

pip install seaborn

```
In [1]: 1 import seaborn as sns
        2 import matplotlib.pyplot as plt
        3 import numpy as np
        4 import pandas as pd
```

```
In [2]: 1 data = pd.read_csv('data.csv')
```

```
In [3]: 1 data.head()
```

Out[3]:

	Unnamed: 0	ID	Name	Age	Photo	Nationality	Flag	Overall	Potential	
0	0	158023	L. Messi	31	https://cdn.sofifa.org/players/4/19/158023.png	Argentina	https://cdn.sofifa.org/flags/52.png	94	94	Bar
1	1	20801	Cristiano Ronaldo	33	https://cdn.sofifa.org/players/4/19/20801.png	Portugal	https://cdn.sofifa.org/flags/38.png	94	94	Ju
2	2	190871	Neymar Jr	26	https://cdn.sofifa.org/players/4/19/190871.png	Brazil	https://cdn.sofifa.org/flags/54.png	92	93	Paris Gé
3	3	193080	De Gea	27	https://cdn.sofifa.org/players/4/19/193080.png	Spain	https://cdn.sofifa.org/flags/45.png	91	93	Manc
4	4	192985	K. De Bruyne	27	https://cdn.sofifa.org/players/4/19/192985.png	Belgium	https://cdn.sofifa.org/flags/7.png	91	92	Manc

5 rows × 89 columns



```
In [4]: 1 data.columns
```

```
Out[4]: Index(['Unnamed: 0', 'ID', 'Name', 'Age', 'Photo', 'Nationality', 'Flag',  
             'Overall', 'Potential', 'Club', 'Club Logo', 'Value', 'Wage', 'Special',  
             'Preferred Foot', 'International Reputation', 'Weak Foot',  
             'Skill Moves', 'Work Rate', 'Body Type', 'Real Face', 'Position',  
             'Jersey Number', 'Joined', 'Loaned From', 'Contract Valid Until',  
             'Height', 'Weight', 'LS', 'ST', 'RS', 'LW', 'LF', 'CF', 'RF', 'RW',  
             'LAM', 'CAM', 'RAM', 'LM', 'LCM', 'CM', 'RCM', 'RM', 'LWB', 'LDM',  
             'CDM', 'RDM', 'RWB', 'LB', 'LCB', 'CB', 'RCB', 'RB', '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', 'GKDividing', 'GKHandling',  
             'GK Kicking', 'GK Positioning', 'GK Reflexes', 'Release Clause'],  
            dtype='object')
```

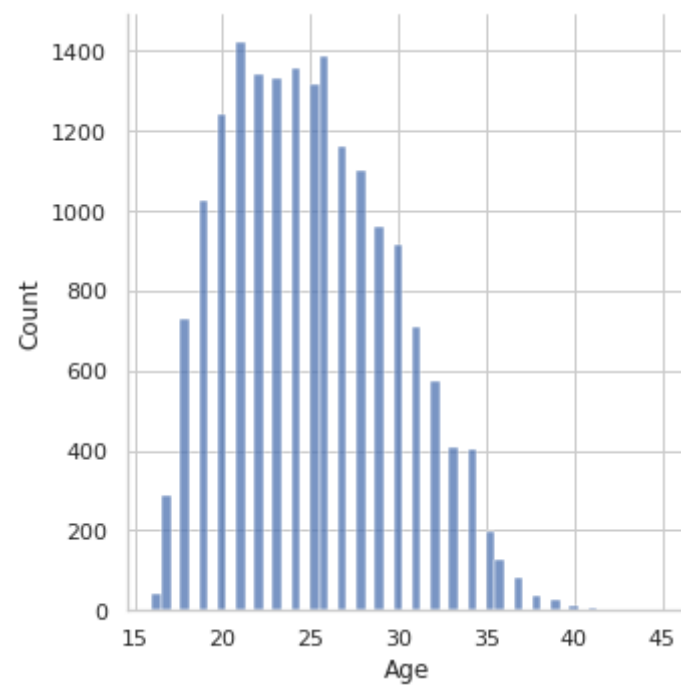
```
In [ ]: 1
```

```
In [5]: 1 sns.set(style='whitegrid')
```

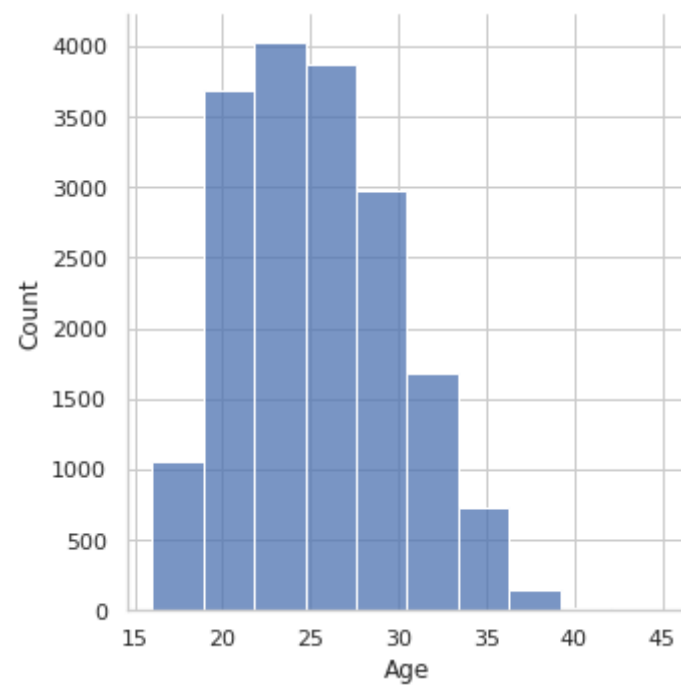
```
In [6]: 1 x = data['Age']
```

distplot -- distribution plot

```
In [7]: 1 sns.displot(data['Age'])  
        2 plt.show()
```

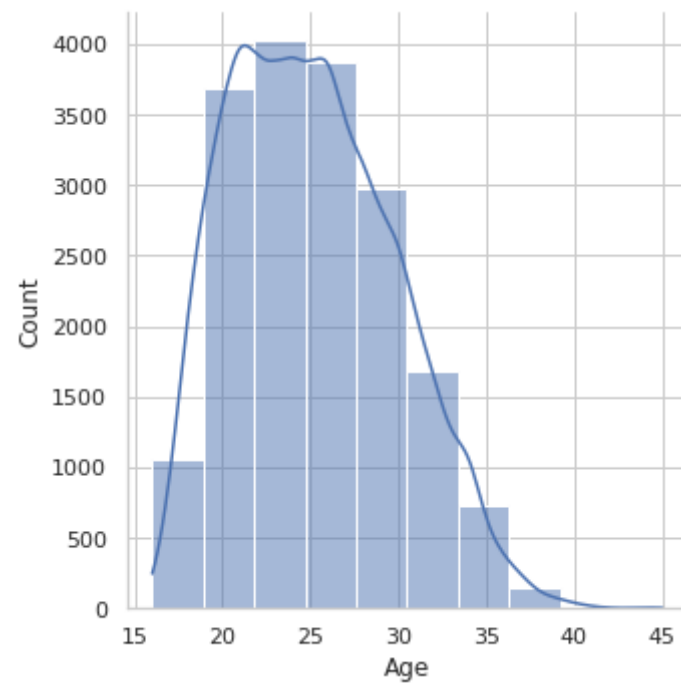


```
In [8]: 1 sns.displot(data['Age'],bins=10)
        2 plt.show()
```

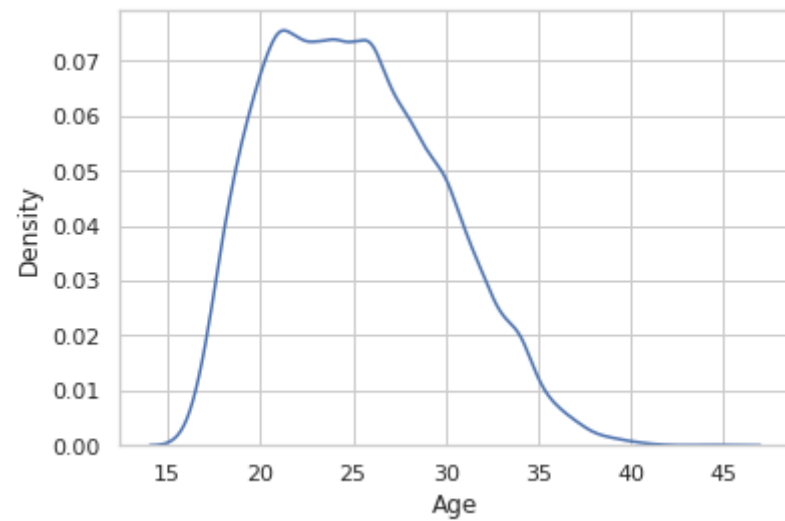


KDE

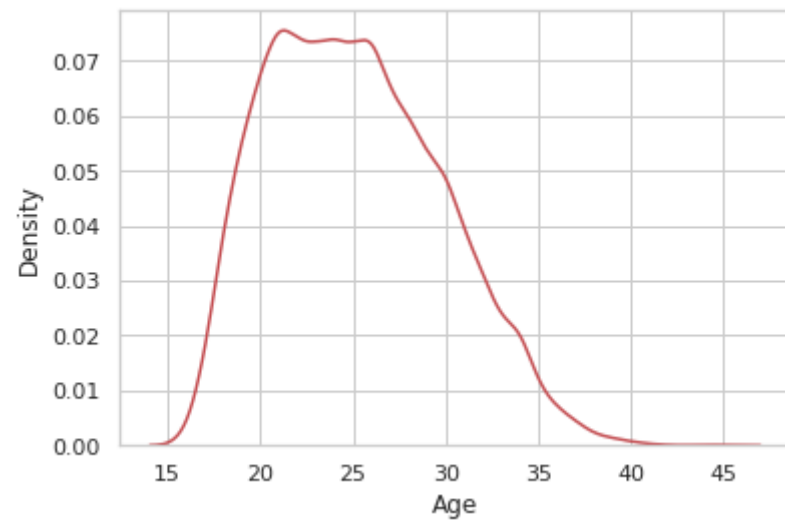
```
In [9]: 1 sns.displot(data['Age'],bins=10,kde=True)  
      2 plt.show()
```



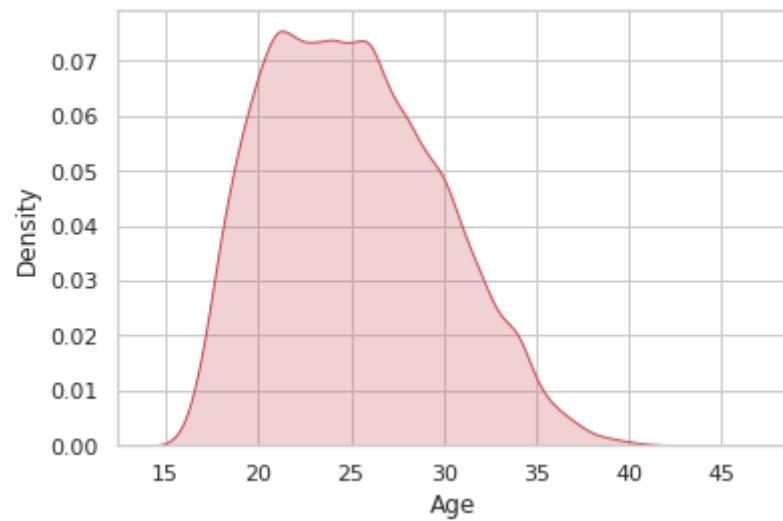
```
In [10]: 1 sns.kdeplot(data['Age'])  
        2 plt.show()
```



```
In [11]: 1 sns.kdeplot(data['Age'],color='r')  
        2 plt.show()
```



```
In [12]: 1 sns.kdeplot(data['Age'],color='r',shade=True)  
        2 plt.show()
```

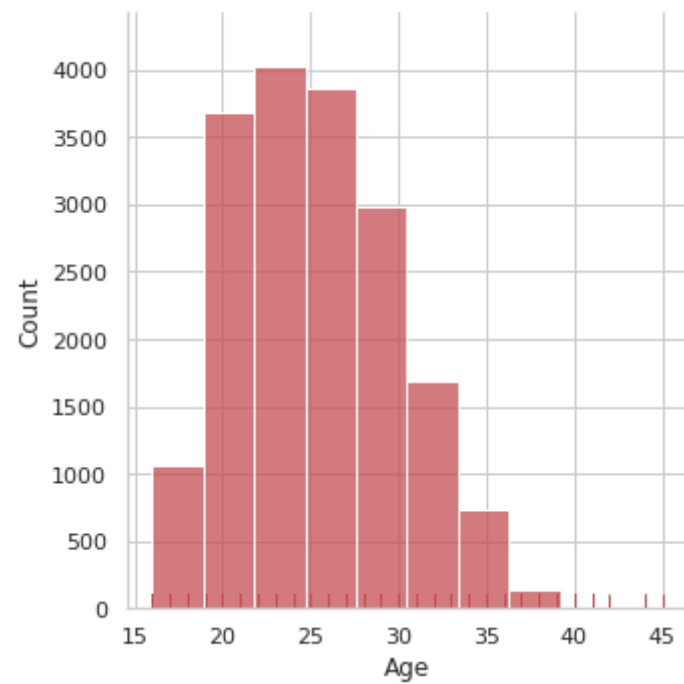


```
In [ ]:
```

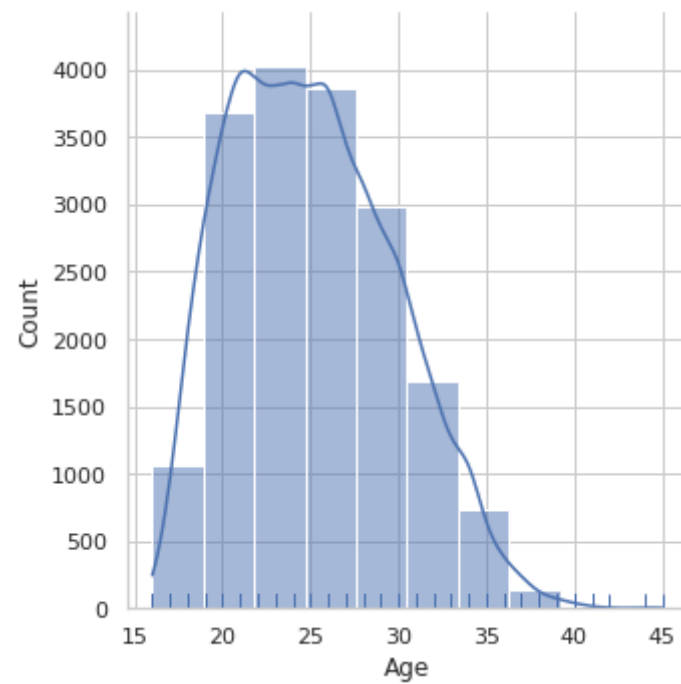
```
1
```

Histogram


```
In [13]: 1 sns.displot(data['Age'],kde=False,bins=10,rug=True,color='r')  
        2 plt.show()
```



```
In [14]: 1 sns.displot(data['Age'], kde=True, bins=10, rug=True)
        2 plt.show()
```



```
In [ ]:
```

```
1
```

```
In [15]: 1 data.columns
```

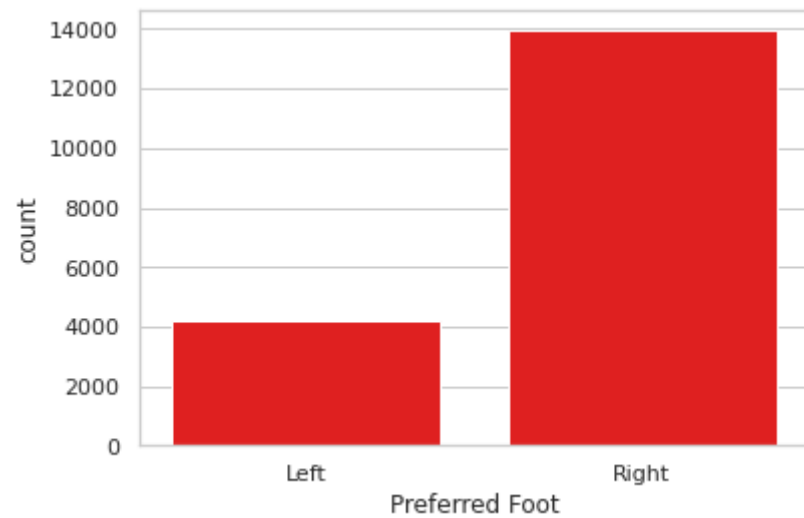
```
Out[15]: Index(['Unnamed: 0', 'ID', 'Name', 'Age', 'Photo', 'Nationality', 'Flag',  
              'Overall', 'Potential', 'Club', 'Club Logo', 'Value', 'Wage', 'Special',  
              'Preferred Foot', 'International Reputation', 'Weak Foot',  
              'Skill Moves', 'Work Rate', 'Body Type', 'Real Face', 'Position',  
              'Jersey Number', 'Joined', 'Loaned From', 'Contract Valid Until',  
              'Height', 'Weight', 'LS', 'ST', 'RS', 'LW', 'LF', 'CF', 'RF', 'RW',  
              'LAM', 'CAM', 'RAM', 'LM', 'LCM', 'CM', 'RCM', 'RM', 'LWB', 'LDM',  
              'CDM', 'RDM', 'RWB', 'LB', 'LCB', 'CB', 'RCB', 'RB', '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', 'GKDividing', 'GKHandling',  
              'GKKicking', 'GKPositioning', 'GKReflexes', 'Release Clause'],  
              dtype='object')
```

```
In [16]: 1 play_data = data['Preferred Foot'].value_counts()
```

```
In [17]: 1 play_data
```

```
Out[17]: Right    13948  
         Left     4211  
         Name: Preferred Foot, dtype: int64
```

```
In [18]: 1 sns.countplot(x="Preferred Foot",data=data,color='red')  
        2 plt.show()
```



```
In [ ]:
```

```
1
```

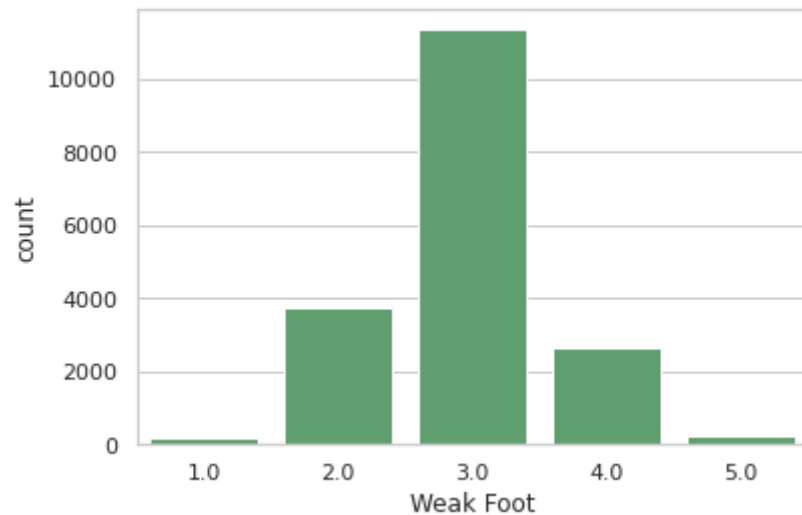
```
In [19]: 1 data.columns
```

```
Out[19]: Index(['Unnamed: 0', 'ID', 'Name', 'Age', 'Photo', 'Nationality', 'Flag',  
              'Overall', 'Potential', 'Club', 'Club Logo', 'Value', 'Wage', 'Special',  
              'Preferred Foot', 'International Reputation', 'Weak Foot',  
              'Skill Moves', 'Work Rate', 'Body Type', 'Real Face', 'Position',  
              'Jersey Number', 'Joined', 'Loaned From', 'Contract Valid Until',  
              'Height', 'Weight', 'LS', 'ST', 'RS', 'LW', 'LF', 'CF', 'RF', 'RW',  
              'LAM', 'CAM', 'RAM', 'LM', 'LCM', 'CM', 'RCM', 'RM', 'LWB', 'LDM',  
              'CDM', 'RDM', 'RWB', 'LB', 'LCB', 'CB', 'RCB', 'RB', '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', 'GKDividing', 'GKHandling',  
              'GK Kicking', 'GK Positioning', 'GK Reflexes', 'Release Clause'],  
              dtype='object')
```

```
In [20]: 1 data['Weak Foot'].value_counts()
```

```
Out[20]: 3.0    11349  
         2.0     3761  
         4.0     2662  
         5.0      229  
         1.0     158  
         Name: Weak Foot, dtype: int64
```

```
In [21]: 1 sns.countplot(x='Weak Foot',data=data,color='g')  
        2 plt.show()
```

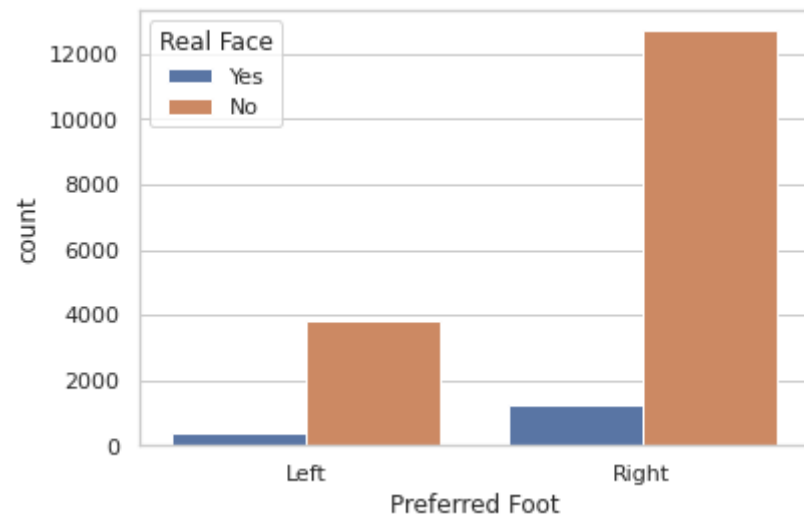


```
In [22]: 1 data['Real Face'].value_counts()
```

```
Out[22]: No      16505  
        Yes      1654  
        Name: Real Face, dtype: int64
```

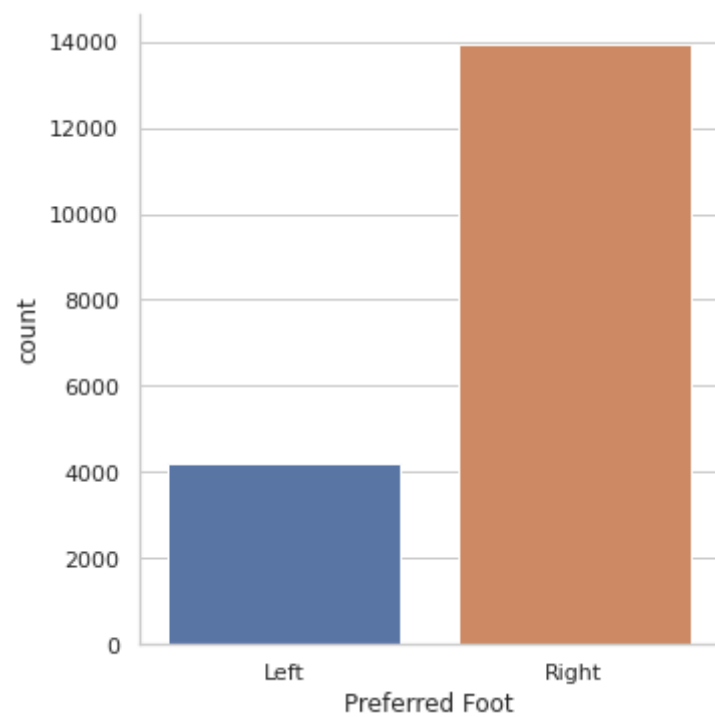
```
In [ ]: 1
```

```
In [23]: 1 sns.countplot(x="Preferred Foot",hue='Real Face',data=data)  
2 plt.show()
```

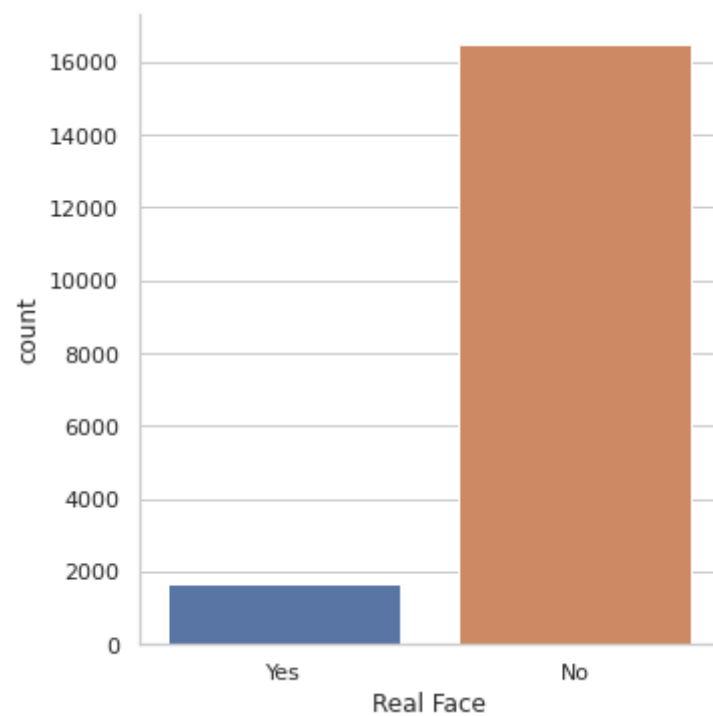


Catplot

```
In [24]: 1 sns.catplot(x="Preferred Foot",kind="count",data=data)  
        2 plt.show()
```




```
In [25]: 1 sns.catplot(x="Real Face",kind="count",data=data)  
2 plt.show()
```



In [26]: 1 data.columns

Out[26]: Index(['Unnamed: 0', 'ID', 'Name', 'Age', 'Photo', 'Nationality', 'Flag',
 'Overall', 'Potential', 'Club', 'Club Logo', 'Value', 'Wage', 'Special',
 'Preferred Foot', 'International Reputation', 'Weak Foot',
 'Skill Moves', 'Work Rate', 'Body Type', 'Real Face', 'Position',
 'Jersey Number', 'Joined', 'Loaned From', 'Contract Valid Until',
 'Height', 'Weight', 'LS', 'ST', 'RS', 'LW', 'LF', 'CF', 'RF', 'RW',
 'LAM', 'CAM', 'RAM', 'LM', 'LCM', 'CM', 'RCM', 'RM', 'LWB', 'LDM',
 'CDM', 'RDM', 'RWB', 'LB', 'LCB', 'CB', 'RCB', 'RB', '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', 'GKDividing', 'GKHandling',
 'GKKicking', 'GKPositioning', 'GKReflexes', 'Release Clause'],
 dtype='object')

In []:

1

Strip-plot

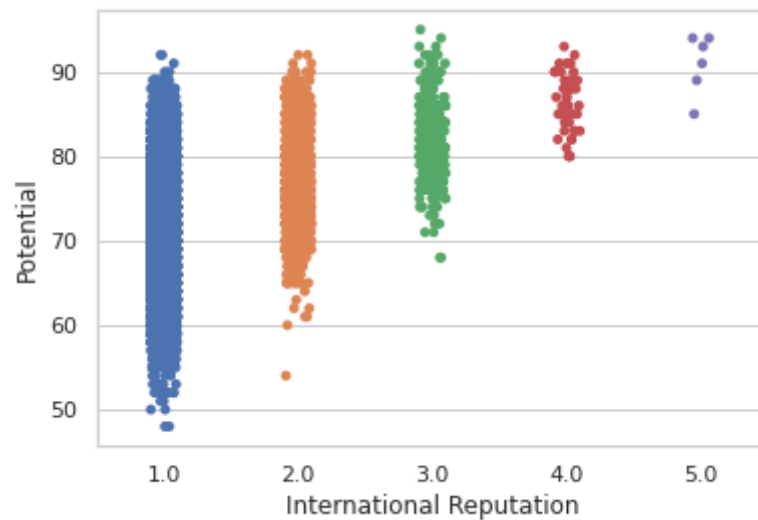
In [27]: 1 data['International Reputation'].value_counts()

Out[27]: 1.0 16532
 2.0 1261
 3.0 309
 4.0 51
 5.0 6
 Name: International Reputation, dtype: int64

```
In [28]: 1 data['Potential']
```

```
Out[28]: 0      94  
1      94  
2      93  
3      93  
4      92  
      ..  
18202   65  
18203   63  
18204   67  
18205   66  
18206   66  
Name: Potential, Length: 18207, dtype: int64
```

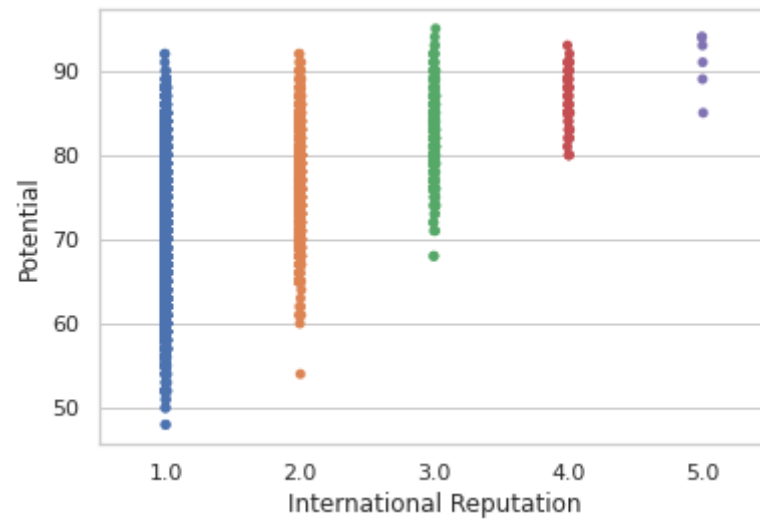
```
In [29]: 1 sns.stripplot(x="International Reputation",y="Potential",data=data)  
2 plt.show()
```



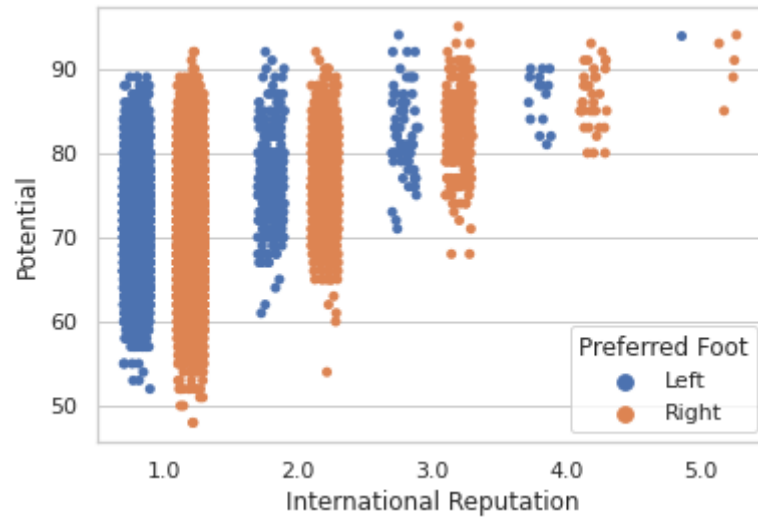
```
In [ ]:
```

```
1
```

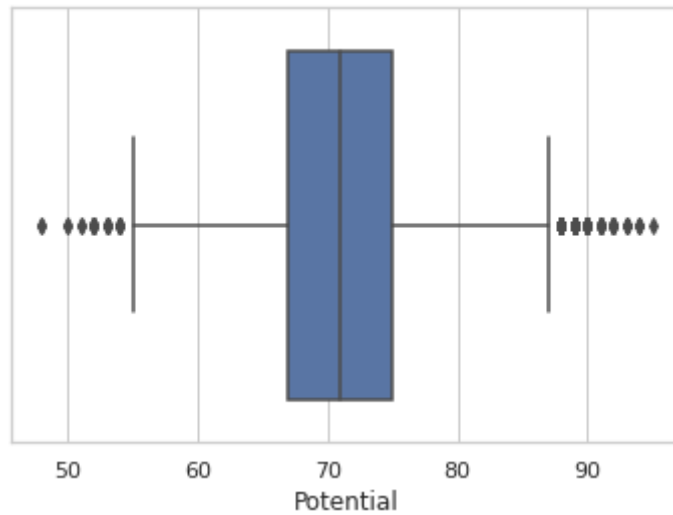
```
In [30]: 1 sns.stripplot(x="International Reputation",y="Potential",data=data,jitter=0.01)  
        2 plt.show()
```



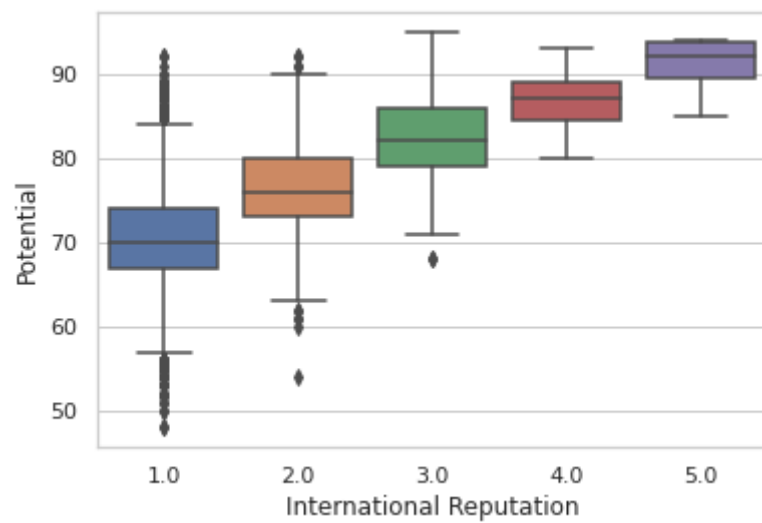
```
In [31]: 1 sns.stripplot(x="International Reputation", y="Potential", hue="Preferred Foot", data=data, jitter=0.2, dodge  
2 plt.show())
```



```
In [32]: 1 sns.boxplot(x=data['Potential'])  
2 plt.show()
```



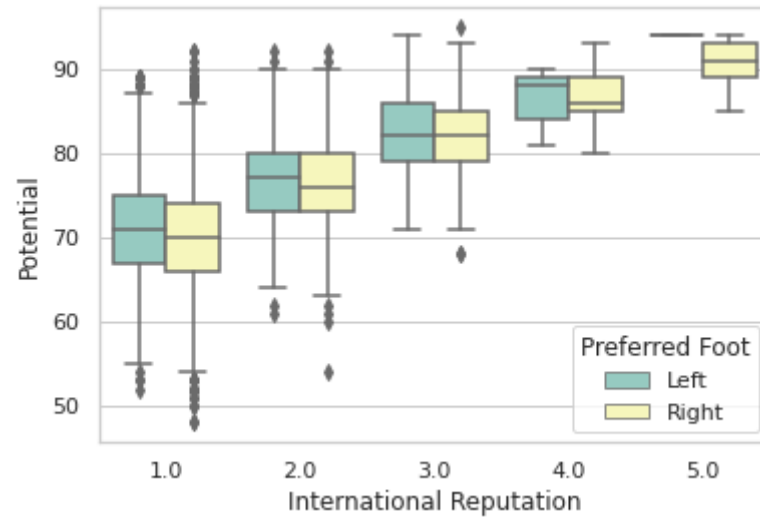
```
In [33]: 1 sns.boxplot(x="International Reputation",y="Potential",data=data)  
2 plt.show()
```



```
In [ ]:
```

```
1
```

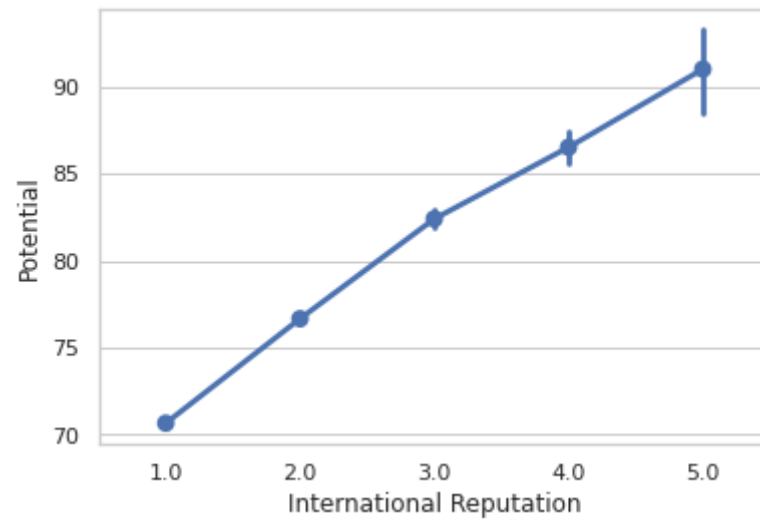
```
In [34]: 1 sns.boxplot(x="International Reputation",y="Potential",hue="Preferred Foot",data=data,palette="Set3")  
2 plt.show()
```



```
In [ ]: 1
```

Point-Plot

```
In [35]: 1 sns.pointplot(x="International Reputation", y="Potential", data=data)  
2 plt.show()
```

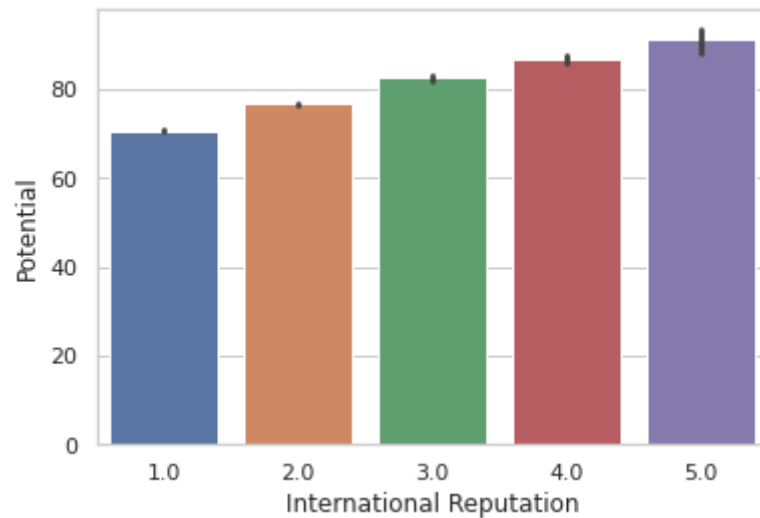


```
In [ ]:
```

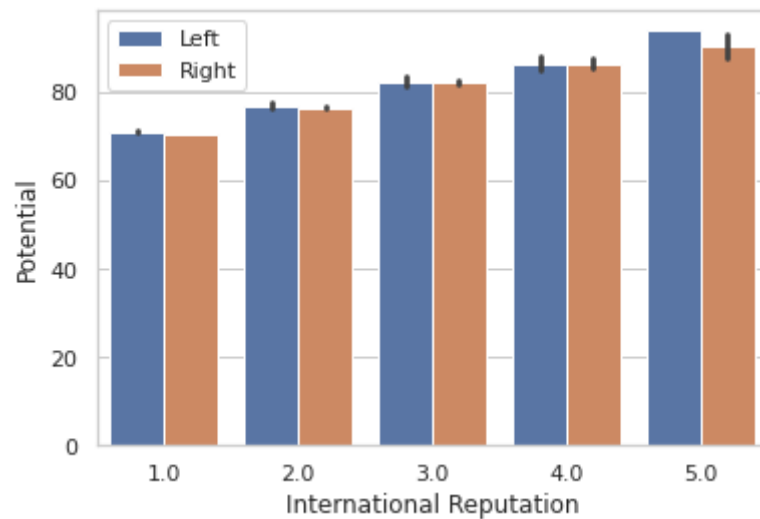
```
1
```

bar-plot


```
In [36]: 1 sns.barplot(x='International Reputation',y='Potential',data=data)  
2 plt.show()
```



```
In [37]: 1 sns.barplot(x='International Reputation',y='Potential',hue="Preferred Foot",data=data)  
2 plt.legend(loc='upper left')  
3 plt.show()
```

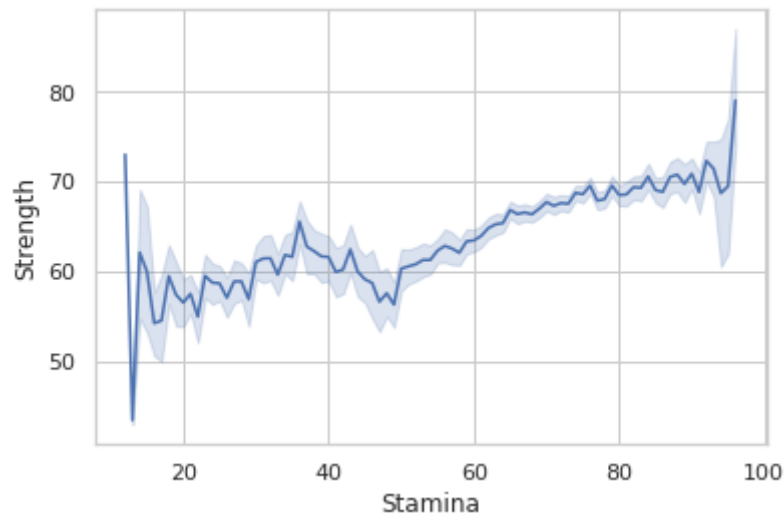


line plot

```
In [38]: 1 data.columns
```

```
Out[38]: Index(['Unnamed: 0', 'ID', 'Name', 'Age', 'Photo', 'Nationality', 'Flag',  
              'Overall', 'Potential', 'Club', 'Club Logo', 'Value', 'Wage', 'Special',  
              'Preferred Foot', 'International Reputation', 'Weak Foot',  
              'Skill Moves', 'Work Rate', 'Body Type', 'Real Face', 'Position',  
              'Jersey Number', 'Joined', 'Loaned From', 'Contract Valid Until',  
              'Height', 'Weight', 'LS', 'ST', 'RS', 'LW', 'LF', 'CF', 'RF', 'RW',  
              'LAM', 'CAM', 'RAM', 'LM', 'LCM', 'CM', 'RCM', 'RM', 'LWB', 'LDM',  
              'CDM', 'RDM', 'RWB', 'LB', 'LCB', 'CB', 'RCB', 'RB', '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', 'GKDividing', 'GKHandling',  
              'GKKicking', 'GKPositioning', 'GKReflexes', 'Release Clause'],  
             dtype='object')
```

```
In [39]: 1 sns.lineplot(x="Stamina",y="Strength",data=data)  
        2 plt.show()
```

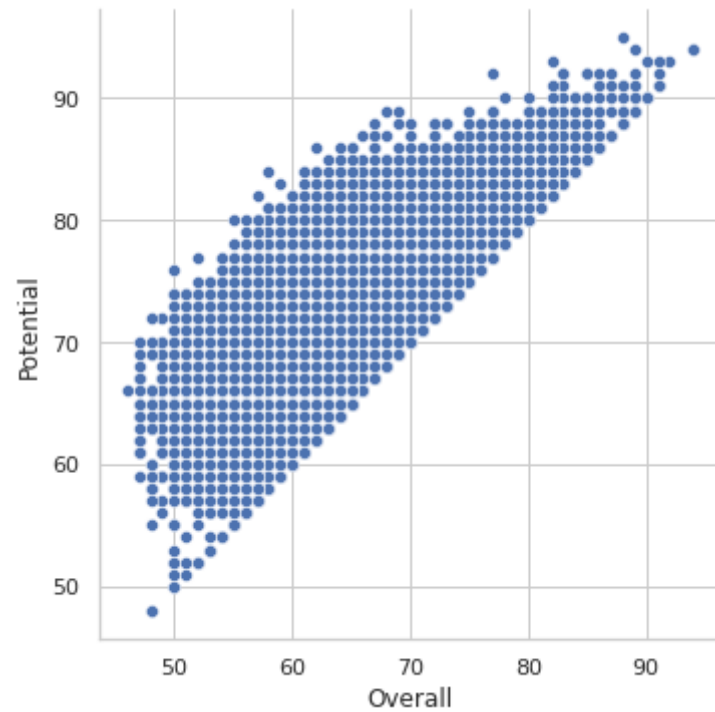


```
In [40]: 1 data.columns
```

```
Out[40]: Index(['Unnamed: 0', 'ID', 'Name', 'Age', 'Photo', 'Nationality', 'Flag',  
              'Overall', 'Potential', 'Club', 'Club Logo', 'Value', 'Wage', 'Special',  
              'Preferred Foot', 'International Reputation', 'Weak Foot',  
              'Skill Moves', 'Work Rate', 'Body Type', 'Real Face', 'Position',  
              'Jersey Number', 'Joined', 'Loaned From', 'Contract Valid Until',  
              'Height', 'Weight', 'LS', 'ST', 'RS', 'LW', 'LF', 'CF', 'RF', 'RW',  
              'LAM', 'CAM', 'RAM', 'LM', 'LCM', 'CM', 'RCM', 'RM', 'LWB', 'LDM',  
              'CDM', 'RDM', 'RWB', 'LB', 'LCB', 'CB', 'RCB', 'RB', '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', 'GKDividing', 'GKHandling',  
              'GK Kicking', 'GK Positioning', 'GK Reflexes', 'Release Clause'],  
              dtype='object')
```

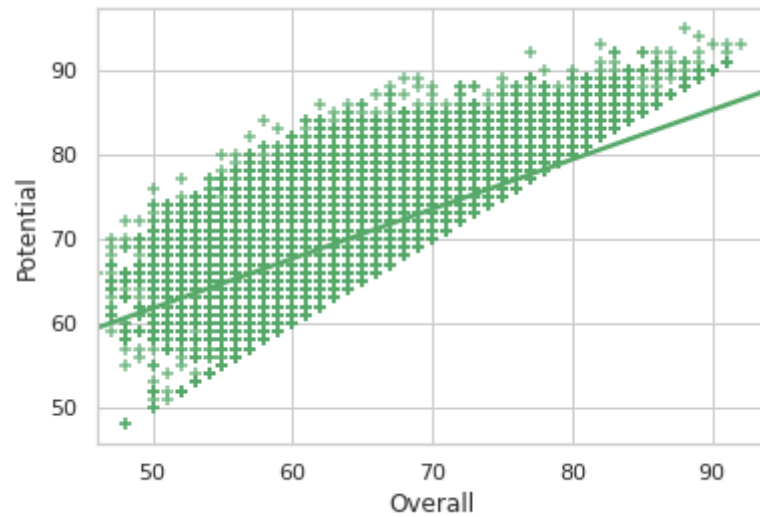
relplot

```
In [41]: 1 sns.relplot(x='Overall',y='Potential',data=data)  
2 plt.show()
```



reg plot

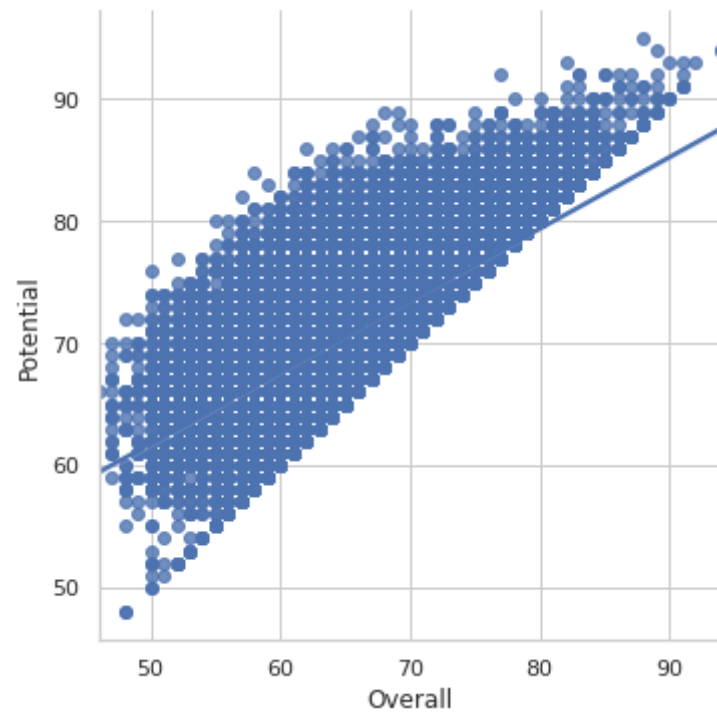
```
In [42]: 1 sns.regplot(x='Overall',y='Potential',data=data,color='g',marker='+')  
        2 plt.show()
```



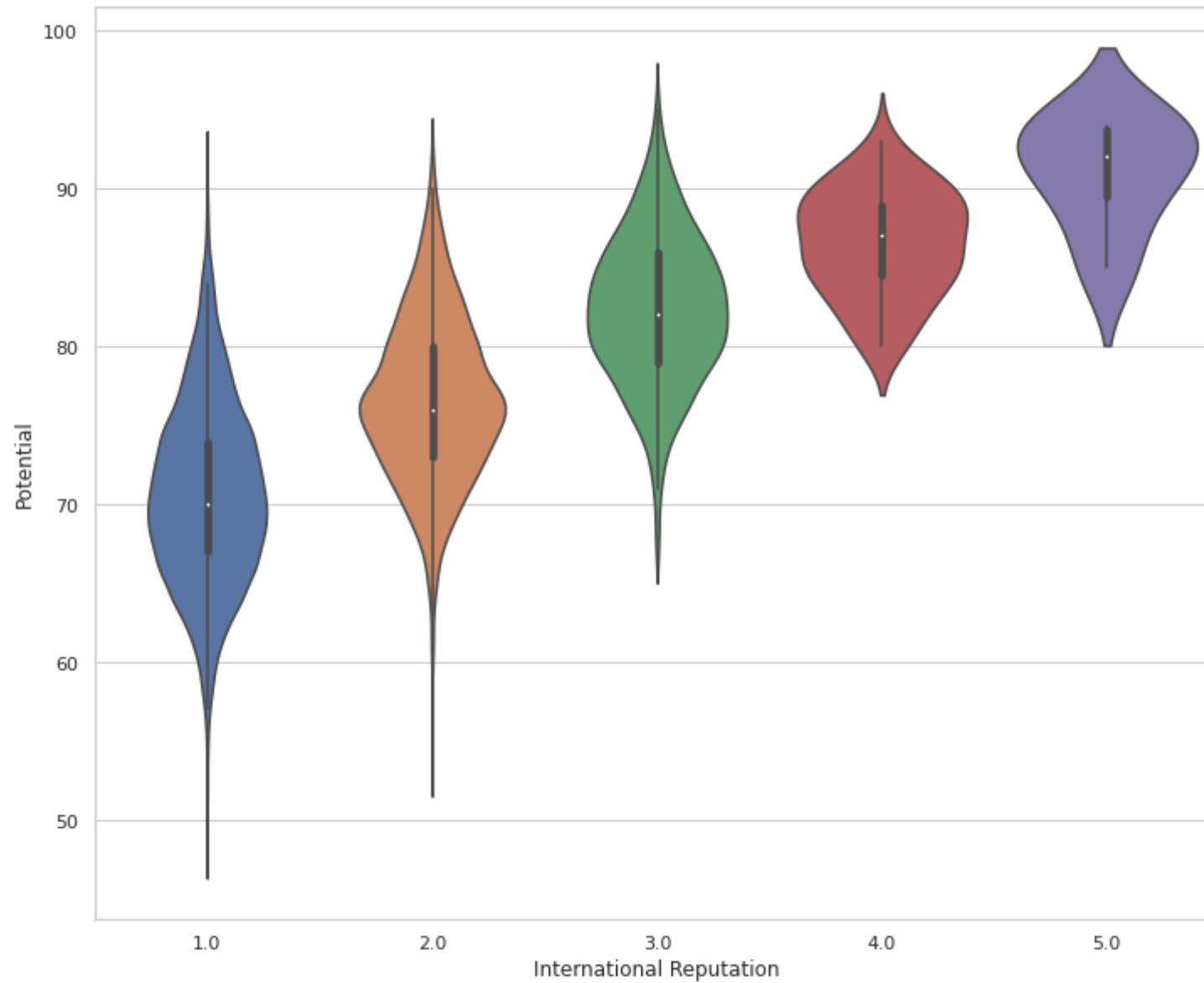
```
In [ ]:
```

```
1
```

```
In [43]: 1 sns.lmplot(x='Overall',y='Potential',data=data)  
        2 plt.show()
```



```
In [48]: 1 plt.figure(figsize=(12,10))  
2 sns.violinplot(x="International Reputation",y="Potential",data=data)  
3 plt.show()
```

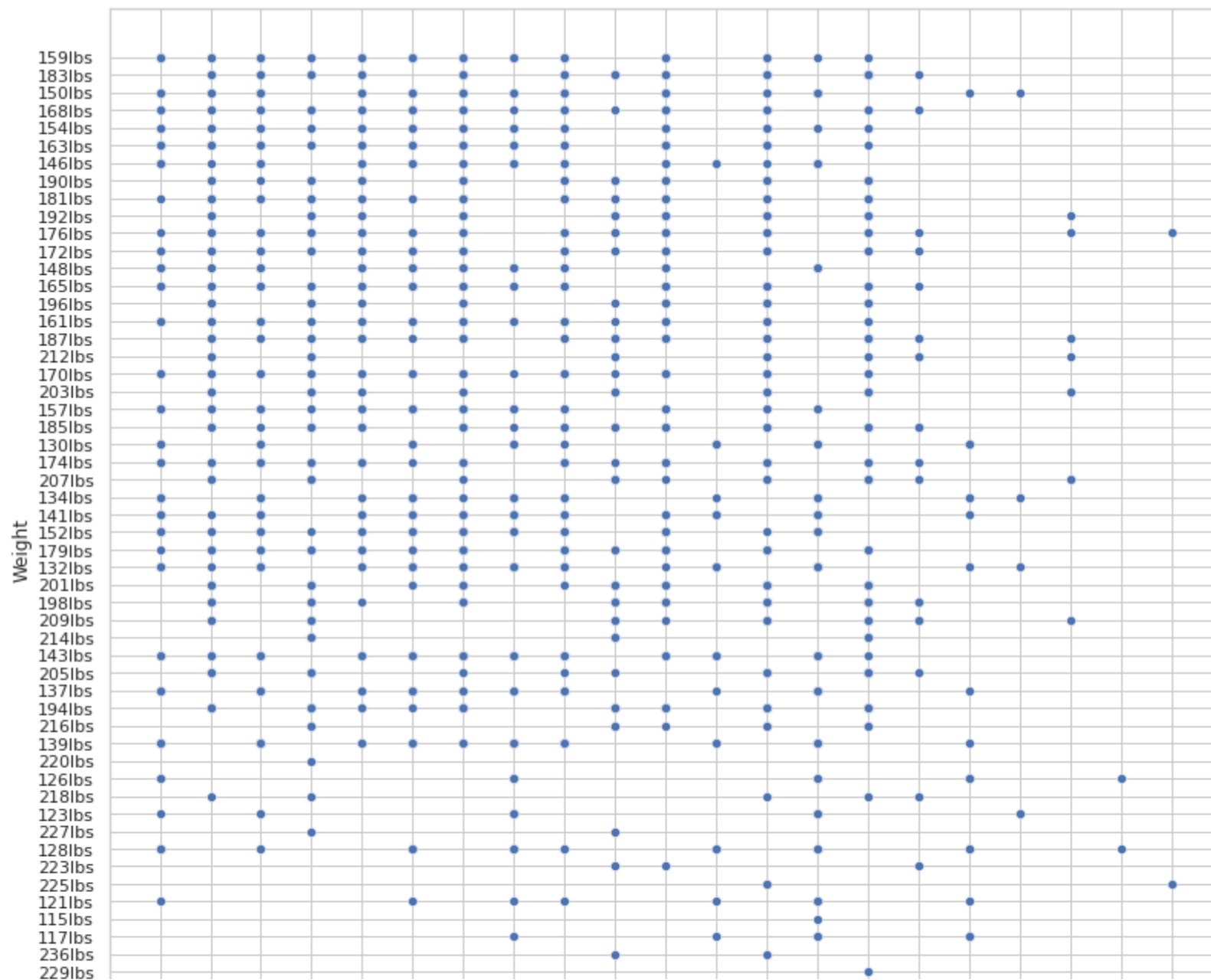


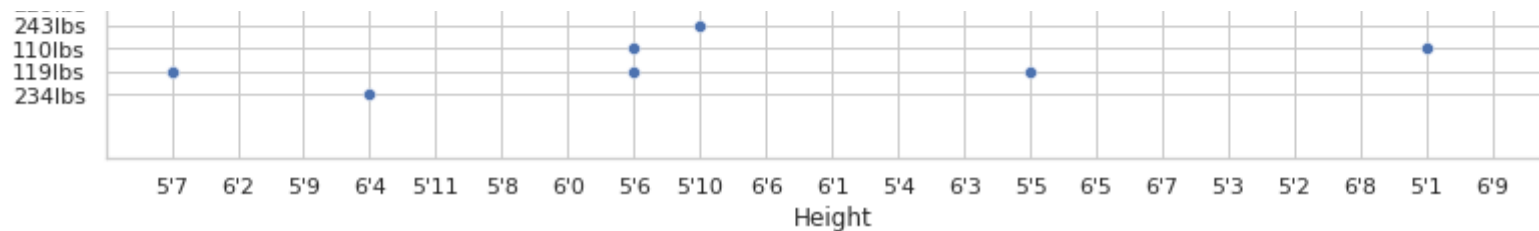
```
In [83]: 1 len(data.columns)
```

```
Out[83]: 89
```

Scatter Plot


```
In [51]: 1 plt.figure(figsize=(13,13))
2         sns.scatterplot(x="Height",y="Weight",data=data)
3         plt.show()
```





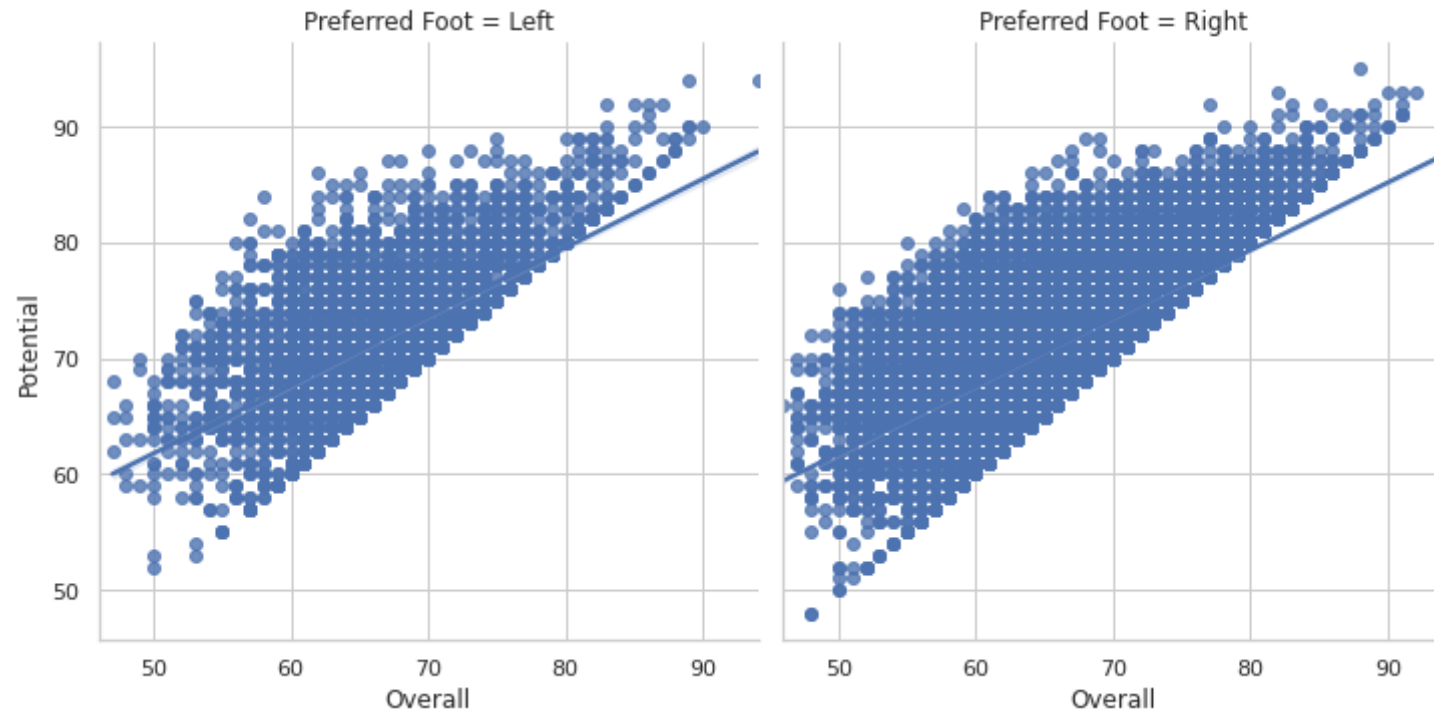
In [53]: 1 data["Height"].value_counts()

```
Out[53]: 6'0      2881
5'10     2479
5'9      2238
5'11     2159
6'2      2015
6'1      1908
6'3       990
5'8       946
5'7       905
6'4       749
5'6       316
6'5       246
5'5       145
6'6        93
5'4        30
6'7        21
5'3        18
6'8        10
5'2         5
5'1         3
6'9         2
Name: Height, dtype: int64
```

In []:

1

```
In [54]: 1 sns.lmplot(x='Overall',y='Potential',col='Preferred Foot',data=data)  
2 plt.show()
```

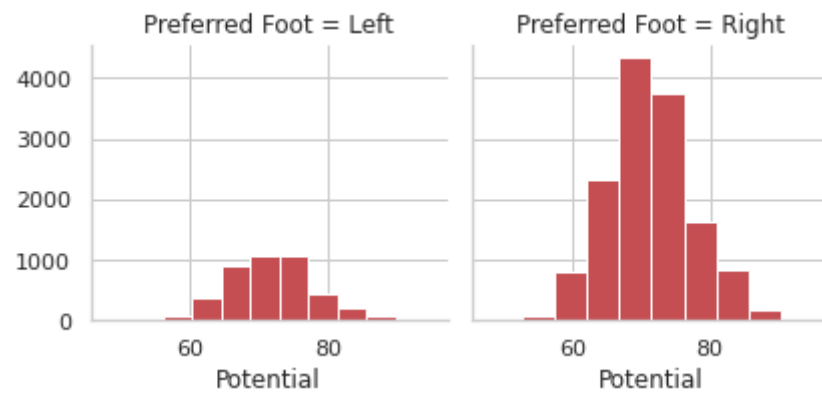


```
In [59]: 1 data['Preferred Foot'].value_counts()
```

```
Out[59]: Right    13948  
Left      4211  
Name: Preferred Foot, dtype: int64
```

```
In [64]: 1 g = sns.FacetGrid(data,col="Preferred Foot")  
        2 g.map(plt.hist,"Potential",color='r')
```

Out[64]: <seaborn.axisgrid.FacetGrid at 0x7f9ff53133a0>

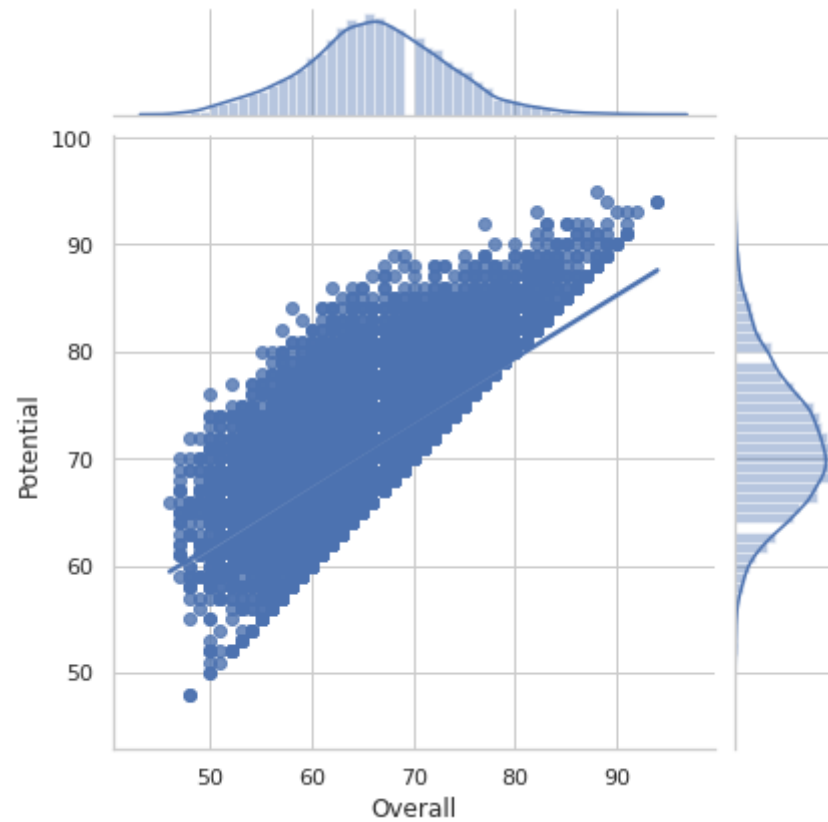


```
In [67]: 1 import warnings  
        2 warnings.filterwarnings('ignore')
```

```
In [70]: 1 plt.figure(figsize=(13,13))
          2 g = sns.JointGrid(x="Overall",y="Potential",data=data)
          3 g.plot(sns.regplot,sns.distplot)
```

Out[70]: <seaborn.axisgrid.JointGrid at 0x7f9ff49f85b0>

<Figure size 936x936 with 0 Axes>



In []:

1

```
In [81]: 1 plt.figure(figsize=(13,13))
2 data1 = [15,25,25,30,5]
3 label = ['G1','G2','G3','G4','G5']
4 color = sns.color_palette('bright')[0:5]
5 plt.pie(data1,labels=label,colors=color,explode=[0.05]*5)
6 plt.show()
```





In []:

1

In [84]:

1 df = sns.load_dataset('tips')

In [85]:

1 df

Out[85]:

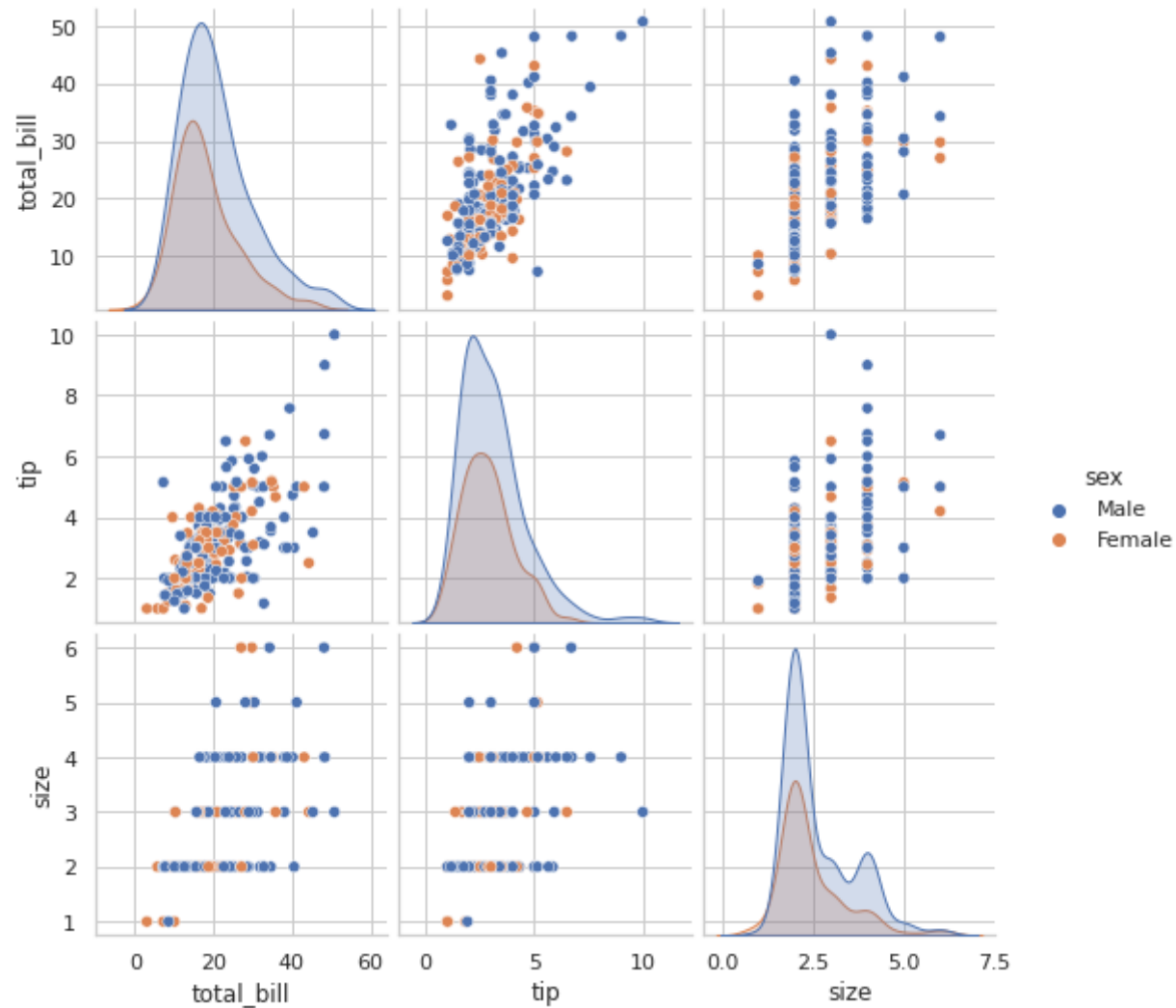
	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4
...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

244 rows × 7 columns

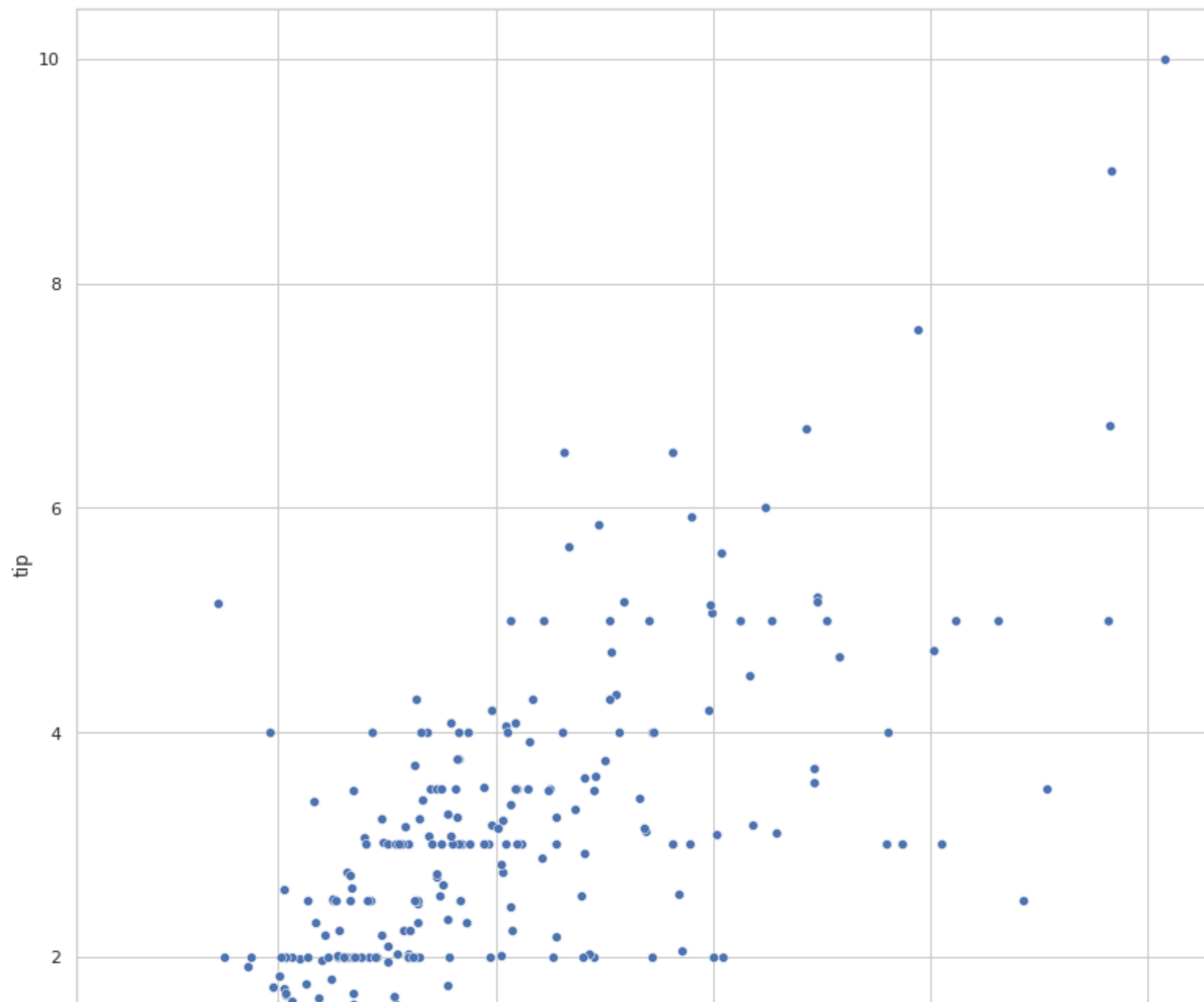

```
In [88]: 1 plt.figure(figsize=(13,13))
        2 sns.pairplot(df,hue='sex',kind='scatter')
```

Out[88]: <seaborn.axisgrid.PairGrid at 0x7f9fbfff8400>

<Figure size 936x936 with 0 Axes>




```
In [90]: 1 plt.figure(figsize=(13,13))  
2 sns.scatterplot(x="total_bill",y="tip",data=df)  
3 plt.show()
```





1