

```
In [1]: import numpy as np # Linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import seaborn as sns
sns.set(style="whitegrid")
import matplotlib.pyplot as plt
from collections import Counter
%matplotlib inline
```

```
In [2]: import warnings
warnings.filterwarnings('ignore')
```

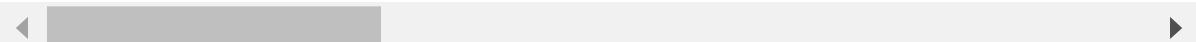
```
In [3]: fifa19=pd.read_csv(r"C:\Users\Admin\Downloads\15th\15th\Seaborn\FIFA.csv", in
```

```
In [4]: fifa19
```

Out[4]:

	ID	Name	Age	Photo	Nationality	
0	158023	L. Messi	31	https://cdn.sofifa.org/players/4/19/158023.png	Argentina	https://
1	20801	Cristiano Ronaldo	33	https://cdn.sofifa.org/players/4/19/20801.png	Portugal	https://
2	190871	Neymar Jr	26	https://cdn.sofifa.org/players/4/19/190871.png	Brazil	https://
3	193080	De Gea	27	https://cdn.sofifa.org/players/4/19/193080.png	Spain	https://
4	192985	K. De Bruyne	27	https://cdn.sofifa.org/players/4/19/192985.png	Belgium	https:
...
18202	238813	J. Lundstram	19	https://cdn.sofifa.org/players/4/19/238813.png	England	https://
18203	243165	N. Christoffersson	19	https://cdn.sofifa.org/players/4/19/243165.png	Sweden	https://
18204	241638	B. Worman	16	https://cdn.sofifa.org/players/4/19/241638.png	England	https://
18205	246268	D. Walker-Rice	17	https://cdn.sofifa.org/players/4/19/246268.png	England	https://
18206	246269	G. Nugent	16	https://cdn.sofifa.org/players/4/19/246269.png	England	https://

18207 rows × 88 columns

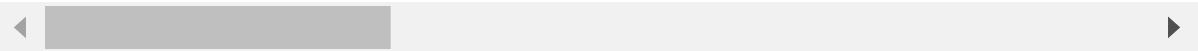


In [5]: `fifa19.head()`

Out[5]:

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

5 rows × 88 columns



```
In [6]: fifa19.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 18207 entries, 0 to 18206
Data columns (total 88 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   ID               18207 non-null   int64  
 1   Name              18207 non-null   object  
 2   Age               18207 non-null   int64  
 3   Photo              18207 non-null   object  
 4   Nationality        18207 non-null   object  
 5   Flag               18207 non-null   object  
 6   Overall            18207 non-null   int64  
 7   Potential          18207 non-null   int64  
 8   Club               17966 non-null   object  
 9   Club Logo           18207 non-null   object  
 10  Value              18207 non-null   object  
 11  Wage               18207 non-null   object  
 12  Special             18207 non-null   int64  
 13  Preferred Foot      18159 non-null   object  
 14  International Reputation 18159 non-null   float64 
 15  Weak Foot           18159 non-null   float64 
 16  Skill Moves          18159 non-null   float64 
 17  Work Rate            18159 non-null   object  
 18  Body Type            18159 non-null   object  
 19  Real Face            18159 non-null   object  
 20  Position             18147 non-null   object  
 21  Jersey Number         18147 non-null   float64 
 22  Joined               16654 non-null   object  
 23  Loaned From          1264 non-null    object  
 24  Contract Valid Until 17918 non-null   object  
 25  Height               18159 non-null   object  
 26  Weight               18159 non-null   object  
 27  LS                   16122 non-null   object  
 28  ST                   16122 non-null   object  
 29  RS                   16122 non-null   object  
 30  LW                   16122 non-null   object  
 31  LF                   16122 non-null   object  
 32  CF                   16122 non-null   object  
 33  RF                   16122 non-null   object  
 34  RW                   16122 non-null   object  
 35  LAM                  16122 non-null   object  
 36  CAM                  16122 non-null   object  
 37  RAM                  16122 non-null   object  
 38  LM                   16122 non-null   object  
 39  LCM                  16122 non-null   object  
 40  CM                   16122 non-null   object  
 41  RCM                  16122 non-null   object  
 42  RM                   16122 non-null   object  
 43  LWB                  16122 non-null   object  
 44  LDM                  16122 non-null   object  
 45  CDM                  16122 non-null   object  
 46  RDM                  16122 non-null   object  
 47  RWB                  16122 non-null   object  
 48  LB                   16122 non-null   object  
 49  LCB                  16122 non-null   object  
 50  CB                   16122 non-null   object  
 51  RCB                  16122 non-null   object
```

```

52   RB           16122 non-null  object
53   Crossing     18159 non-null  float64
54   Finishing    18159 non-null  float64
55   HeadingAccuracy  18159 non-null  float64
56   ShortPassing  18159 non-null  float64
57   Volleys      18159 non-null  float64
58   Dribbling    18159 non-null  float64
59   Curve        18159 non-null  float64
60   FKAccuracy   18159 non-null  float64
61   LongPassing  18159 non-null  float64
62   BallControl   18159 non-null  float64
63   Acceleration 18159 non-null  float64
64   SprintSpeed  18159 non-null  float64
65   Agility       18159 non-null  float64
66   Reactions     18159 non-null  float64
67   Balance       18159 non-null  float64
68   ShotPower     18159 non-null  float64
69   Jumping       18159 non-null  float64
70   Stamina       18159 non-null  float64
71   Strength      18159 non-null  float64
72   LongShots     18159 non-null  float64
73   Aggression    18159 non-null  float64
74   Interceptions 18159 non-null  float64
75   Positioning   18159 non-null  float64
76   Vision        18159 non-null  float64
77   Penalties     18159 non-null  float64
78   Composure     18159 non-null  float64
79   Marking       18159 non-null  float64
80   StandingTackle 18159 non-null  float64
81   SlidingTackle 18159 non-null  float64
82   GKDiving      18159 non-null  float64
83   GKHandling    18159 non-null  float64
84   GKKicking      18159 non-null  float64
85   GKPositioning 18159 non-null  float64
86   GKReflexes    18159 non-null  float64
87   Release Clause 16643 non-null  object
dtypes: float64(38), int64(5), object(45)
memory usage: 12.4+ MB

```

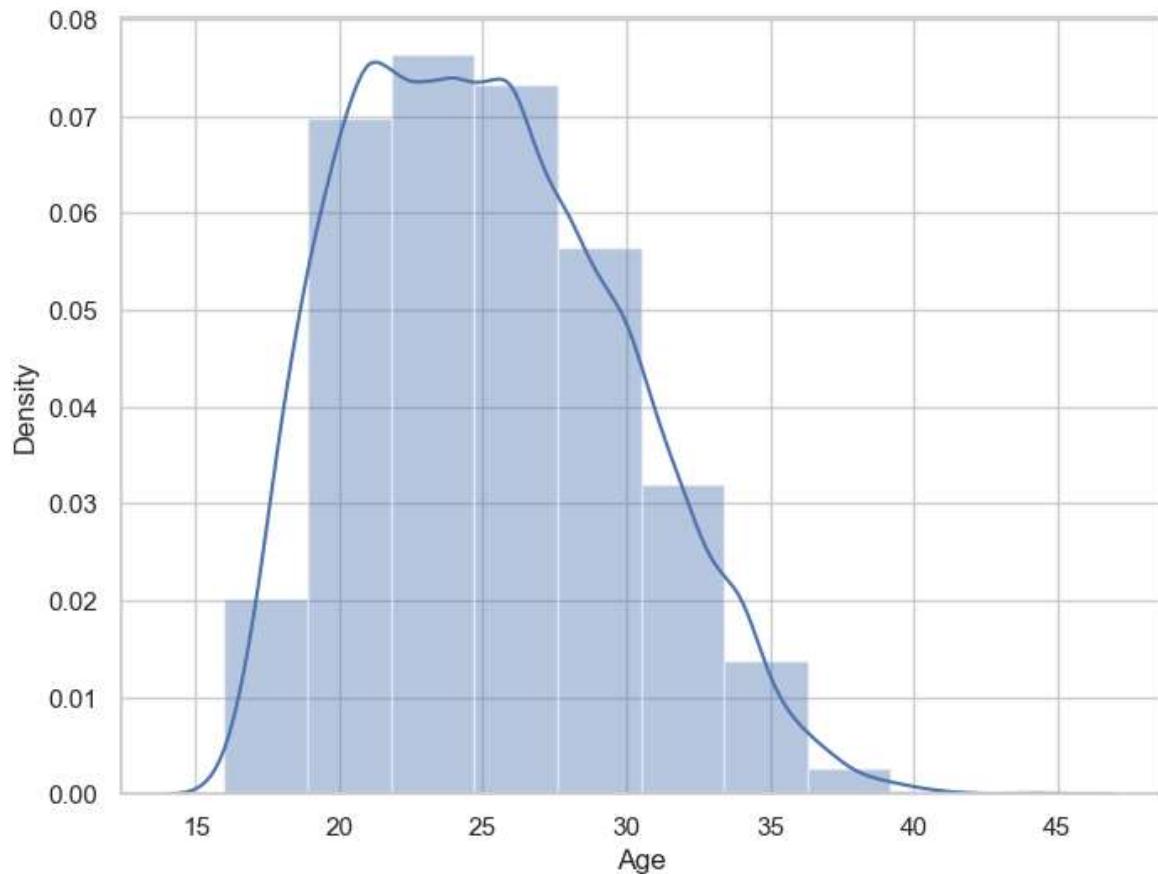
In [7]: `fifa19['Body Type'].value_counts()`

```

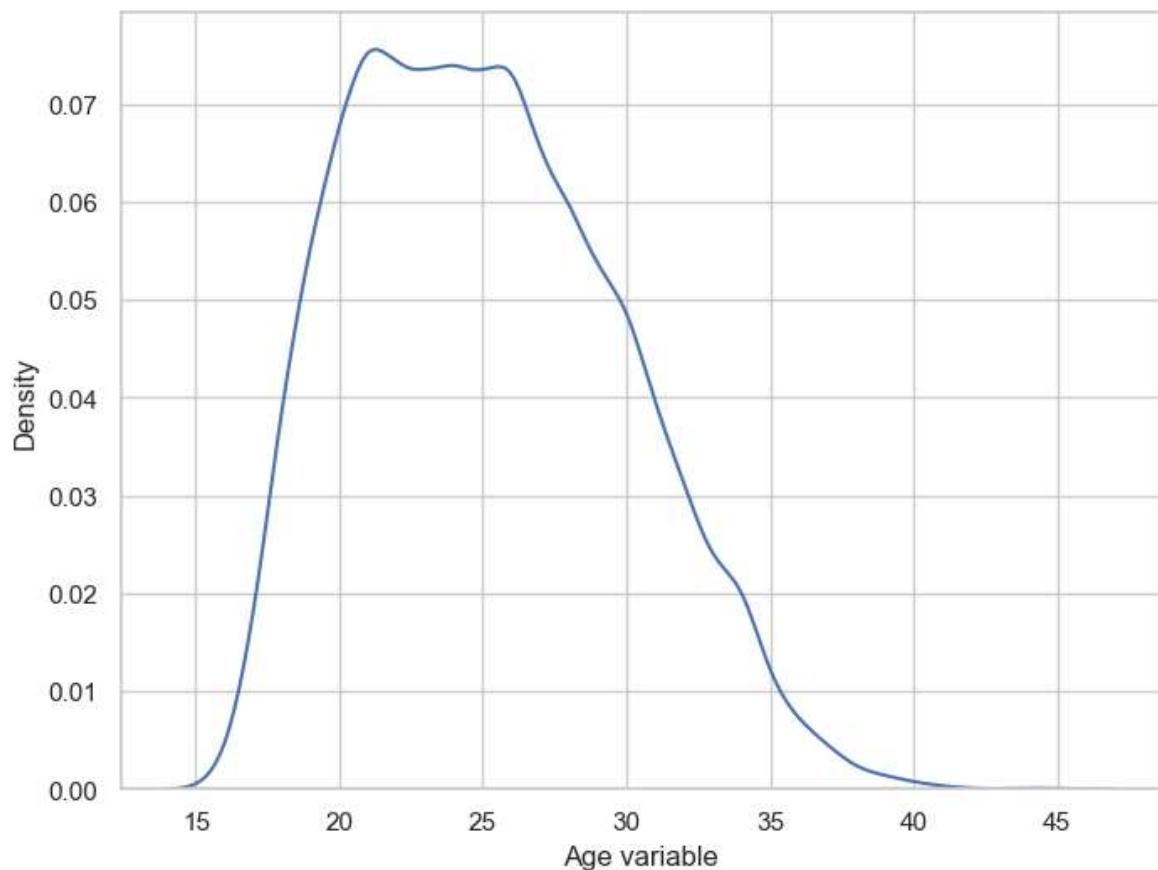
Out[7]: Normal          10595
        Lean            6417
        Stocky          1140
        Messi           1
        C. Ronaldo      1
        Neymar          1
        Courtois         1
        PLAYER_BODY_TYPE_25 1
        Shaqiri          1
        Akinfenwa        1
        Name: Body Type, dtype: int64

```

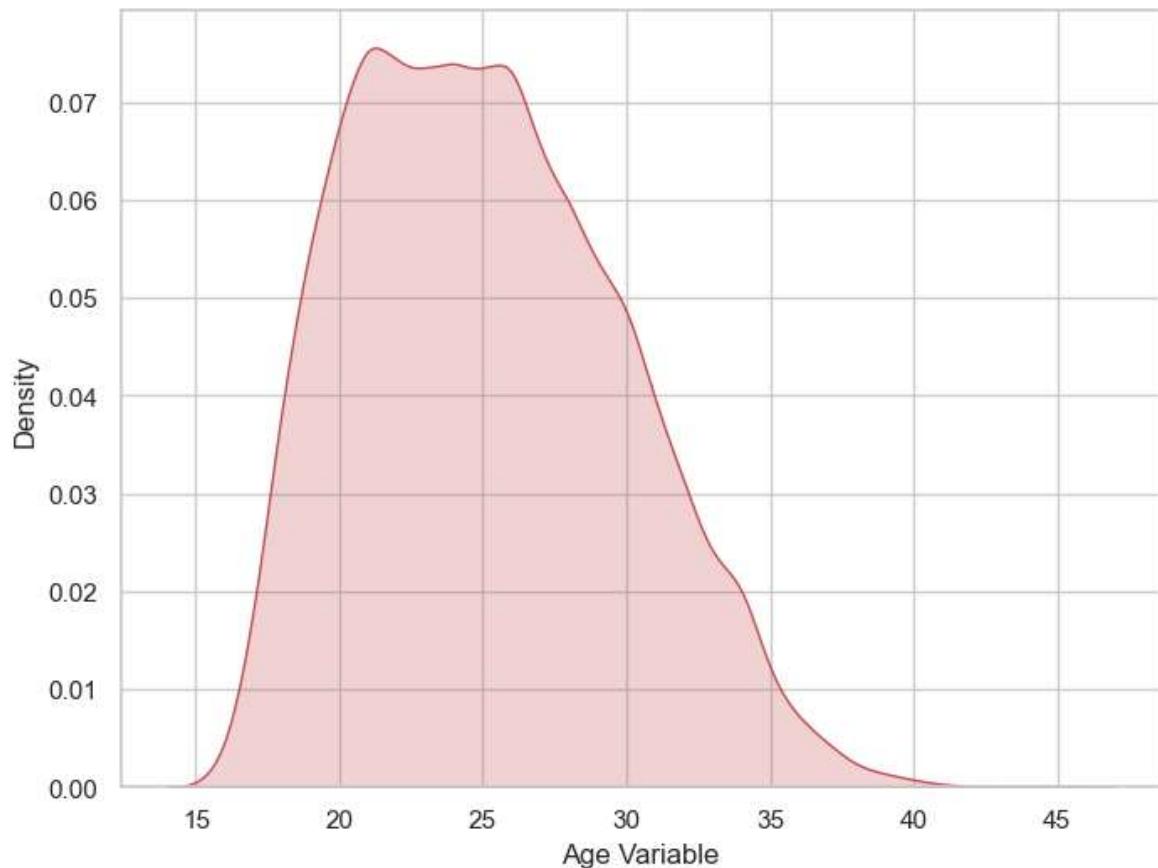
```
In [8]: f,ax=plt.subplots(figsize=(8,6))
x=fifa19['Age']
ax=sns.distplot(x,bins=10)
plt.show()
```



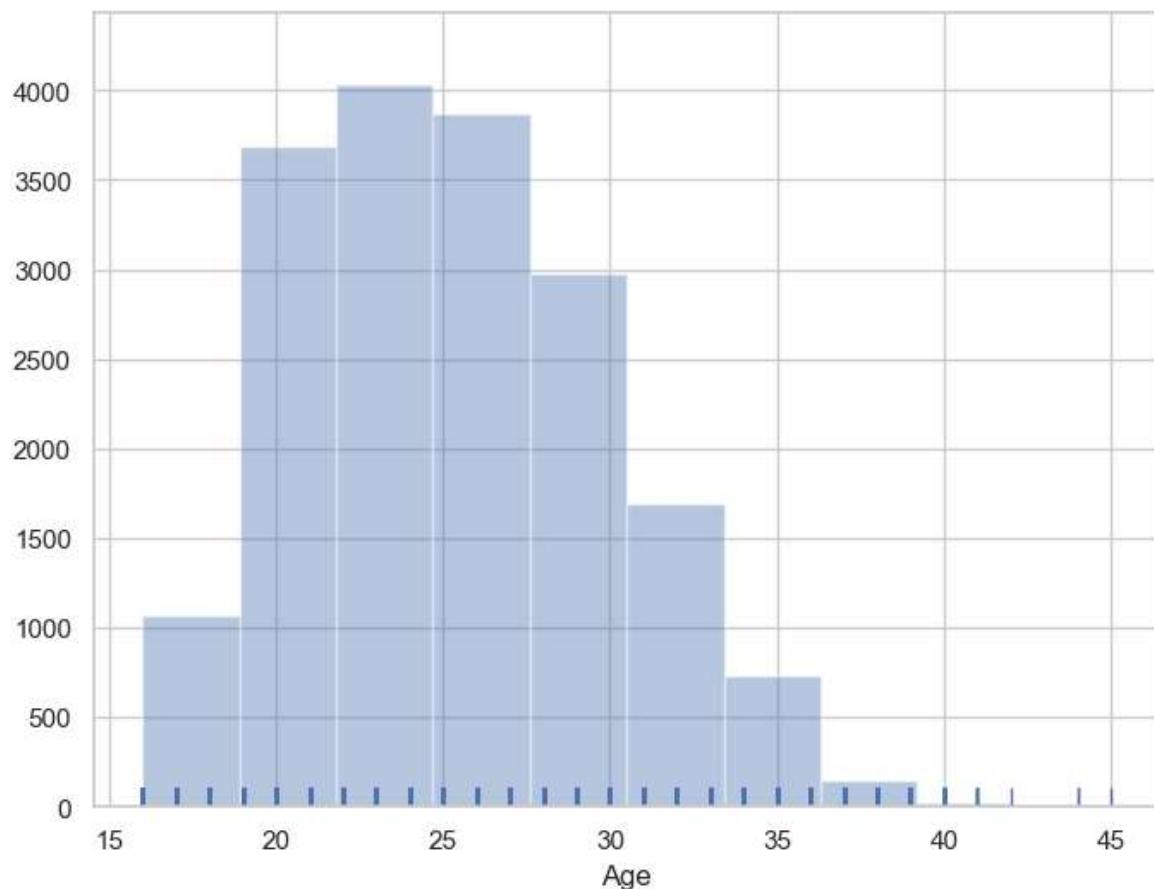
```
In [9]: f, ax = plt.subplots(figsize=(8,6))
x = fifa19['Age']
x = pd.Series(x, name="Age variable")
ax = sns.kdeplot(x)
plt.show()
```



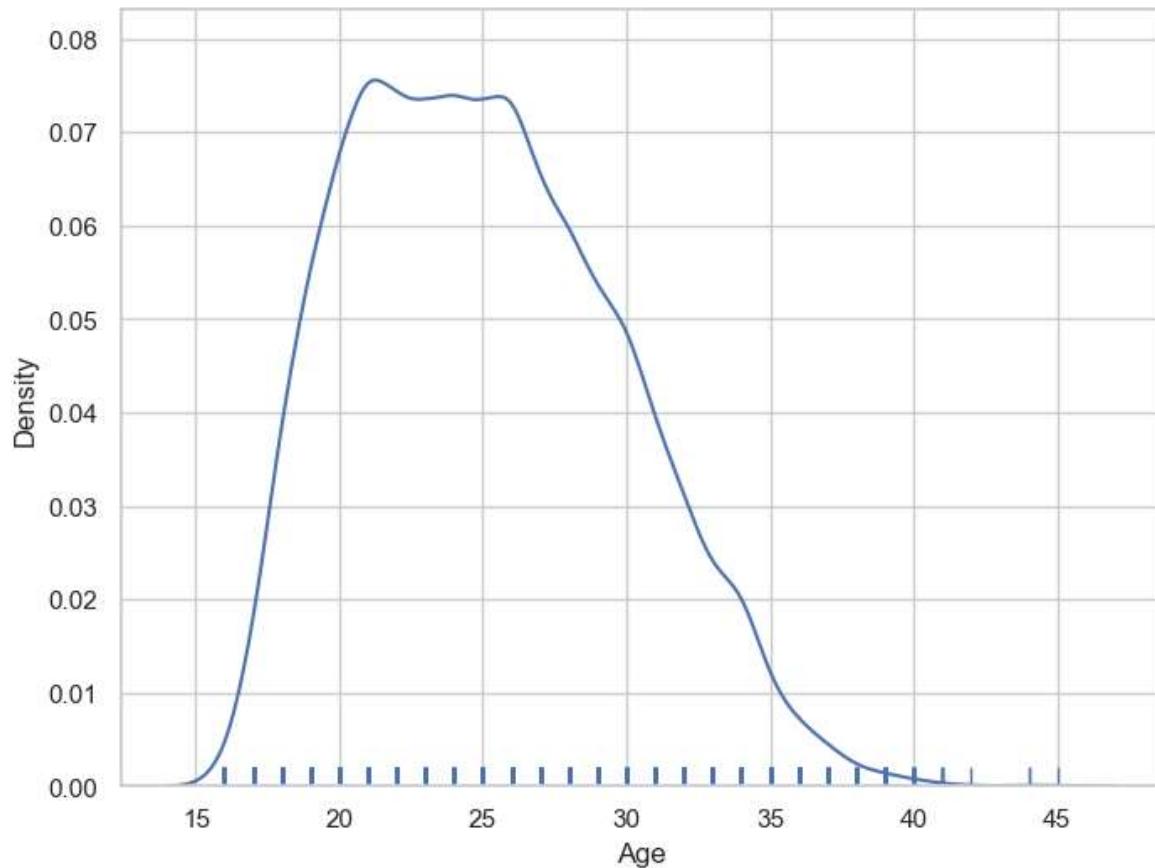
```
In [10]: f,ax=plt.subplots(figsize=(8,6))
x=fifa19["Age"]
x=pd.Series(x,name="Age Variable")
ax=sns.kdeplot(x,shade=True,color="r")
plt.show()
```



```
In [11]: f,ax=plt.subplots(figsize=(8,6))
x=fifa19["Age"]
x=sns.distplot(x,kde=False,rug=True,bins=10)
plt.show()
```



```
In [12]: f, ax = plt.subplots(figsize=(8,6))
x = fifa19['Age']
ax = sns.distplot(x, hist=False, rug=True, bins=10)
plt.show()
```



```
In [13]: fifa19['Preferred Foot'].nunique()
```

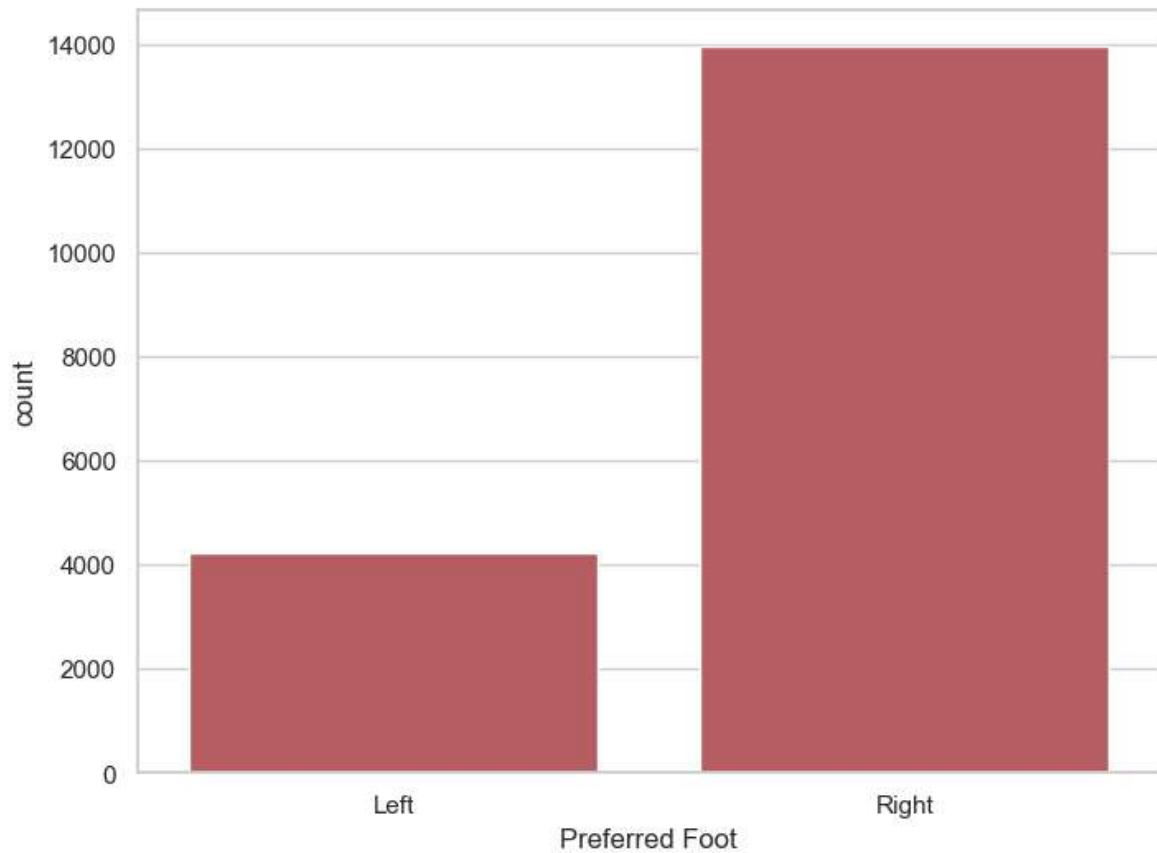
```
Out[13]: 2
```

```
In [14]: fifa19['Preferred Foot'].value_counts()
```

```
Out[14]: Right    13948
Left      4211
Name: Preferred Foot, dtype: int64
```

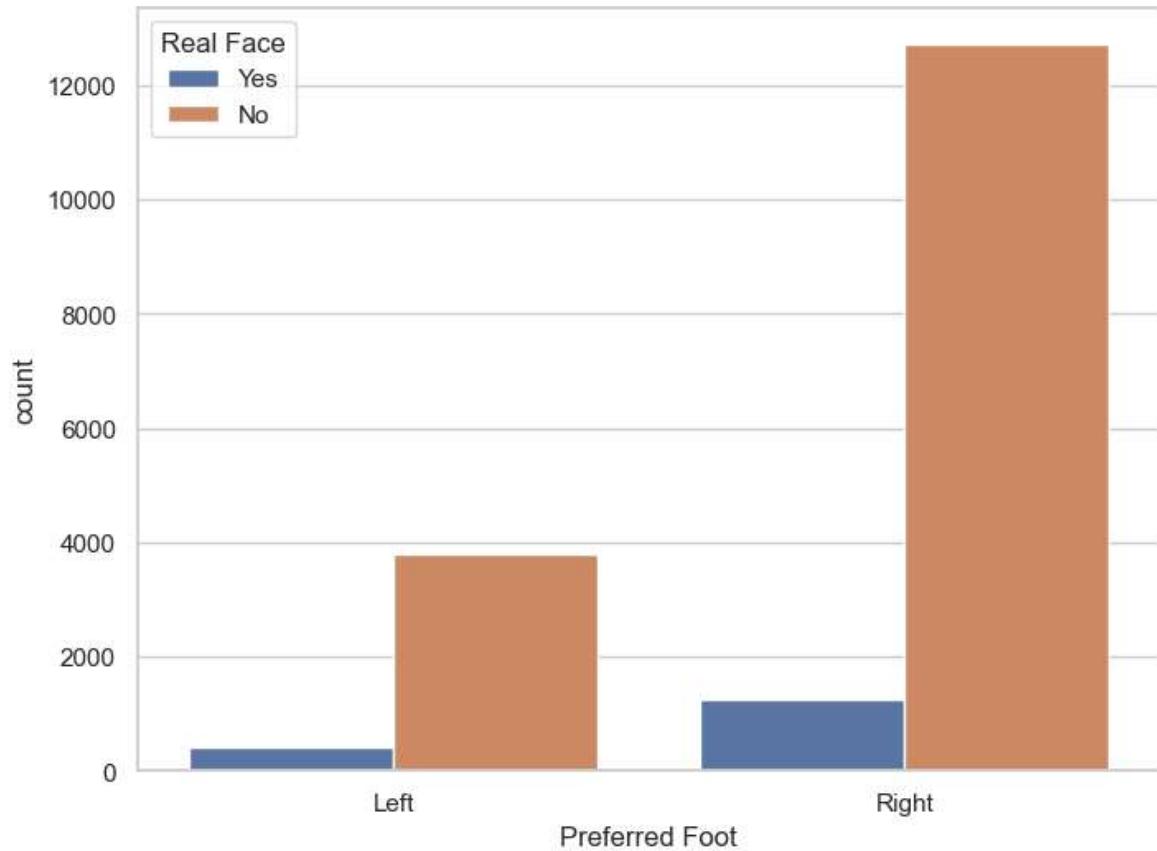
```
In [15]: f,ax=plt.subplots(figsize=(8,6))
sns.countplot(x='Preferred Foot',data=fifa19,color="r")
```

```
Out[15]: <Axes: xlabel='Preferred Foot', ylabel='count'>
```



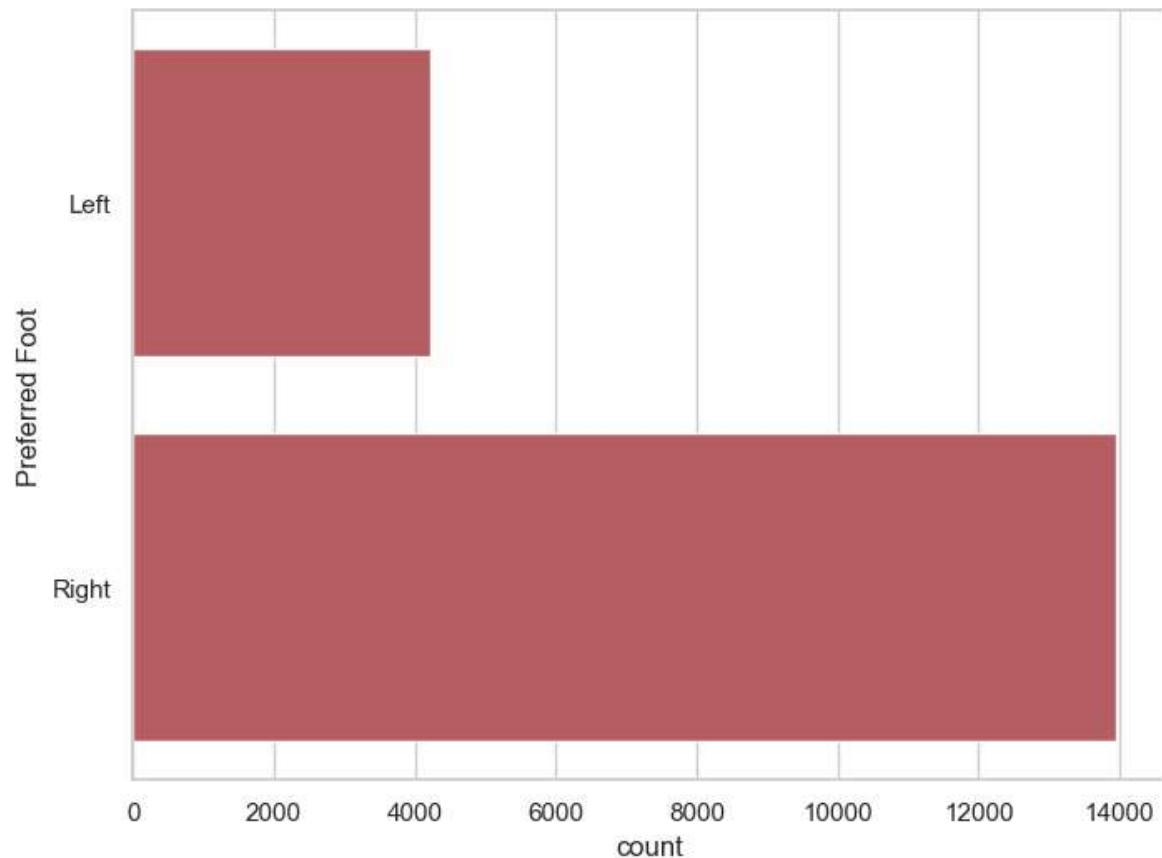
```
In [16]: f,ax=plt.subplots(figsize=(8,6))
sns.countplot(x='Preferred Foot',hue="Real Face",data=fifa19)
```

```
Out[16]: <Axes: xlabel='Preferred Foot', ylabel='count'>
```

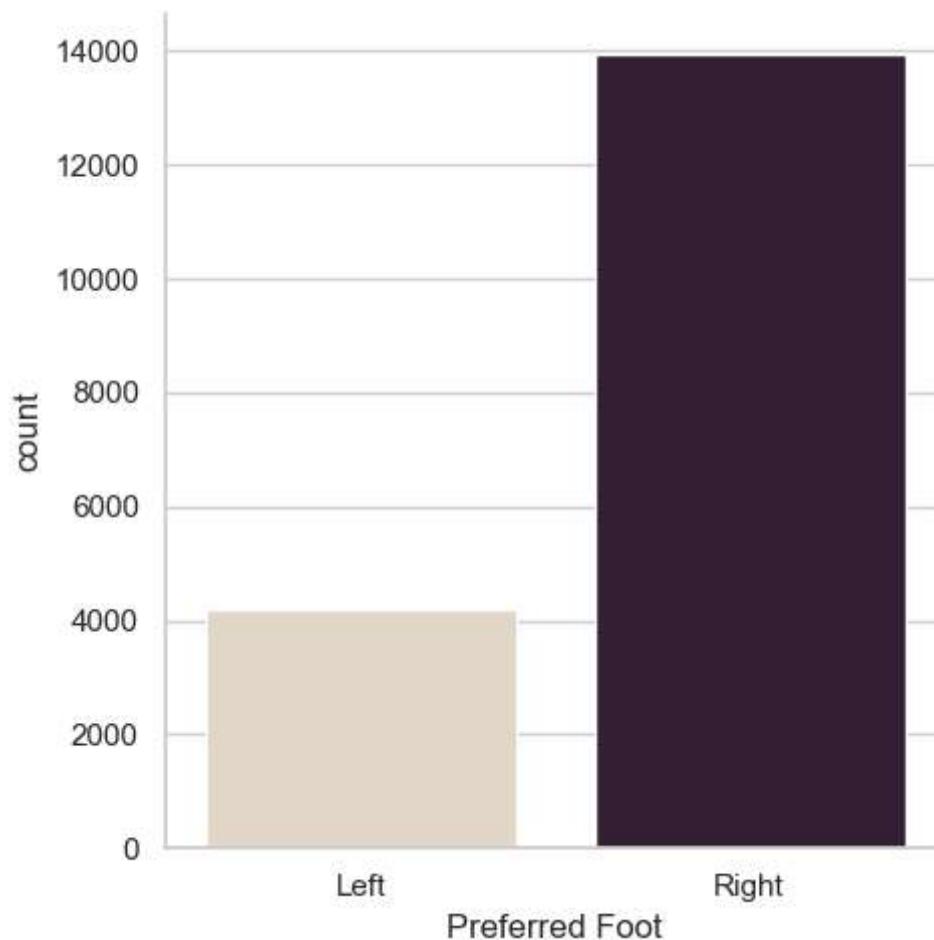


```
In [17]: f, x = plt.subplots(figsize=(8,6))
sns.countplot(y='Preferred Foot', data=fifa19, color="r")
```

```
Out[17]: <Axes: xlabel='count', ylabel='Preferred Foot'>
```



```
In [18]: g=sns.catplot(x="Preferred Foot",kind="count",data=fifa19,palette="ch:.25")
```



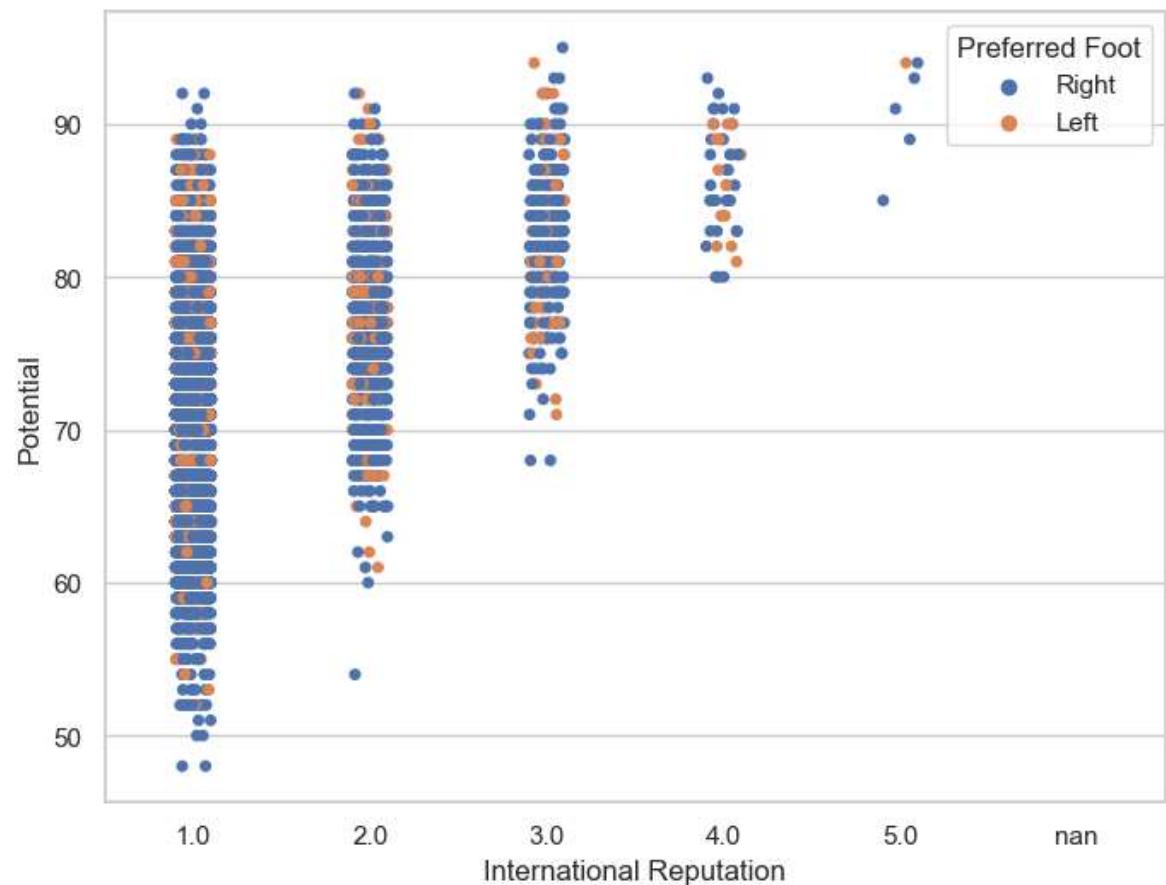
```
In [19]: fifa19["International Reputation"].nunique()
```

```
Out[19]: 5
```

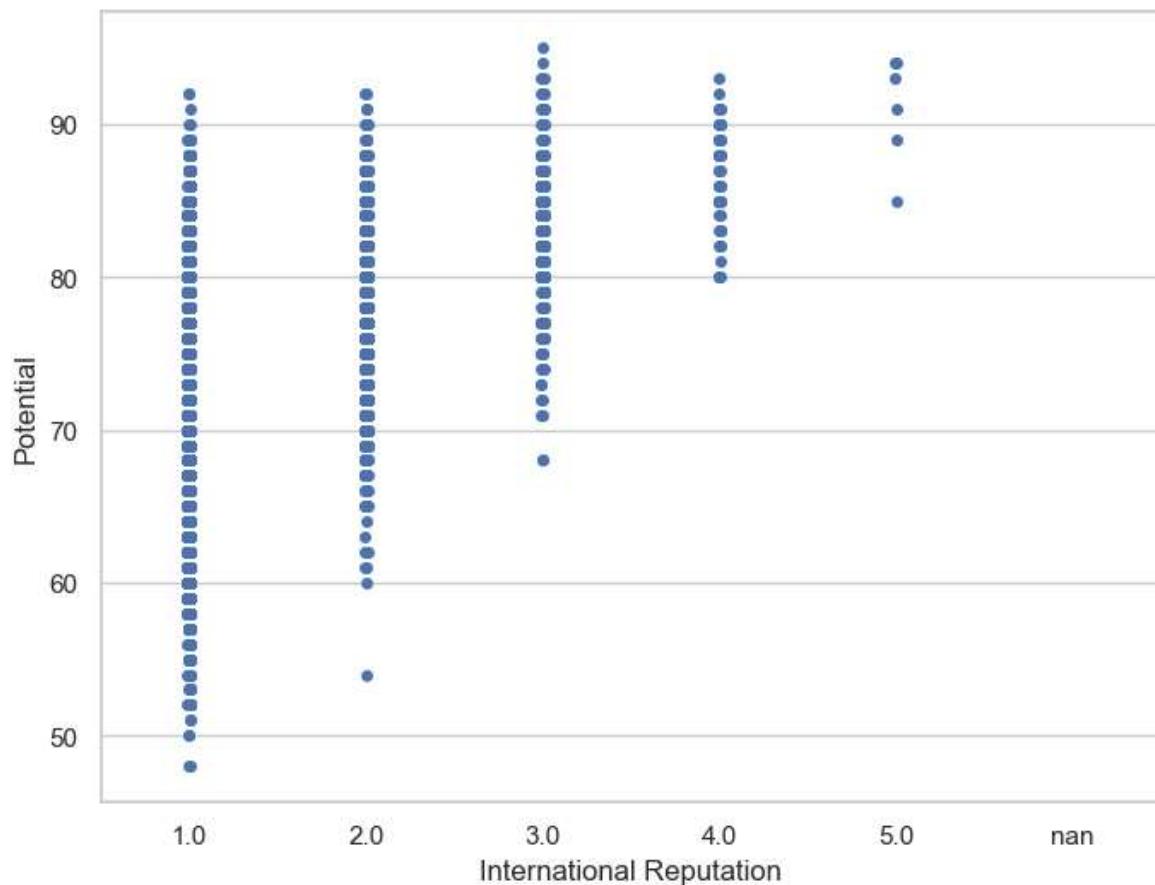
```
In [20]: fifa19['International Reputation'].value_counts()
```

```
Out[20]: 1.0    16532
2.0    1261
3.0     309
4.0      51
5.0      6
Name: International Reputation, dtype: int64
```

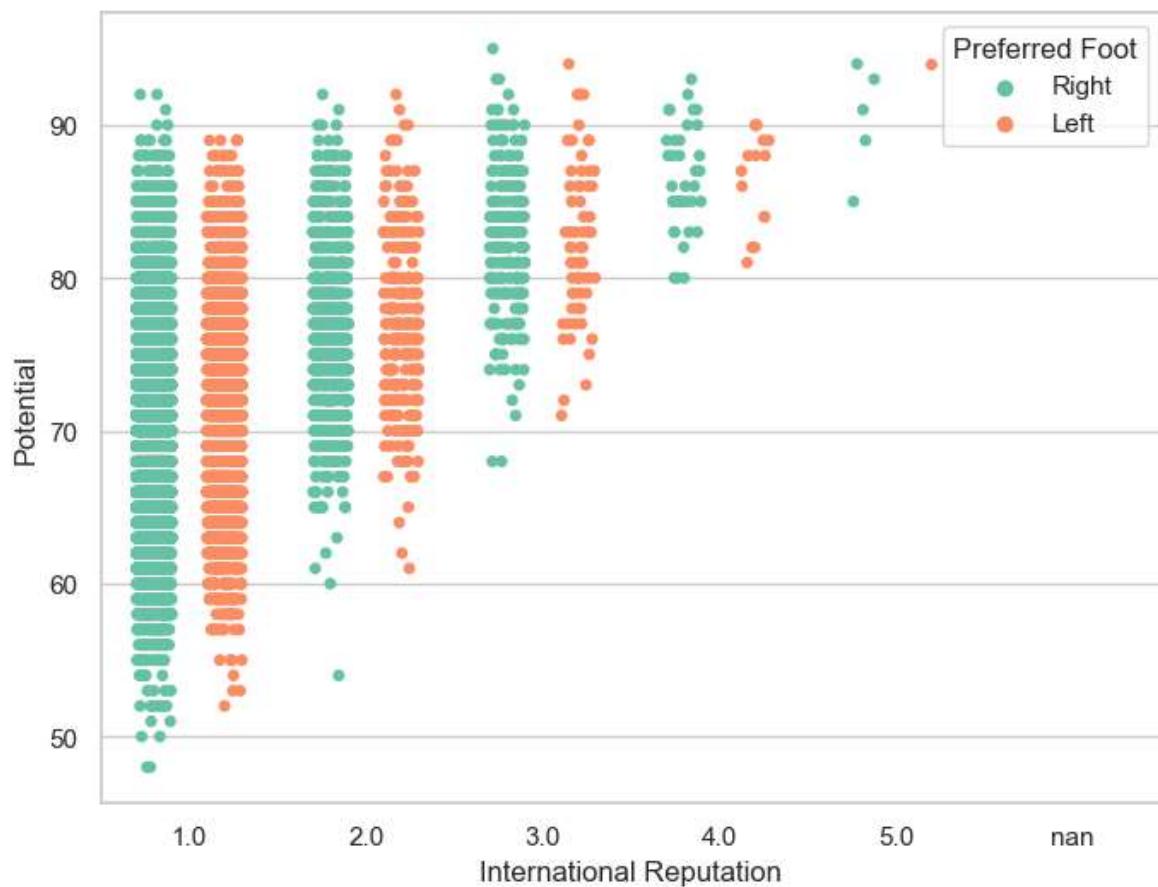
```
In [21]: f,ax=plt.subplots(figsize=(8,6))
sns.stripplot(x="International Reputation",y="Potential",data=fifa19, hue="Preferred Foot")
plt.show()
```



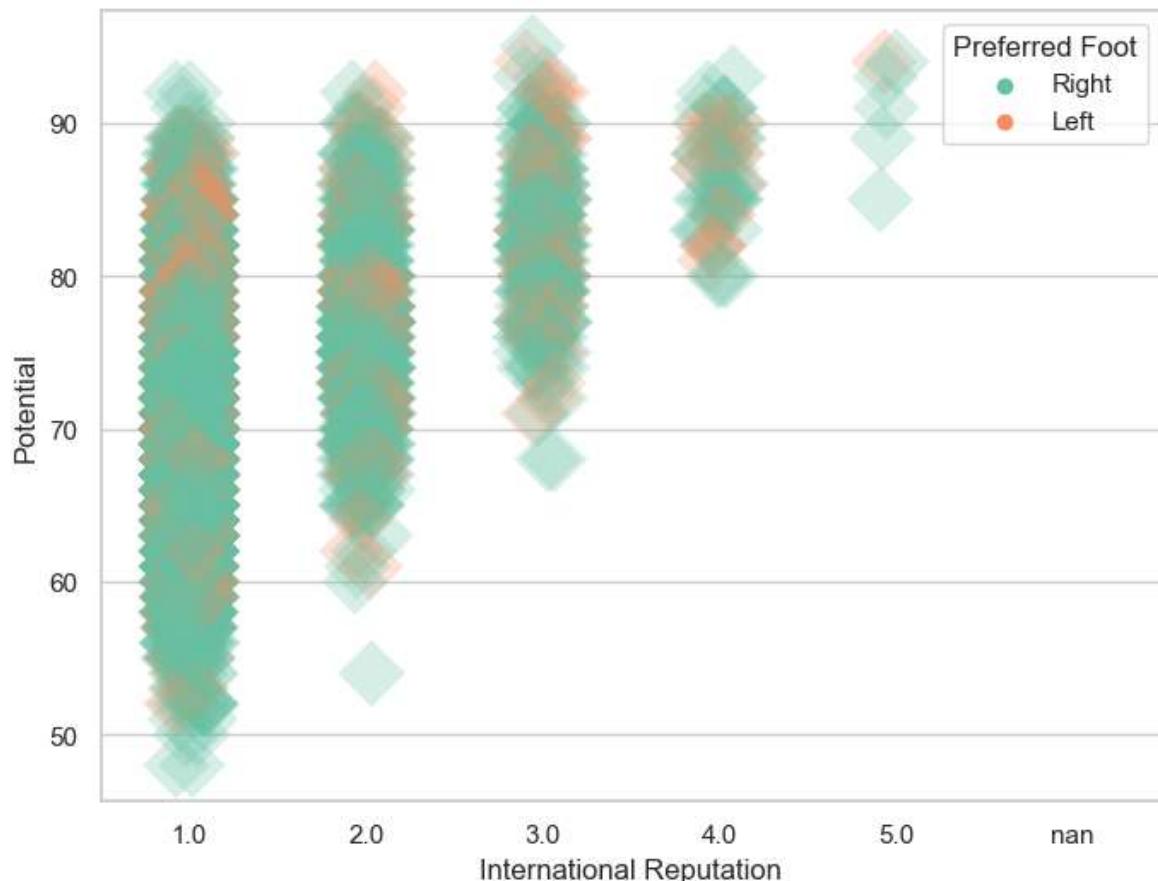
```
In [22]: f, ax = plt.subplots(figsize=(8, 6))
sns.stripplot(x="International Reputation", y="Potential", data=fifa19, jitter=True)
plt.show()
```



```
In [23]: f, ax = plt.subplots(figsize=(8, 6))
sns.stripplot(x="International Reputation", y="Potential", hue="Preferred Foot",
               data=fifa19, jitter=0.2, palette="Set2", dodge=True)
plt.show()
```

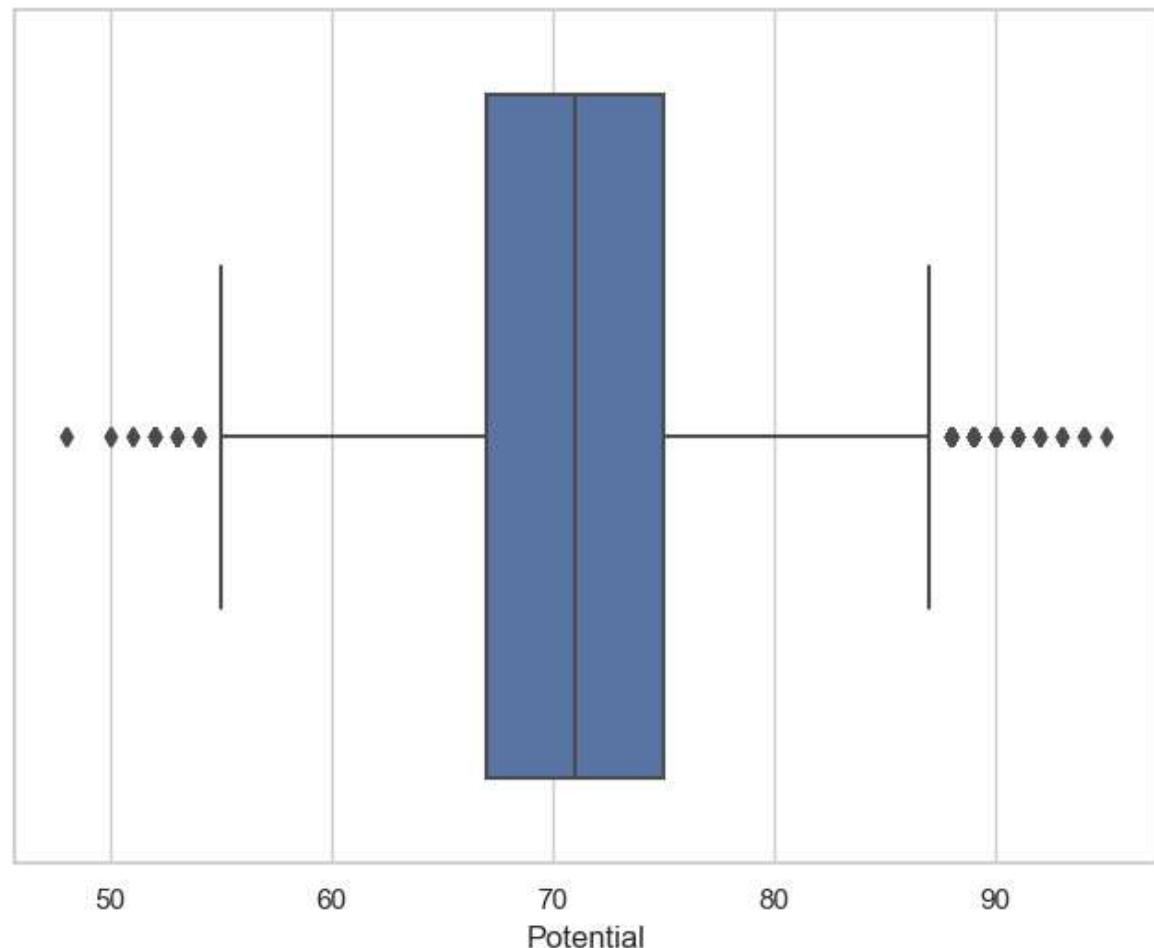


```
In [24]: f, ax = plt.subplots(figsize=(8, 6))
sns.stripplot(x="International Reputation", y="Potential", hue="Preferred Foot",
               data=fifa19, palette="Set2", size=20, marker="D",
               edgecolor="gray", alpha=.25)
plt.show()
```

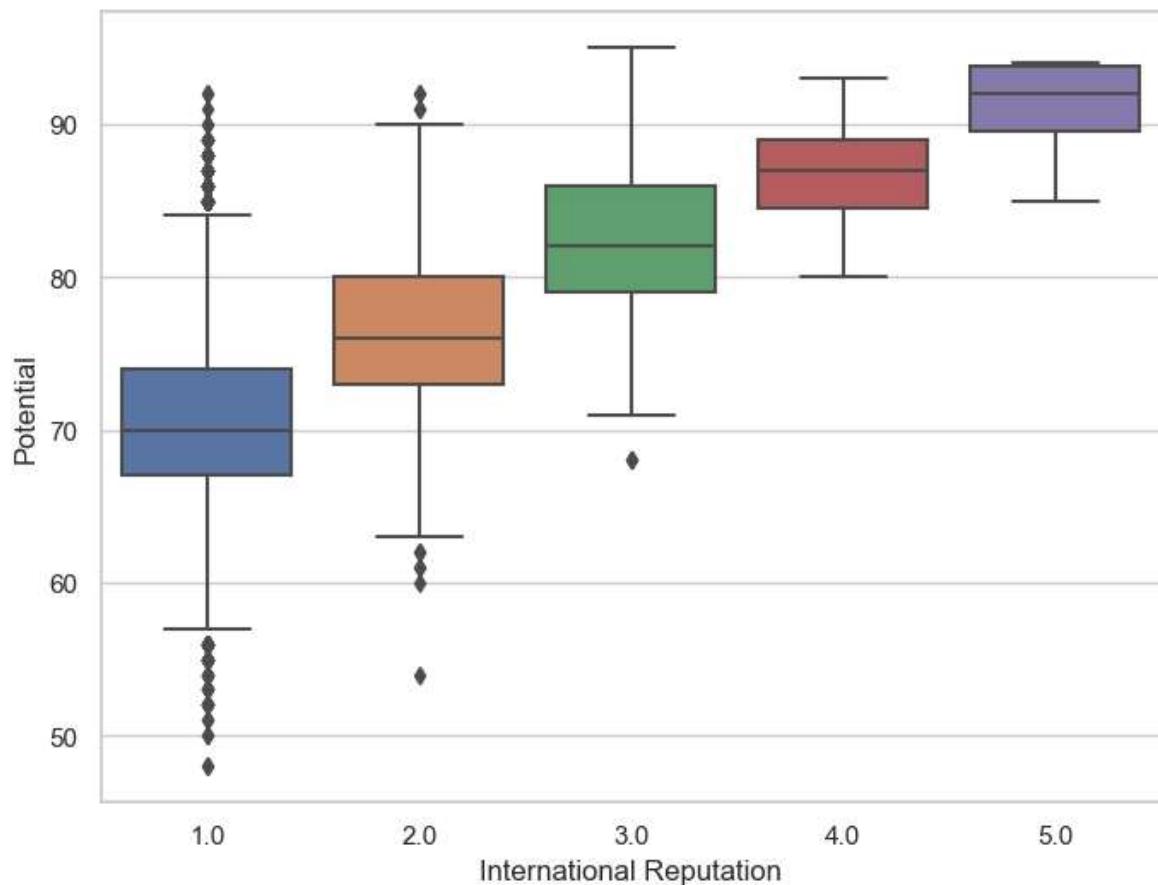


```
In [25]: f,ax=plt.subplots(figsize=(8,6))
sns.boxplot(x=fifa19["Potential"])
```

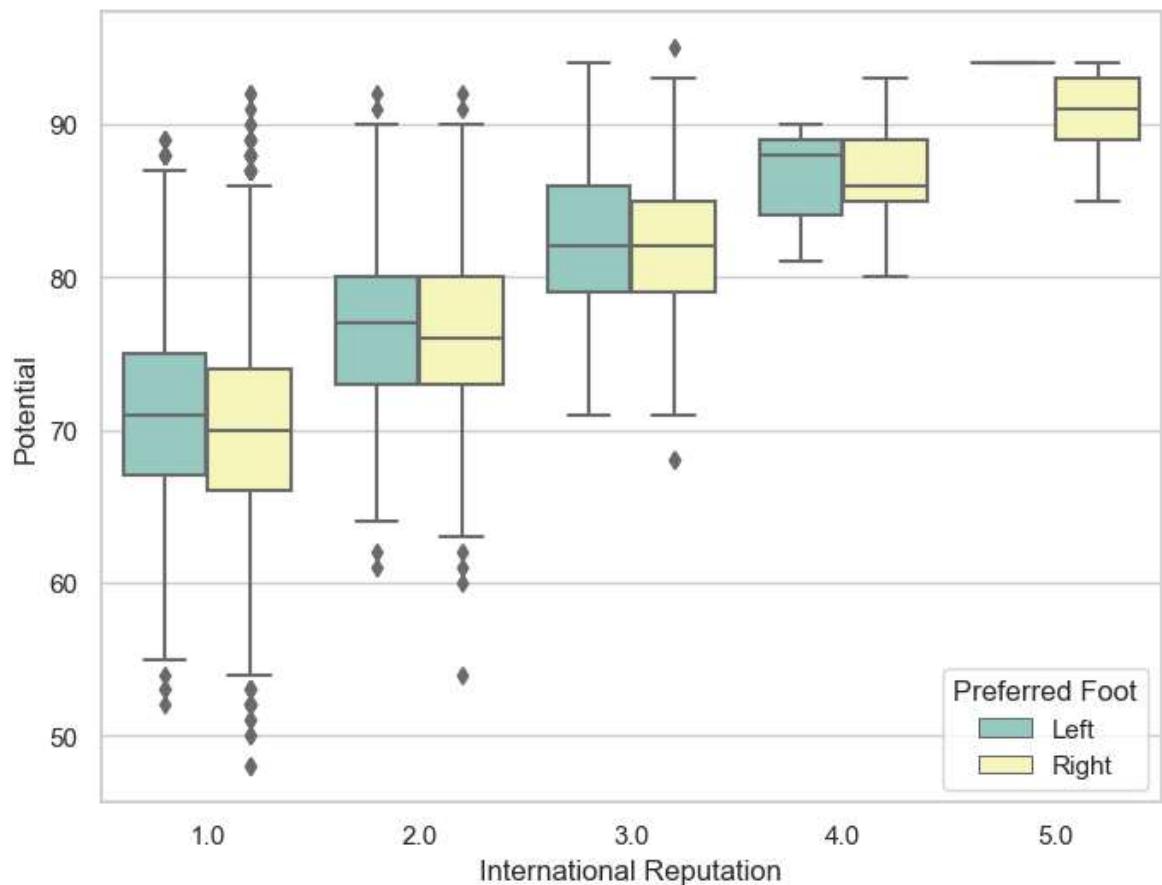
```
Out[25]: <Axes: xlabel='Potential'>
```



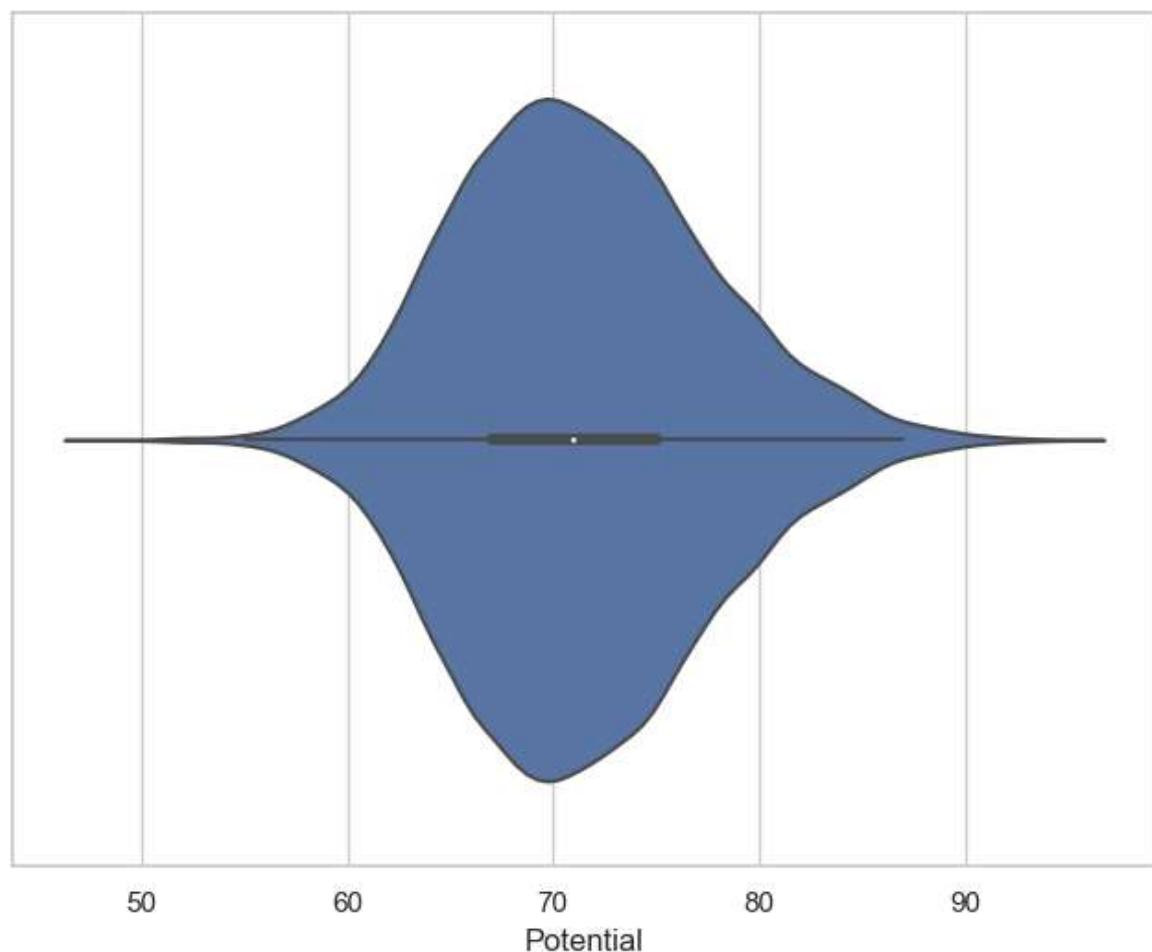
```
In [26]: f,ax = plt.subplots(figsize=(8,6))
sns.boxplot(x="International Reputation",y="Potential",data=fifa19)
plt.show()
```



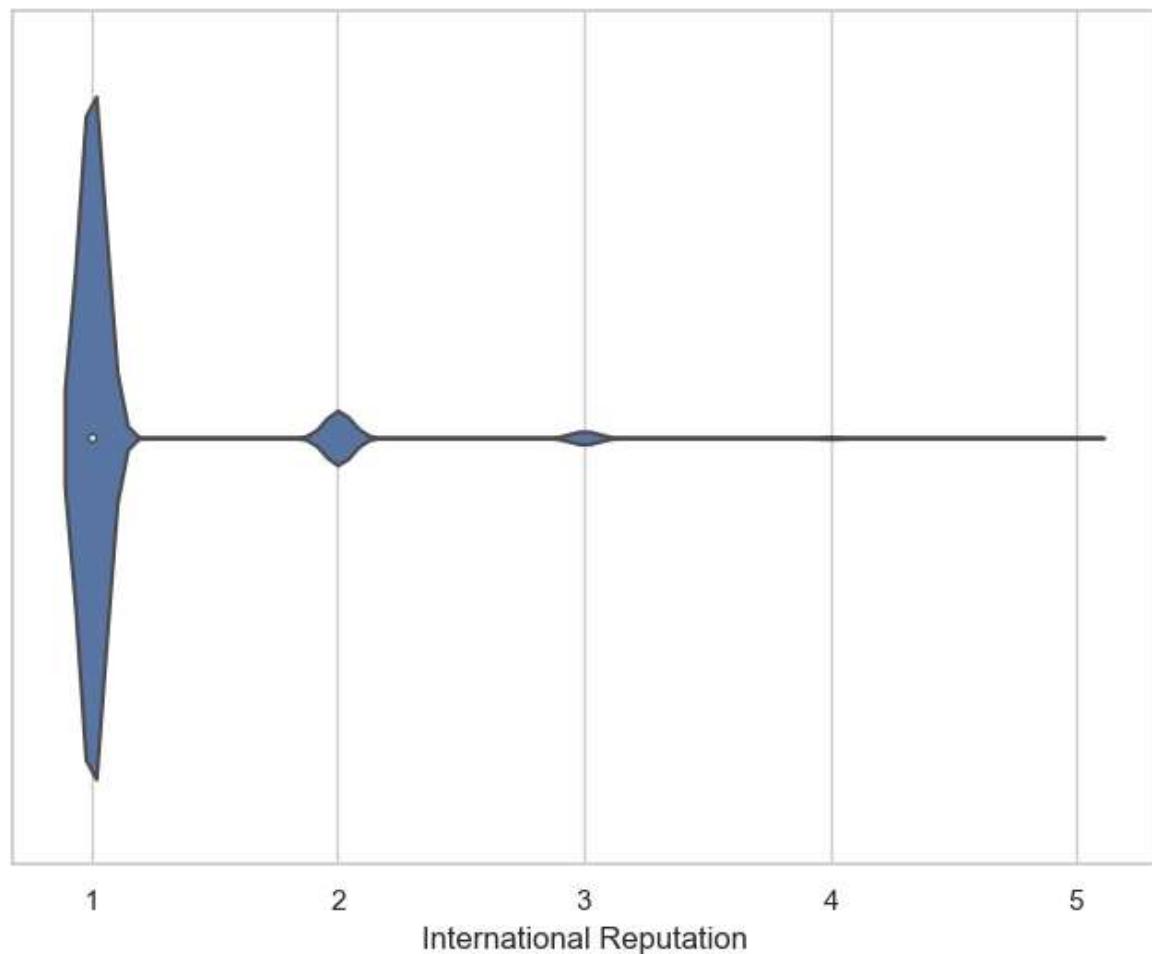
```
In [27]: f,ax = plt.subplots(figsize=(8,6))
sns.boxplot(x="International Reputation",y="Potential",hue="Preferred Foot",data=d)
plt.show()
```



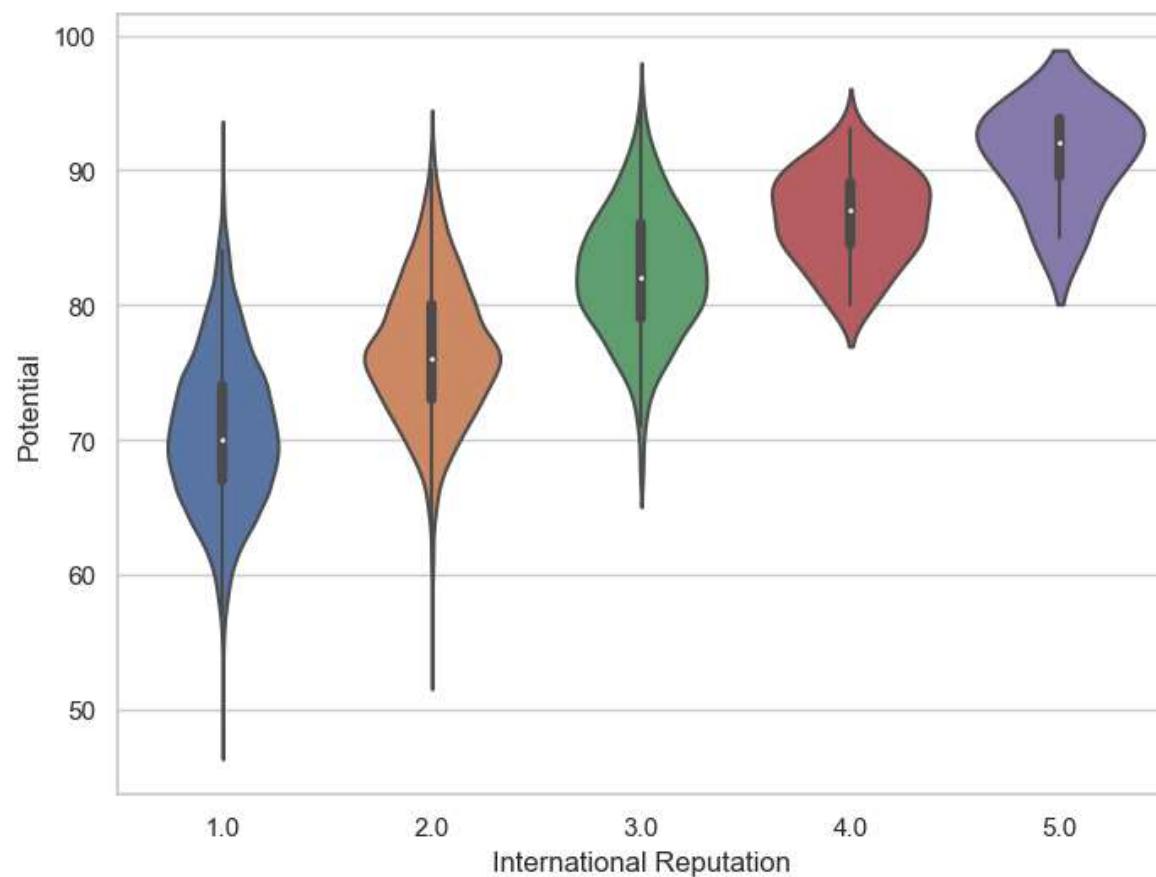
```
In [28]: f, ax = plt.subplots(figsize=(8, 6))
sns.violinplot(x=fifa19["Potential"])
plt.show()
```



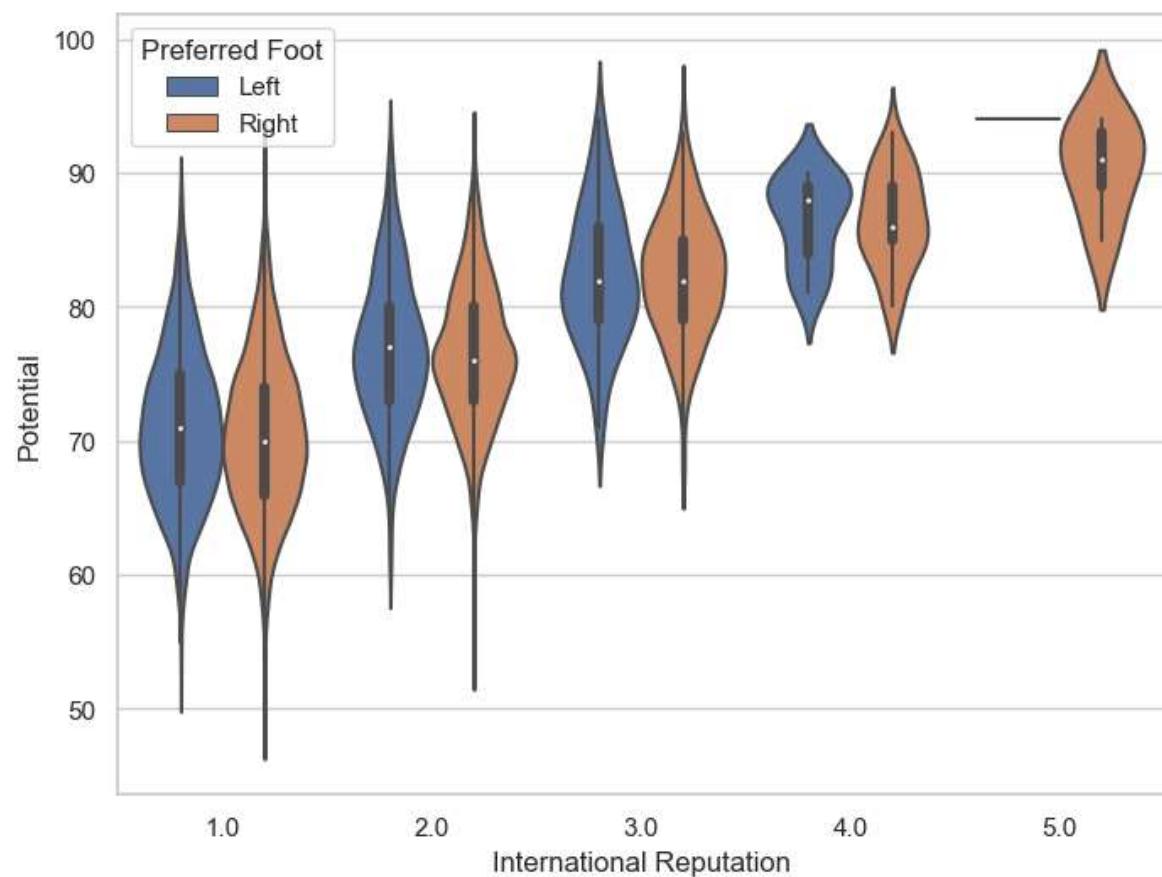
```
In [29]: f, ax = plt.subplots(figsize=(8, 6))
sns.violinplot(x=fifa19["International Reputation"])
plt.show()
```



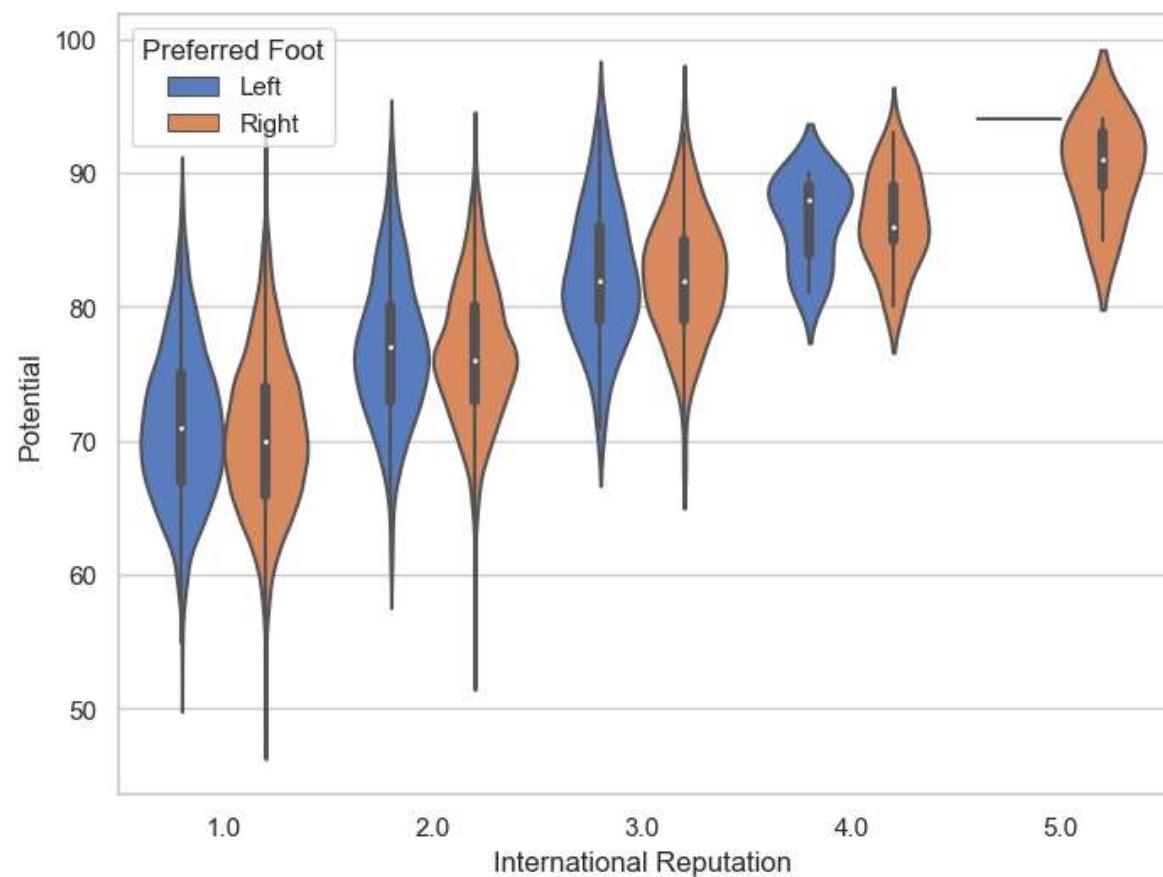
```
In [30]: f, ax = plt.subplots(figsize=(8, 6))
sns.violinplot(x="International Reputation", y="Potential", data=fifa19)
plt.show()
```



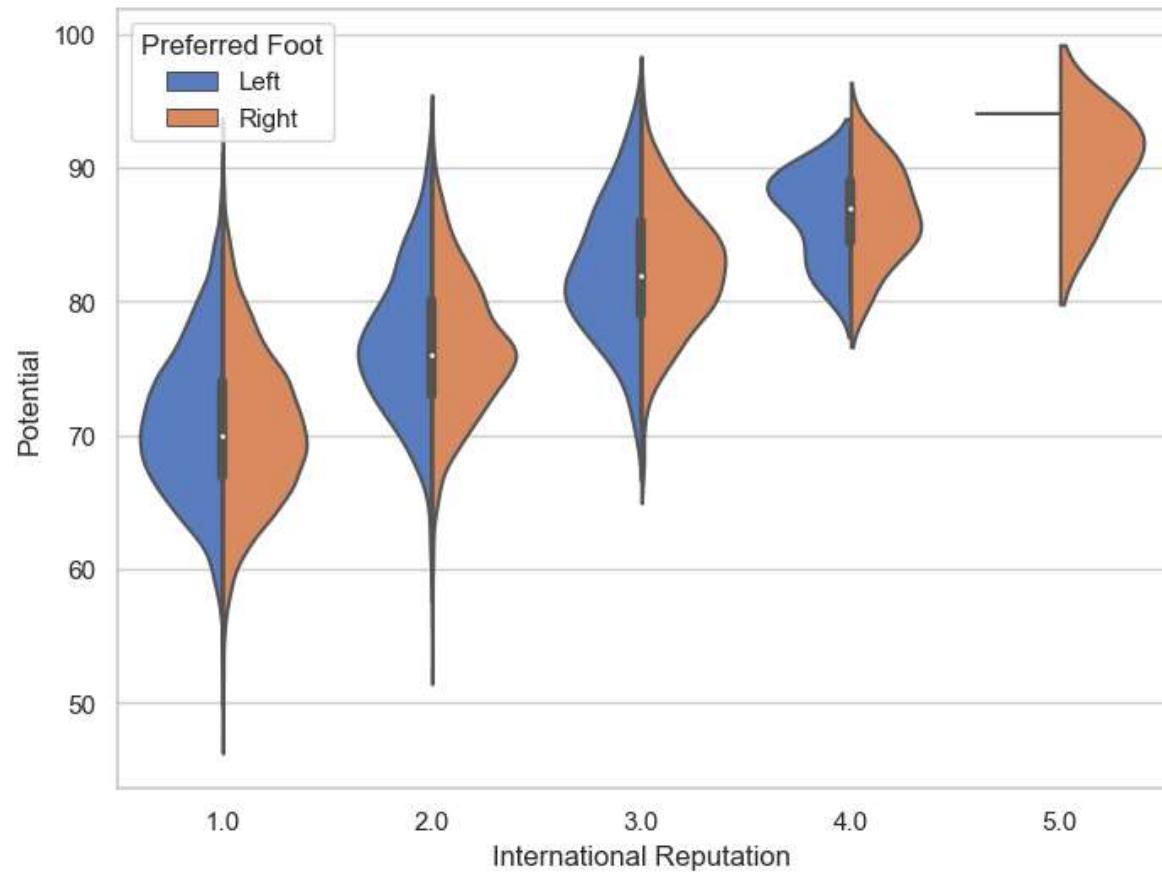
```
In [31]: f, ax = plt.subplots(figsize=(8, 6))
sns.violinplot(x="International Reputation", y="Potential", hue="Preferred Foot")
plt.show()
```



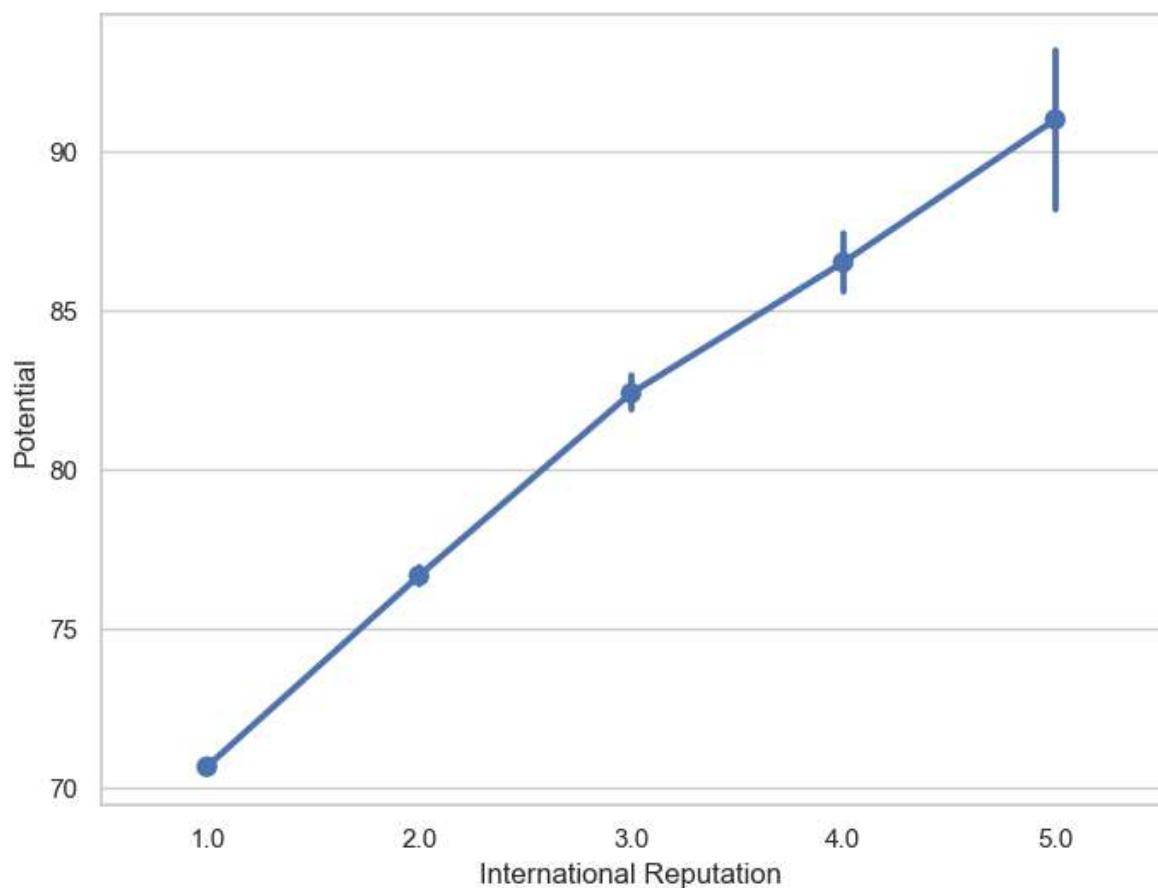
```
In [32]: f, ax = plt.subplots(figsize=(8, 6))
sns.violinplot(x="International Reputation", y="Potential", hue="Preferred Foot")
plt.show()
```



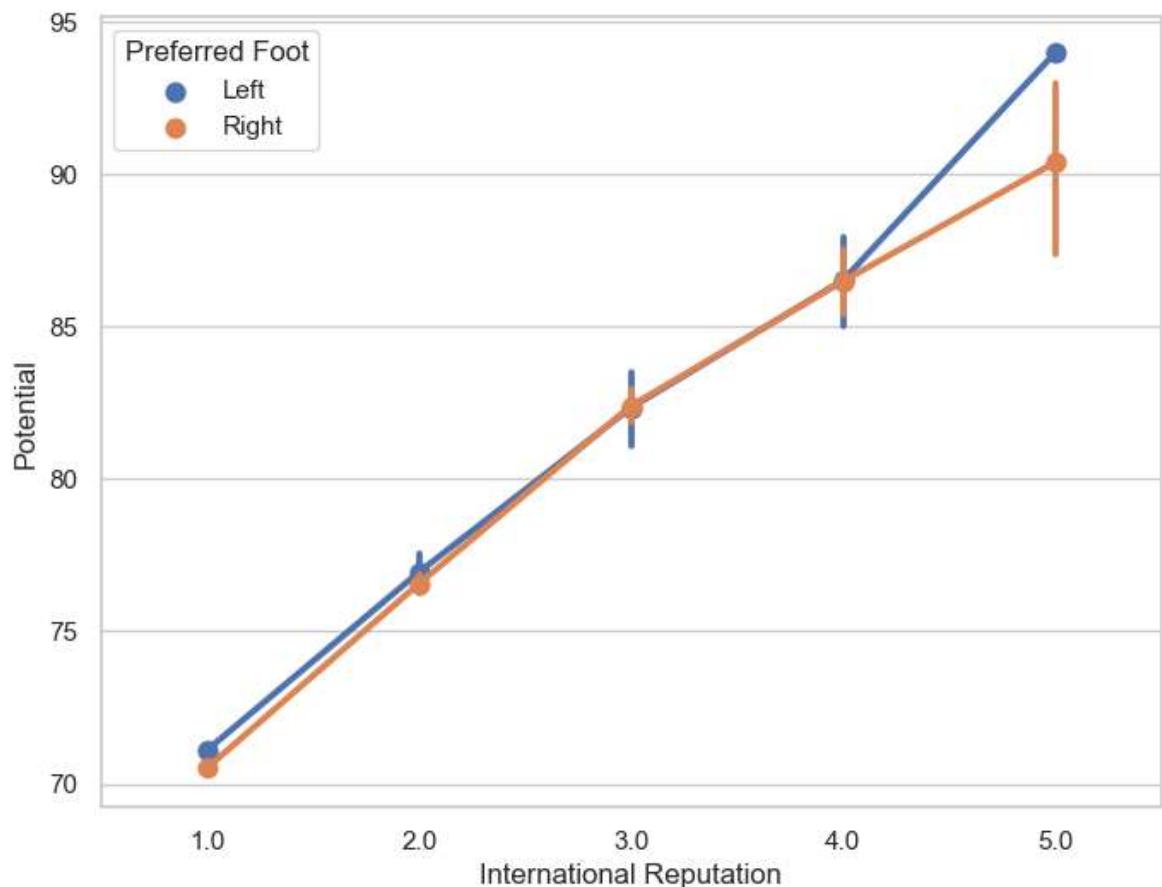
```
In [33]: f, ax = plt.subplots(figsize=(8, 6))
sns.violinplot(x="International Reputation", y="Potential", hue="Preferred Foot",
                data=fifa19, palette="muted", split=True)
plt.show()
```



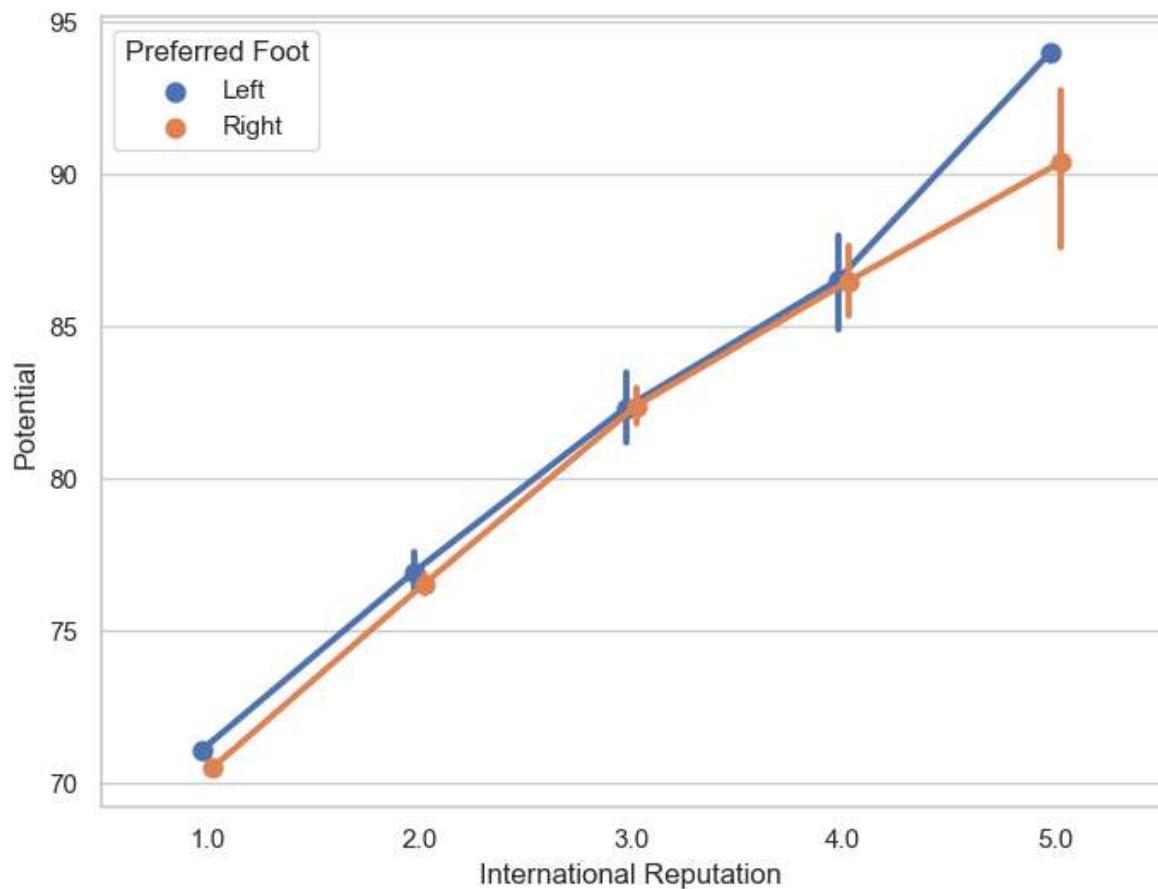
```
In [34]: f, ax = plt.subplots(figsize=(8, 6))
sns.pointplot(x="International Reputation", y="Potential", data=fifa19)
plt.show()
```



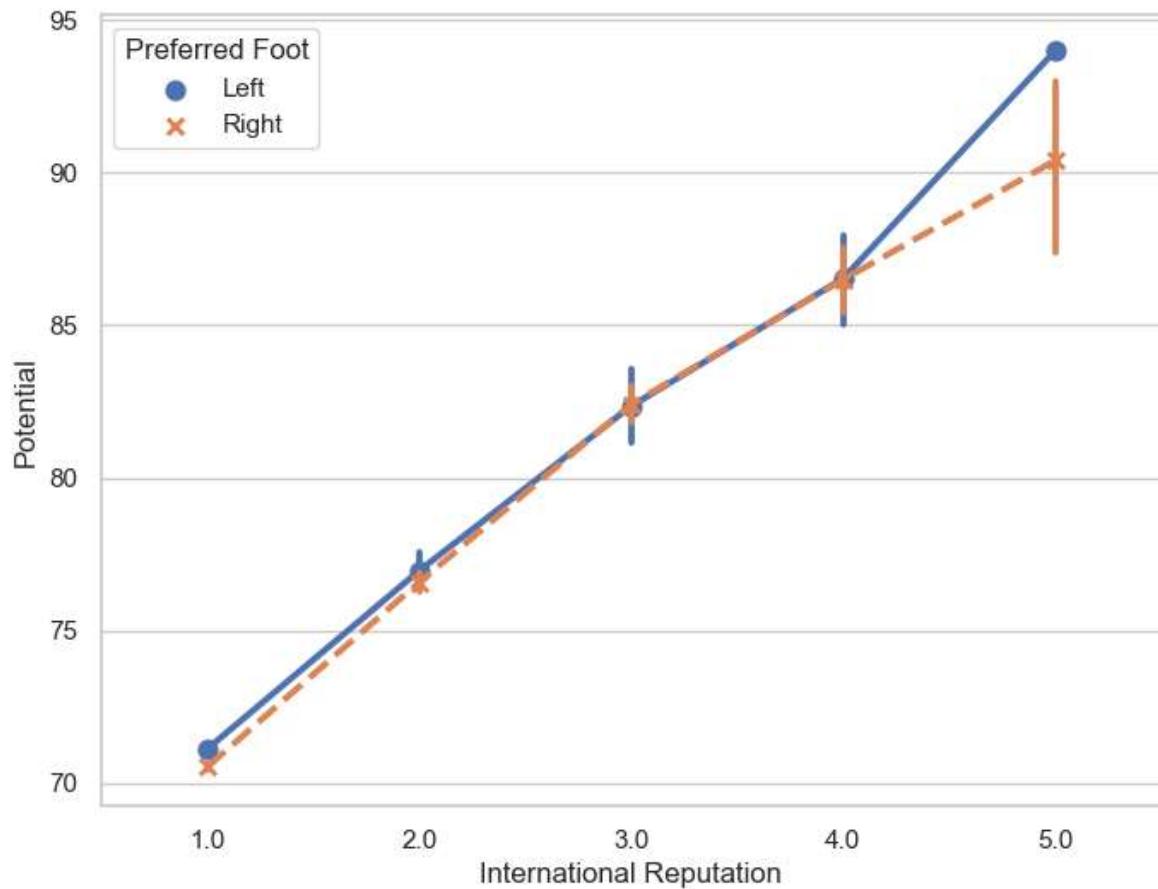
```
In [35]: f, ax = plt.subplots(figsize=(8, 6))
sns.pointplot(x="International Reputation", y="Potential", hue="Preferred Foot",
plt.show()
```



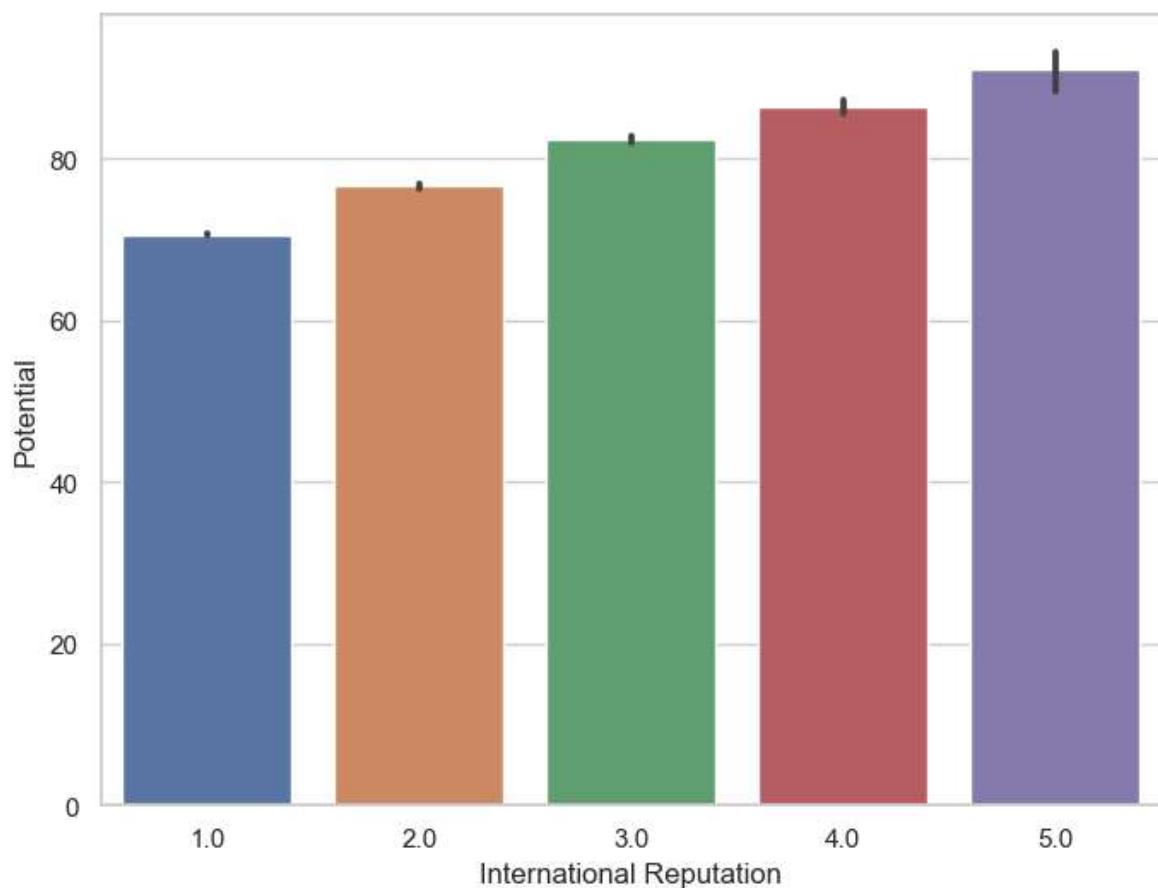
```
In [36]: f, ax = plt.subplots(figsize=(8, 6))
sns.pointplot(x="International Reputation", y="Potential", hue="Preferred Foot",
plt.show()
```



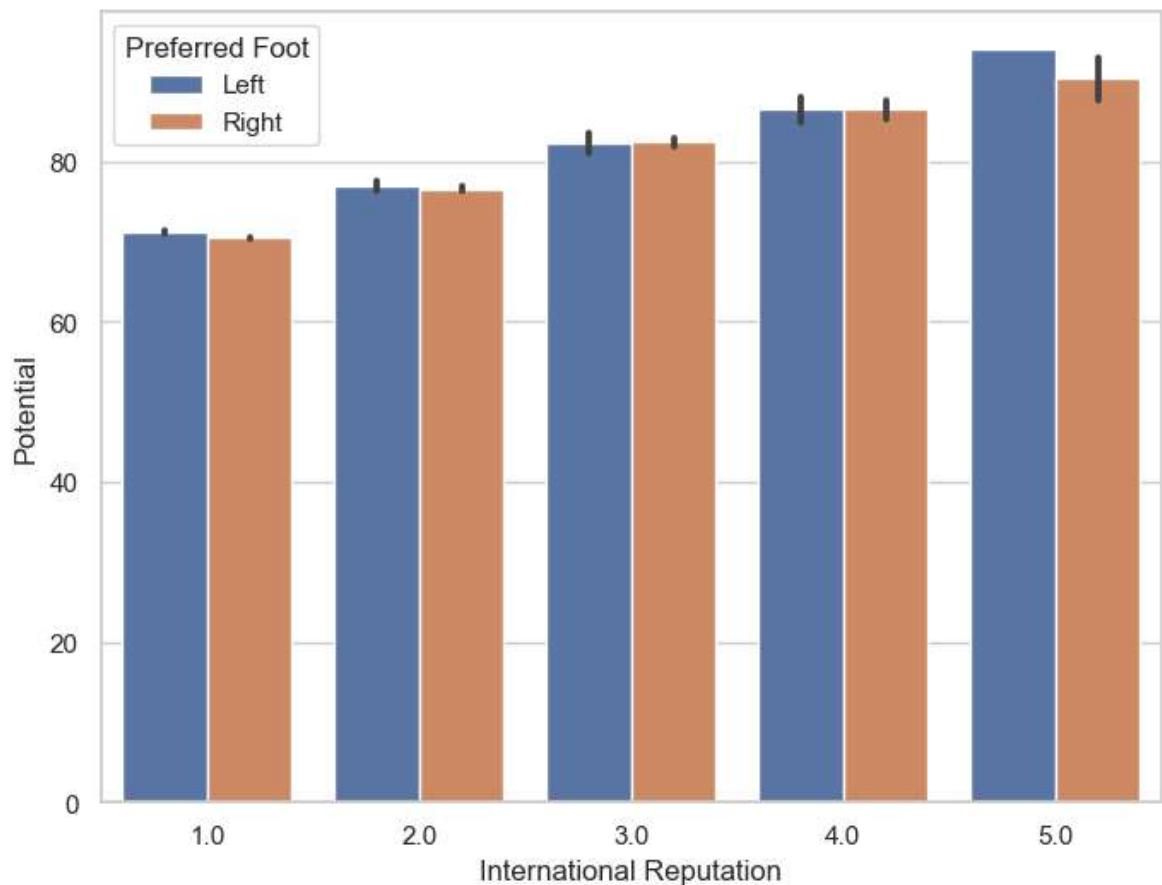
```
In [37]: f, ax = plt.subplots(figsize=(8, 6))
sns.pointplot(x="International Reputation", y="Potential", hue="Preferred Foot",
               data=fifa19, markers=["o", "x"], linestyles=["-", "--"])
plt.show()
```



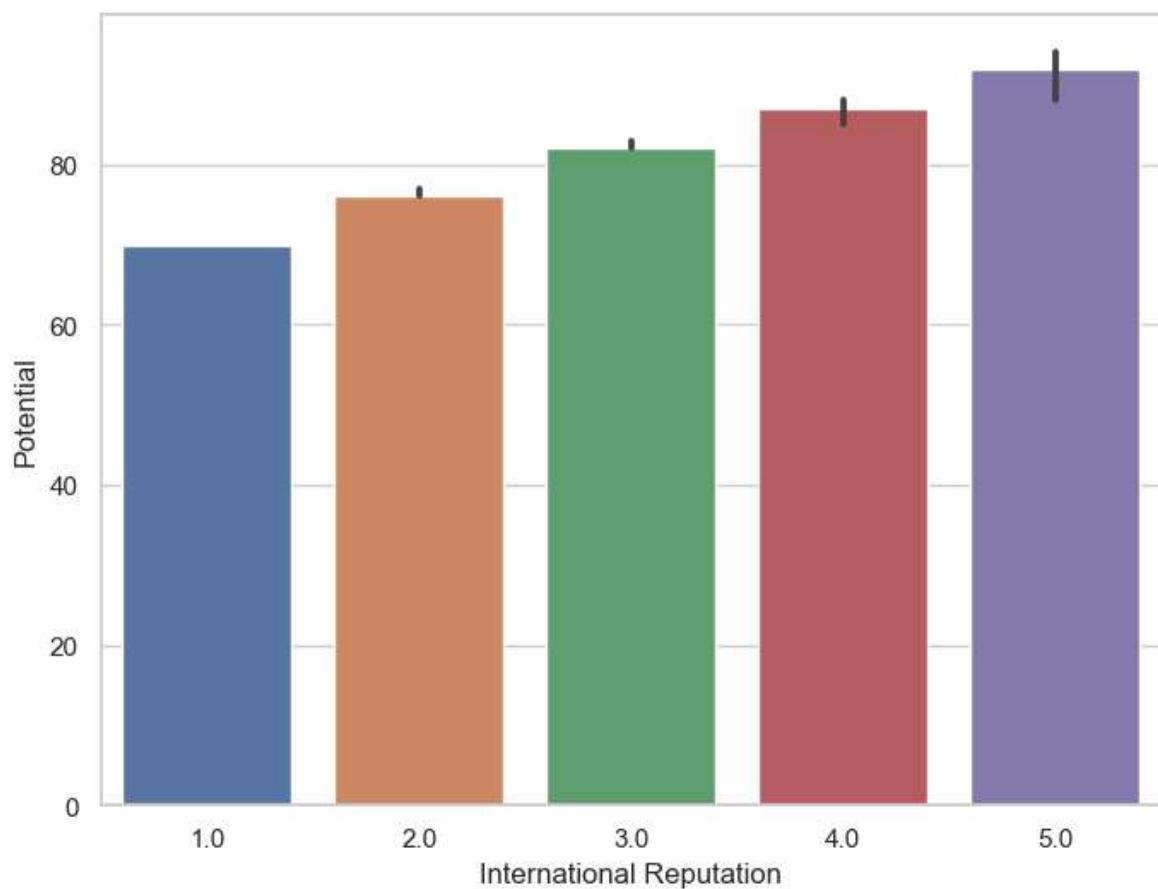
```
In [38]: f, ax = plt.subplots(figsize=(8, 6))
sns.barplot(x="International Reputation", y="Potential", data=fifa19)
plt.show()
```



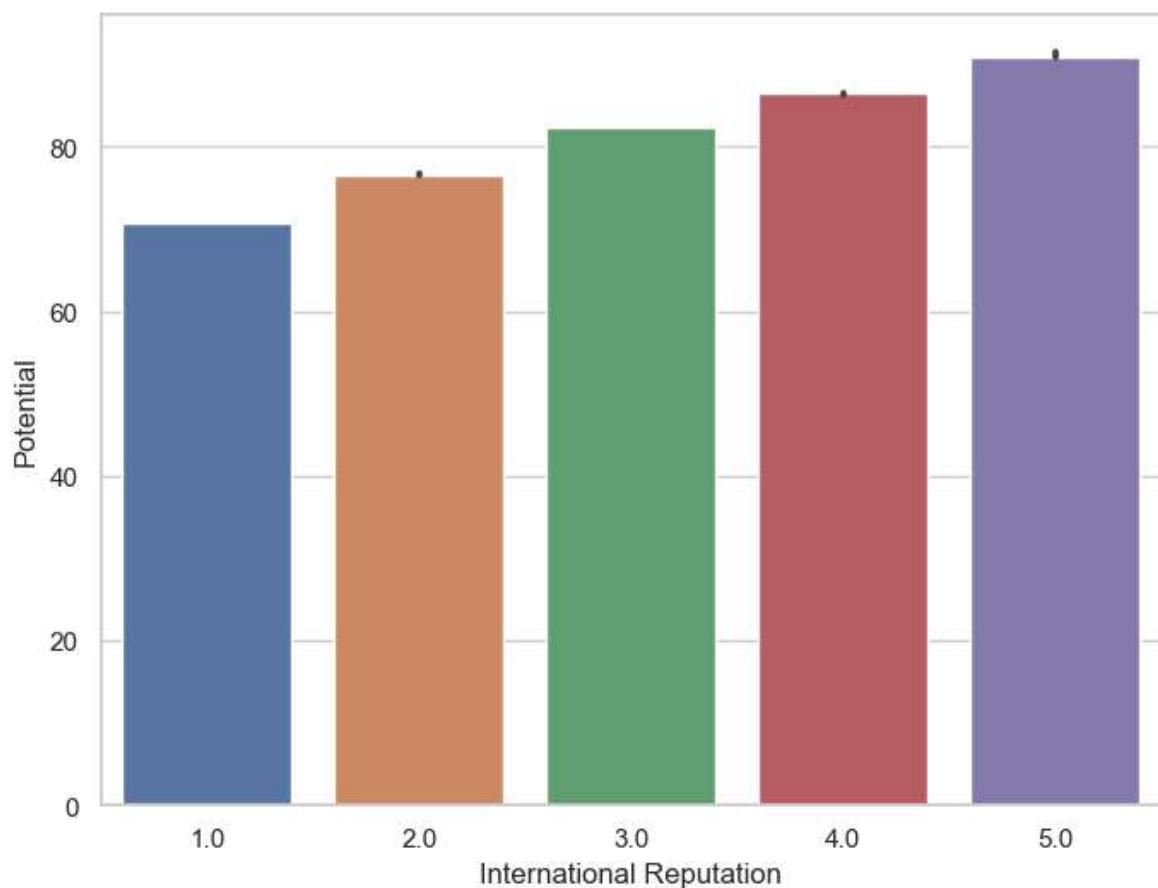
```
In [39]: f, ax = plt.subplots(figsize=(8, 6))
sns.barplot(x="International Reputation", y="Potential", hue="Preferred Foot")
plt.show()
```



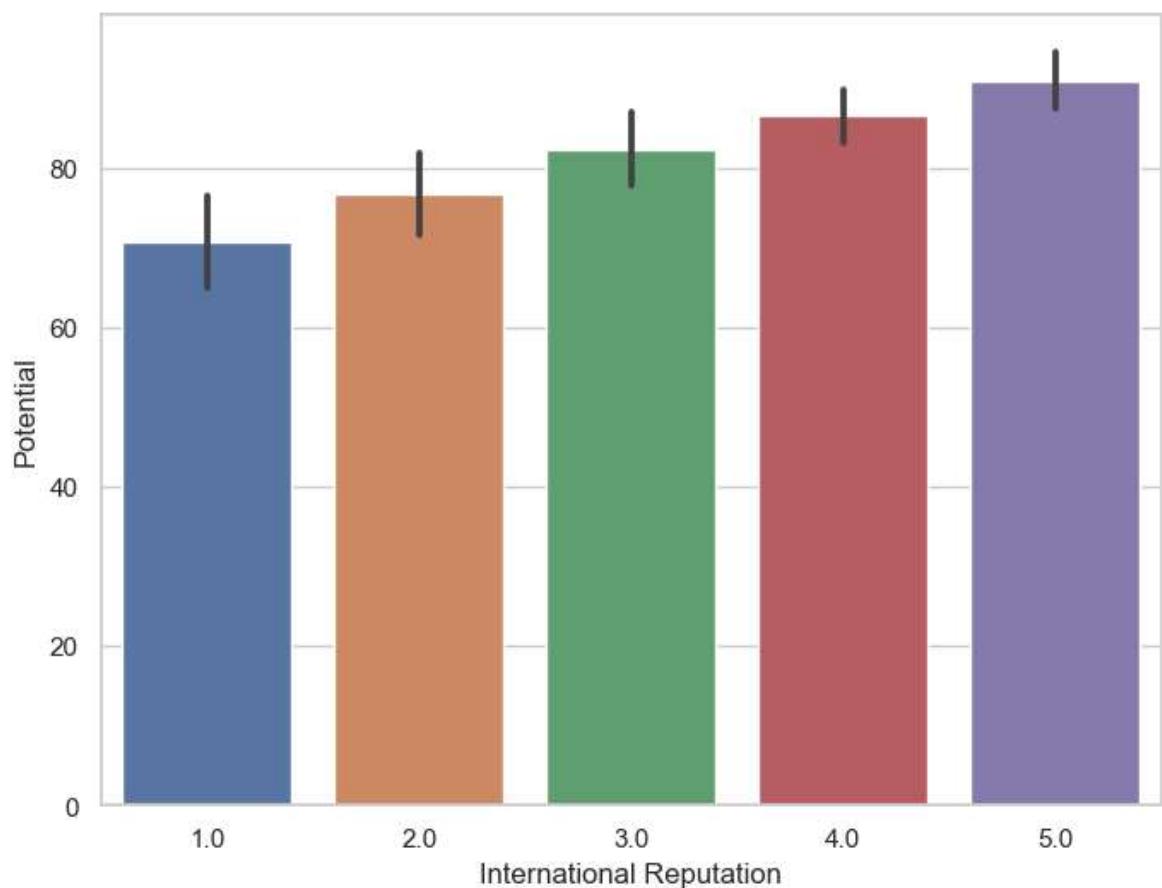
```
In [40]: from numpy import median
f, ax = plt.subplots(figsize=(8, 6))
sns.barplot(x="International Reputation", y="Potential", data=fifa19, estimate="median")
plt.show()
```



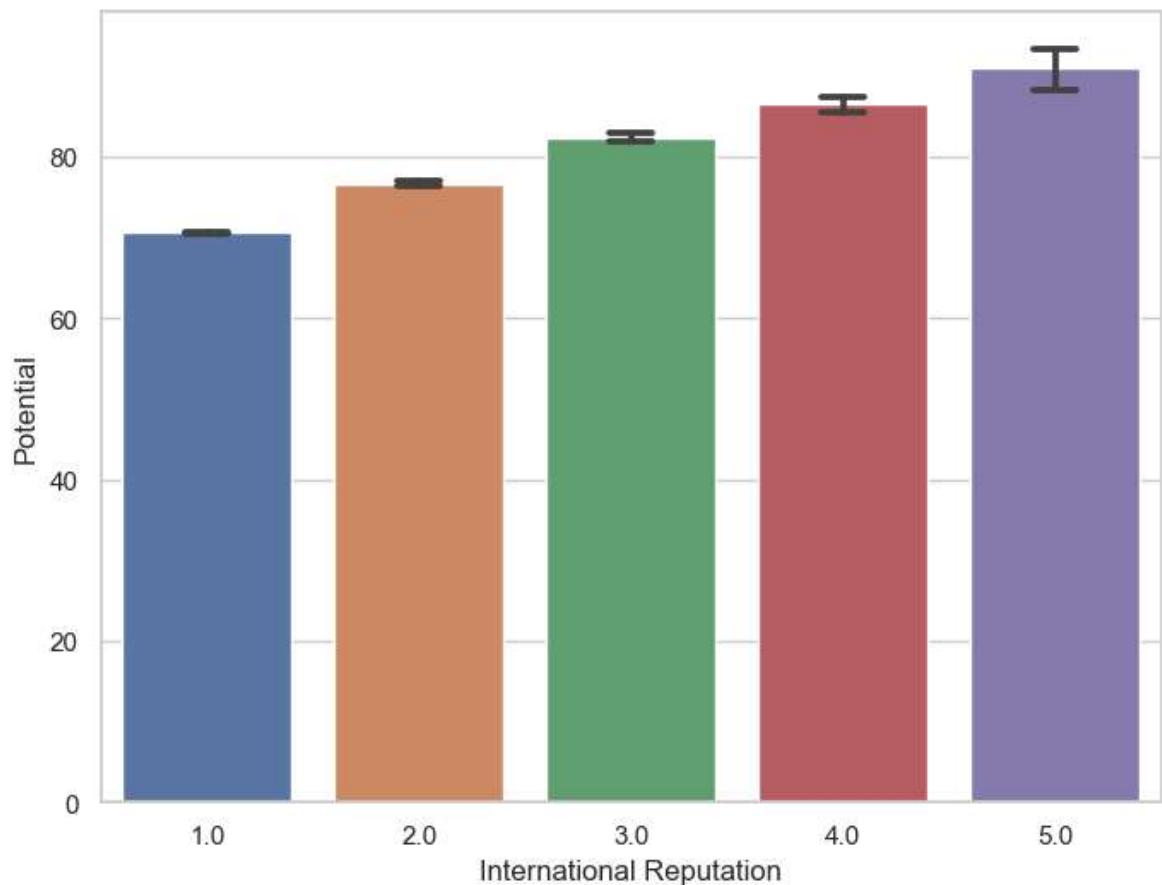
```
In [41]: f, ax = plt.subplots(figsize=(8, 6))
sns.barplot(x="International Reputation", y="Potential", data=fifa19, ci=19)
plt.show()
```



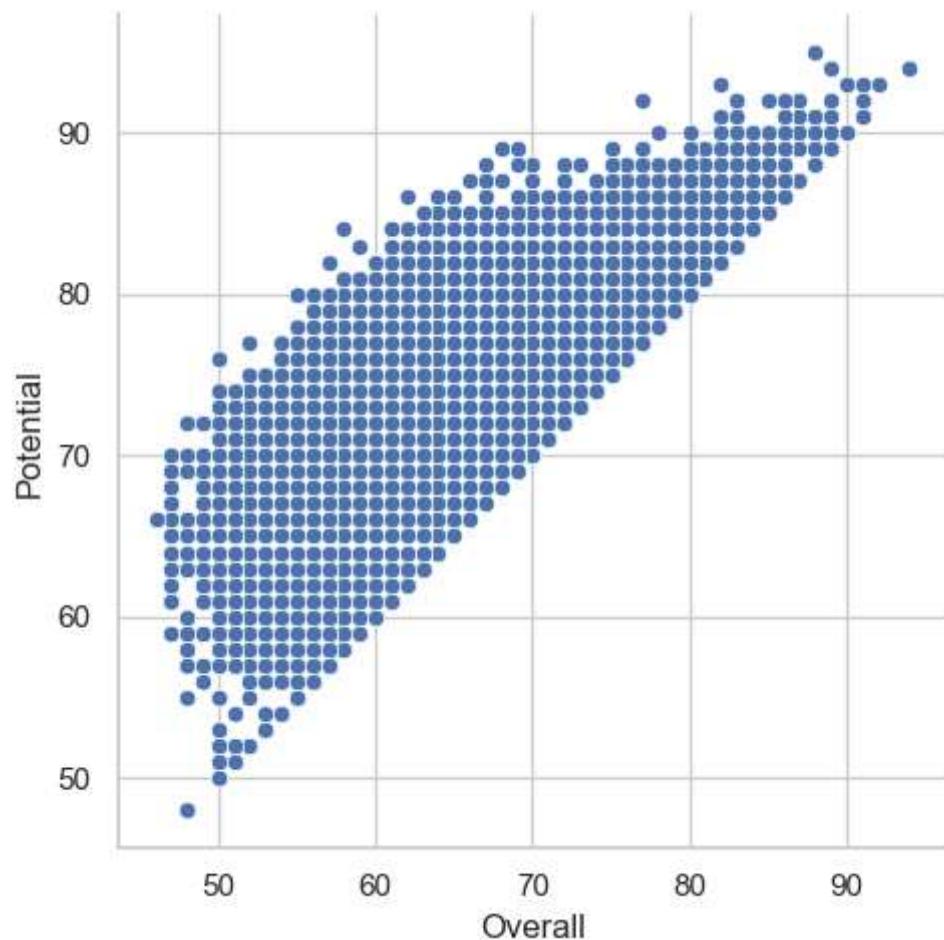
```
In [42]: f, ax = plt.subplots(figsize=(8, 6))
sns.barplot(x="International Reputation", y="Potential", data=fifa19, ci="sd"
plt.show()
```



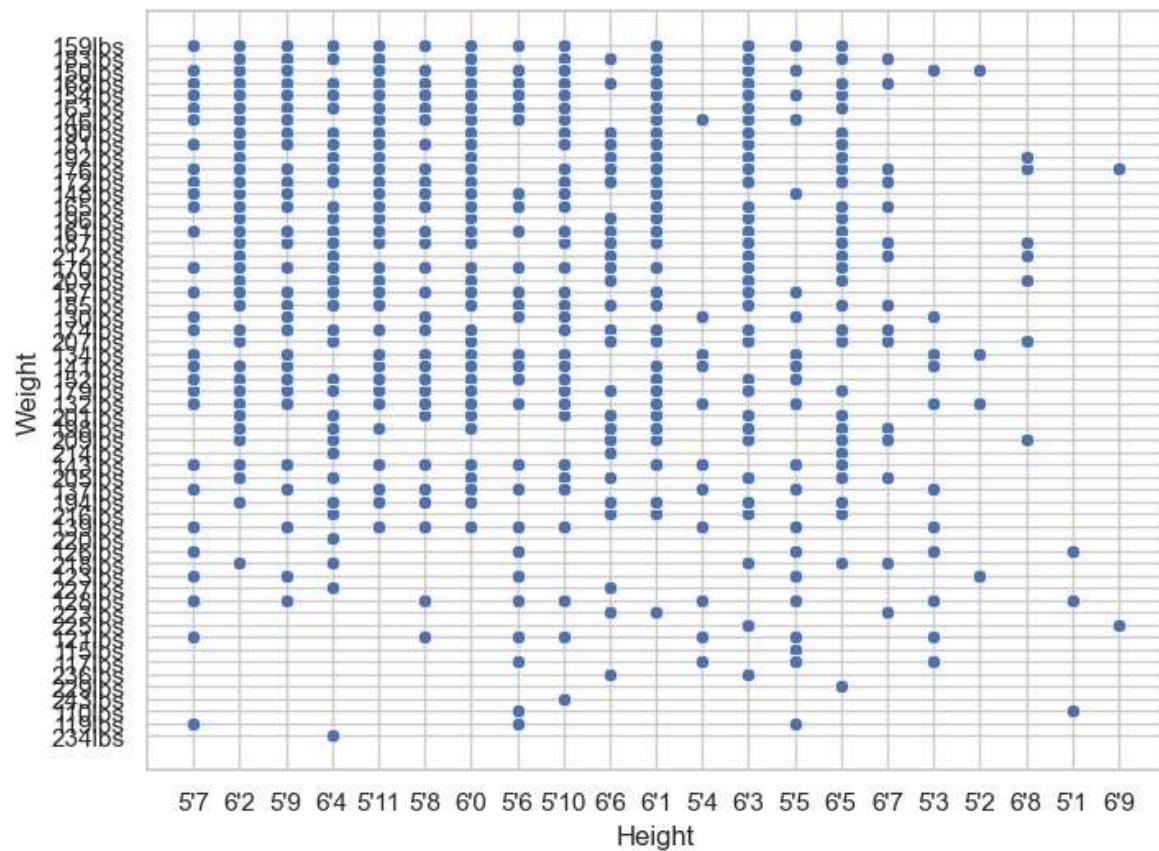
```
In [43]: f, ax = plt.subplots(figsize=(8, 6))
sns.barplot(x="International Reputation", y="Potential", data=fifa19, capsize=2)
plt.show()
```



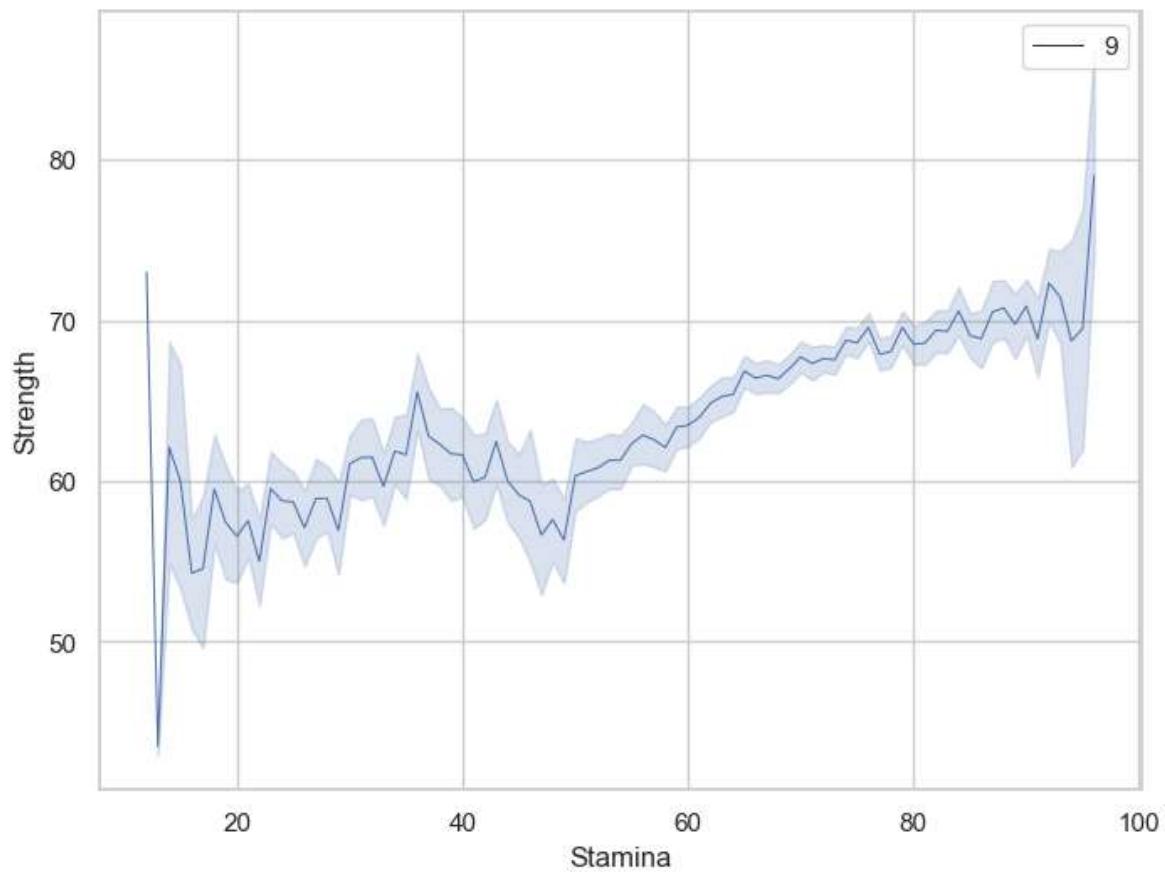
```
In [44]: g = sns.relplot(x="Overall", y="Potential", data=fifa19)
```



```
In [45]: f, ax = plt.subplots(figsize=(8, 6))
sns.scatterplot(x="Height", y="Weight", data=fifa19)
plt.show()
```

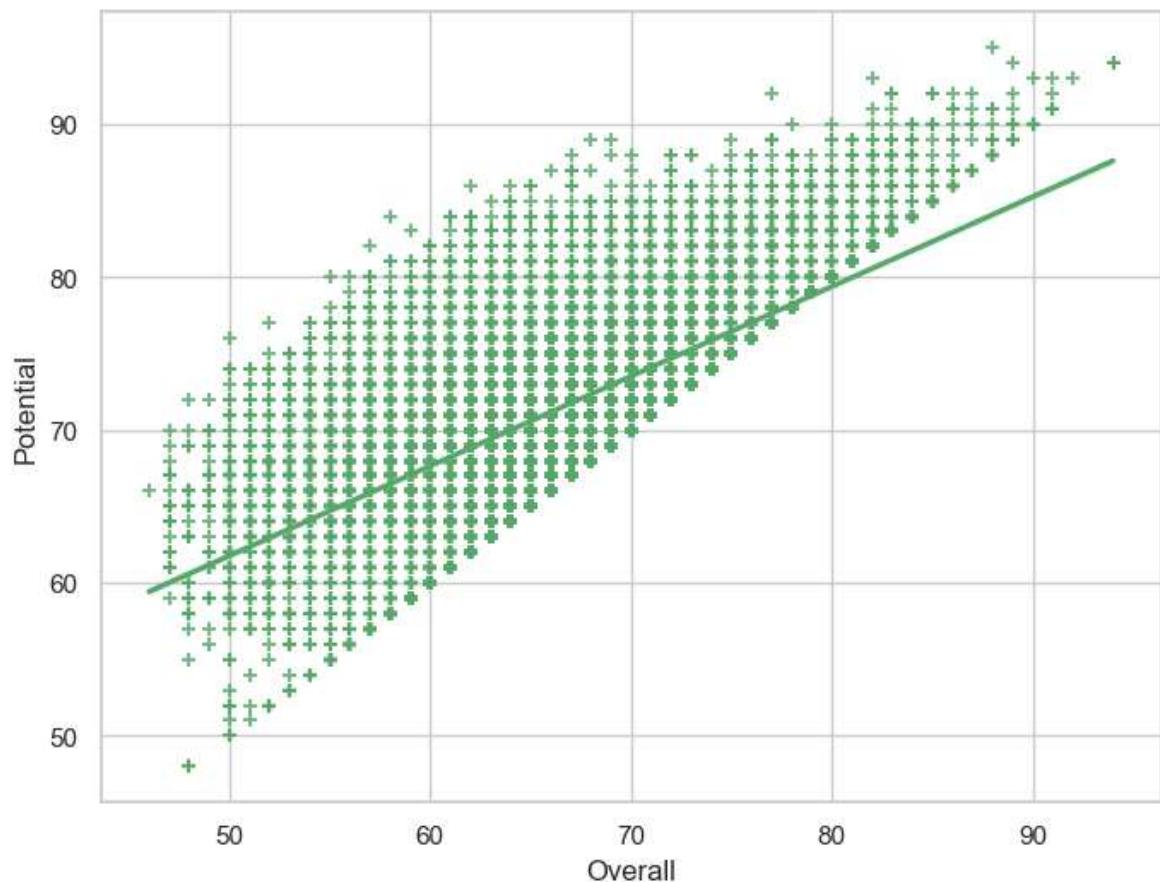


```
In [46]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.lineplot(x="Stamina", y="Strength", data=fifa19, size=9)
plt.show()
```

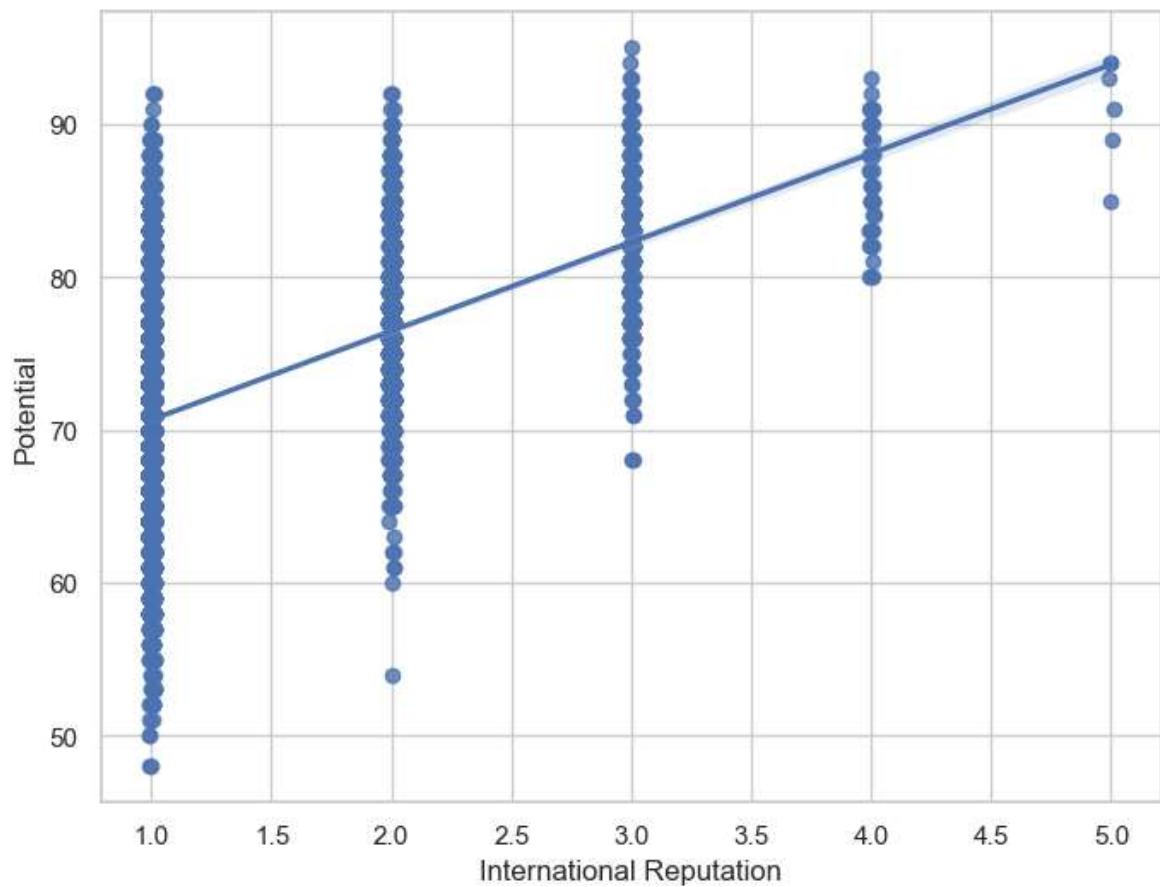


```
In [ ]:
```

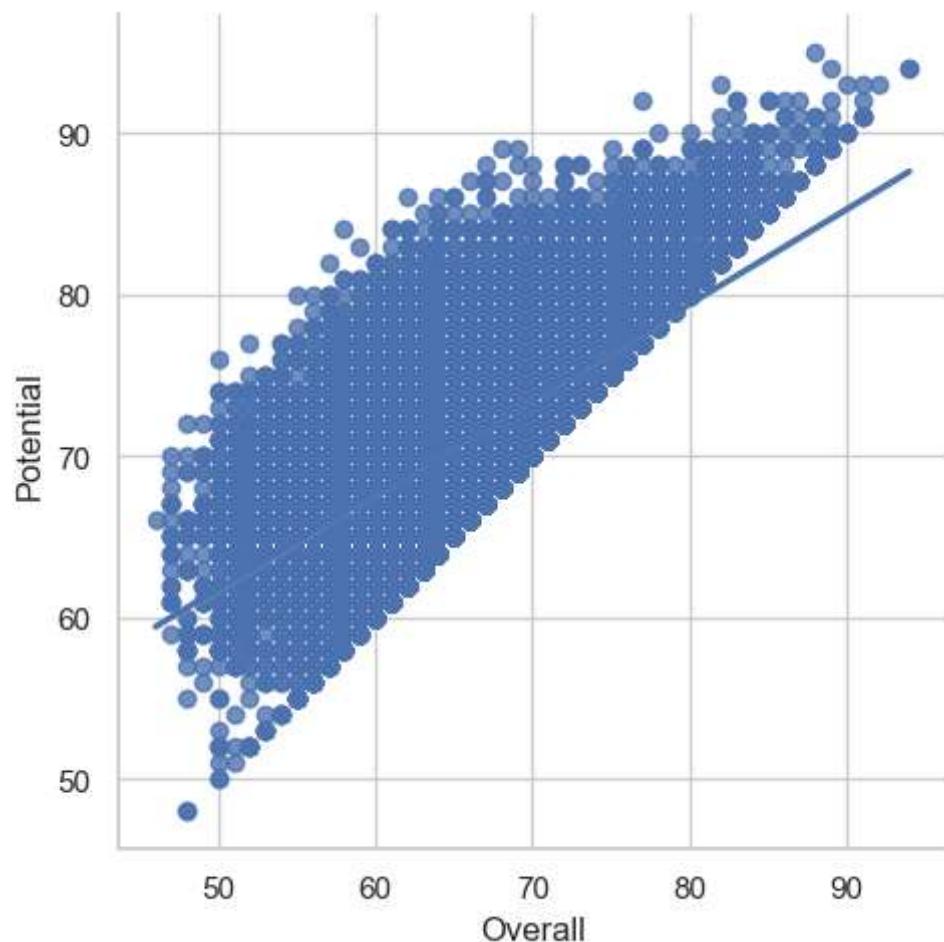
```
In [47]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.regplot(x="Overall", y="Potential", data=fifa19, color= "g", marker= "+", fit_reg=True)
```



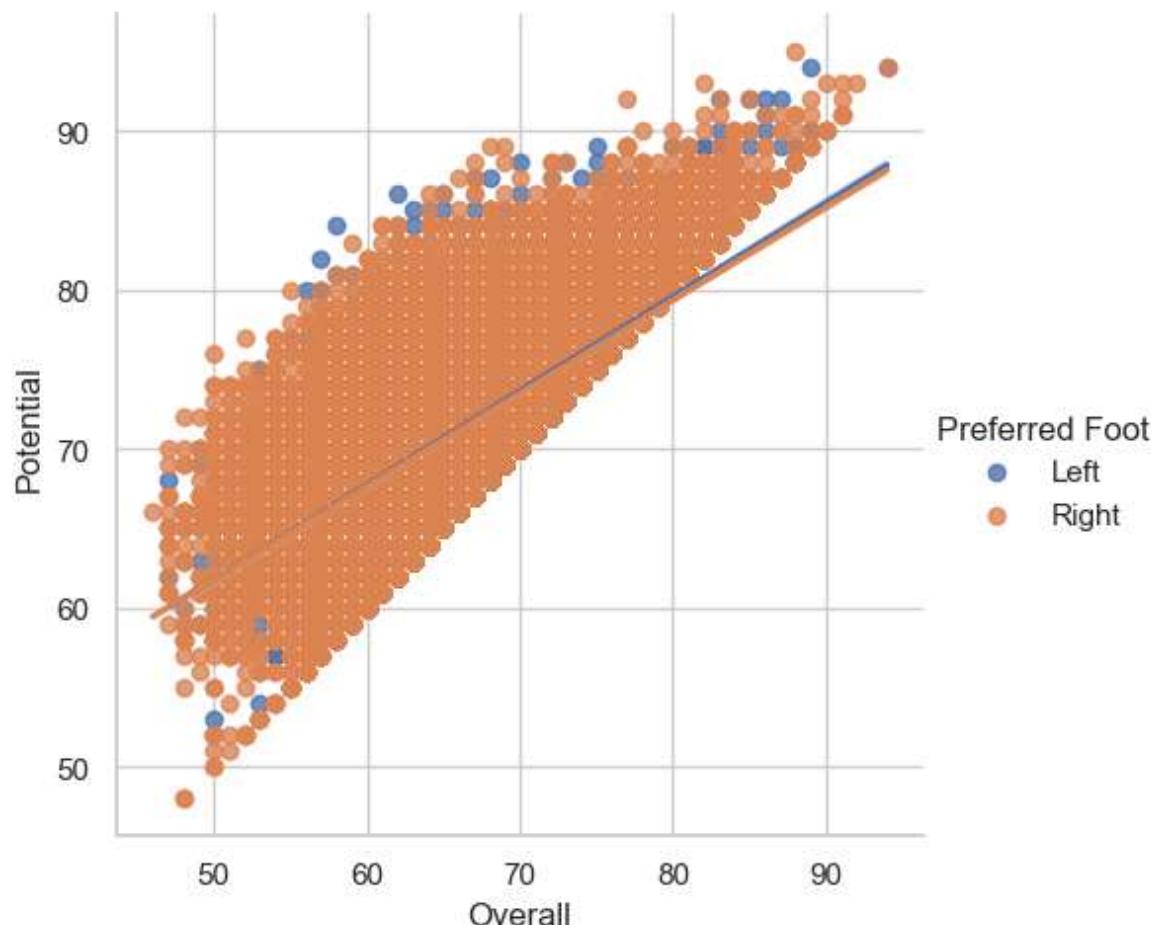
```
In [48]: f, ax = plt.subplots(figsize=(8, 6))
sns.regplot(x="International Reputation", y="Potential", data=fifa19, x_jitte
plt.show()
```



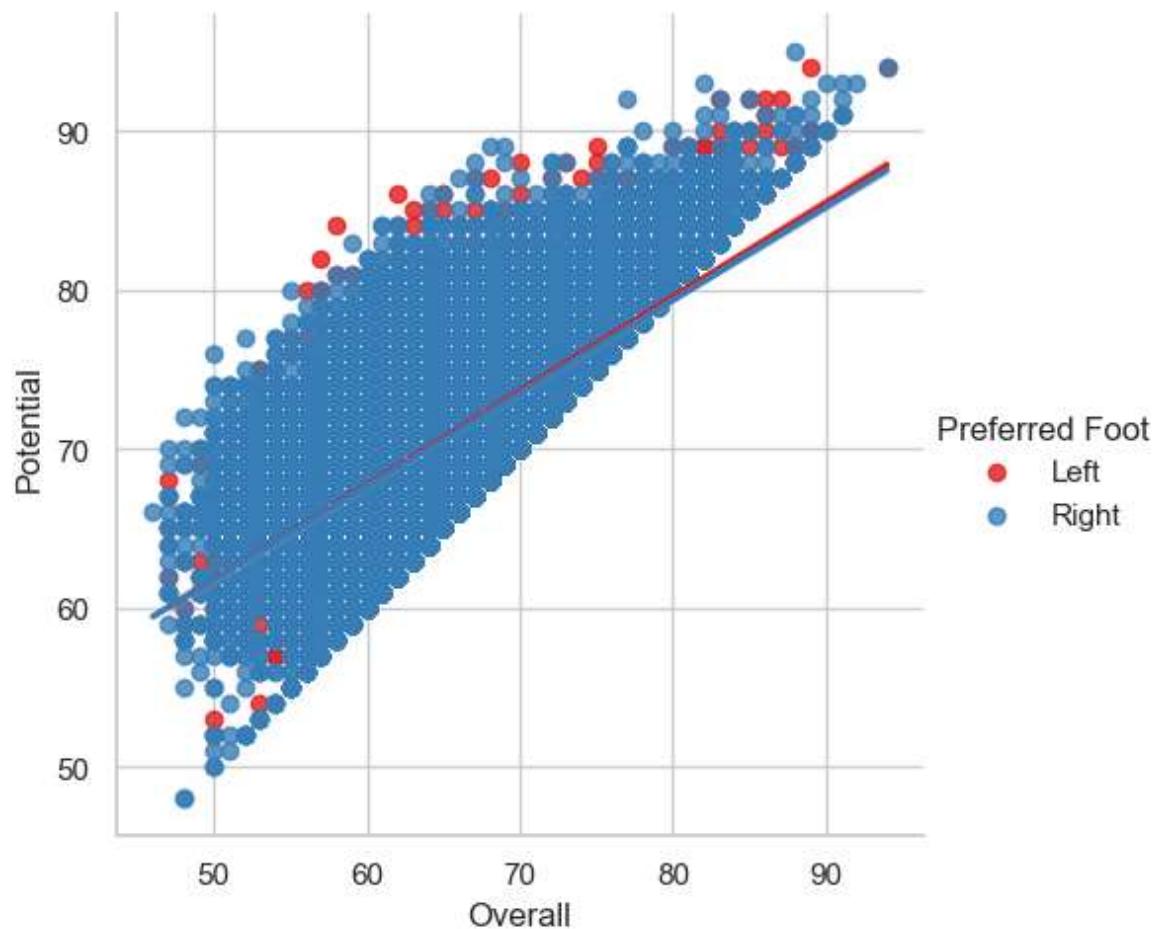
```
In [49]: g = sns.lmplot(x="Overall", y="Potential", data=fifa19)
```



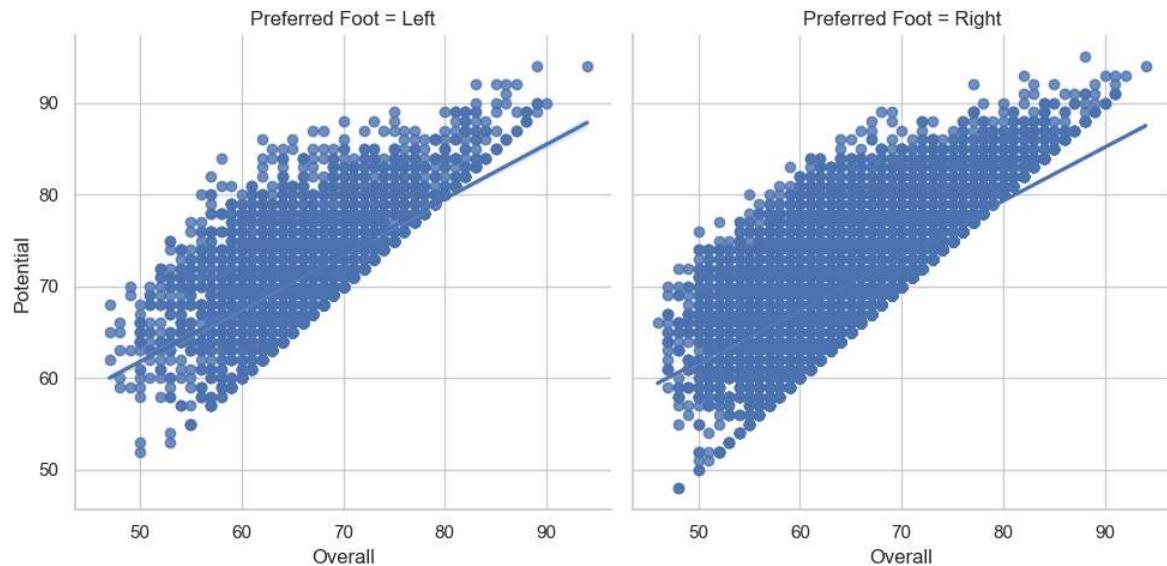
```
In [50]: g= sns.lmplot(x="Overall", y="Potential", hue="Preferred Foot", data=fifa19)
```



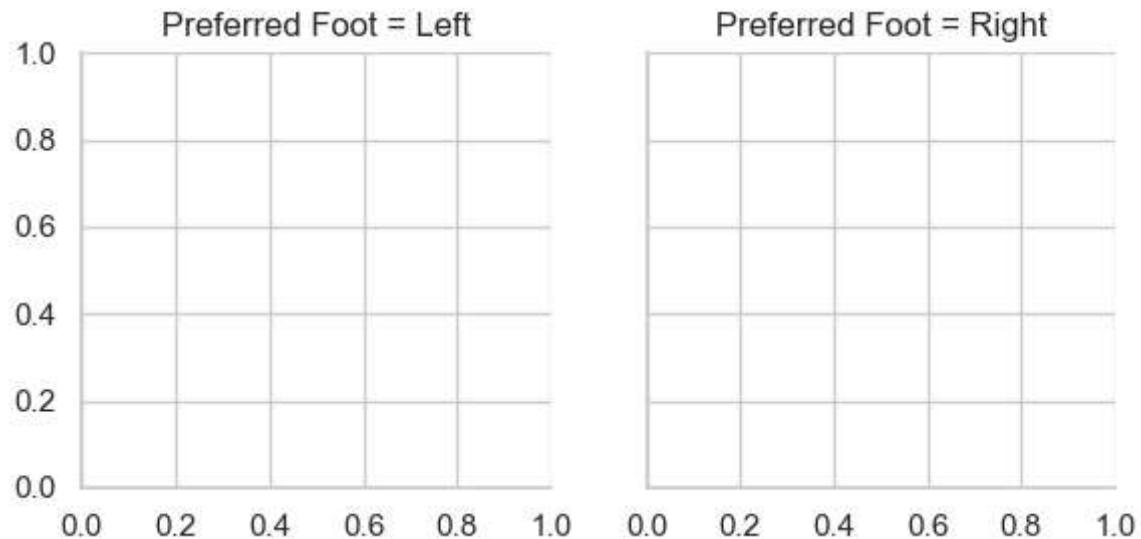
```
In [51]: g = sns.lmplot(x="Overall", y="Potential", hue="Preferred Foot", data=fifa19,
```



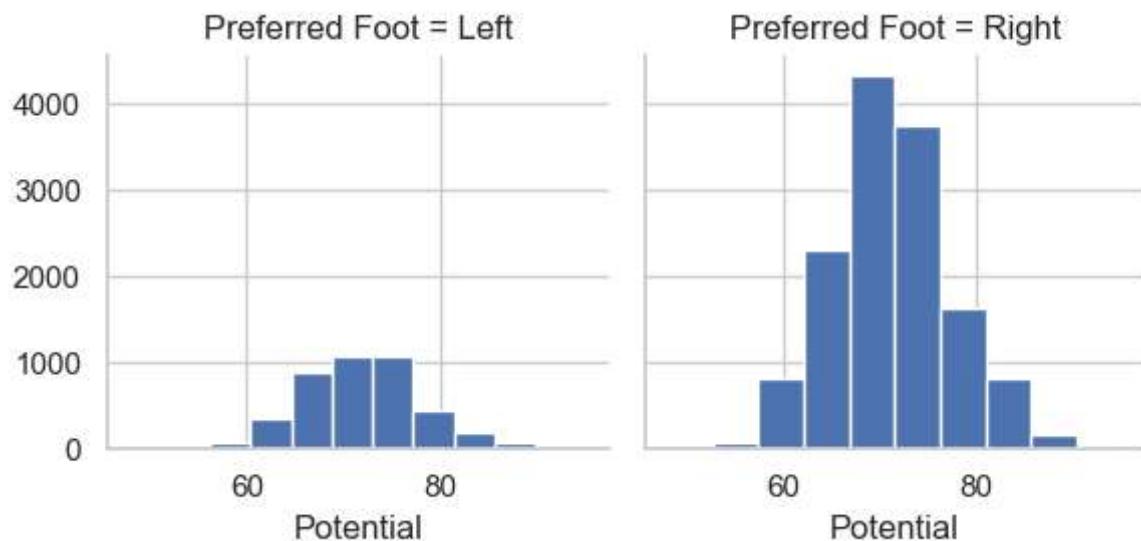
```
In [52]: g = sns.lmplot(x="Overall", y="Potential", col="Preferred Foot", data=fifa19)
```



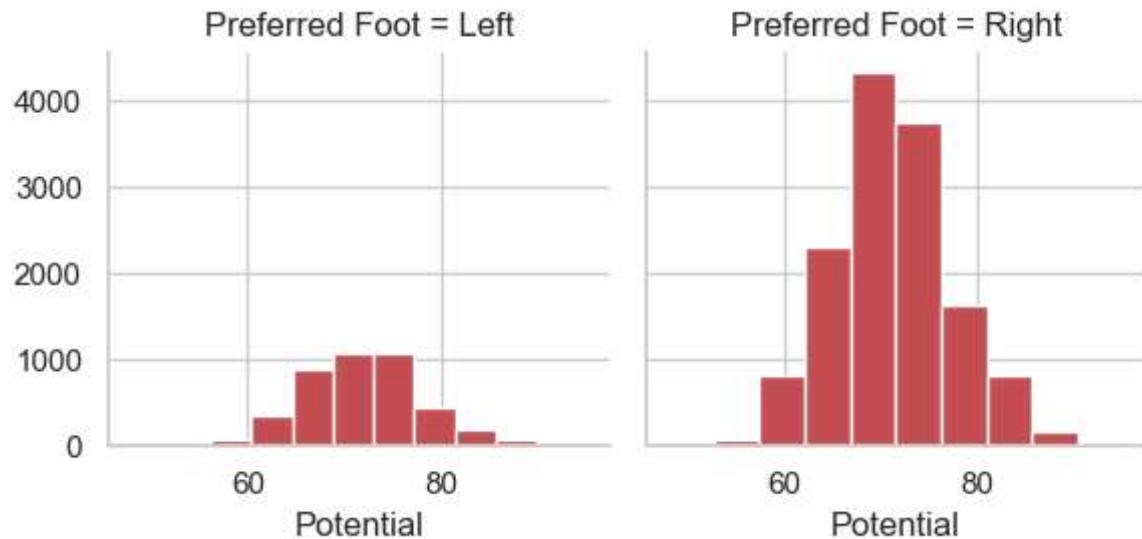
```
In [53]: g = sns.FacetGrid(fifa19, col="Preferred Foot")
```



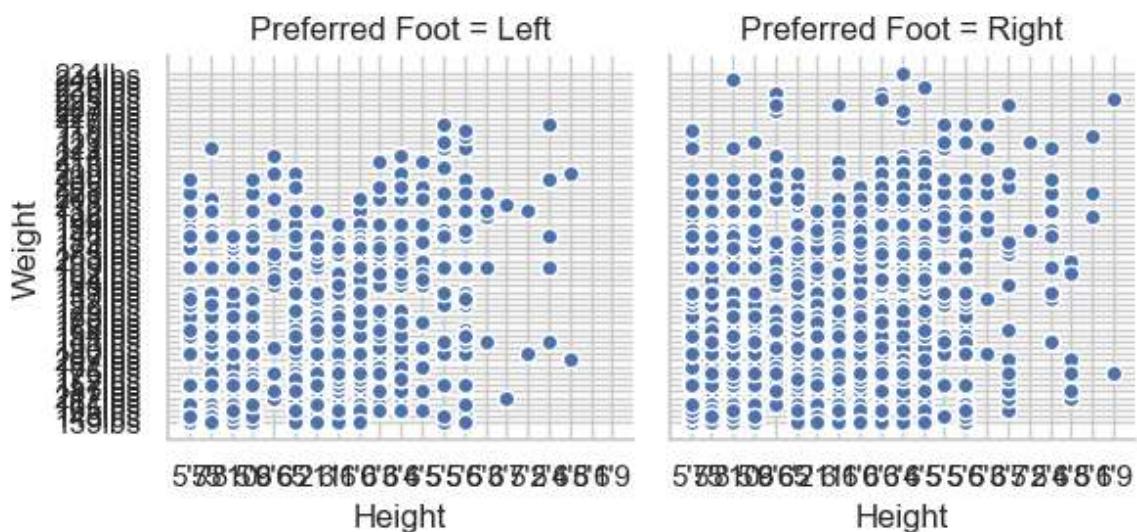
```
In [54]: g=sns.FacetGrid(fifa19,col="Preferred Foot")
g = g.map(plt.hist,"Potential")
```



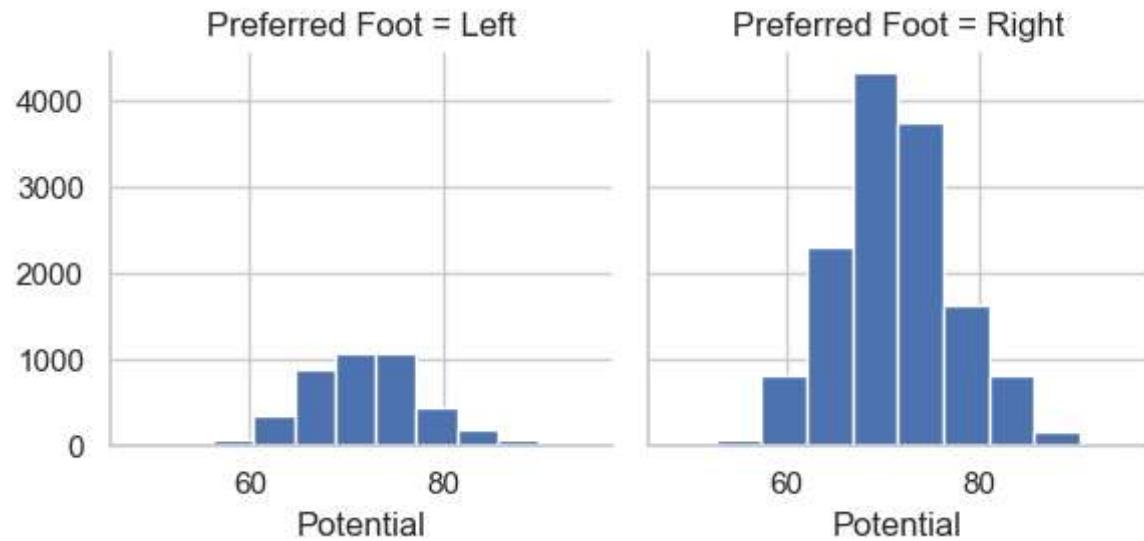
```
In [55]: g = sns.FacetGrid(fifa19, col="Preferred Foot")
          g = g.map(plt.hist, "Potential", bins=10, color="r")
```



```
In [56]: g = sns.FacetGrid(fifa19, col="Preferred Foot")
g=g.map(plt.scatter,'Height','Weight',edgecolor="w").add_legend()
```

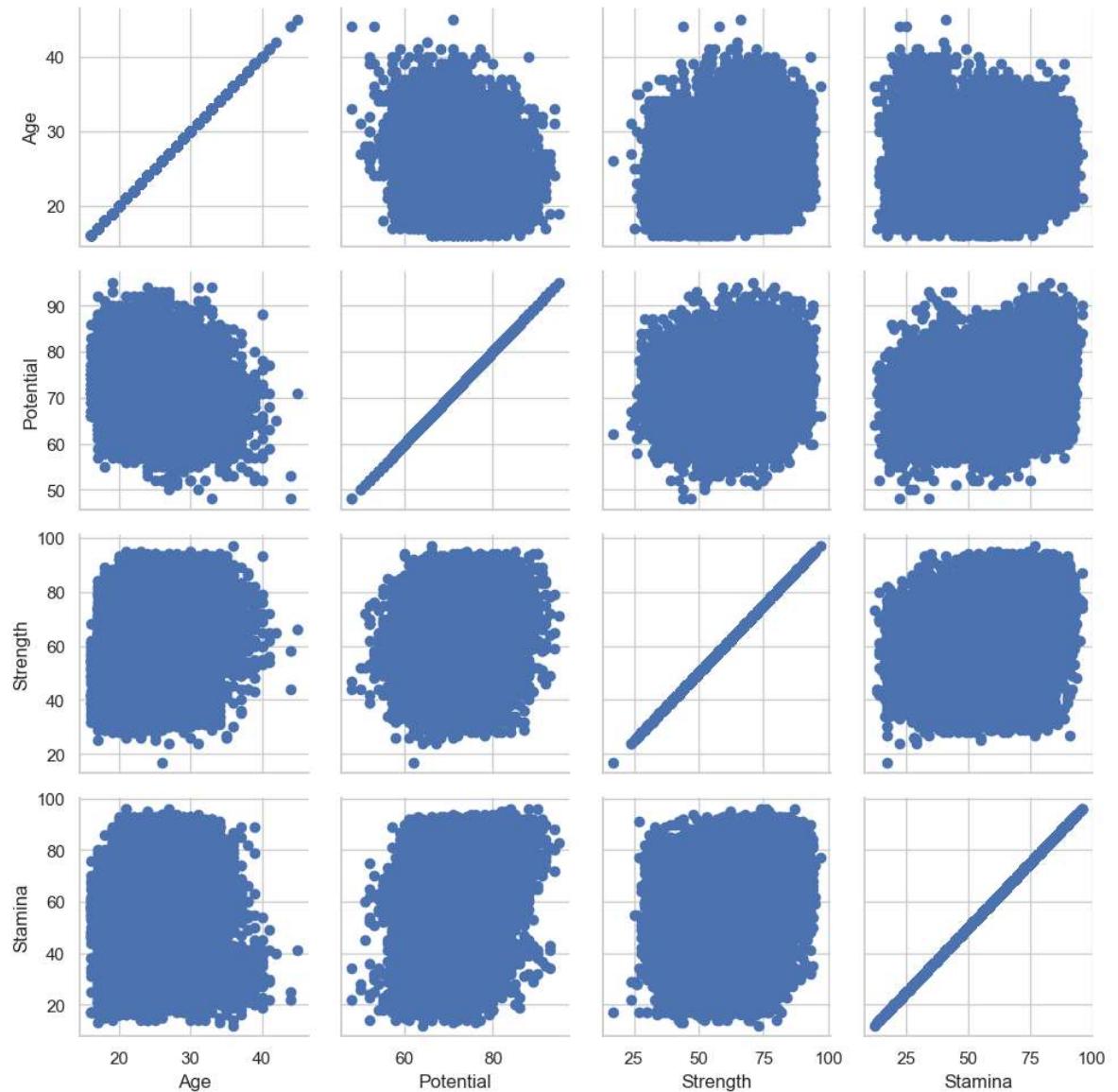


```
In [57]: g = sns.FacetGrid(fifa19, col="Preferred Foot", height=3, aspect=1)
g = g.map(plt.hist, "Potential")
```

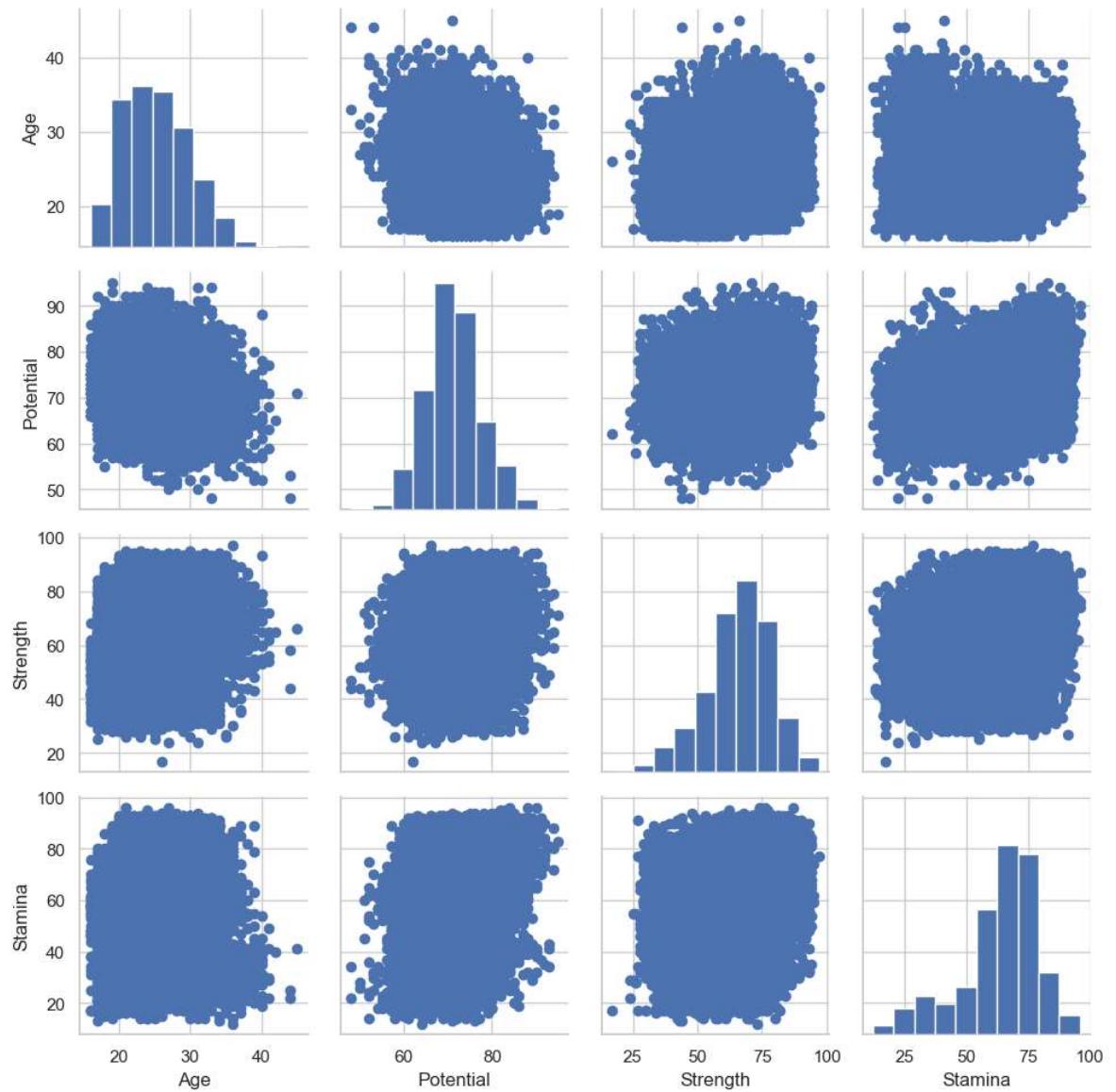


```
In [58]: fifa19_new = fifa19[['Age', 'Potential', 'Strength', 'Stamina', 'Preferred Fo
```

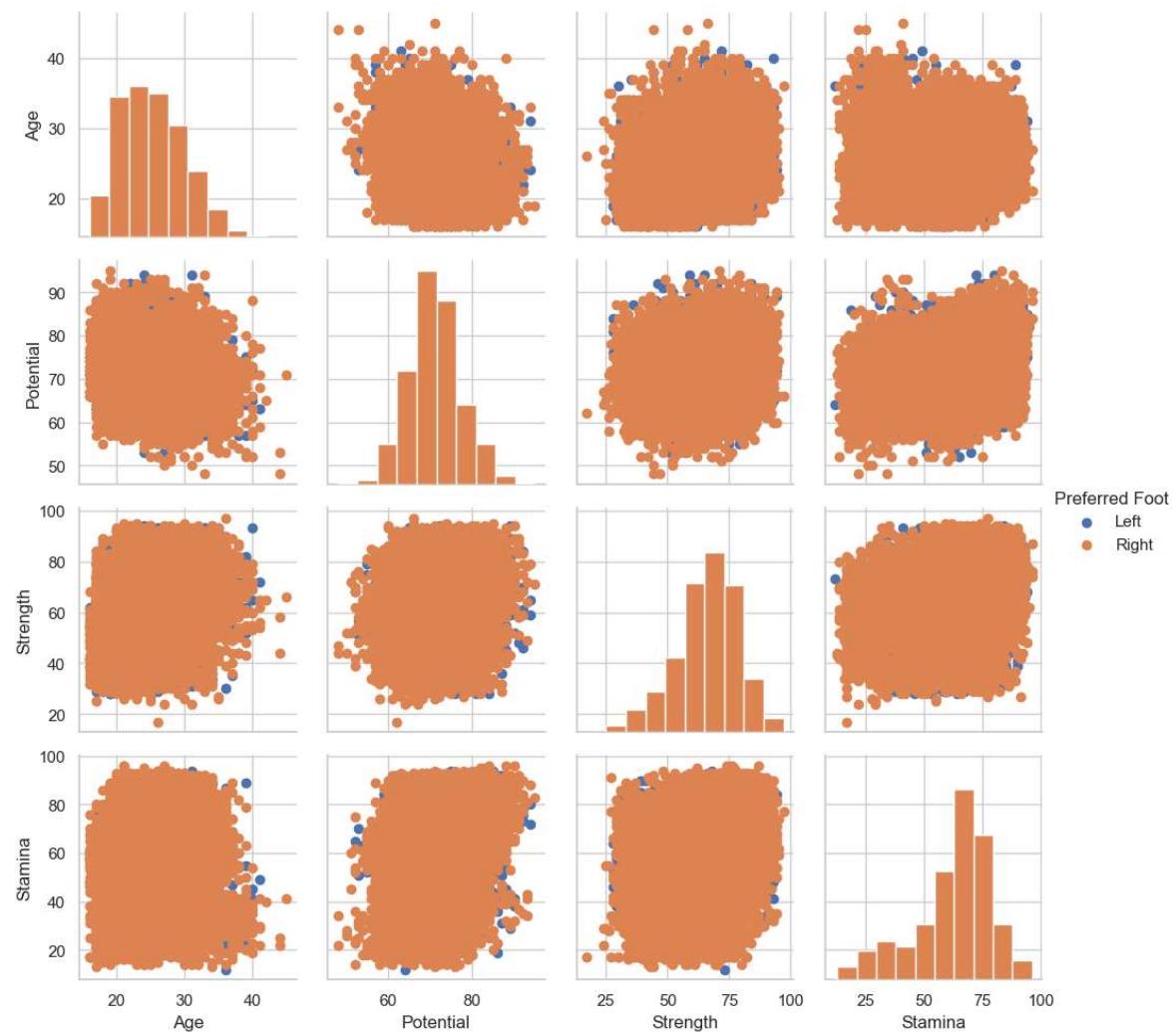
```
In [59]: g = sns.PairGrid(fifa19_new)
g = g.map(plt.scatter)
```



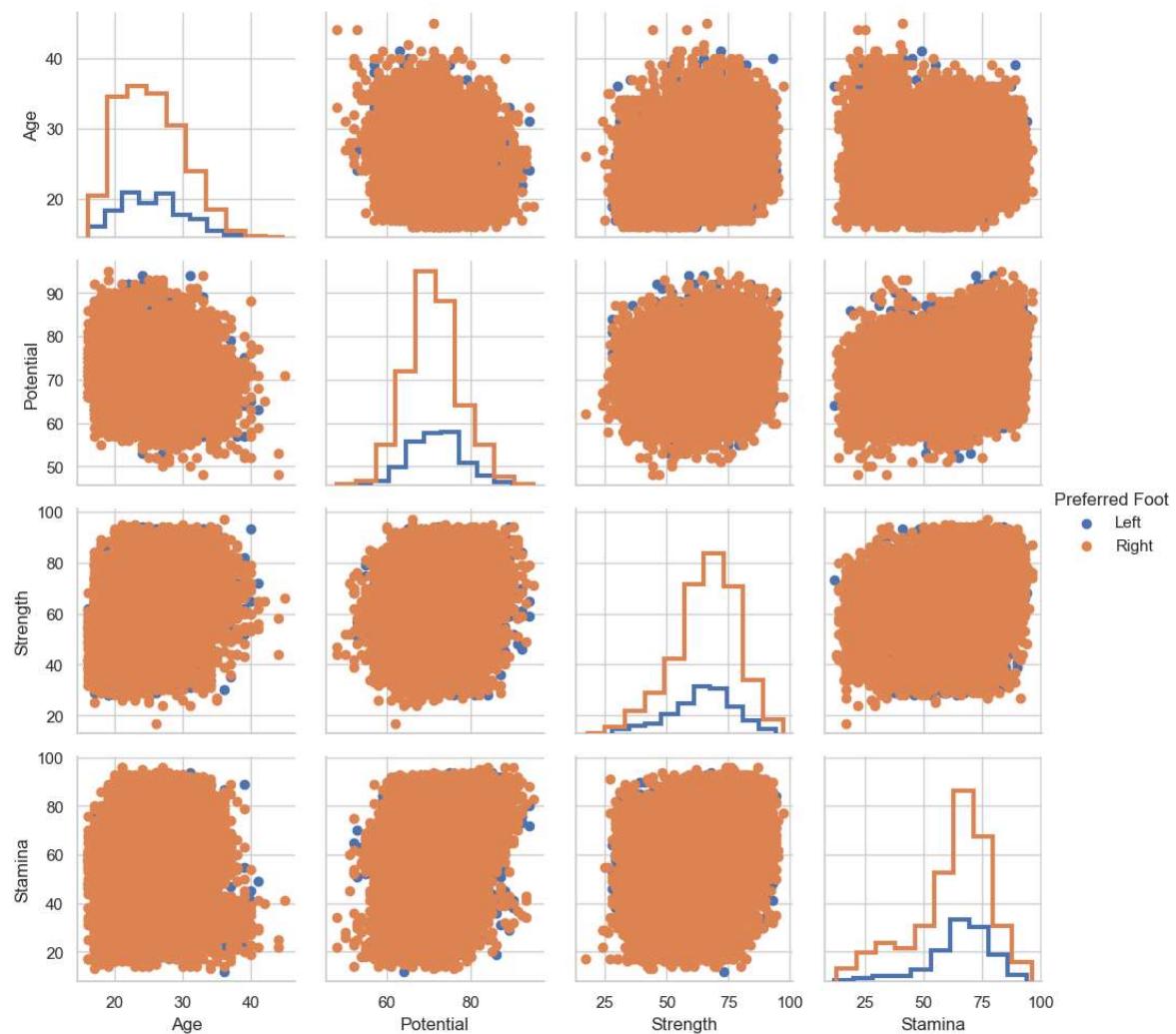
```
In [60]: g = sns.PairGrid(fifa19_new)
g = g.map_diag(plt.hist)
g = g.map_offdiag(plt.scatter)
```



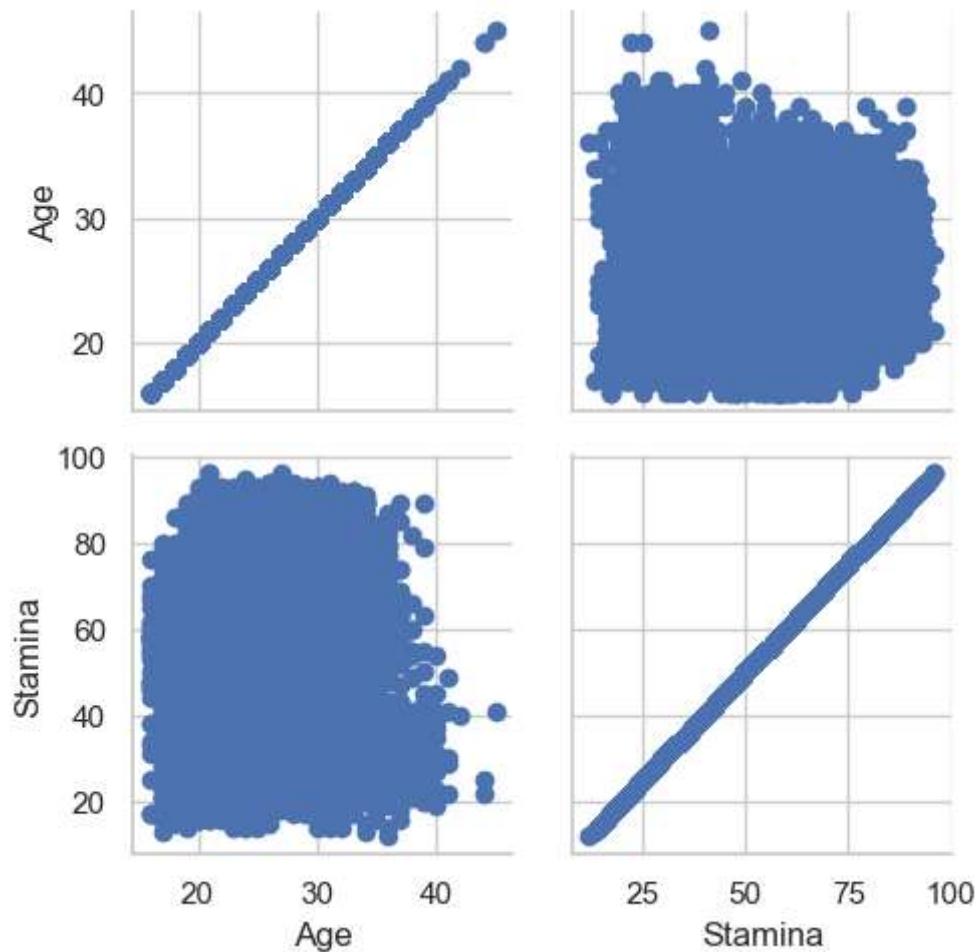
```
In [61]: g = sns.PairGrid(fifa19_new, hue="Preferred Foot")
g = g.map_diag(plt.hist)
g = g.map_offdiag(plt.scatter)
g = g.add_legend()
```



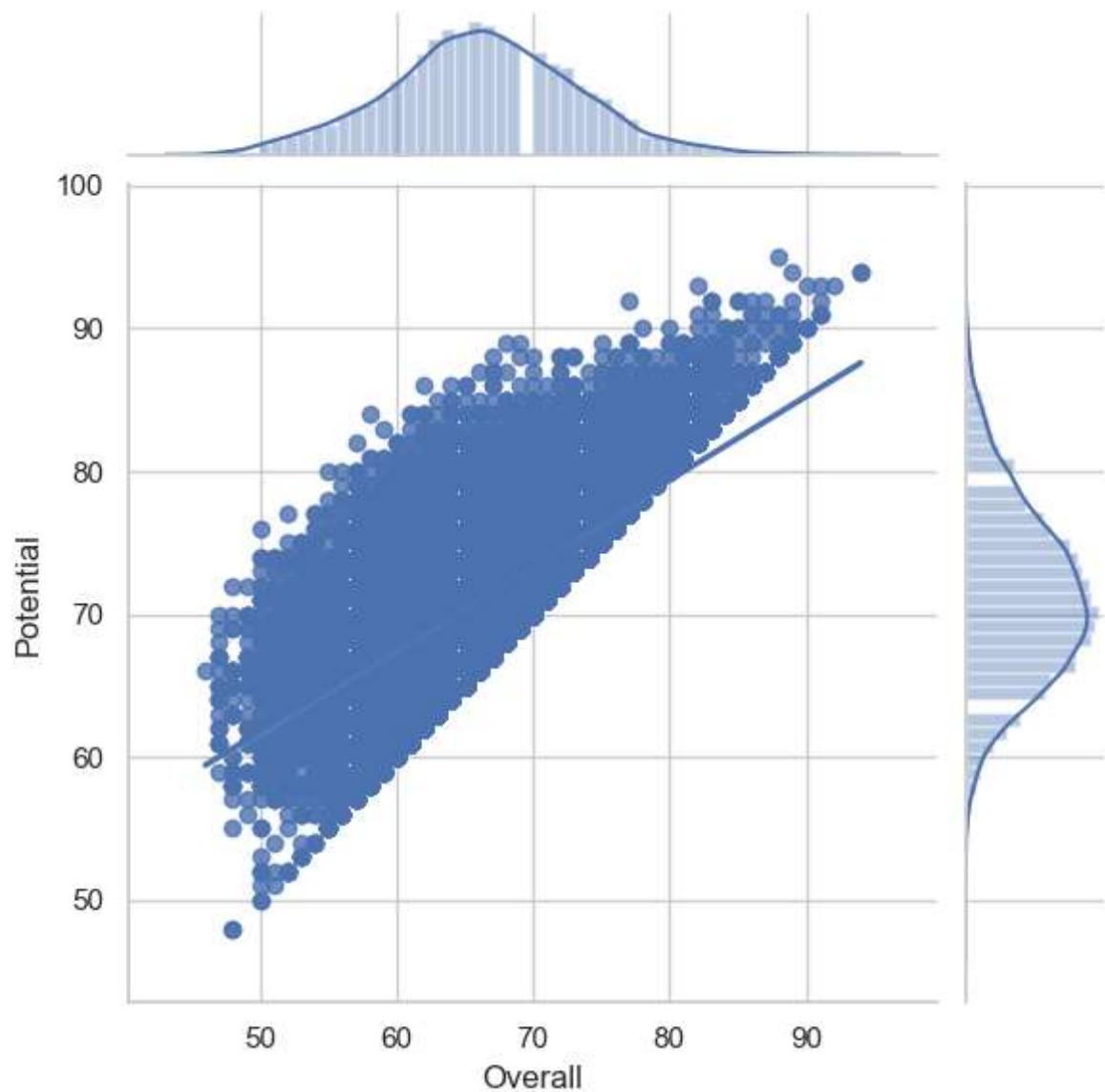
```
In [62]: g = sns.PairGrid(fifa19_new, hue="Preferred Foot")
g = g.map_diag(plt.hist, histtype="step", linewidth=3)
g = g.map_offdiag(plt.scatter)
g = g.add_legend()
```



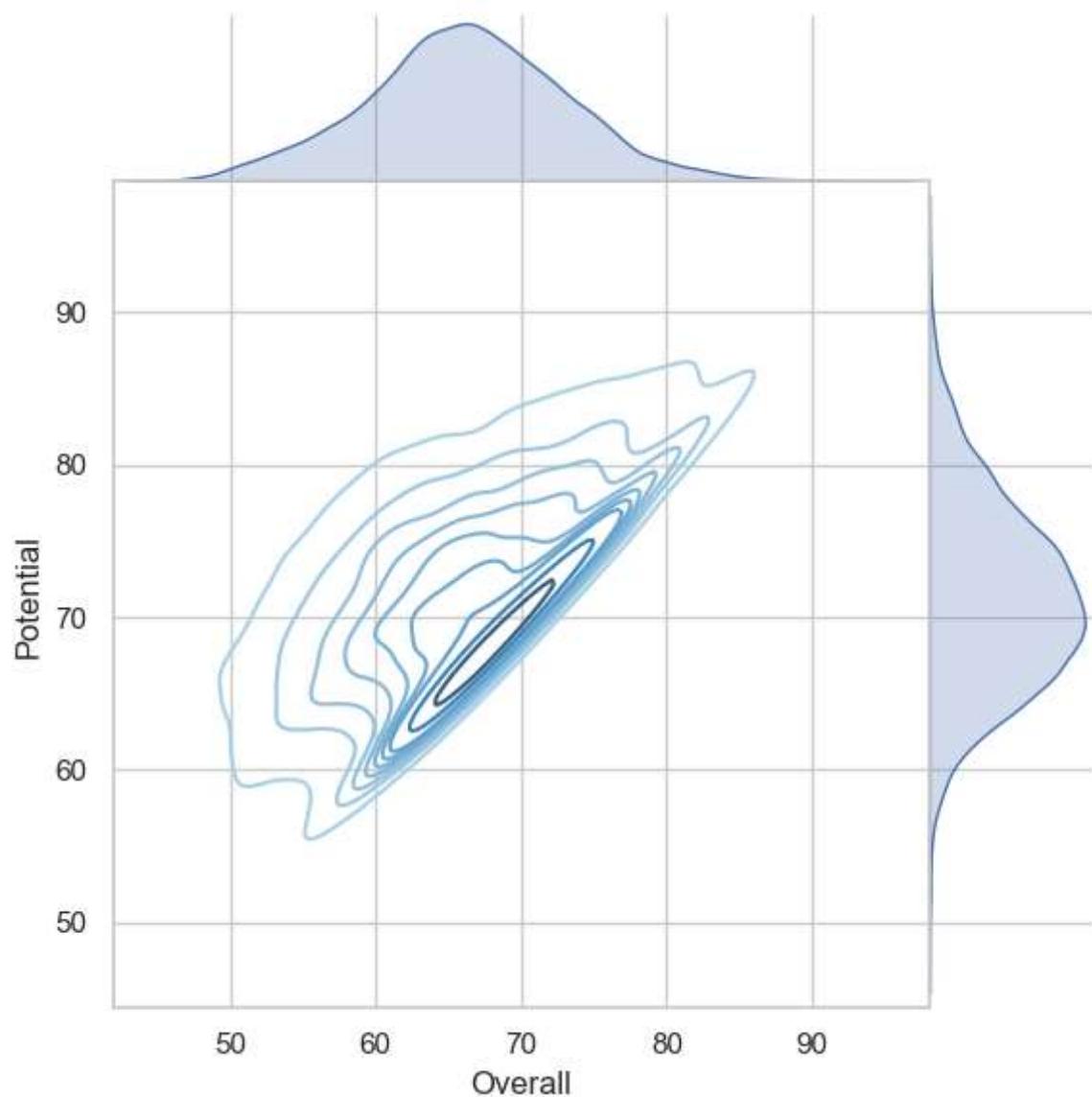
```
In [63]: g = sns.PairGrid(fifa19_new, vars=['Age', 'Stamina'])
g = g.map(plt.scatter)
```



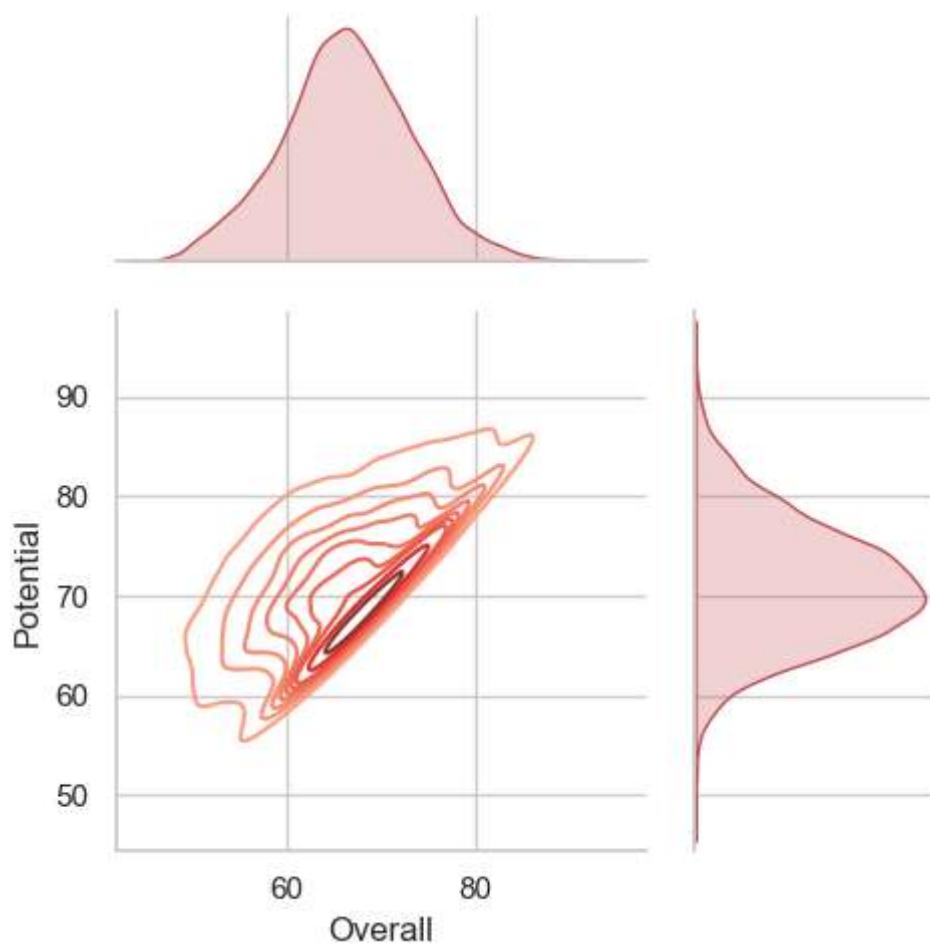
```
In [69]: g=sns.JointGrid(x="Overall", y="Potential",data=fifa19)
g=g.plot(sns.regplot,sns.distplot)
```



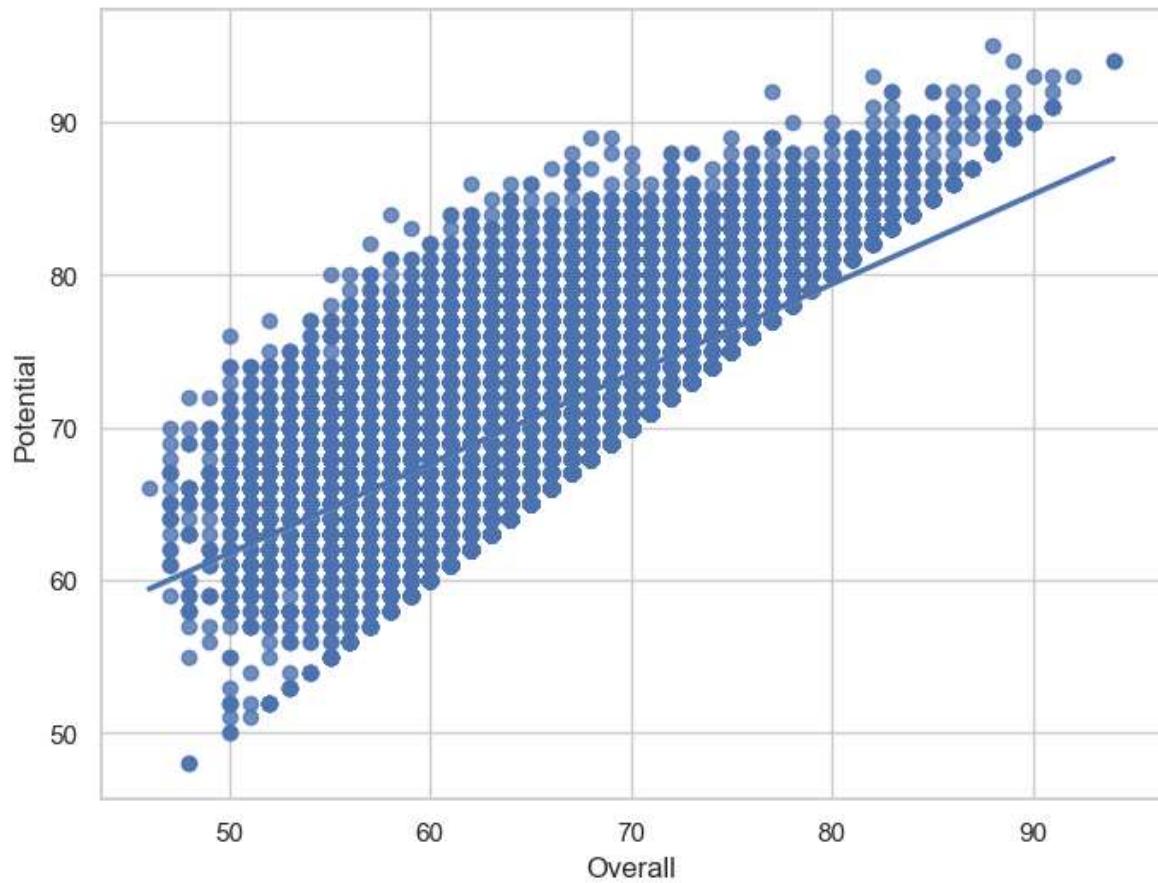
```
In [70]: g = sns.JointGrid(x="Overall", y="Potential", data=fifa19, space=0)
g = g.plot_joint(sns.kdeplot, cmap="Blues_d")
g = g.plot_marginals(sns.kdeplot, shade=True)
```



```
In [71]: g = sns.JointGrid(x="Overall", y="Potential", data=fifa19, height=5, ratio=2)
g = g.plot_joint(sns.kdeplot, cmap="Reds_d")
g = g.plot_marginals(sns.kdeplot, color="r", shade=True)
```

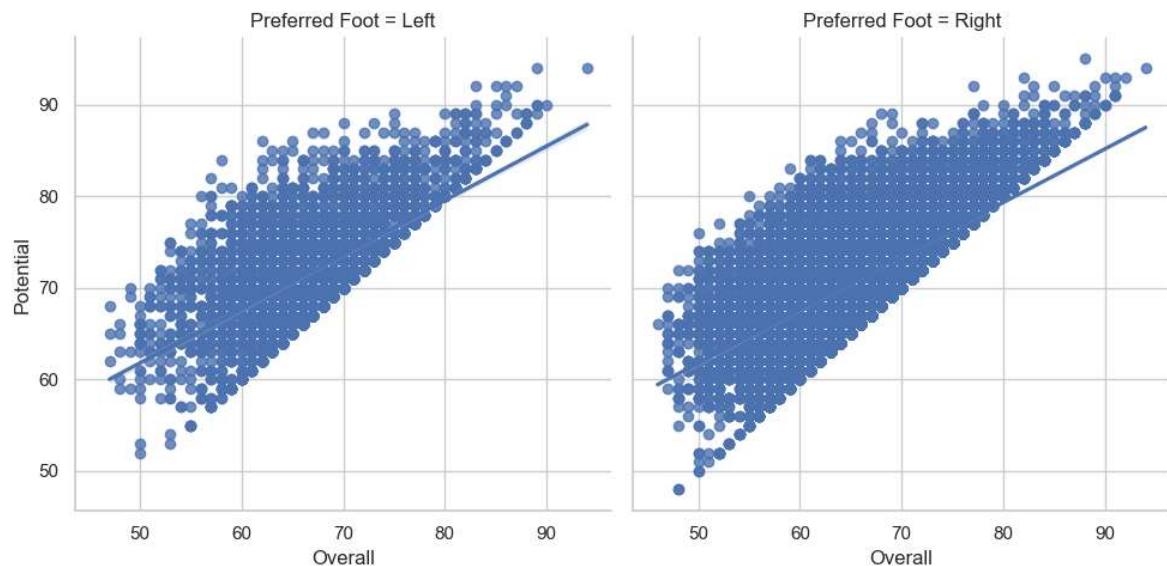


```
In [72]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.regplot(x="Overall", y="Potential", data=fifa19);
```



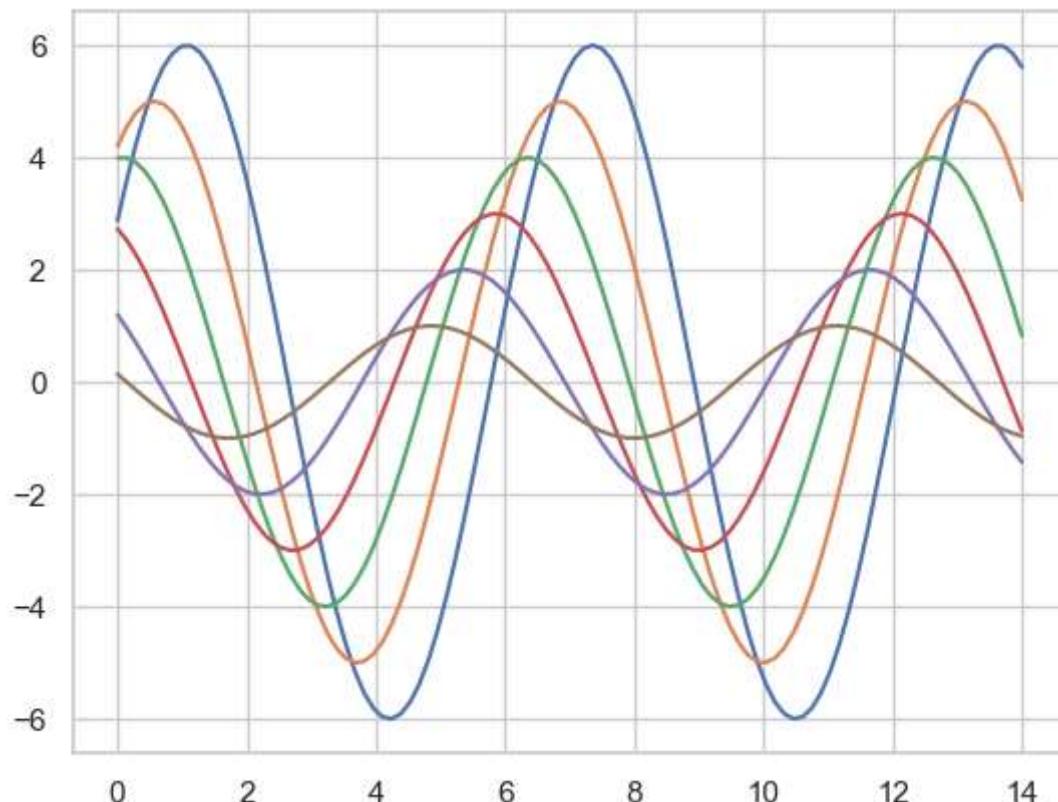
```
In [73]: sns.lmplot(x="Overall", y="Potential", col="Preferred Foot", data=fifa19, col
```

```
Out[73]: <seaborn.axisgrid.FacetGrid at 0x232b8e77c70>
```

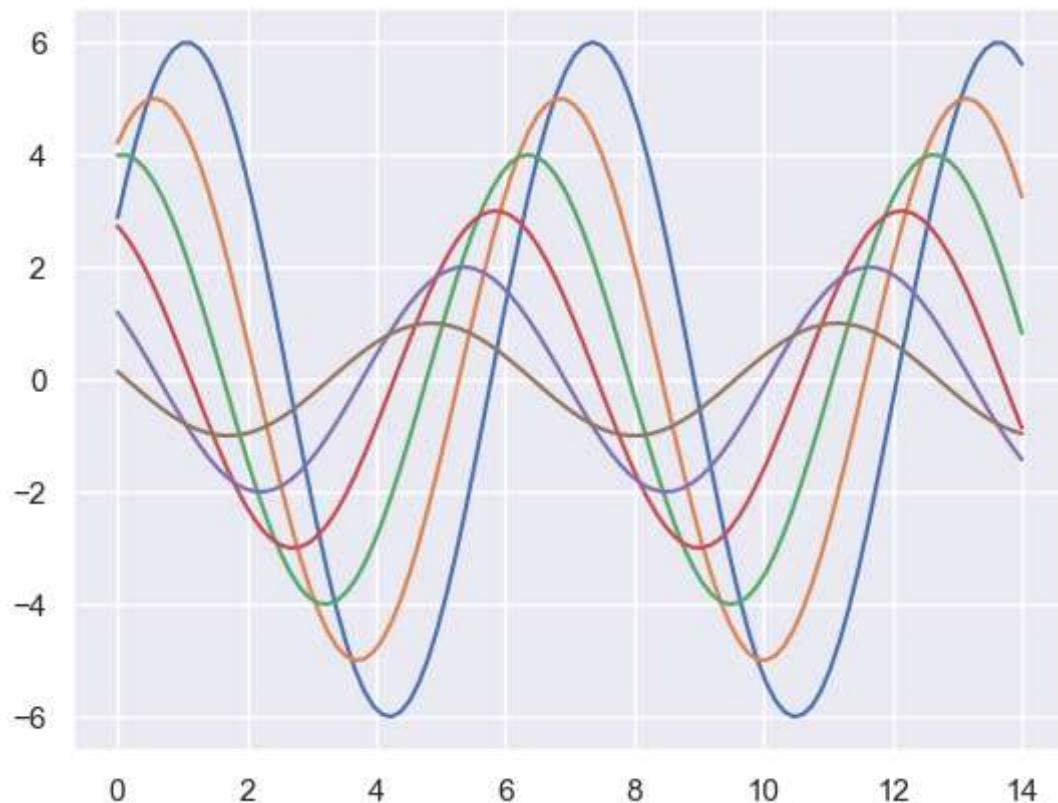


```
In [74]: def sinplot(flip=1):
    x = np.linspace(0, 14, 100)
    for i in range(1, 7):
        plt.plot(x, np.sin(x + i * .5) * (7 - i) * flip)
```

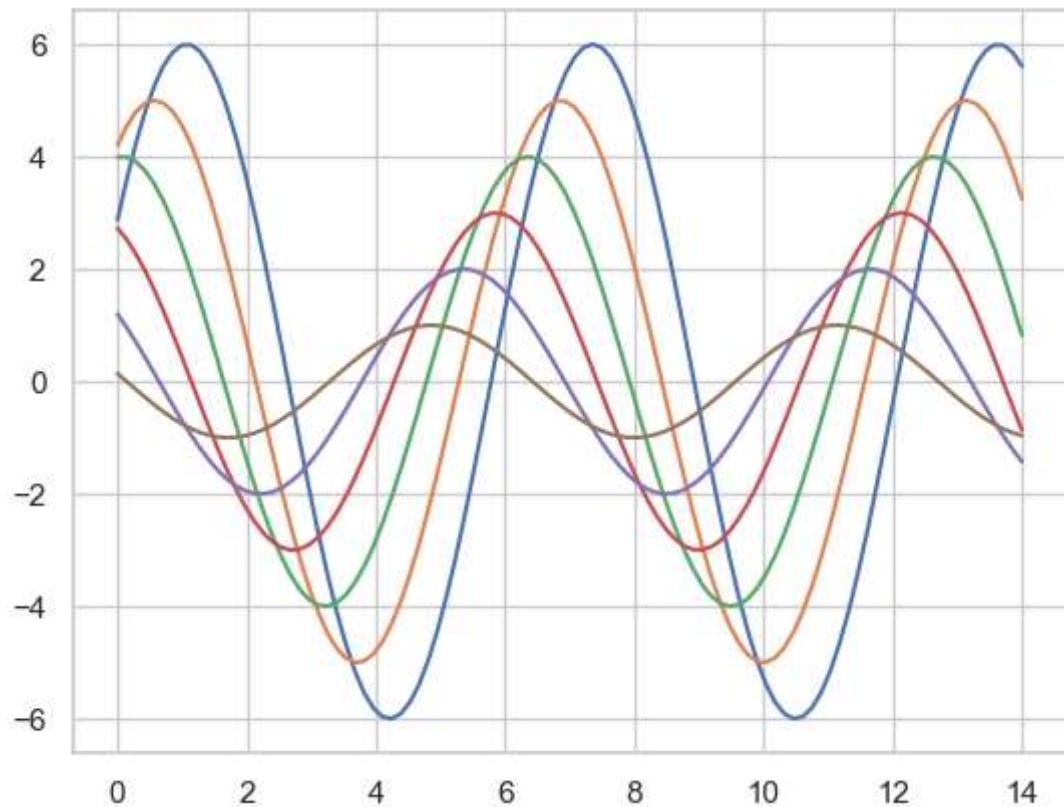
```
In [75]: sinplot()
```



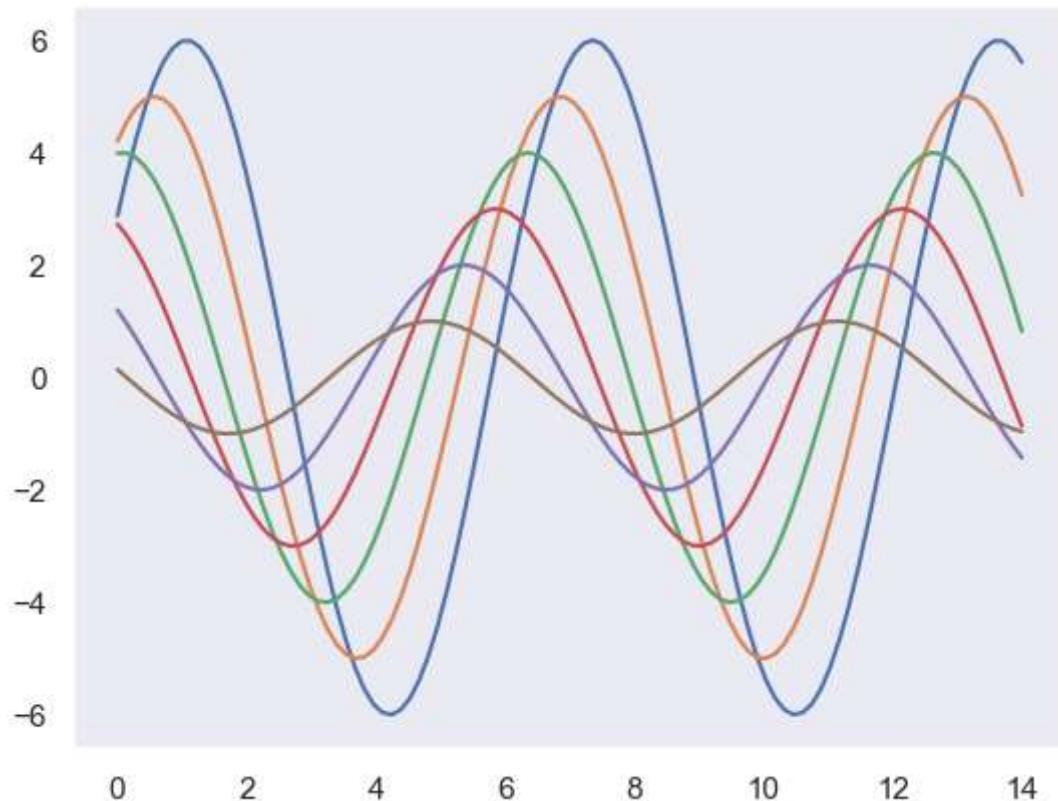
```
In [76]: sns.set()  
sinplot()
```



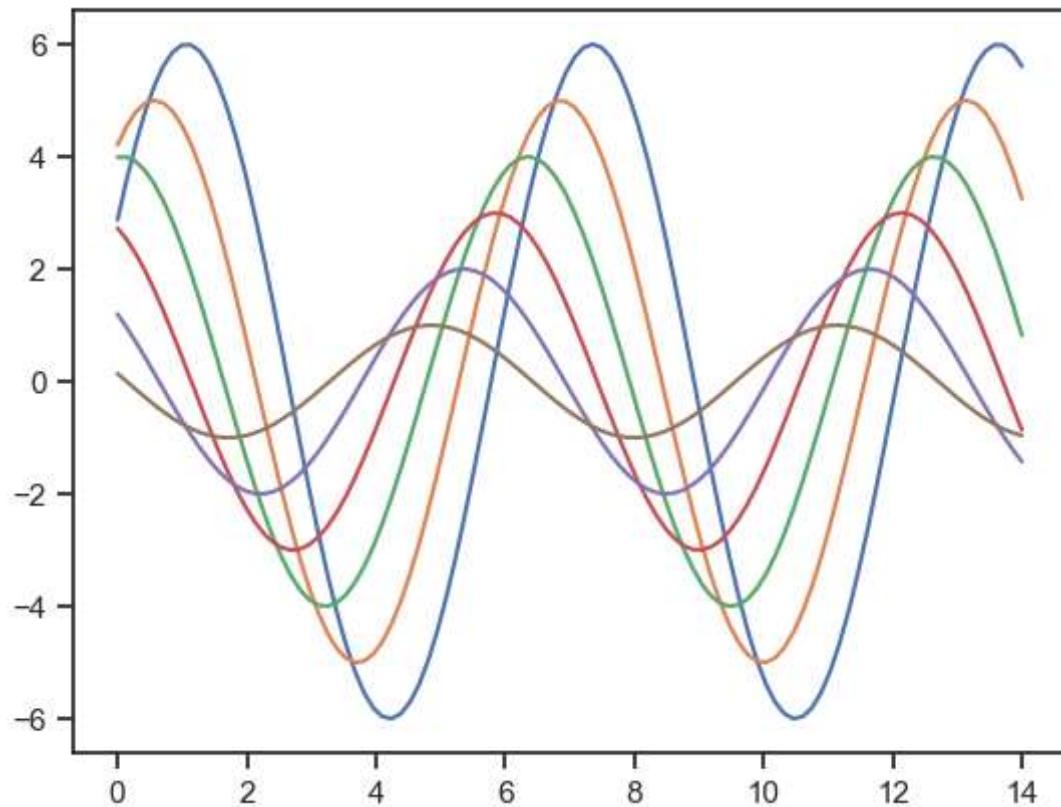
```
In [77]: sns.set_style("whitegrid")
sinplot()
```



```
In [78]: sns.set_style("dark")
sinplot()
```



```
In [79]: sns.set_style("ticks")
sinplot()
```



```
In [ ]:
```