In [2]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19: FutureWarning: p
andas.util.testing is deprecated. Use the functions in the public API at pandas.testing i
nstead.
  import pandas.util.testing as tm
```

In [3]:

```
df = pd.read_csv('FIFA_data.csv', index_col = 0)
```

The Fifa Dataset includes detailed attributes of every player registered in FIFA 2019 database.

In [4]:

```
df.head()
```

Out[4]:

	ID	Name	Age	Photo	Nationality	Flag	Overall
0	158023	L. Messi	31	https://cdn.sofifa.org/players/4/19/158023.png	Argentina	https://cdn.sofifa.org/flags/52.png	94
1	20801	Cristiano Ronaldo	33	https://cdn.sofifa.org/players/4/19/20801.png	Portugal	https://cdn.sofifa.org/flags/38.png	94
2	190871	Neymar Jr	26	https://cdn.sofifa.org/players/4/19/190871.png	Brazil	https://cdn.sofifa.org/flags/54.png	92
3	193080	De Gea	27	https://cdn.sofifa.org/players/4/19/193080.png	Spain	https://cdn.sofifa.org/flags/45.png	91
4	192985	K. De Bruyne	27	https://cdn.sofifa.org/players/4/19/192985.png	Belgium	https://cdn.sofifa.org/flags/7.png	91

5 rows × 88 columns

In [5]:

```
df.shape
```

Out[5]:

```
(18207, 88)
```

The FiFa dataset contains 88 varables. Lets remove some of the unwanted columns in the dataset.

Dropping unnecessary columns in the dataset.

Some dataset contains photos which are not necessary. So, these columns are dropped.

In [6]:

```
col = ['Photo', 'Flag', 'Club Logo']
```

```
df.drop(col, axis = 1, inplace=True)
```

Check and Filling up missing values

In [7]:

```
for col in df.columns:
  print(col, ":" ,df[col].isnull().sum())
```

```
ID : 0
Name : 0
Age : 0
Nationality : 0
Overall : 0
Potential : 0
Club : 241
Value : 0
Wage : 0
Special : 0
Preferred Foot : 48
International Reputation : 48
Weak Foot : 48
Skill Moves : 48
Work Rate : 48
Body Type : 48
Real Face : 48
Position : 60
Jersey Number : 60
Joined : 1553
Loaned From : 16943
Contract Valid Until : 289
Height : 48
Weight : 48
LS : 2085
ST : 2085
RS : 2085
LW : 2085
LF : 2085
CF : 2085
RF : 2085
RW : 2085
LAM : 2085
CAM : 2085
RAM : 2085
LM : 2085
LCM : 2085
CM : 2085
RCM : 2085
RM : 2085
LWB : 2085
LDM : 2085
CDM : 2085
RDM : 2085
RWB : 2085
LB : 2085
LCB : 2085
CB : 2085
RCB : 2085
RB : 2085
Crossing : 48
Finishing : 48
HeadingAccuracy : 48
ShortPassing : 48
Volleys : 48
Dribbling : 48
Curve : 48
FKAccuracy : 48
LongPassing : 48
BallControl : 48
```

```
Acceleration : 48
SprintSpeed : 48
Agility : 48
Reactions : 48
Balance : 48
ShotPower : 48
Jumping : 48
Stamina : 48
Strength : 48
LongShots : 48
Aggression : 48
Interceptions : 48
Positioning : 48
Vision : 48
Penalties : 48
Composure : 48
Marking : 48
StandingTackle : 48
SlidingTackle : 48
GKDiving : 48
GKHandling : 48
GKKicking : 48
GKPositioning : 48
GKReflexes : 48
Release Clause : 1564
```

The dataset contains lots of null values. The mean function is used for filling up the null values.

In [8]:

```
df.fillna(df.mean(), inplace = True)
for col in df.columns:
  print(col, ":" ,df[col].isnull().sum())
```

```
ID : 0
Name : 0
Age : 0
Nationality : 0
Overall : 0
Potential : 0
Club : 241
Value : 0
Wage : 0
Special : 0
Preferred Foot : 48
International Reputation : 0
Weak Foot : 0
Skill Moves : 0
Work Rate : 48
Body Type : 48
Real Face : 48
Position : 60
Jersey Number : 0
Joined : 1553
Loaned From : 16943
Contract Valid Until : 289
Height : 48
Weight : 48
LS : 2085
ST : 2085
RS : 2085
LW : 2085
LF : 2085
CF : 2085
RF : 2085
RW : 2085
LAM : 2085
CAM : 2085
RAM : 2085
LM : 2085
LCM : 2085
CM : 2085
```

```
CM : 2085
RCM : 2085
RM : 2085
LWB : 2085
LDM : 2085
CDM : 2085
RDM : 2085
RWB : 2085
LB : 2085
LCB : 2085
CB : 2085
RCB : 2085
RB : 2085
Crossing : 0
Finishing : 0
HeadingAccuracy : 0
ShortPassing : 0
Volleys : 0
Dribbling : 0
Curve : 0
FKAccuracy : 0
LongPassing : 0
BallControl : 0
Acceleration : 0
SprintSpeed : 0
Agility : 0
Reactions : 0
Balance : 0
ShotPower : 0
Jumping : 0
Stamina : 0
Strength : 0
LongShots : 0
Aggression : 0
Interceptions : 0
Positioning : 0
Vision : 0
Penalties : 0
Composure : 0
Marking : 0
StandingTackle : 0
SlidingTackle : 0
GKDiving : 0
GKHandling : 0
GKKicking : 0
GKPositioning : 0
GKReflexes : 0
Release Clause : 1564
```

There are stil some varaibles which contains null values and it could not be addressed by mean since it may be a string. So, we shall assign it as "Unassigned".

In [9]:

```
df.fillna("Unassigned", inplace = True)
```

In [10]:

```
for col in df.columns:
  print(col, ":" ,df[col].isnull().sum())
```

```
ID : 0
Name : 0
Age : 0
Nationality : 0
Overall : 0
Potential : 0
Club : 0
Value : 0
Wage : 0
Special : 0
Preferred Foot : 0
```

Preferred Foot : 0
International Reputation : 0
Weak Foot : 0
Skill Moves : 0
Work Rate : 0
Body Type : 0
Real Face : 0
Position : 0
Jersey Number : 0
Joined : 0
Loaned From : 0
Contract Valid Until : 0
Height : 0
Weight : 0
LS : 0
ST : 0
RS : 0
LW : 0
LF : 0
CF : 0
RF : 0
RW : 0
LAM : 0
CAM : 0
RAM : 0
LM : 0
LCM : 0
CM : 0
RCM : 0
RM : 0
LWB : 0
LDM : 0
CDM : 0
RDM : 0
RWB : 0
LB : 0
LCB : 0
CB : 0
RCB : 0
RB : 0
Crossing : 0
Finishing : 0
HeadingAccuracy : 0
ShortPassing : 0
Volleys : 0
Dribbling : 0
Curve : 0
FKAccuracy : 0
LongPassing : 0
BallControl : 0
Acceleration : 0
SprintSpeed : 0
Agility : 0
Reactions : 0
Balance : 0
ShotPower : 0
Jumping : 0
Stamina : 0
Strength : 0
LongShots : 0
Aggression : 0
Interceptions : 0
Positioning : 0
Vision : 0
Penalties : 0
Composure : 0
Marking : 0
StandingTackle : 0
SlidingTackle : 0
GKDiving : 0
GKHandling : 0
GKKicking : 0
GKPositioning : 0

GKPositioning : 0
GKReflexes : 0
Release Clause : 0

Data Analysis and Visualization

In [11]:

```
df.describe()
```

Out[11]:

	ID	Age	Overall	Potential	Special	International Reputation	Weak Foot	Skill Moves	18
count	18207.000000	18207.000000	18207.000000	18207.000000	18207.000000	18207.000000	18207.000000	18207.000000	
mean	214298.338606	25.122206	66.238699	71.307299	1597.809908	1.113222	2.947299	2.361308	
std	29965.244204	4.669943	6.908930	6.136496	272.586016	0.393511	0.659585	0.755167	
min	16.000000	16.000000	46.000000	48.000000	731.000000	1.000000	1.000000	1.000000	
25%	200315.500000	21.000000	62.000000	67.000000	1457.000000	1.000000	3.000000	2.000000	
50%	221759.000000	25.000000	66.000000	71.000000	1635.000000	1.000000	3.000000	2.000000	
75%	236529.500000	28.000000	71.000000	75.000000	1787.000000	1.000000	3.000000	3.000000	
max	246620.000000	45.000000	94.000000	95.000000	2346.000000	5.000000	5.000000	5.000000	

In [12]:

```
df.head(2)
```

Out[12]:

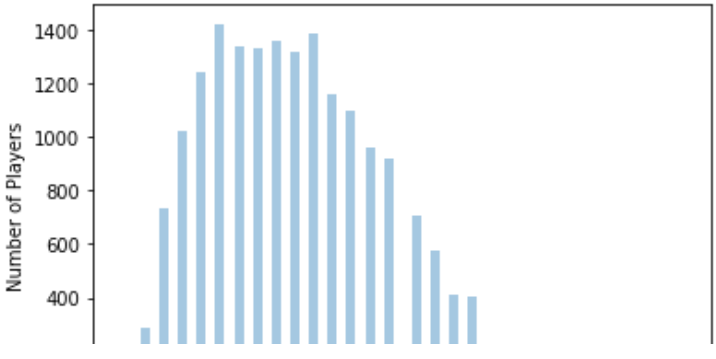
	ID	Name	Age	Nationality	Overall	Potential	Club	Value	Wage	Special	Preferred Foot	International Reputation	Weak Foot
0	158023	L. Messi	31	Argentina	94	94	FC Barcelona	€110.5M	€565K	2202	Left	5.0	4.0
1	20801	Cristiano Ronaldo	33	Portugal	94	94	Juventus	€77M	€405K	2228	Right	5.0	4.0

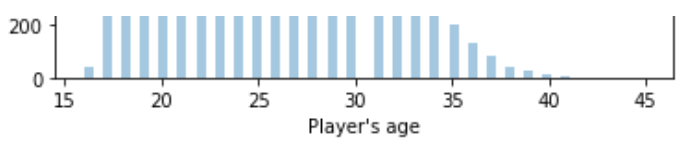
2 rows × 85 columns

Histogram of Age

In [13]:

```
sns.distplot(df.Age, bins = 60, kde = False)
plt.xlabel('Player\'s age')
plt.ylabel('Number of Players')
plt.show()
```





From this analysis, we can see that players with age 26 are more. There are more than 600 players from age 24 to 30.

In [14]:

```
# Top ten eldest players with name, club and Wage.

df_elder = df.sort_values('Age', ascending=False)[['Name', 'Age', 'Club', 'Wage']].head(
10)
df_elder.set_index('Name', inplace=True)
print(df_elder)
```

```
               Age                          Club Wage
Name
O. Pérez        45                       Pachuca  €8K
K. Pilkington   44                Cambridge United  €1K
T. Warner       44             Accrington Stanley  €1K
S. Narazaki     42                 Nagoya Grampus  €1K
C. Muñoz        41  CD Universidad de Concepción  €1K
J. Villar       41                     Unassigned   €0
H. Sulaimani    41                      Ohod Club  €3K
M. Tyler        41            Peterborough United  €1K
B. Nivet        41                  ESTAC Troyes  €5K
F. Kippe        40                 Lillestrøm SK  €1K
```

In [15]:

```
# Similarly 10 youngest players with Name, Age, Nationality and Value
# Top ten eldest players with name, club and Wage.

df_youngest = df.sort_values('Age', ascending=True)[['Name', 'Age', 'Nationality', 'Valu
e']].head(10)
df_youngest.set_index('Name', inplace=True)
print(df_youngest)
```

```
               Age          Nationality  Value
Name
G. Nugent       16              England   €60K
J. Olstad       16               Norway  €100K
H. Massengo     16               France  €450K
J. Italiano     16            Australia  €280K
N. Ayéva        16               Sweden   €70K
K. Broda        16               Poland  €110K
L. D'Arrigo     16            Australia  €130K
Y. Verschaeren  16              Belgium  €650K
B. Nygren       16               Sweden  €180K
B. O'Gorman     16  Republic of Ireland   €60K
```

1. **We can observe that age with 45 is the eldest player followed by 41 and 40.**
2. **Age with 17 is the youngest player.**

In [16]:

```
df.head(2)
```

Out[16]:

	ID	Name	Age	Nationality	Overall	Potential	Club	Value	Wage	Special	Preferred Foot	International Reputation	Weak Foot
0	**158023**	**L. Messi**	**31**	**Argentina**	**94**	**94**	**FC Barcelona**	**€110.5M**	**€565K**	**2202**	**Left**	**5.0**	**4.0**
1	**20801**	**Cristiano**	**33**	**Portugal**	**94**	**94**	**Juventus**	**€77M**	**€405K**	**2208**	**Right**	**5.0**	**4.0**

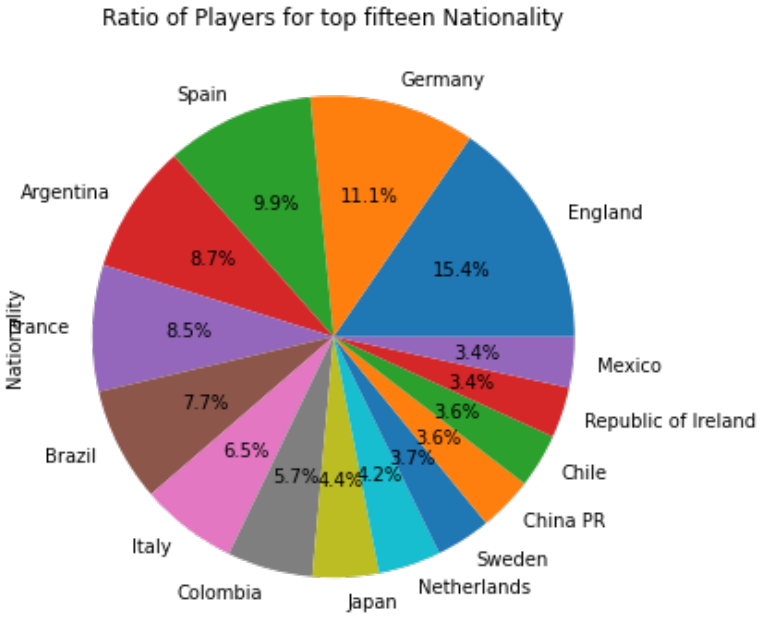
	ID	Name	Age	Nationality	Overall	Potential	Club	Value	Wage	Special	Preferred Foot	International Reputation	Weak Foot
1	20801	Cristiano Ronaldo	33	Portugal	94	94	Juventus	€77M	€405K	2228	Right	5.0	4.0

2 rows × 85 columns

Players from different Nationality

In [17]:

```
data = df['Nationality'].value_counts()[:15] # Top 15 nationality
labels = data.index
plt.figure(figsize = (6,6))
data.plot(kind = 'pie', labels = labels, shadow = False, autopct = '%1.1f%%')
plt.title('Ratio of Players for top fifteen Nationality')
plt.show()
```



From the analysis, we can see that there are about 12.5% players from Brazil Nationality followed by Argentina with 10.9%. Among top 15 Nationality, USA has least players with 2.4%.

Top 10 Right and Left Foot Footballers

In [18]:

```
# Top 10 players with Right Leg Footballers

df_right_players = df[df['Preferred Foot'] == 'Right'][['Name', 'Age', 'Nationality', 'Value']].head(10)
df_right_players.set_index('Name', inplace=True)
print(df_right_players)
```

```
                  Age Nationality    Value
Name
Cristiano Ronaldo  33    Portugal     €77M
Neymar Jr          26      Brazil  €118.5M
De Gea             27       Spain     €72M
K. De Bruyne       27     Belgium    €102M
E. Hazard          27     Belgium     €93M
L. Modrić          32     Croatia     €67M
L. Suárez          31     Uruguay     €80M
Sergio Ramos       32       Spain     €51M
J. Oblak           25    Slovenia     €68M
R. Lewandowski     29      Poland     €77M
```

In [19]:

```
# Top 10 players with Left Leg Footballers

df_right_players = df[df['Preferred Foot'] == 'Left'][['Name', 'Age', 'Nationality', 'Va
lue']].head(10)
df_right_players.set_index('Name', inplace=True)
print(df_right_players)
```

```
              Age Nationality    Value
Name
L. Messi       31   Argentina  €110.5M
David Silva    32       Spain    €60M
P. Dybala      24   Argentina    €89M
A. Griezmann   27      France    €78M
T. Courtois    26     Belgium   €53.5M
G. Chiellini   33       Italy    €27M
M. Salah       26       Egypt   €69.5M
J. Rodríguez   26    Colombia   €69.5M
Marcelo        30      Brazil    €43M
G. Bale        28       Wales    €60M
```

Percentage of Preferred foot

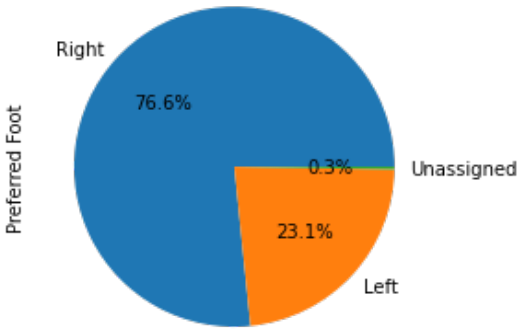
In [20]:

```
pref_foot = df['Preferred Foot'].value_counts()
print(pref_foot)
labels1 = pref_foot.index
```

```
Right         13948
Left           4211
Unassigned       48
Name: Preferred Foot, dtype: int64
```

In [21]:

```
pref_foot.plot(kind = 'pie', labels = labels1, shadow = False, autopct = '%1.1f%%')
plt.show()
```



We can see that there are about 75.6% Right foot players and 24.4% Left foot players.

In [22]:

```
df.head(2)
```

Out[22]:

	ID	Name	Age	Nationality	Overall	Potential	Club	Value	Wage	Special	Preferred Foot	International Reputation	Weak Foot
0	**158023**	**L. Messi**	**31**	**Argentina**	**94**	**94**	**FC Barcelona**	**€110.5M**	**€565K**	**2202**	**Left**	**5.0**	**4.0**
1	**20801**	**Cristiano Ronaldo**	**33**	**Portugal**	**94**	**94**	**Juventus**	**€77M**	**€405K**	**2228**	**Right**	**5.0**	**4.0**

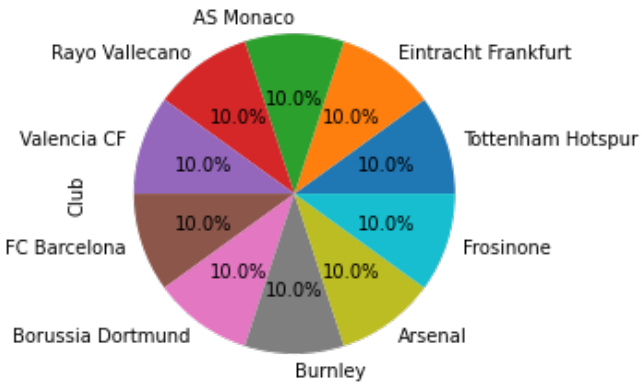
2 rows × 25 columns

2 rows × 85 columns

Ratio of Players from different clubs

In [23]:

```
data_club = df['Club'].value_counts()[1:11]
labels2 = data_club.index
data_club.plot(kind = 'pie', labels = labels2, shadow = False, autopct = '%1.1f%%')
plt.show()
```



From this, we can see that Players from Frankfurt and Guimaraes has highest percentage which is around 10.5%.

Position of Players

In [24]:

```
# New Dataframe with position = ST and Preferred foot = Right
data = df[(df['Position'] == 'ST') & (df['Preferred Foot'] == 'Right')]
data
```

Out[24]:

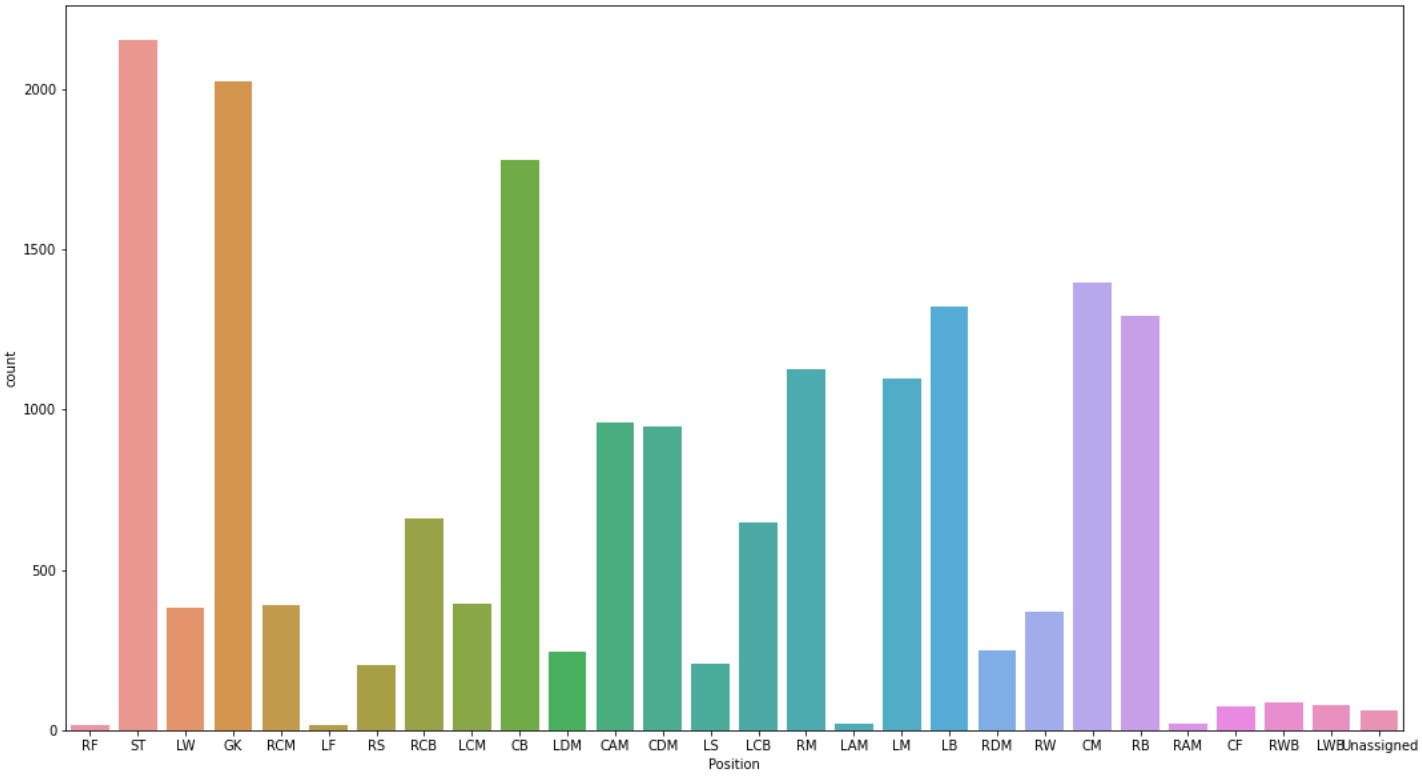
	ID	Name	Age	Nationality	Overall	Potential	Club	Value	Wage	Special	Preferred Foot	Internat Reputa
1	20801	Cristiano Ronaldo	33	Portugal	94	94	Juventus	€77M	€405K	2228	Right	
10	188545	R. Lewandowski	29	Poland	90	90	FC Bayern München	€77M	€205K	2152	Right	
16	202126	H. Kane	24	England	89	91	Tottenham Hotspur	€83.5M	€205K	2165	Right	
23	153079	S. Agüero	30	Argentina	89	89	Manchester City	€64.5M	€300K	2107	Right	
43	201399	M. Icardi	25	Argentina	87	90	Inter	€64.5M	€130K	1940	Right	
...	...	...	...	...	...	...	...	...	...	...	...	...
18166	243621	N. Ayéva	16	Sweden	48	72	Örebro SK	€70K	€1K	1286	Right	
18177	238550	R. Roache	18	Republic of Ireland	48	69	Blackpool	€70K	€1K	1178	Right	
18189	240160	A. Kaltner	18	Germany	47	61	SpVgg Unterhaching	€60K	€1K	1290	Right	
		N.					Trelleborgs					

	ID	Name	Age	Nationality	Overall	Potential	Club	Value	Wage	Special	Preferred Foot	Internat Reputa
18203	243165	Christoffersson	19	Sweden	47	63	FF	€60K	€1K	1098	Right	
18204	241638	B. Worman	16	England	47	67	Cambridge United	€60K	€1K	1189	Right	

1859 rows × 85 columns

In [25]:

```
plt.figure(figsize = (18, 10))
sns.countplot(x = 'Position', data = df)
plt.show()
```



From the above analysis, we can see that players with ST position are highest.

Finding Best players with Postion considering Overall score.

In [26]:

```
data = df.iloc[df.groupby(['Position'])['Overall'].idxmax()][['Name', 'Position']]
data
```

Out[26]:

	Name	Position
17	A. Griezmann	CAM
12	D. Godín	CB
20	Sergio Busquets	CDM
271	Luis Alberto	CF
67	Thiago	CM
3	De Gea	GK
28	J. Rodríguez	LAM
35	Marcelo	LB
24	G. Chiellini	LCB
11	T. Kroos	LCM
14	N. Kanté	LDM

	Name	Position
5	E. Hazard	LF
33	P. Aubameyang	LM
21	E. Cavani	LS
2	Neymar Jr	LW
474	N. Schulz	LWB
129	J. Cuadrado	RAM
69	Azpilicueta	RB
8	Sergio Ramos	RCB
4	K. De Bruyne	RCM
45	P. Pogba	RDM
0	L. Messi	RF
25	K. Mbappé	RM
7	L. Suárez	RS
56	Bernardo Silva	RW
450	M. Ginter	RWB
1	Cristiano Ronaldo	ST
5018	R. Raldes	Unassigned

In [27]:

```
df.head(2)
```

Out[27]:

	ID	Name	Age	Nationality	Overall	Potential	Club	Value	Wage	Special	Preferred Foot	International Reputation	Weak Foot
0	158023	L. Messi	31	Argentina	94	94	FC Barcelona	€110.5M	€565K	2202	Left	5.0	4.0
1	20801	Cristiano Ronaldo	33	Portugal	94	94	Juventus	€77M	€405K	2228	Right	5.0	4.0

2 rows × 85 columns

Top players in Different Abilities

In [45]:

```
# Top Dribblers

df_dribblers = df[df['Dribbling']> 90][['Name' , 'Dribbling']]
df_dribblers
```

Out[45]:

	Name	Dribbling
0	L. Messi	97.0
2	Neymar Jr	96.0
5	E. Hazard	95.0
15	P. Dybala	92.0
30	Isco	94.0
32	Coutinho	91.0

	Name	Dribbling
50	D. Mertens	91.0
56	Bernardo Silva	92.0
65	Douglas Costa	92.0
84	R. Mahrez	91.0
94	Y. Brahimi	93.0
371	J. Corona	91.0

In [44]:

```
# Top Stamina players
df_stamina = df[df['Stamina']> 93][['Name' , 'Stamina']]
df_stamina
```

Out[44]:

	Name	Stamina
14	N. Kanté	96.0
103	B. Matuidi	94.0
190	Allan	95.0
406	H. Herrera	94.0
562	F. Kessié	96.0
587	R. Battaglia	94.0
751	M. Eggestein	96.0
836	V. Darida	94.0
1192	A. Schöpf	95.0
2290	G. Shinnie	94.0
5676	R. Murawski	94.0
5796	M. Prietl	94.0

Conclusion

The dataset has been analysed for different players based on different attributes.

1. **First the dataset is cleaned by addressing the missing values and also by dropping unnecessary columns.**
2. **Histogram analysis of Age has been done.**
3. **Players from different Nationality have been obtained.**
4. **Top left and right foot players have been obtained.**
5. **Ratio of players from different clubs have been done.**
6. **Position of players and top players in Different abilities have been obtained.**