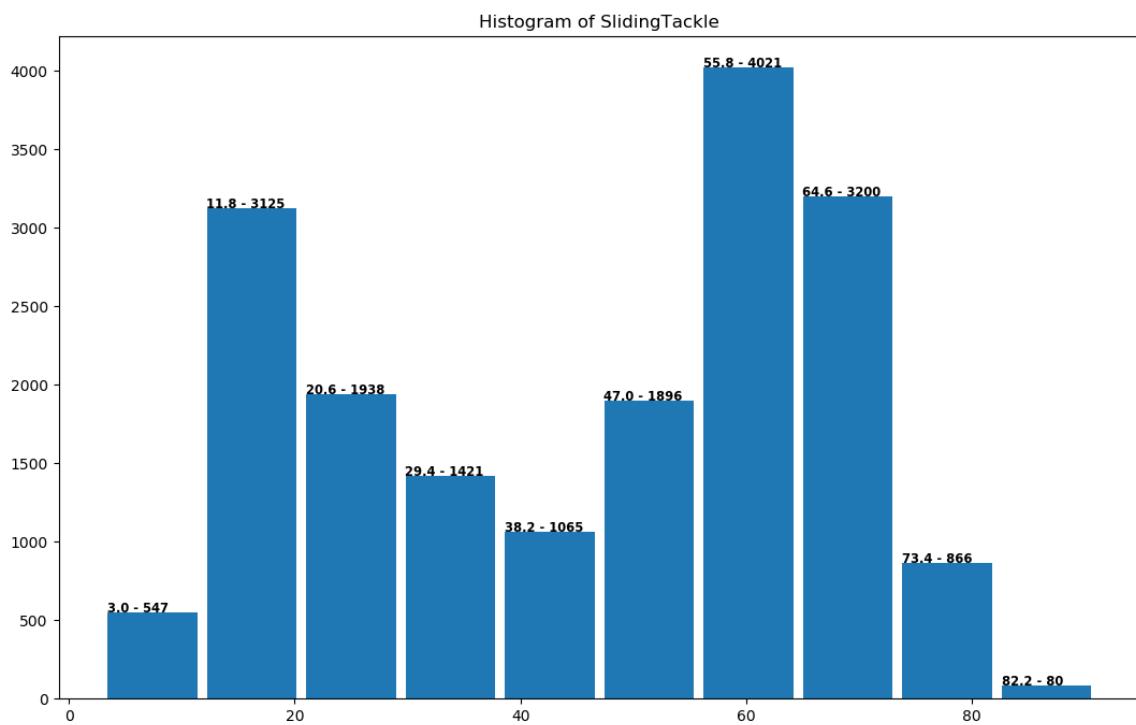
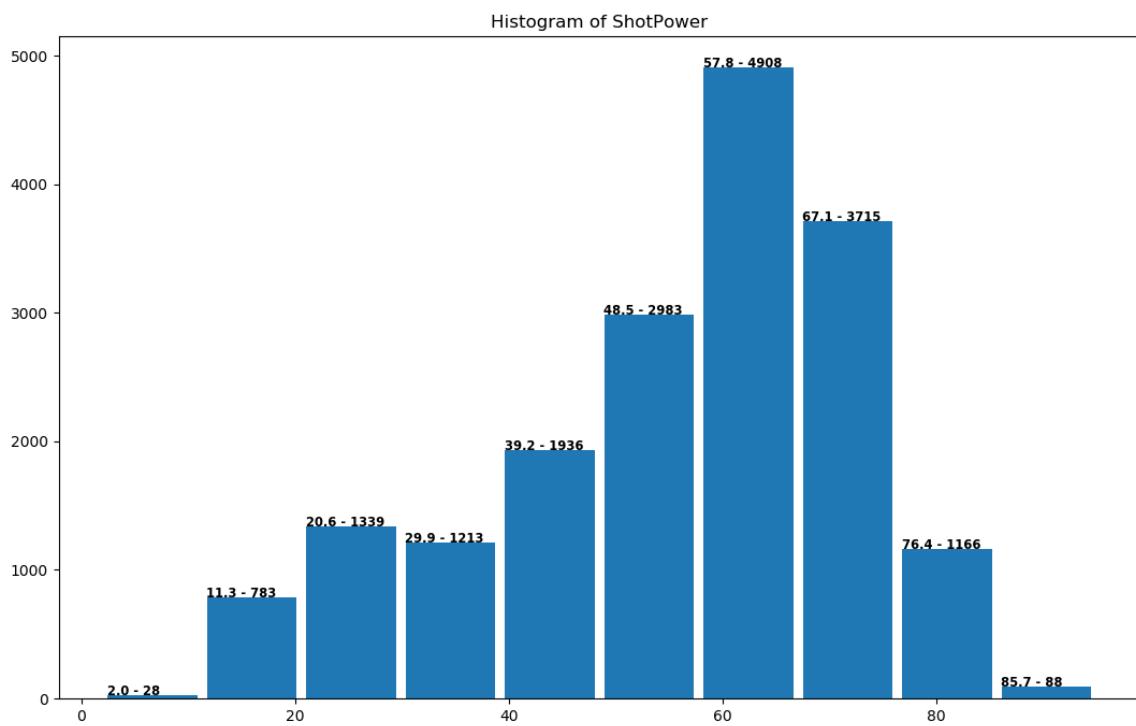


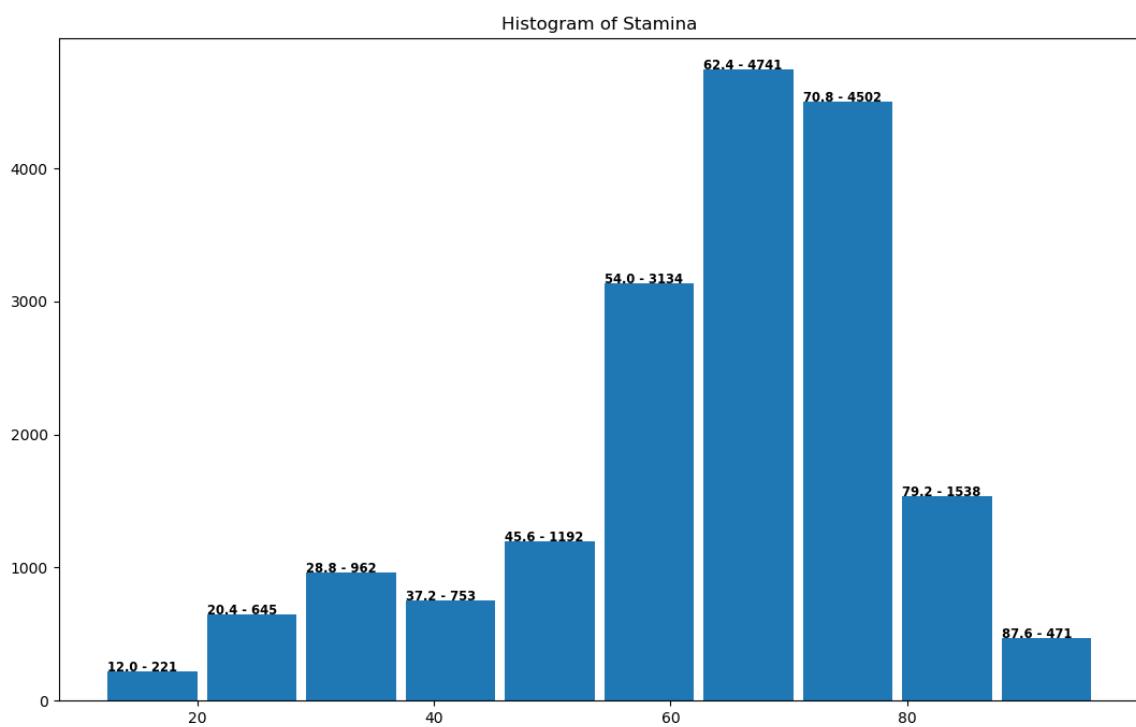
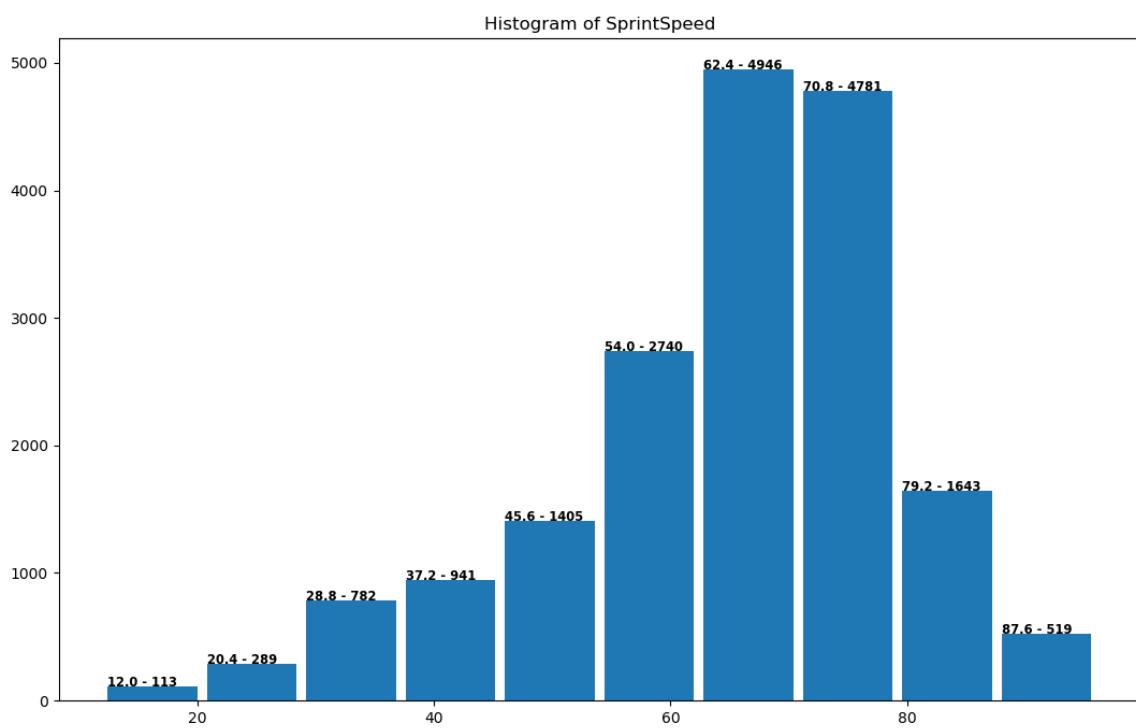
Histogram of Reactions

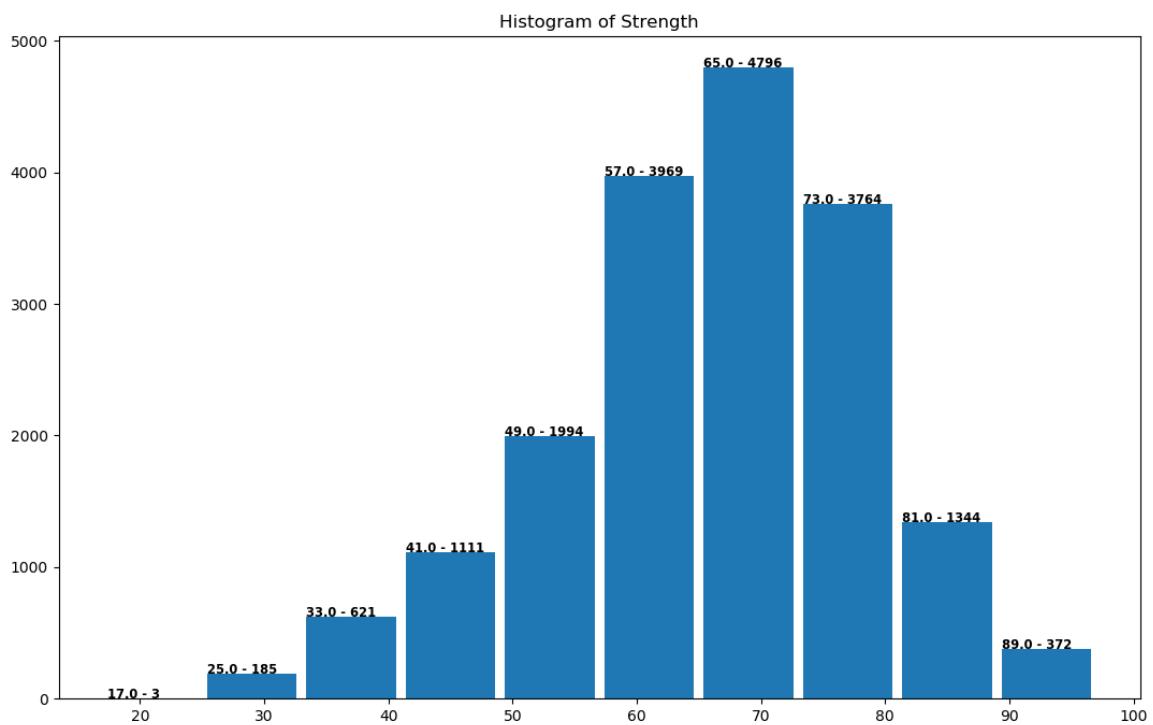
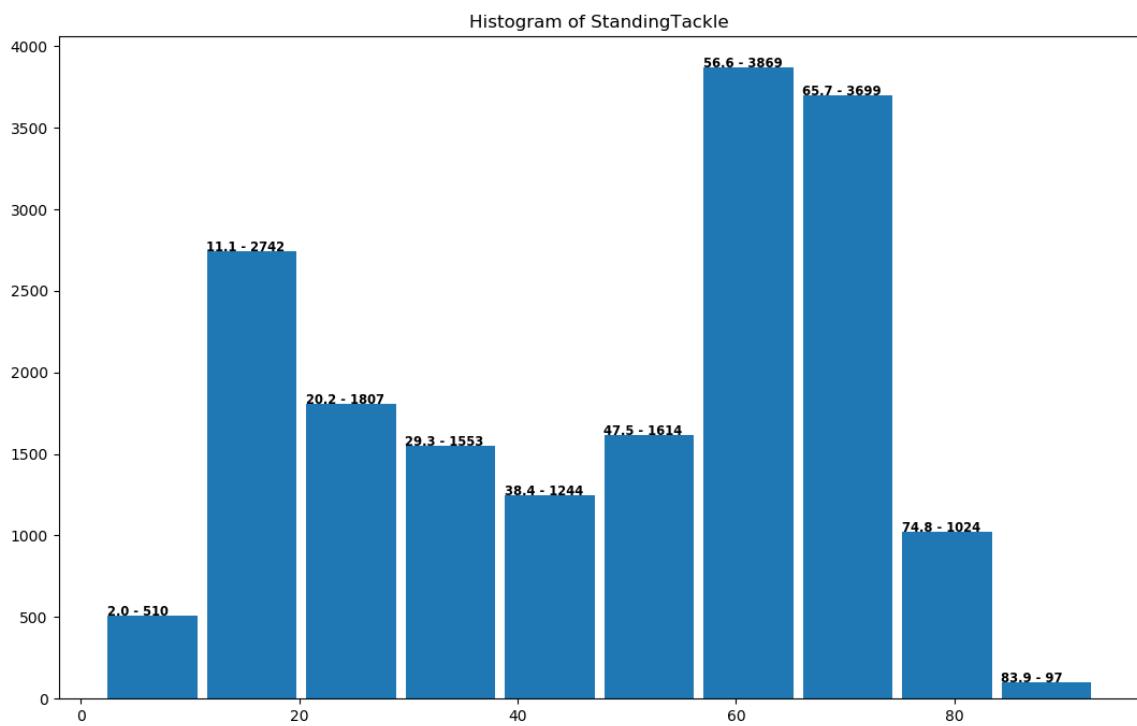


Histogram of ShortPassing

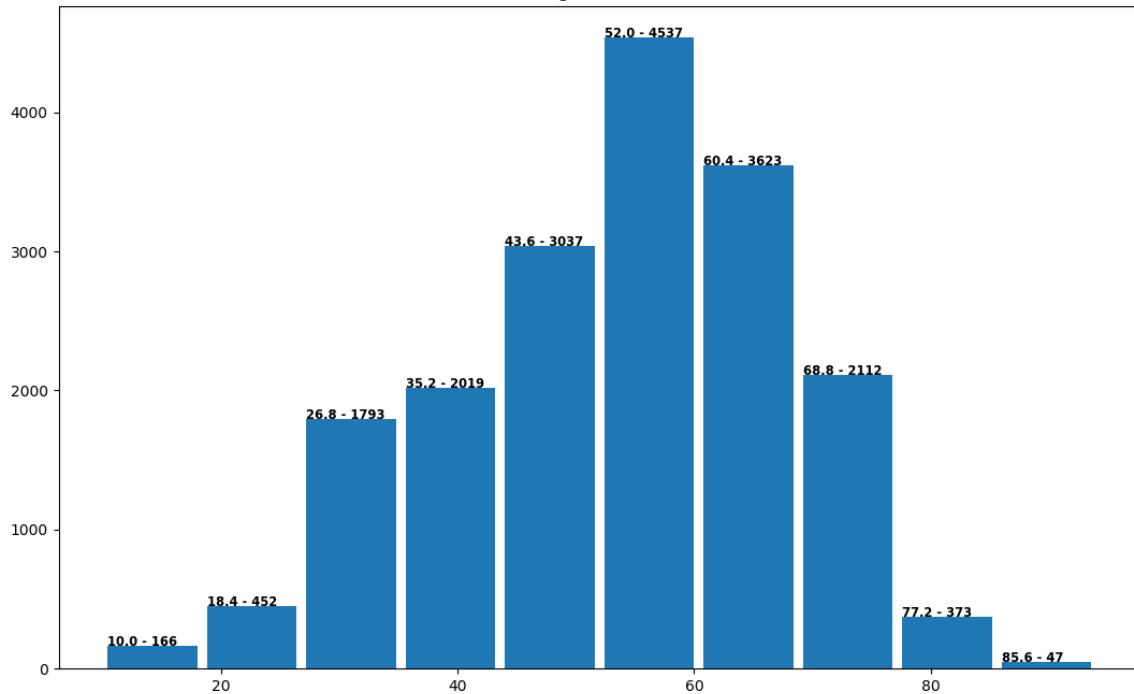




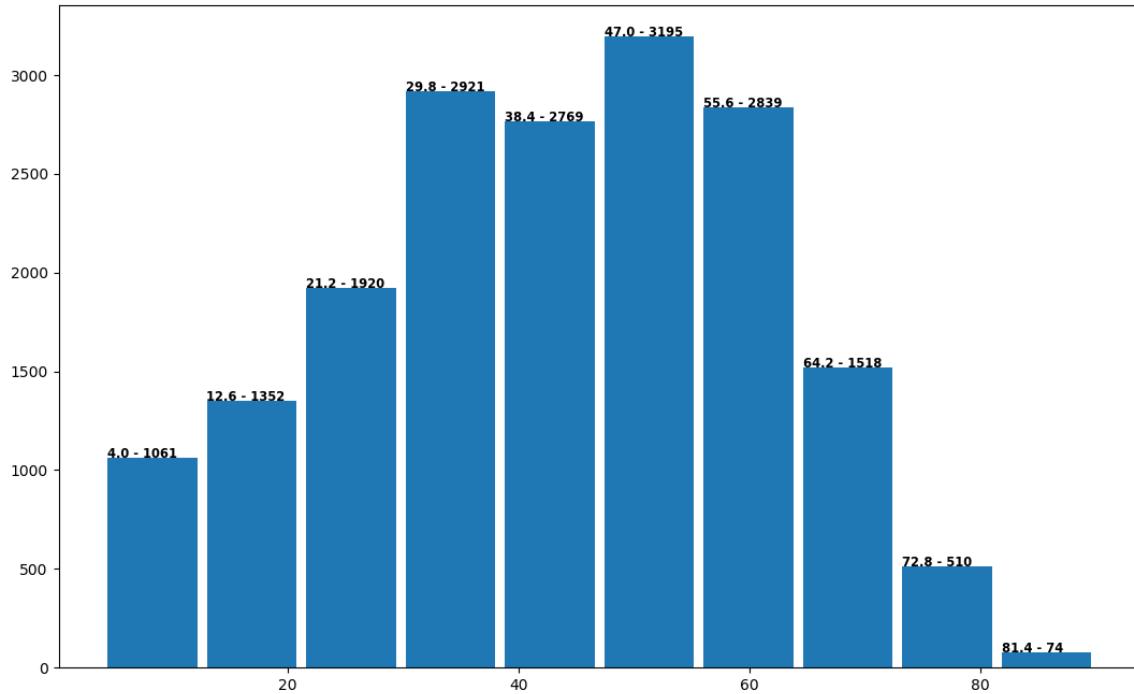




Histogram of Vision



Histogram of Volleys



Conclusion :

Total 34 different tasks or special skills are used in Football from that in

Acceleration we have very good number of players which are good in it.

Aggression we have average number of player not very much players can do it.

Agility it is better than aggression.

Balance of player is also same as Agility.

BallControll we have good number of players

Composure we have very high number of player with good composure.

Crossing we have very less amount of player with very good or very bad crossing ability, but in we have good number of players with average crossing quality.

Curve is a very special type of ability in football which is neede in every st positioned player, in histogram there are very less player in very high quality, average number of player in bad and average quality of curv.

Dribbling very large number of player with very bad quality of dribbling very less player with good and very high quality.

Finishing very high amount of players with bad quality in finishing, very less amount of players are good in finishing.

FKAccuracy stands for Free Kick accuracy, in this very very large amount of player are having bad quality.

GKDiving stands for GoalKeeper Diving, in this we have seen previously second most players are GK but the quality of diving is very very bad by maximum number of players.

GKHandling, GKKicking and GKPositioning also the same thing here very very large amount of players with very bad quality.

HeadingAccuracy, maximum players have average heading accuracy.

Interception, maximum players have average quality and very less players have high quality in interception.

Jumping, large amount of players have good jumping quality.

LongPassing, maximum players have average quality and very small amount of players are having good quality.

LongShots, maximum players are having average performance, 50% players are good in longshot.

Marking, 53 players are very good in marking

Penalties, small amount of players very good in penalties.

Positioning, half of the players in fifa are good in positioning.

Reacting, very large amount of players having good reacting on time of particular situation.

ShortPassing, large number of players are very good in short passing.

Shotpower, more than 50% of players are having good shotpower.

SlidingTackle, it is both side equal as many are not good same count of players are good in Tackling.

SprintSpeed, 519 players are having very good SprintSpeed.

Stamina, large amount of player having very good stamina.

StadingTackle, near about 50% player having good skill in StadingTackle.

Strength, maximum players have good strength.

Vision, more than 50% players having good Vision from that 47 are very good.

Maximum players are not very good in Volley but many nearly 40% are good and 74 are extremely good in volley.

- **Overall**

Q.1 Top 20 players of fifa 19.

```
import pandas as pd
import sys
import pandasql as q
import pylab as g

sys.__stdout__=sys.stdout

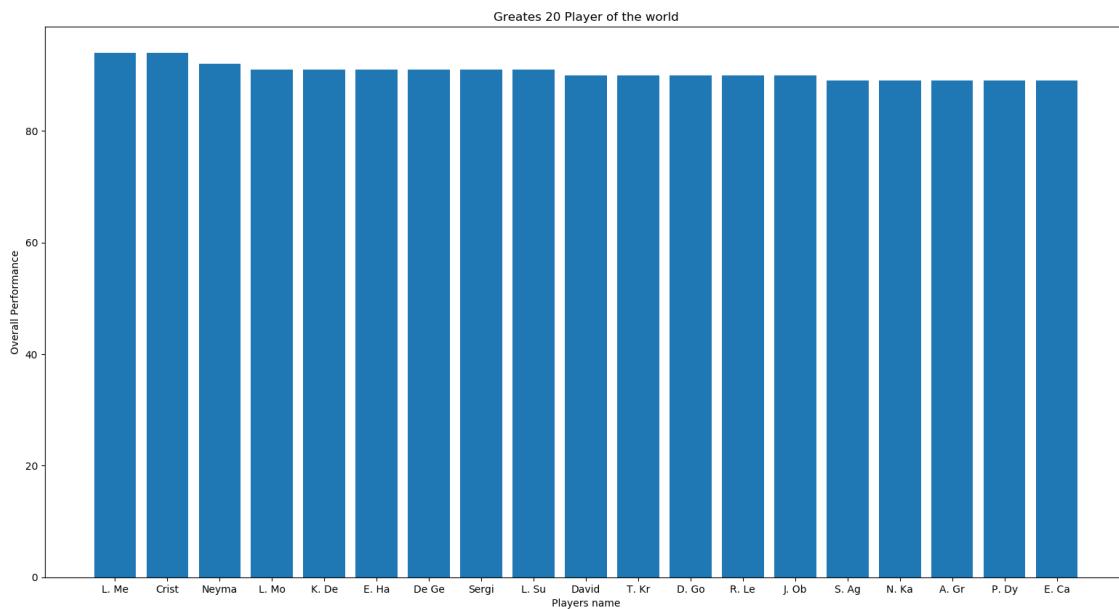
players = pd.read_excel(r"C:\Users\Hp\Documents\Data Science Project\data\Football Data New Excel Format.xlsx","data")

q = q.sqlldf

data = q("Select Name,Overall from players order by Overall desc limit 20;")

name1 = list(data["Name"])
name = []
for x in name1:
    name.append(x[:5])
overall = list(data["Overall"])

g.bar(name,overall)
g.xlabel("Players name")
g.ylabel("Overall Performance")
g.title("Greatest 20 Player of the world")
g.show()
```



Conclusion :

1st player in world is L. Messi then C. Ronaldo and then Neymar.

Q.2 Poor 20 players of fifa 19.

```
import pandas as pd
import sys
import pandasql as q
import pylab as g

sys.__stdout__=sys.stdout

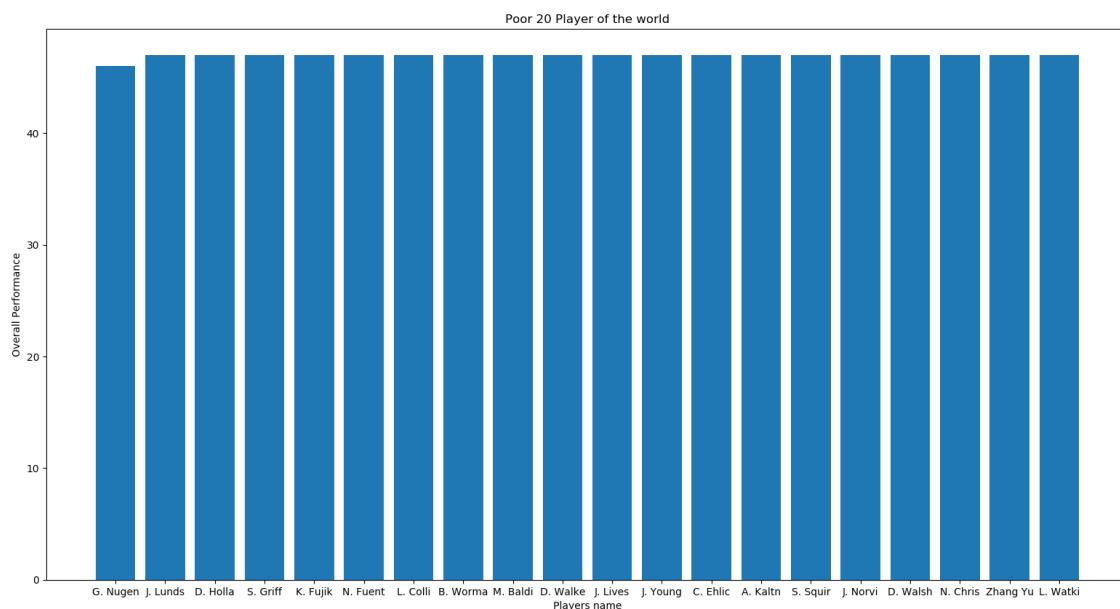
players = pd.read_excel(r"C:\Users\Hp\Documents\Data Science Project\data\Football Data New Excel Format.xlsx","data")

q = q.sql df

data = q("Select Name,Overall from players order by Overall limit 20;")

name1 = list(data["Name"])
name = []
for x in name1:
    name.append(x[:8])
overall = list(data["Overall"])

g.bar(name,overall)
g.xlabel("Players name")
g.ylabel("Overall Performance")
g.title("Poor 20 Player of the world")
g.show()
```



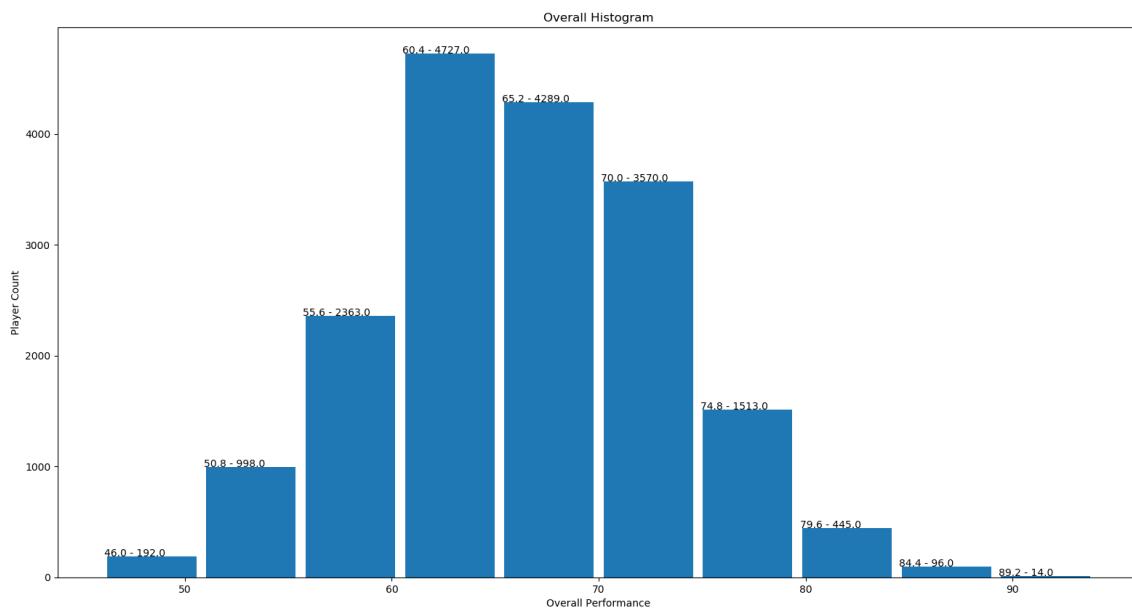
Conclusion :

Poor player are shown in graph.

Q.3 Histogram of Overall

```
import pandas as pd
import sys
import pandasql as q
import pylab as g

sys.__stdout__=sys.stdout
players = pd.read_excel(r"C:\Users\Hp\Documents\Data Science Project\data\Football Data New Excel Format.xlsx","Overall")
q = q.sqldf
data = q("Select Overall from players")
data = data.round(2)
overall = list(data["Overall"])
a,b,c = g.hist(overall,rwidth=0.9)
cnt = 0
while cnt<len(a):
    g.text(b.item(cnt)+0.12,a.item(cnt),str(round(b.item(cnt),2))+" - "+str(a.item(cnt)))
    cnt+=1
g.xlabel("Overall Performance")
g.ylabel("Player Count")
g.title("Overall Histogram")
g.show()
```



Conclusion :

More than 55% of fifa players are good in overall performance.

Heat map – Correlation Analysis

Python Code:

```

import seaborn as sns

df = pd.read_excel(r"Football Data New Excel Format.xlsx", "data")

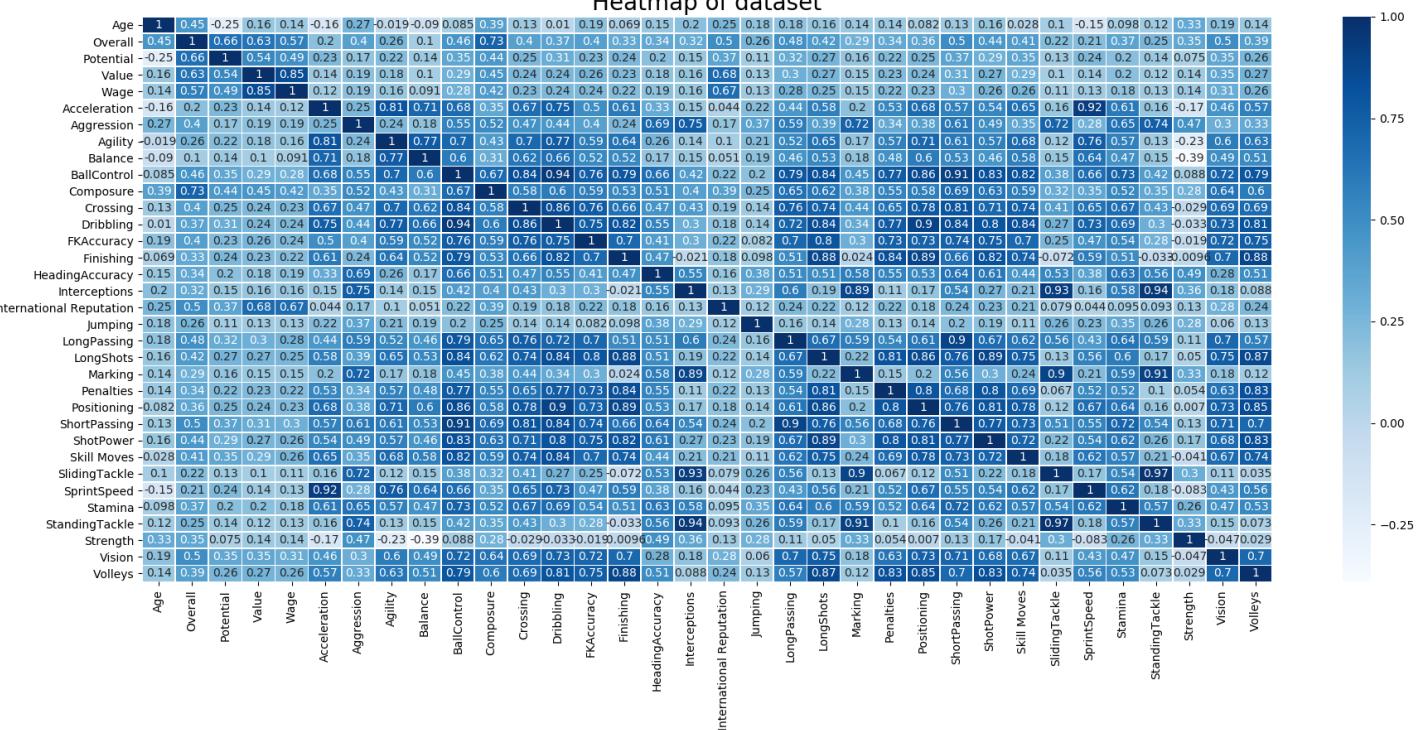
plt.rcParams['figure.figsize']=(25,16)

hm=sns.heatmap(df[['Age', 'Overall', 'Potential', 'Value', 'Wage',
'Acceleration', 'Aggression', 'Agility', 'Balance', 'BallControl',
'Body Type','Composure', 'Crossing','Dribbling', 'FKAccuracy', 'Finishing',
'HeadingAccuracy', 'Interceptions','International Reputation',
'Joined', 'Jumping', 'LongPassing', 'LongShots',
'Marking', 'Penalties', 'Position', 'Positioning',
'ShortPassing', 'ShotPower', 'Skill Moves', 'SlidingTackle',#WeightLbs',
'SprintSpeed', 'Stamina', 'StandingTackle', 'Strength', 'Vision',#JoinYear', 'ValidYear',
'Volleys']].corr(), annot = True, linewidths=.5, cmap='Blues')

hm.set_title(label='Heatmap of dataset', fontsize=20);

plt.show()

```



Conclusion:

The above graph is represents the co-relation between each data in dataset.

Overall, it is related with composure, potential and value.

Potential is related with overall.

Value, it is related with Wage, International Reputation and overall.

Wage, it is related with value and international reputation.

Acceleration, it is depend on Sprint Speed and agility.

Aggression is related with interception.

Agility is related with acceleration.

Balance is related with agility very highly oppositely related with Strength.

BallControll is related with dribbling.

Composure is related with overall.

Crossing is related with dribbling and BallControll.

Dribbling is related with BallControll and Positioning.

Free Kick Accuracy is related with Longshots.

Finishing is related with Positioning.

Heading Accuracy is related with Aggression.

Interceptions are related with Tackle and Marking.

International Reputation is related with value and wage of player.

Long Passing is highly related with Short Passing

Long Shots is related with Shot Power.

Marking is related with Tackle and highly related with Standing Tackle.

Penalties related with shot power and finishing.

Positioning is related with dribbling and finishing, vision, volley, skill moves passing and shots.

Short Passing is related with Skill Moves and BallControll.

Shot Power related with Long Shot and Volley.

Skill Moves are related with composure and dribbling.

Tackles are related with Interception and player marking.

Stamina is related with shot power and sprint speed.

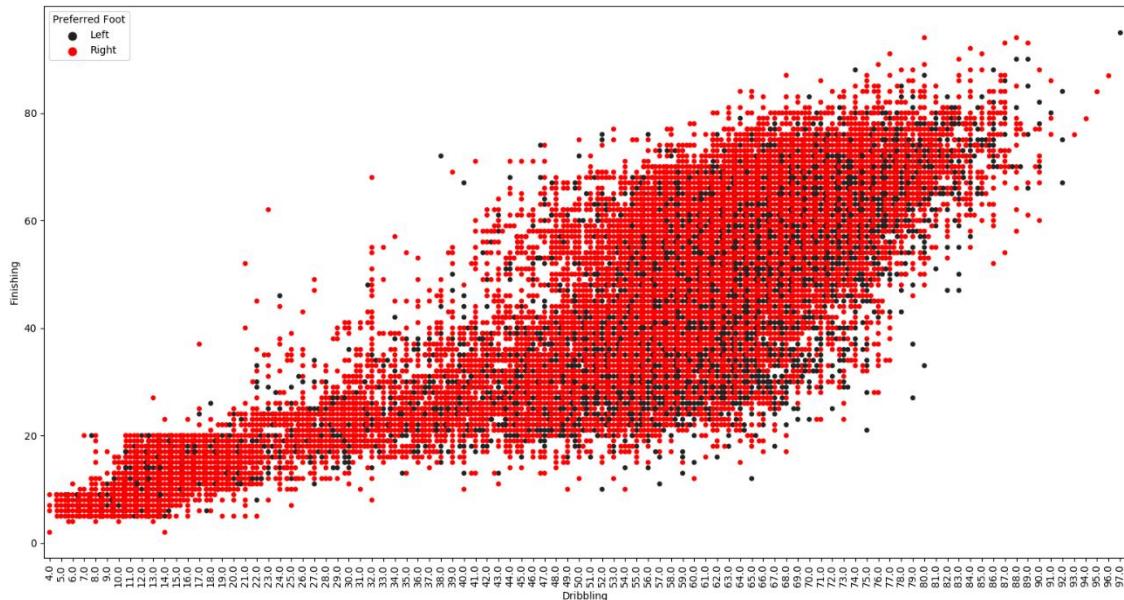
Vision is related with long shots, finishing and volleys.

Volley is related with finishing, shot power and penalties, vision.

Swarm Graph – Dribbling and Finishing by lefty and righty players

Python Code:

```
import seaborn as sns  
  
sns.swarmplot(x="Dribbling", y="Finishing", hue="Preferred Foot", data = df, color = 'red')  
  
plt.xticks(rotation=90)  
  
plt.show()
```



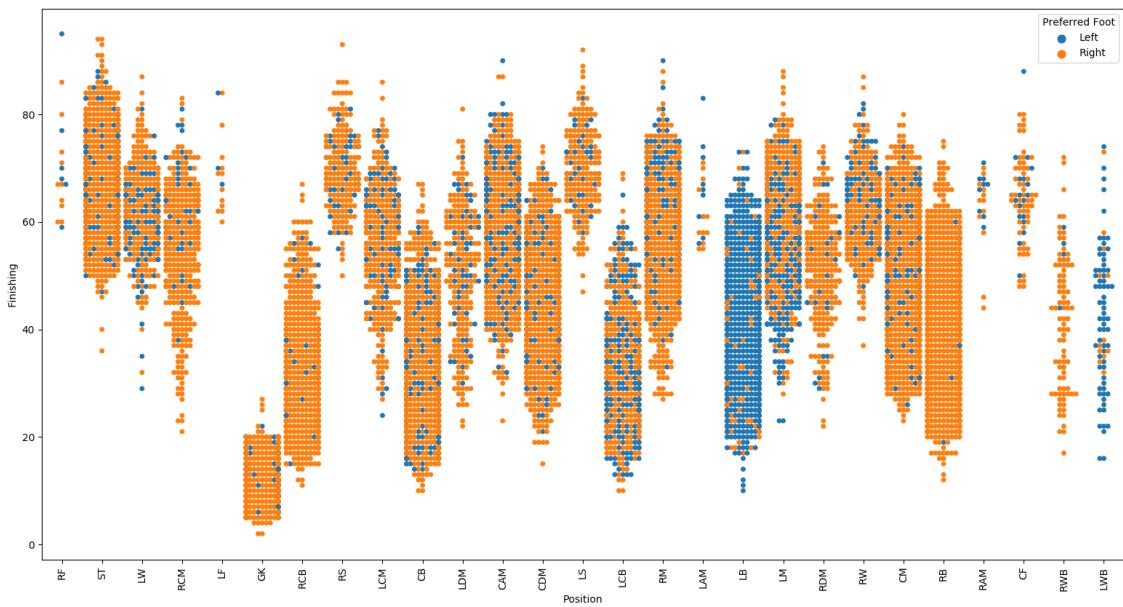
Conclusion:

The above graph is representing the red as righty and blue as lefty players and this dots are representing the dribbling and finishing of each player, so we say that left players are very less and one lefty player having maximum dribbling and finishing ability which is L. Messi.

Swarm Graph – Position and Finishing

Python Code:

```
import seaborn as sns  
  
sns.swarmplot(x="Position", y="Finishing", hue="Preferred Foot", data=df)  
  
plt.xticks(rotation=90)  
  
plt.show()
```



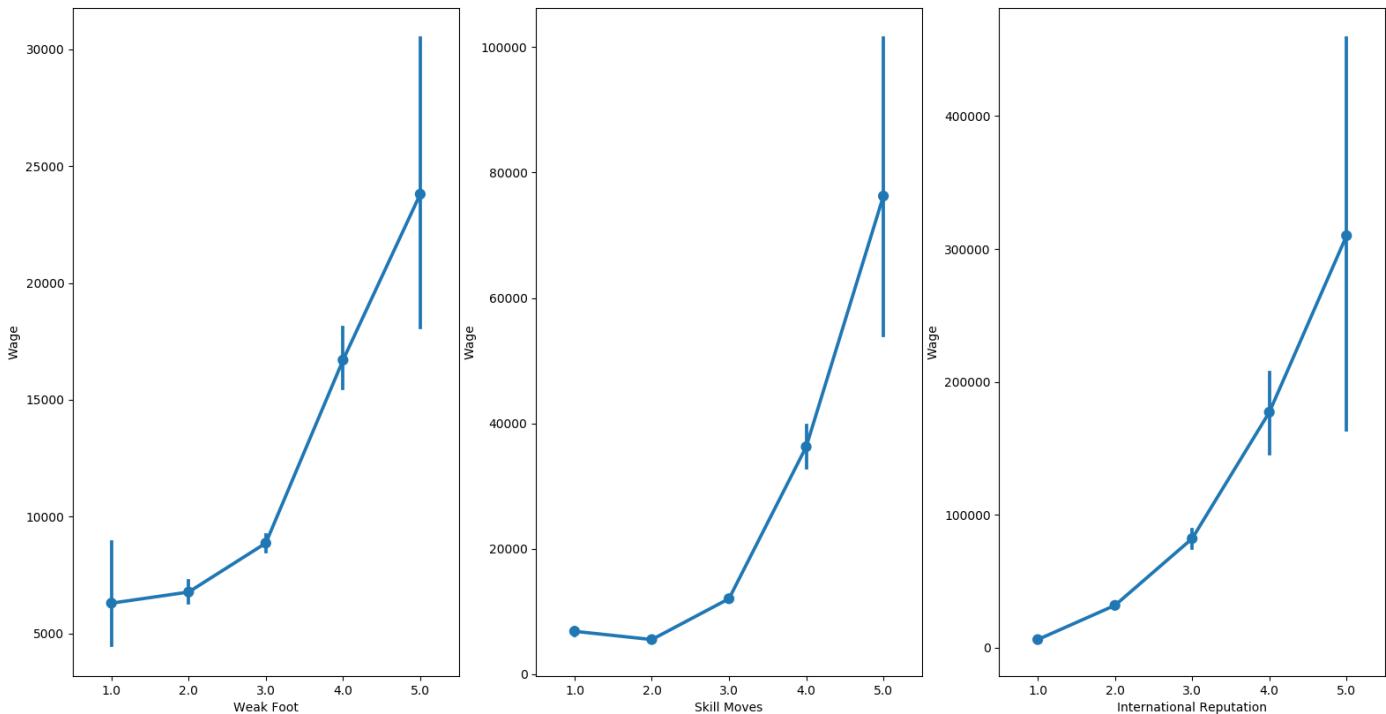
Conclusion:

The above graph shows that at left back position, maximum players are with left foot. ST position having righty players which are brilliant in finishing. All right forward players are good in finishing.

Factor Plots – Wage Diff. of Weak Foot, Skill Move and International Reputation

Python Code

```
Import seaborn as sns  
Import matplotlib.pyplot as plt  
a,axis = plt.subplots(1,3)  
sns.factorplot("Weak Foot", "Wage", data=df,ax=axis[0])  
sns.factorplot("Skill Moves", "Wage", data=df,ax=axis[1])  
sns.factorplot("International Reputation", "Wage", data=df,ax=axis[2])  
plt.show()
```



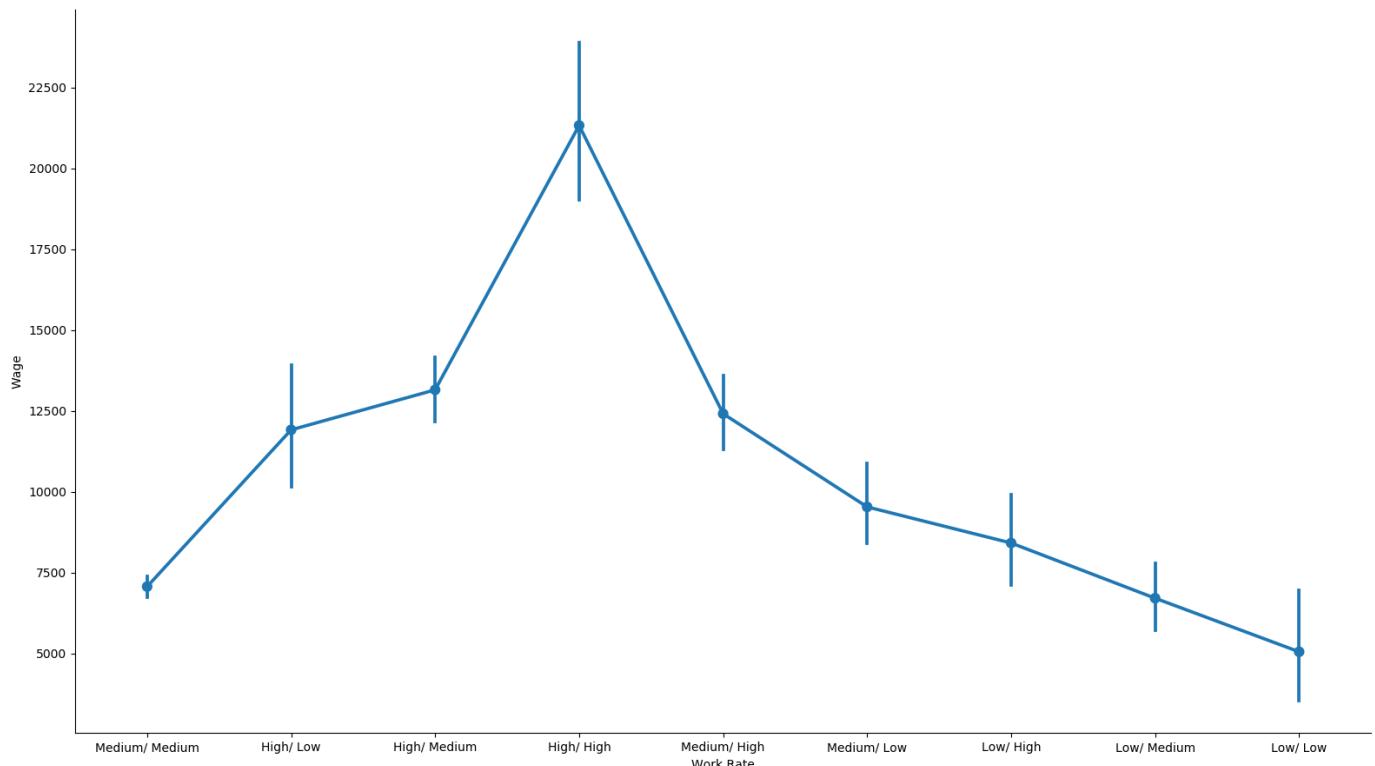
Conclusion:

The relation of wage is depend on International Reputation of player, but at the level 5 means the highest weak foot, skill moves players and int. rep. player having very large amount of difference in wage.

Factor Plots – Wage Difference between various Work Rates

Python Code:

```
Import seaborn as sns  
Import matplotlib.pyplot as plt  
sns.factorplot("Position","Wage",data=df)  
plt.show()
```



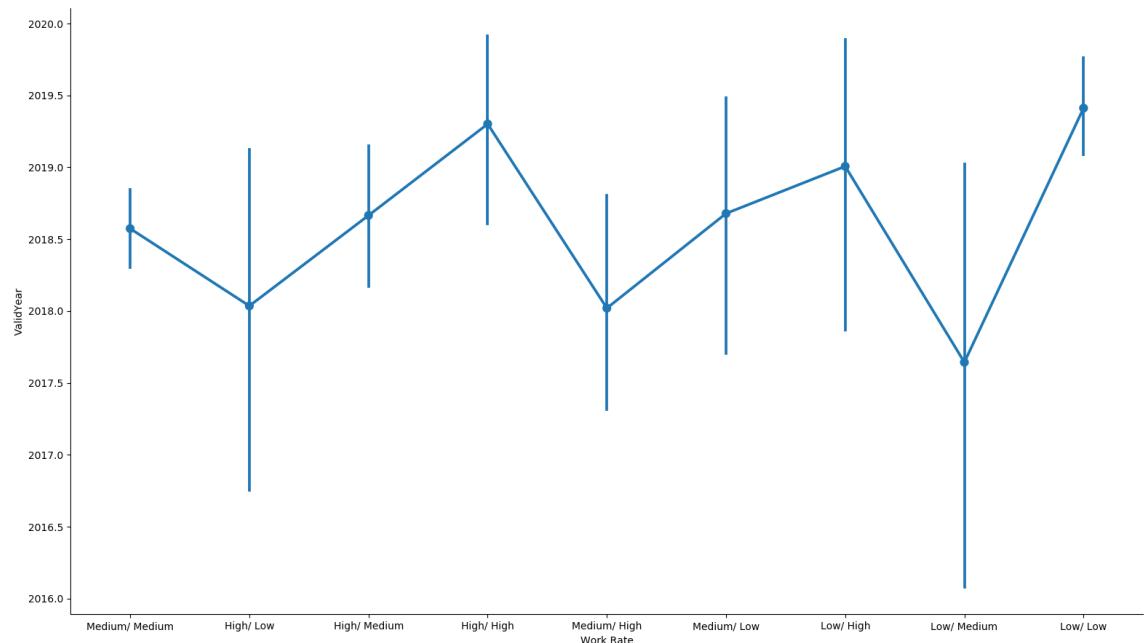
Conclusion:

The highest wage is given to those players who has High work rate.

Factor Plots – Valid year with Work Rates

Python Code:

```
Import seaborn as sns  
Import matplotlib.pyplot as plt  
sns.factorplot("Work Rate","ValidYear",data=df)  
plt.show()
```



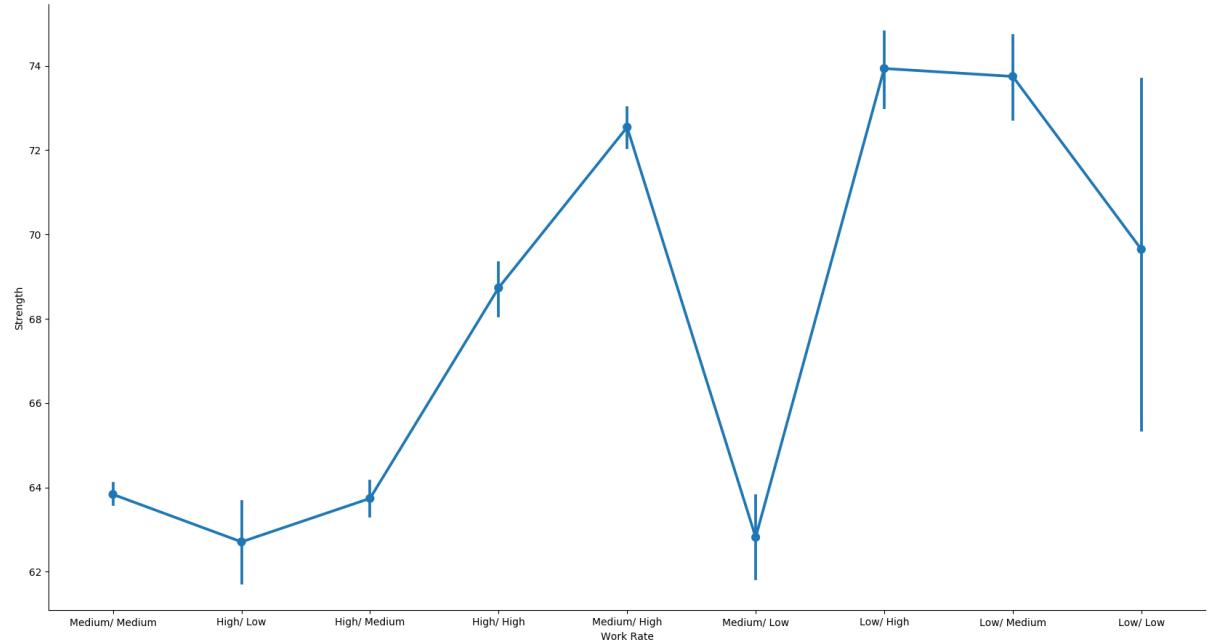
Conclusion:

Low / Medium players' validity on club is till 2019, after 2020 all players Contact will be over in club.

Factor Plots – Strength Difference between various Work Rates

Python Code:

```
Import seaborn as sns  
Import matplotlib.pyplot as plt  
sns.factorplot("Work Rate","Strength",data=df)  
plt.show()
```



Conclusion:

High / Low Work Rate players having very less strength, Low / High work rate players having max strength.

Conclusion

- Total 18207 players, 48 players with unknown preferred foot, more than 76% of player are Right preferred foot.
- There are 651 clubs in Fifa 19 but yet 241 players are not member of any club.
- The age range of player is between 16 and 45, most of the player are in age class 20 to 30.
- Total 164 nation's player are part of Fifa 19. The nation England, Germany, Spain are nation from maximum player are in Fifa 19, and Fiji, Nicaragua, Namibia are nation from minimum player are in Fifa 19. The maximum wage is paid by Spain, England and Brazil to their players. England having maximum clubs in Fifa. Portugal is having world's best player Ronaldo.
- Total 651 clubs are there from them minimum player count in club is 18 and maximum is 33 players. Maximum clubs having 28 players. If we see performance of club players Juveture is at 1st then it is Napoli, Inter, and Real Madrid. In Wage of player Real Madrid is at 1st then Barcelona, in investment vs performance of club Juveture is very smart. From all players in Fifa 1.32 % players which is near about 241 players haven't joined any club.
- Total wage paid to all players is 177.1 Million Euros, in wage and performance L. Messi having similar performance to C. Ronaldo but L. Messi taking more wage then C. Ronaldo.
- Skill Moves, maximum players having rating 2 and rating 3, Brazil and Spain having top skilled players. In clubs Liverpool, Lazio, Borussia Dortmund having most skilled players.
- Most skilled nation in average of Skill Move is United Arab Emirates, Libya, Bermunda and so on. Most poor skilled player are from Belize, Oman and then Guatmala nation. Most poor in skill moves are player who haven't joined any club. 47 % players have rating 2 and 36 % players have rating 3 the improvement needed for them.
- Face, only 9 % near about 1.6k players have real face other have no real face.
- Body Type, Maximum player having normal body and very less players have stocky body type.
- Positions, Total 27 different positions are there and maximum players playing at ST position, then GK and CB. Maximum wage is paid to ST positioned players near about 20M Euros.
- Jersey Number, There are 1 to 99 Available jersey number players, and if we see number of players maximum player are having number 8, after 40 number very less players are there.
- Total loaned players in Fifa 19 are 1264 and they are from 341 different clubs.
- Atalanta club has loaned maximum players then after Sassuolo and then Juventus.
- Maximum player joined in 2018 and the ratio starts growing very increasingly from year 2013. It means the speedy rise of awareness of fifa is started from 2013. As a graph shows ogive curve, it possible that next year will create new record of player join in fifa.
- Contact Valid Till, a very large number of player's contact is end in 2019 near about 5820 players, so in this year there will be very high demand of players, and if we see next that is 2021 which is second top demand year for players, and in previous year 2018 there was very less demand of players.
- Week foot details represents the average performance of player maximum players are with level 3 in week foot it is near to average.
- Maximum players age 20 – 26. Very less players of age 16 and range of 33-46. Maximum age of player in fifa is 45.

- In height players are in range of 5.1 to 6.9 (ft. in.) height. The height histogram graph shows highest number of player in fifa are in height class 5.10 to 5.28 or 6.00 to 6.18.
- Weight, min is 110lbs and max is 243lbs. A histogram shows maximum players having weight class 136.6 to 189.8.
- Total 34 different tasks or special skills are used in Football from that in
 - a. Acceleration we have very good number of players which are good in it.
 - b. Aggression we have average number of player not very much players can do it.
 - c. Agility it is better than aggression.
 - d. Balance of player is also same as Agility.
 - e. BallControll we have good number of players with very good quality of BallControll.
 - f. Composure we have very high number of player with good composure.
 - g. Crossing we have very less amount of player with very good or very bad crossing ability, but in we have good number of players with average crossing quality.
 - h. Curve is a very special type of ability in football which is needed in every ST positioned player, in histogram there are very less player in very high quality, average number of player in bad and average quality of curve.
 - i. Dribbling very large number of player with very bad quality of dribbling very less player with good and very high quality.
 - j. Finishing very high amount of players with bad quality in finishing, very less amount of players are good in finishing.
 - k. FKAccuracy stands for Free Kick accuracy, in this very large amount of player are having bad quality.
 - l. GKDiving stands for Goalkeeper Diving, in this we have seen previously second most players are GK but the quality of diving is very bad by maximum number of players.
 - m. GKHandling, GKKicking and GKPositioning also the same thing here very large amount of players with very bad quality.
 - n. HeadingAccuracy, maximum players have average heading accuracy.
 - o. Interception, maximum players have average quality and very less players have high quality in interception.
 - p. Jumping, large amount of players have good jumping quality.
 - q. LongPassing, maximum players have average quality and very small amount of players are having good quality.
 - r. LongShots, maximum players are having average performance, 50% players are good in longshot.
 - s. Marking, 53 players are very good in marking
 - t. Penalties, small amount of players very good in penalties.
 - u. Positioning, half of the players in fifa are good in positioning.
 - v. Reacting, very large amount of players having good reacting on time of particular situation.
 - w. ShortPassing, large number of players are very good in short passing.
 - x. Shotpower, more than 50% of players are having good shotpower.
 - y. SlidingTackle, it is both side equal as many are not good same count of players are good in Tackling.
 - z. SprintSpeed, 519 players are having very good SprintSpeed.
 - aa. Stamina, large amount of player having very good stamina.
 - bb. StadingTackle, near about 50% player having good skill in StadingTackle.
 - cc. Strength, maximum players have good strength.

- dd. Vision, more than 50% players having good Vision from that 47 are very good.
- ee. Maximum players are not very good in Volley but many nearly 40% are good and 74 are extremely good in volley.
- More than 55% of fifa players are good in overall performance.

Bibliography

Kaggle.com

For finding, downloading and support of Dataset.

proz.com/kudoz/english-to-french/sports-fitness-recreation/695698-list-of-abbreviations-football-positions.html

For finding the information of different position abbreviation of football.

Wikipedia.com

Information about Fifa.

ifd.uci.edu/~gohlke/pythonlibs/

For downloading python modules.