FIFA DV Project

`Group-9`

Prepared and Presented by:

- Archana S Ajith (20181CSE0071)
- Ashritha L (20181CSE0079)
- B Sai Rahul Reddy (20181CSE0087)
- Balu Anush A (20181CSE0091)
- Buthalapalli Lohith Reddy (20181CSE0119)

Importing libraries

- **NumPy: NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.**
- **Pandas: Pandas is a Python library which is used to analyze data.**

- **Matplotlib: Matplotlib is a low level graph plotting library in python that serves as a visualization utility.**
- **Seaborn: Seaborn is a visualization library for statistical graphics plotting in Python. It provides beautiful default styles and color palettes to make statistical plots more attractive. It is built on the top of matplotlib library and also closely integrated to the data structures from pandas.**

In [ ]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use('fivethirtyeight')
```

Read the data from CSV file:

- **data is a variable that is used to store the data which is on the CSV file by using the read_csv function avaliable in pandas library.**
- **%time is used to measure the amount of time which is needed to load the CSV file on to the data set.**

In [ ]:

```
%time data = pd.read_csv('/content/data.csv')
print(data.shape)
```

```
CPU times: user 235 ms, sys: 50.7 ms, total: 285 ms
Wall time: 288 ms
(18207, 89)
```

Checking first 5 rows and columns of the dataset:

In [ ]:

```
data.head()
```

Out[ ]:

Unnamed:

	Unnamed: 0	ID	Name	Age	Photo	Nationality	Fl
0	0	158023	L. Messi	31	https://cdn.sofifa.org/players/4/19/158023.png	Argentina	https://cdn.sofifa.org/flags/52.p
1	1	20801	Cristiano Ronaldo	33	https://cdn.sofifa.org/players/4/19/20801.png	Portugal	https://cdn.sofifa.org/flags/38.p
2	2	190871	Neymar Jr	26	https://cdn.sofifa.org/players/4/19/190871.png	Brazil	https://cdn.sofifa.org/flags/54.p
3	3	193080	De Gea	27	https://cdn.sofifa.org/players/4/19/193080.png	Spain	https://cdn.sofifa.org/flags/45.p
4	4	192985	K. De Bruyne	27	https://cdn.sofifa.org/players/4/19/192985.png	Belgium	https://cdn.sofifa.org/flags/7.p

5 rows × 89 columns

Describing the data:

- We can determine the description of the dataset such as count, mean, min, etc using the describe function.

In [ ]:

```
data.describe()
```

Out[ ]:

	Unnamed: 0	ID	Age	Overall	Potential	Special	International Reputation	Weak Foot	S
count	18207.000000	18207.000000	18207.000000	18207.000000	18207.000000	18207.000000	18159.000000	18159.000000	18
mean	9103.000000	214298.338606	25.122206	66.238699	71.307299	1597.809908	1.113222	2.947299	
std	5256.052511	29965.244204	4.669943	6.908930	6.136496	272.586016	0.394031	0.660456	
min	0.000000	16.000000	16.000000	46.000000	48.000000	731.000000	1.000000	1.000000	
25%	4551.500000	200315.500000	21.000000	62.000000	67.000000	1457.000000	1.000000	3.000000	
50%	9103.000000	221759.000000	25.000000	66.000000	71.000000	1635.000000	1.000000	3.000000	
75%	13654.500000	236529.500000	28.000000	71.000000	75.000000	1787.000000	1.000000	3.000000	
max	18206.000000	246620.000000	45.000000	94.000000	95.000000	2346.000000	5.000000	5.000000	

Checking for NULL values in the dataset:

- There are chances that our dataset may have NULL values for some reasons.

In [ ]:

```
data.isnull().sum()
```

Out[ ]:

```
Unnamed: 0             0
ID                     0
Name                   0
Age                    0
Photo                  0
                      ...
GKHandling            48
GKKicking             48
GKPositioning         48
GKReflexes            48
GKReflexes          48
Release Clause    1564
Length: 89, dtype: int64
```

- **We need to handle the NULL values by either removing the entire row or by filling the row with some values(mean, sum, etc).**
- **Here, we are following the latter by filling the missing values using the mean of the column.**
- **Also, we are using the inplace function in order to fill the values even in the originally loaded data.**

In [ ]:

```
data['ShortPassing'].fillna(data['ShortPassing'].mean(), inplace = True)
data['Volleys'].fillna(data['Volleys'].mean(), inplace = True)
data['Dribbling'].fillna(data['Dribbling'].mean(), inplace = True)
data['Curve'].fillna(data['Curve'].mean(), inplace = True)
data['FKAccuracy'].fillna(data['FKAccuracy'], inplace = True)
data['LongPassing'].fillna(data['LongPassing'].mean(), inplace = True)
data['BallControl'].fillna(data['BallControl'].mean(), inplace = True)
data['HeadingAccuracy'].fillna(data['HeadingAccuracy'].mean(), inplace = True)
data['Finishing'].fillna(data['Finishing'].mean(), inplace = True)
data['Crossing'].fillna(data['Crossing'].mean(), inplace = True)
data['Weight'].fillna('200lbs', inplace = True)
data['Contract Valid Until'].fillna(2019, inplace = True)
data['Height'].fillna("5'11", inplace = True)
data['Loaned From'].fillna('None', inplace = True)
data['Joined'].fillna('Jul 1, 2018', inplace = True)
data['Jersey Number'].fillna(8, inplace = True)
data['Body Type'].fillna('Normal', inplace = True)
data['Position'].fillna('ST', inplace = True)
data['Club'].fillna('No Club', inplace = True)
data['Work Rate'].fillna('Medium/ Medium', inplace = True)
data['Skill Moves'].fillna(data['Skill Moves'].median(), inplace = True)
data['Weak Foot'].fillna(3, inplace = True)
data['Preferred Foot'].fillna('Right', inplace = True)
data['International Reputation'].fillna(1, inplace = True)
data['Wage'].fillna('€200K', inplace = True)
```

In [ ]:

```
data.fillna(0, inplace = True)
```

DATA VISUALIZATION:

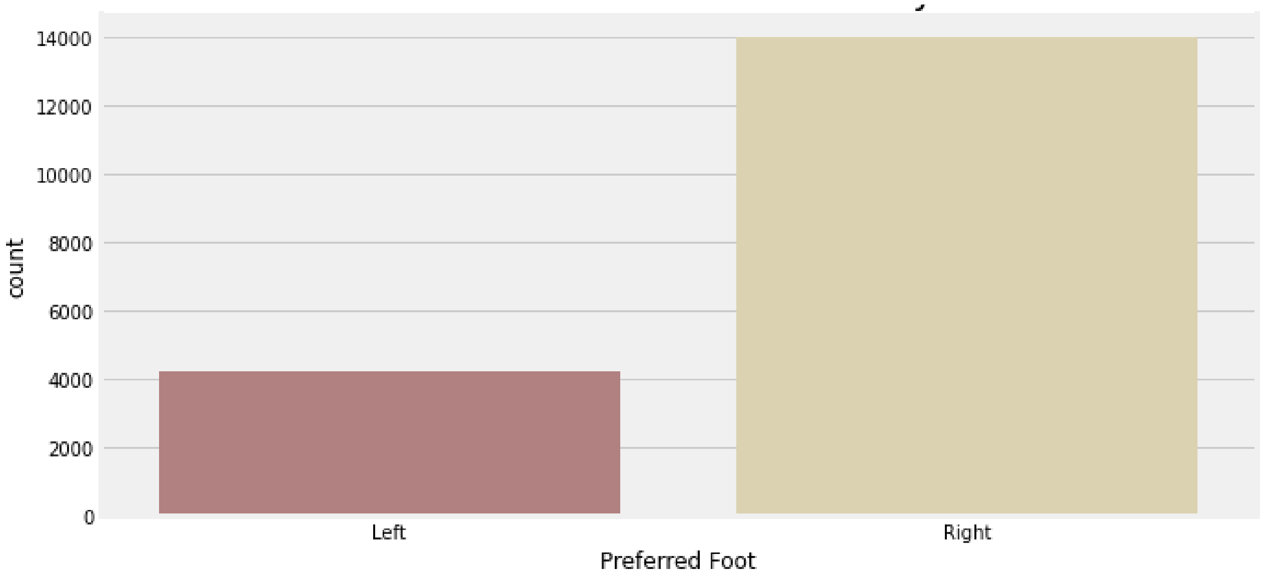
1. Comparison of preferred foot over the different players:

- **Each time Matplotlib loads, it defines a runtime configuration (rc) containing the default styles for every plot element you create. This configuration can be adjusted at any time using the plt.**
- **seaborn.countplot() method is used to Show the counts of observations in each categorical bin using bars.**
- **title is used to display the title of the figure.**
- **show is used to display the plot on the screen.**

In [ ]:

```
plt.rcParams['figure.figsize'] = (10, 5)
sns.countplot(data['Preferred Foot'], palette = 'pink')
plt.title('Most Preferred Foot of the Players', fontsize = 20)
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the
following variable as a keyword arg: x. From version 0.12, the only valid positional argu
ment will be `data`, and passing other arguments without an explicit keyword will result
in an error or misinterpretation.
  FutureWarning
```



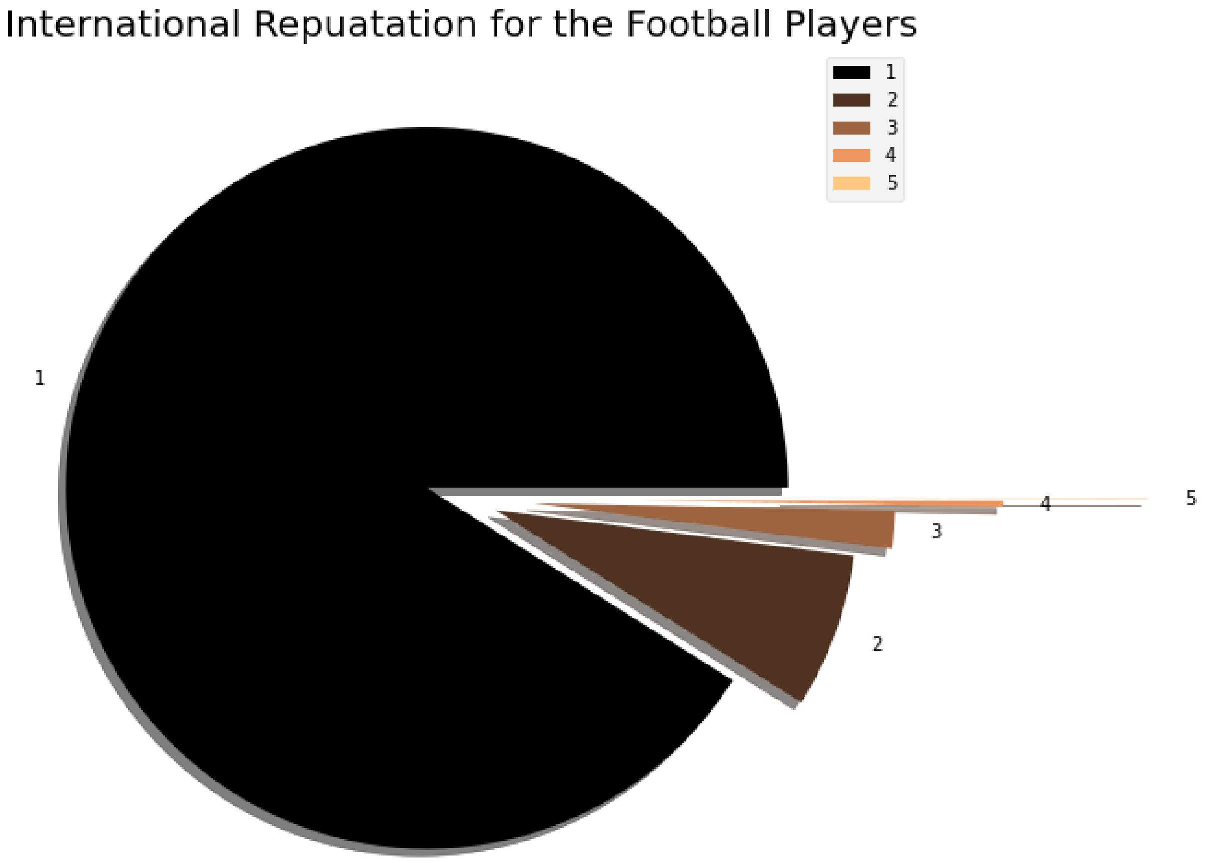
2. Plotting a pie chart to represent share of International Repuatation:

- **Labels is used to show the values that are used in the column of International Reputation.**
- **sizes is used to count the number of values that each label holds.**
- **colours is used to create colour zone on the same template.**
- **We are using a PIE CHART to show the reputation.**

In [ ]:

```
labels = ['1', '2', '3', '4', '5']
sizes = data['International Reputation'].value_counts()
colors = plt.cm.copper(np.linspace(0, 1, 5))
explode = [0.1, 0.1, 0.2, 0.5, 0.9]

plt.rcParams['figure.figsize'] = (9, 9)
plt.pie(sizes, labels = labels, colors = colors, explode = explode, shadow = True)
plt.title('International Repuatation for the Football Players', fontsize = 20)
plt.legend()
plt.show()
```

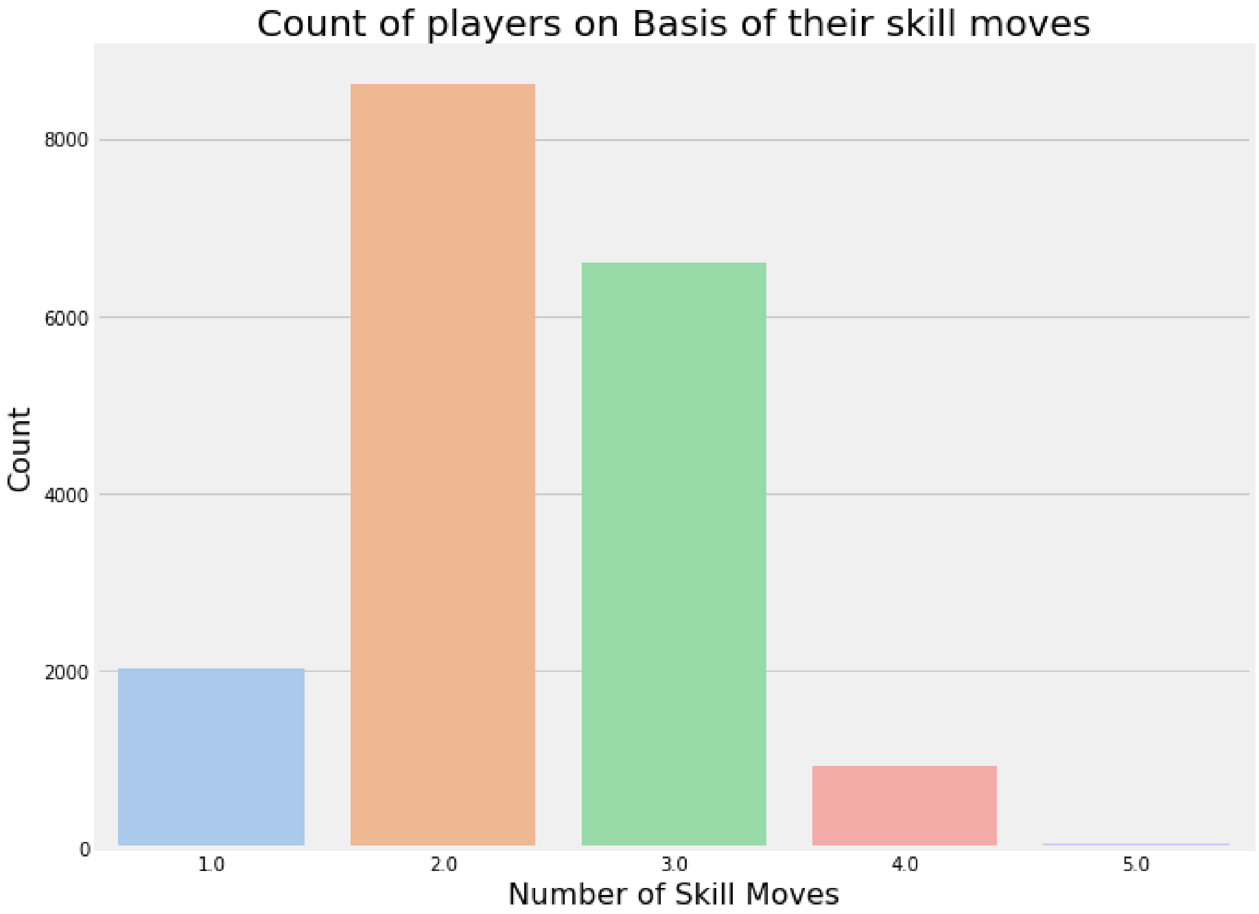


3. Skill Moves of Players:

- We are using countplot to show the number of players who are having their skill set rating for each number.

In [ ]:

```
plt.figure(figsize = (10, 8))
ax = sns.countplot(x = 'Skill Moves', data = data, palette = 'pastel')
ax.set_title(label = 'Count of players on Basis of their skill moves', fontsize = 20)
ax.set_xlabel(xlabel = 'Number of Skill Moves', fontsize = 16)
ax.set_ylabel(ylabel = 'Count', fontsize = 16)
plt.show()
```



4. Histogram of the age of the players:

- A histogram is basically used to represent data provided in a form of some groups.It is accurate method for the graphical representation of numerical data distribution.It is a type of bar plot where X-axis represents the bin ranges while Y-axis gives information about frequency.
- We are setting the style of the histogram using the set function from Seaborn.
- displot is used basically for univariant set of observations and visualizes it through a histogram i.e. only one observation and hence we choose one particular column of the dataset.

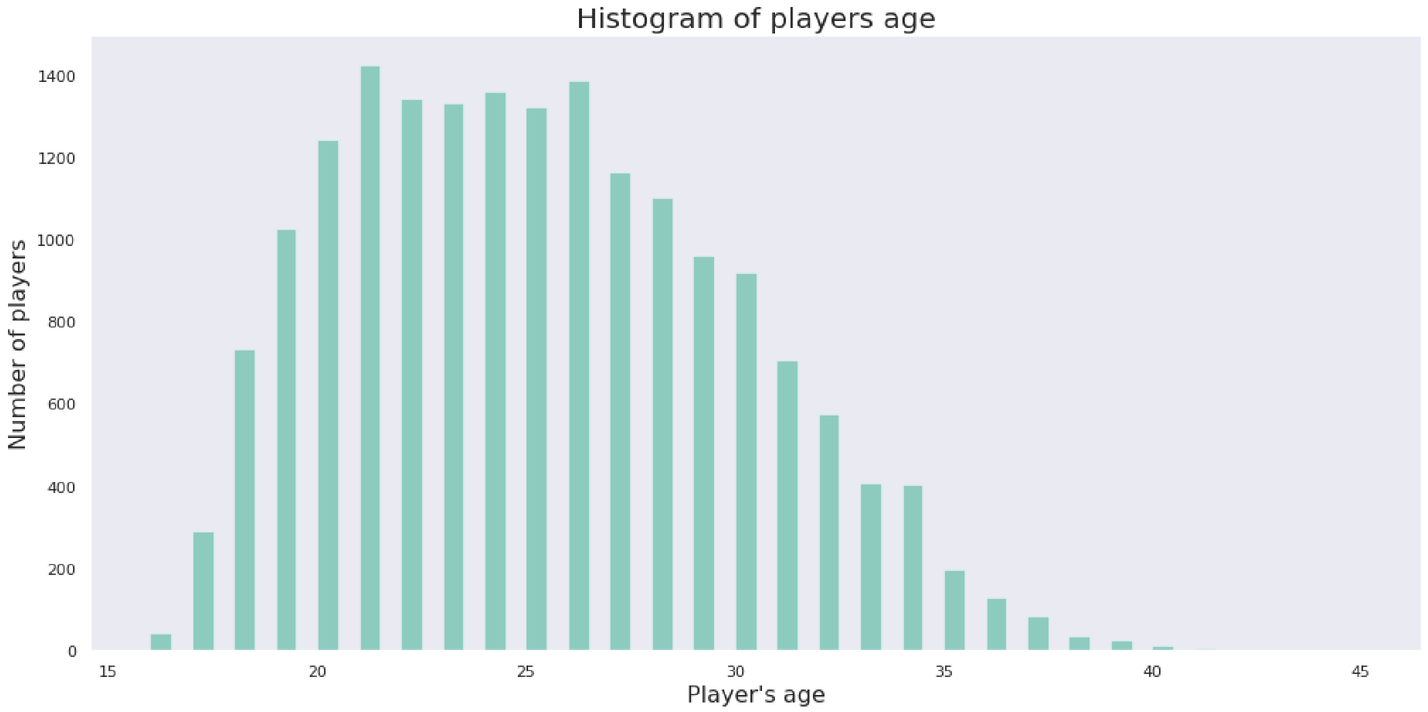
In [ ]:

```
# To show that there are people having same age
# Histogram: number of players's age

sns.set(style = "dark", palette = "colorblind", color_codes = True)
x = data.Age
plt.figure(figsize = (15,8))
ax = sns.distplot(x, bins = 58, kde = False, color = 'g')
```

```
ax.set_xlabel(xlabel = "Player\'s age", fontsize = 16)
ax.set_ylabel(ylabel = 'Number of players', fontsize = 16)
ax.set_title(label = 'Histogram of players age', fontsize = 20)
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `dis
tplot` is a deprecated function and will be removed in a future version. Please adapt you
r code to use either `displot` (a figure-level function with similar flexibility) or `his
tplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



5. Boxenplot/Violin Plot

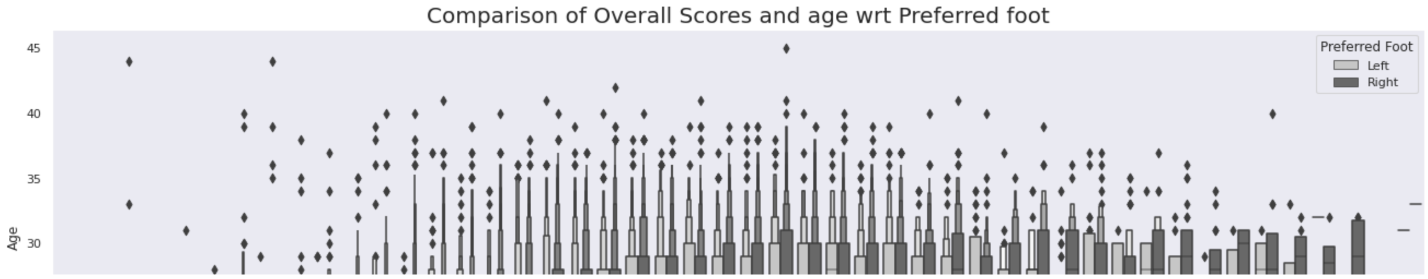
- **A boxenplot is used to show the comparison of distributions for a large dataset.**
- **In the following plot, we are plotting the age of a player with the number of goals scored along with determining the number of left foot and right foot players.**
- **The boxenplot takes the overall as the x-axis parameter and age as the y-axis parameter. The hue parameter is used to plot the categorical levels and here in the following plot, we are using the foot parameter which catergorizes the plot into left foot and right foot.**

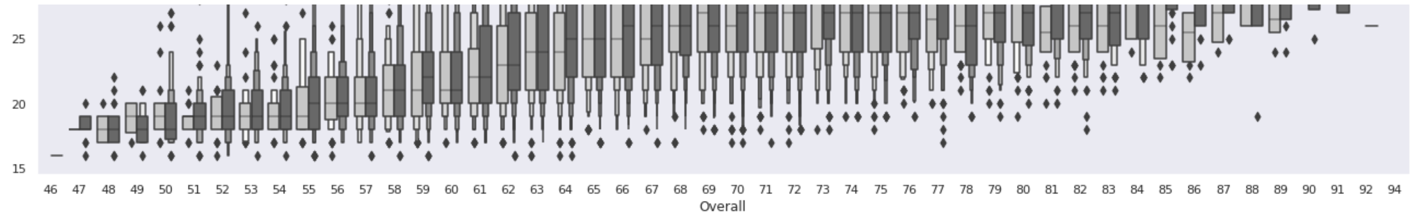
In [ ]:

```
plt.rcParams['figure.figsize'] = (20, 7)
plt.style.use('seaborn-dark-palette')

sns.boxenplot(data['Overall'], data['Age'], hue = data['Preferred Foot'], palette = 'Gre
ys')
plt.title('Comparison of Overall Scores and age wrt Preferred foot', fontsize = 20)
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the
following variables as keyword args: x, y. From version 0.12, the only valid positional a
rgument will be `data`, and passing other arguments without an explicit keyword will resu
lt in an error or misinterpretation.
  FutureWarning
```



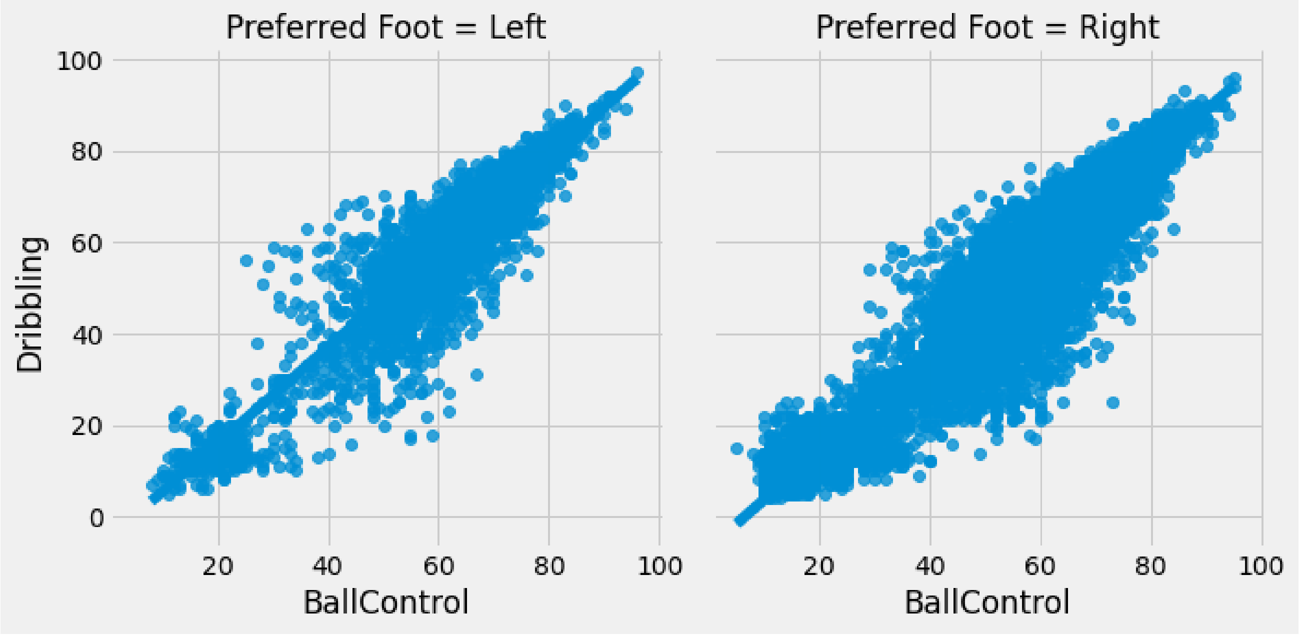


6. Ball Control vs Dribbling for Left Foot and Right Foot players:

- **In order to draw a comparison between the dribbing and ball controlling ability of the left and right foot players, we are using the lmplot.**
- **lmplot is used to draw a scatter plot on a faceted grid to bring out the comparisons.**
- **It simply takes the parameters for x-axis(BallControl here) and y-axis(Dribbling here) and uses the dataframe in the data attribute and plots the subset of plots using the col paramater(Preferred Foot in this case having Left Foot and Right Foot).**

In [ ]:

```
sns.lmplot(x = 'BallControl', y = 'Dribbling', data = data, col = 'Preferred Foot')
plt.show()
```


Preferred Foot = Left
Preferred Foot = Right
Dribbling
BallControl
BallControl