In [1]:
```
##Machin learning module:
#1 - Import data
#2- Clean the data
#3- Split data. Training Set/Test set
#4- Create a Model
#5- Check the output
#6- Improve
```

Import data

In [2]:
```
import pandas as pd
Data_frame = pd.read_csv('data.csv')
Data_frame
```

Out[2]:

	Unnamed: 0	ID	Name	Age	Photo	National
0	0	158023	L. Messi	31	https://cdn.sofifa.org/players/4/19/158023.png	Argenti
1	1	20801	Cristiano Ronaldo	33	https://cdn.sofifa.org/players/4/19/20801.png	Portu
2	2	190871	Neymar Jr	26	https://cdn.sofifa.org/players/4/19/190871.png	Bra
3	3	193080	De Gea	27	https://cdn.sofifa.org/players/4/19/193080.png	Sp
4	4	192985	K. De Bruyne	27	https://cdn.sofifa.org/players/4/19/192985.png	Belgi
...	...	...	...	...	...	
18202	18202	238813	J. Lundstram	19	https://cdn.sofifa.org/players/4/19/238813.png	Engla
18203	18203	243165	N. Christoffersson	19	https://cdn.sofifa.org/players/4/19/243165.png	Swed
18204	18204	241638	B. Worman	16	https://cdn.sofifa.org/players/4/19/241638.png	Engla
18205	18205	246268	D. Walker-Rice	17	https://cdn.sofifa.org/players/4/19/246268.png	Engla
18206	18206	246269	G. Nugent	16	https://cdn.sofifa.org/players/4/19/246269.png	Engla

18207 rows × 89 columns

In [3]:
```
Data_frame.shape
```

Out[3]:
```
(18207, 89)
```

In [4]:
```
1 Data_frame.isnull()
```

Out[4]:

	Unnamed: 0	ID	Name	Age	Photo	Nationality	Flag	Overall	Potential	Club	...	Co
0	False	False	False	False	False	False	False	False	False	False	...	
1	False	False	False	False	False	False	False	False	False	False	...	
2	False	False	False	False	False	False	False	False	False	False	...	
3	False	False	False	False	False	False	False	False	False	False	...	
4	False	False	False	False	False	False	False	False	False	False	...	
...	...	...	...	...	...	...	...	...	...	...	...	
18202	False	False	False	False	False	False	False	False	False	False	...	
18203	False	False	False	False	False	False	False	False	False	False	...	
18204	False	False	False	False	False	False	False	False	False	False	...	
18205	False	False	False	False	False	False	False	False	False	False	...	
18206	False	False	False	False	False	False	False	False	False	False	...	

18207 rows × 89 columns

In [5]:
```
1 Data_frame.describe()
2
```

Out[5]:

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

8 rows × 44 columns

In [6]:
```
1 Data_frame.all()
```

Out[6]:
```
Unnamed: 0        False
ID                 True
Name               True
Age                True
Photo              True
                  ...
GKHandling         True
GKKicking          True
GKPositioning      True
GKReflexes         True
Release Clause     True
Length: 89, dtype: bool
```

In [7]:
```
1 Data_frame.values
```

Out[7]:
```
array([[0, 158023, 'L. Messi', ..., 14.0, 8.0, '€226.5M'],
       [1, 20801, 'Cristiano Ronaldo', ..., 14.0, 11.0, '€127.1M'],
       [2, 190871, 'Neymar Jr', ..., 15.0, 11.0, '€228.1M'],
       ...,
       [18204, 241638, 'B. Worman', ..., 6.0, 13.0, '€165K'],
       [18205, 246268, 'D. Walker-Rice', ..., 8.0, 9.0, '€143K'],
       [18206, 246269, 'G. Nugent', ..., 12.0, 9.0, '€165K']],
      dtype=object)
```

In [8]:
```
1 Data_frame[Data_frame["Age"]>40].head()
```

Out[8]:

	Unnamed: 0	ID	Name	Age	Photo	Nationality
1120	1120	156092	J. Villar	41	https://cdn.sofifa.org/players/4/19/156092.png	Paraguay
4228	4228	3665	B. Nivet	41	https://cdn.sofifa.org/players/4/19/3665.png	France
4741	4741	140029	O. Pérez	45	https://cdn.sofifa.org/players/4/19/140029.png	Mexico
7225	7225	142998	C. Muñoz	41	https://cdn.sofifa.org/players/4/19/142998.png	Argentina
10545	10545	140183	S. Narazaki	42	https://cdn.sofifa.org/players/4/19/140183.png	Japan

5 rows × 89 columns

In [9]:
```
1 # Note you don't need to memorize this codes . often you can check the
2 #pandas documentation for how to manipulate the data
3 # for filtration and grab the data that you need.
```

Cleaning the data.

In [10]:
```
# your boss asking you this?!!
# what player the best in terms of skill but are also lower salary the we
# value - wage - name. (colums)
```

In [11]:
```
df1 = pd.DataFrame(Data_frame, columns=['Name','Wage','Value'])
df1
```

Out[11]:

	Name	Wage	Value
0	L. Messi	€565K	€110.5M
1	Cristiano Ronaldo	€405K	€77M
2	Neymar Jr	€290K	€118.5M
3	De Gea	€260K	€72M
4	K. De Bruyne	€355K	€102M
...	...	...	...
18202	J. Lundstram	€1K	€60K
18203	N. Christoffersson	€1K	€60K
18204	B. Worman	€1K	€60K
18205	D. Walker-Rice	€1K	€60K
18206	G. Nugent	€1K	€60K

18207 rows × 3 columns

In [14]:

```
def value_to_float(x):
    if type(x) == float or type(x) == int:
        return x
    if 'K' in x:
        if len(x) > 1:
            return float(x.replace('K', '')) * 1000
        return 1000.0
    if 'M' in x:
        if len(x) > 1:
            return float(x.replace('M', '')) * 1000000
        return 1000000.0
    if 'B' in x:
        return float(x.replace('B', '')) * 1000000000
    return 0.0

wage = df1['Wage'].replace('[\€,]', '', regex=True).apply(value_to_float)
value = df1['Value'].replace('[\€,]', '', regex=True).apply(value_to_float

df1['Wage'] = wage
df1['Value'] = value

df1['difference'] = df1['Value'] - df1['Wage']
df1.sort_values(by='difference', ascending=False )

```

Out[14]:

	Name	Wage	Value	Vlaue	difference
2	Neymar Jr	290000.0	118500000.0	118500000.0	118210000.0
0	L. Messi	565000.0	110500000.0	110500000.0	109935000.0
4	K. De Bruyne	355000.0	102000000.0	102000000.0	101645000.0
5	E. Hazard	340000.0	93000000.0	93000000.0	92660000.0
15	P. Dybala	205000.0	89000000.0	89000000.0	88795000.0
...	...	...	...	...	...
17752	S. Phillips	1000.0	0.0	0.0	-1000.0
12192	H. Sulaimani	3000.0	0.0	0.0	-3000.0
3550	S. Nakamura	4000.0	0.0	0.0	-4000.0
4228	B. Nivet	5000.0	0.0	0.0	-5000.0
864	Hilton	18000.0	0.0	0.0	-18000.0

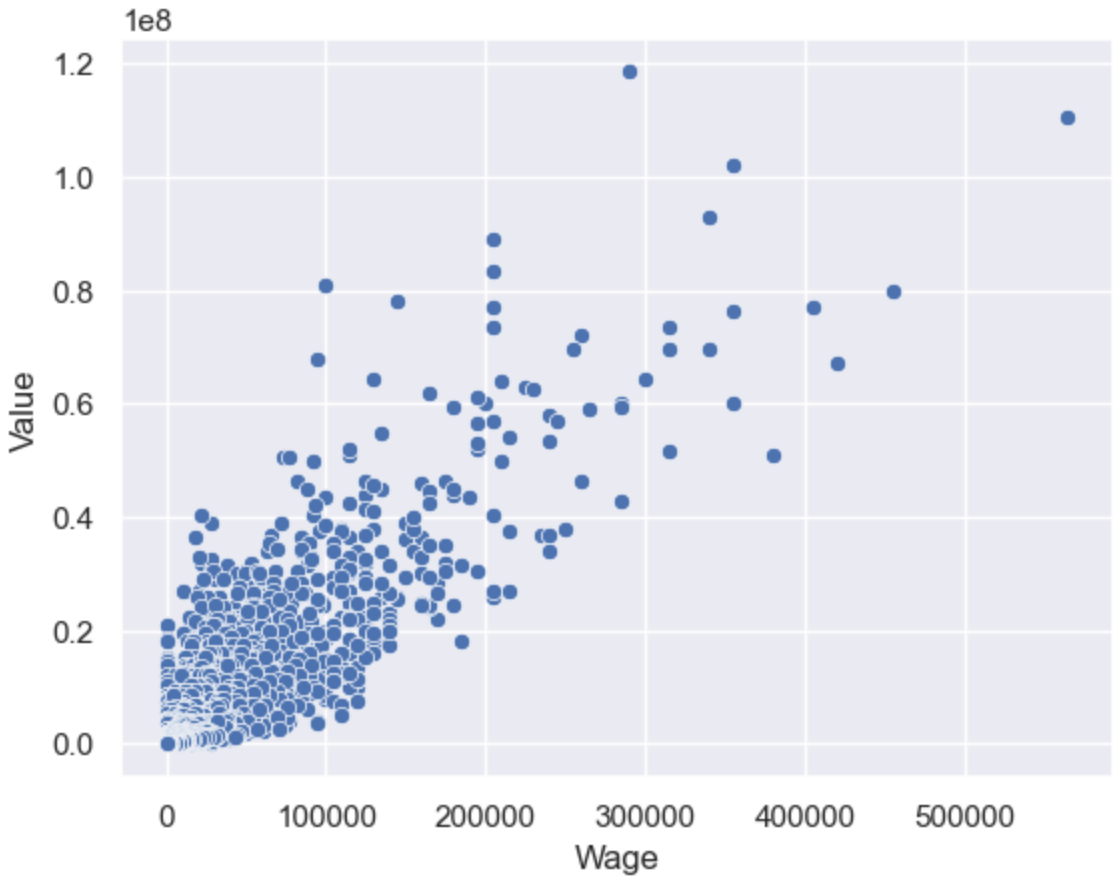
18207 rows × 5 columns

Visualize the Data

In [18]:
```
import seaborn as sns
sns.set()

graph = sns.scatterplot(x='Wage', y='Value',data = df1)
graph
```

Out[18]: <AxesSubplot:xlabel='Wage', ylabel='Value'>



In [30]:
```
from bokeh.plotting import figure, show
from bokeh.models import HoverTool

p = figure(title="Soccer 2019", x_axis_label='Wage', y_axis_label='Value',
p.circle('Wage', 'Value', size=10, source=df1)
show(p)

```

Hover tool to indicates the metrics and interact with points values

In [33]:
```
from bokeh.plotting import figure, show
from bokeh.models import HoverTool

# Create the figure
p = figure(title="Soccer 2019", x_axis_label='Wage', y_axis_label='Value',

# Add the circle glyph
p.circle('Wage', 'Value', size=10, source=df1)

# Create the HoverTool
hover = HoverTool(tooltips=[
    ("Name", "@Name"),
    ("Wage", "@Wage"),
    ("Value", "@Value"),
])

# Add the HoverTool to the figure
p.add_tools(hover)

# Show the plot
show(p)

```

##It will produced in html file in your c drive

In [36]:
```
##file:///C:/Users/Hazem%20EL-Batawy/AppData/Local/Temp/tmprtex0qfq.html
```