

```
In [4]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
In [5]: df=pd.read_csv(r"C:\Users\user\Downloads\16_Sleep_health_and_lifestyle_dataset.
df.fillna(0,inplace=True)
df
```

Out[5]:

	Person ID	Gender	Age	Occupation	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	BMI Category	Blood Pressure
0	1	Male	27	Software Engineer	6.1	6	42	6	Overweight	126/
1	2	Male	28	Doctor	6.2	6	60	8	Normal	125/
2	3	Male	28	Doctor	6.2	6	60	8	Normal	125/
3	4	Male	28	Sales Representative	5.9	4	30	8	Obese	140/
4	5	Male	28	Sales Representative	5.9	4	30	8	Obese	140/
...
369	370	Female	59	Nurse	8.1	9	75	3	Overweight	140/
370	371	Female	59	Nurse	8.0	9	75	3	Overweight	140/
371	372	Female	59	Nurse	8.1	9	75	3	Overweight	140/
372	373	Female	59	Nurse	8.1	9	75	3	Overweight	140/
373	374	Female	59	Nurse	8.1	9	75	3	Overweight	140/

374 rows × 13 columns



```
In [6]: df.head()
```

Out[6]:

	Person ID	Gender	Age	Occupation	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	BMI Category	Blood Pressure
0	1	Male	27	Software Engineer	6.1	6	42	6	Overweight	126/83
1	2	Male	28	Doctor	6.2	6	60	8	Normal	125/80
2	3	Male	28	Doctor	6.2	6	60	8	Normal	125/80
3	4	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90
4	5	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90

```
In [7]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 374 entries, 0 to 373
Data columns (total 13 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Person ID                            374 non-null    int64
1   Gender                               374 non-null    object
2   Age                                   374 non-null    int64
3   Occupation                           374 non-null    object
4   Sleep Duration                       374 non-null    float64
5   Quality of Sleep                     374 non-null    int64
6   Physical Activity Level              374 non-null    int64
7   Stress Level                         374 non-null    int64
8   BMI Category                         374 non-null    object
9   Blood Pressure                       374 non-null    object
10  Heart Rate                           374 non-null    int64
11  Daily Steps                          374 non-null    int64
12  Sleep Disorder                       374 non-null    object
dtypes: float64(1), int64(7), object(5)
memory usage: 38.1+ KB
```

```
In [8]: import seaborn as sns
```

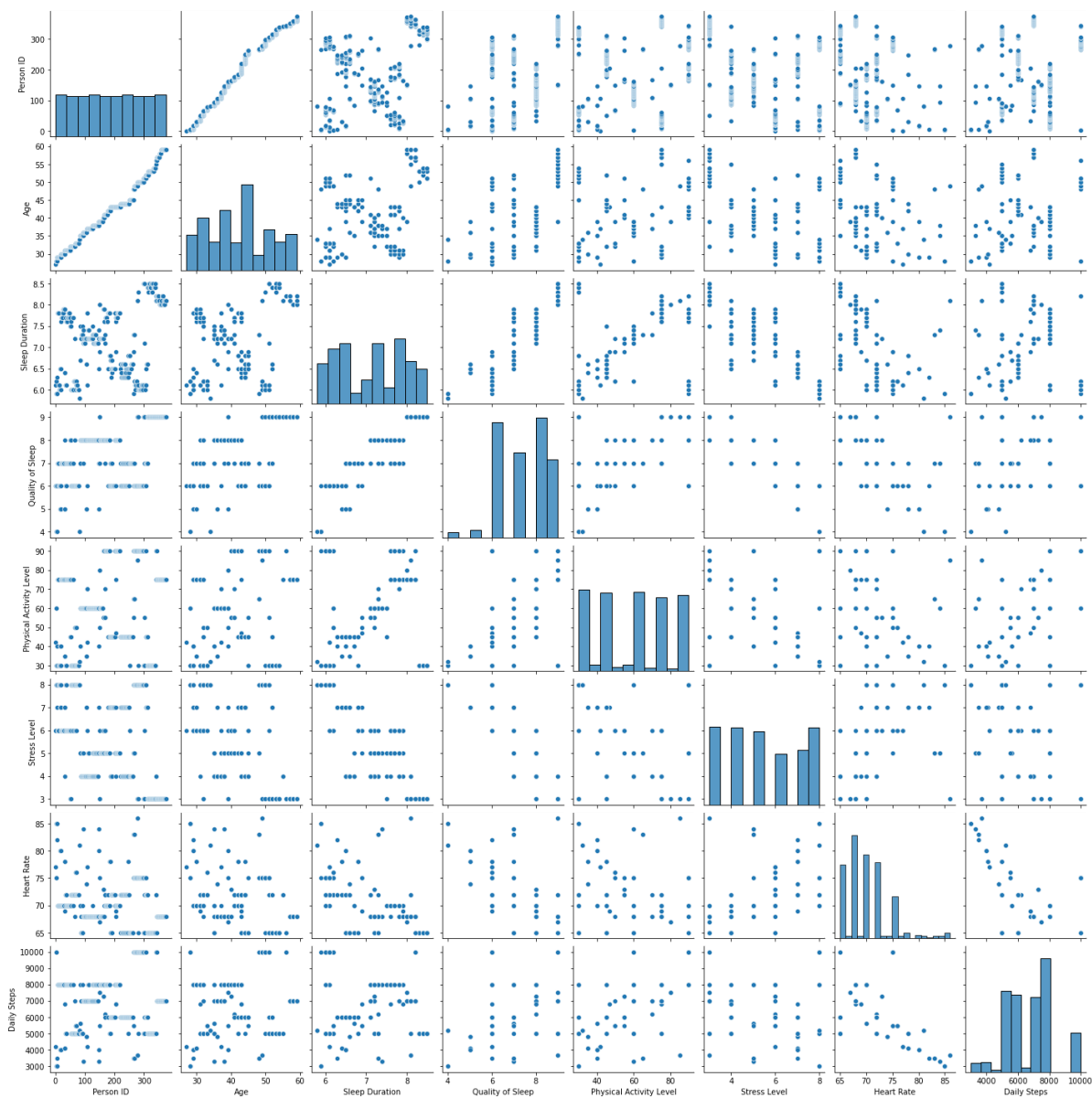
```
In [9]: df.describe()
```

Out[9]:

	Person ID	Age	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	Heart Rate	Da
count	374.000000	374.000000	374.000000	374.000000	374.000000	374.000000	374.000000	37
mean	187.500000	42.184492	7.132086	7.312834	59.171123	5.385027	70.165775	681
std	108.108742	8.673133	0.795657	1.196956	20.830804	1.774526	4.135676	161
min	1.000000	27.000000	5.800000	4.000000	30.000000	3.000000	65.000000	300
25%	94.250000	35.250000	6.400000	6.000000	45.000000	4.000000	68.000000	560
50%	187.500000	43.000000	7.200000	7.000000	60.000000	5.000000	70.000000	700
75%	280.750000	50.000000	7.800000	8.000000	75.000000	7.000000	72.000000	800
max	374.000000	59.000000	8.500000	9.000000	90.000000	8.000000	86.000000	1000

```
sns.pairplot(df)
```

```
Out[10]: <seaborn.axisgrid.PairGrid at 0x1a818136c70>
```

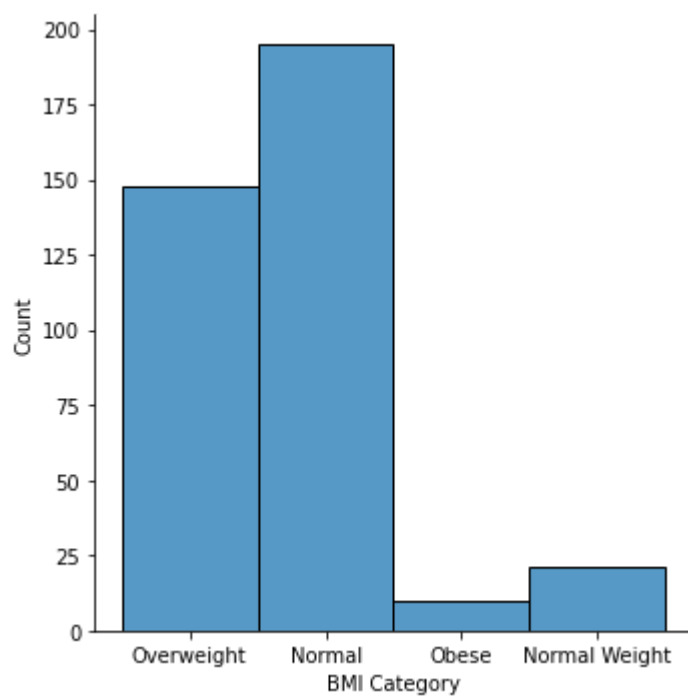


```
In [11]: df1=df.drop(['Stress Level'],axis=1)
df1
df1=df1.drop(df1.index[1537:])
df1.isna().sum()
```

```
Out[11]: Person ID          0
Gender          0
Age            0
Occupation      0
Sleep Duration  0
Quality of Sleep 0
Physical Activity Level 0
BMI Category    0
Blood Pressure  0
Heart Rate      0
Daily Steps     0
Sleep Disorder  0
dtype: int64
```

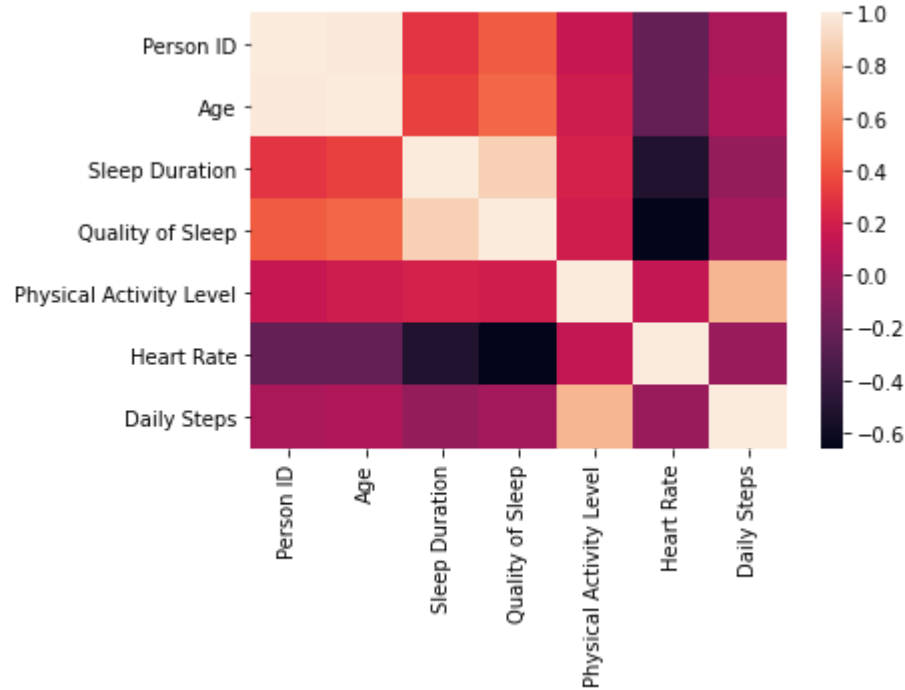
```
In [13]: sns.displot(df['BMI Category'])
```

```
Out[13]: <seaborn.axisgrid.FacetGrid at 0x1a81b7e9a90>
```



```
In [14]: sns.heatmap(df1.corr())
```

```
Out[14]: <AxesSubplot:>
```



```
In [15]: from sklearn.model_selection import train_test_split  
from sklearn.linear_model import LinearRegression
```

```
In [16]: df1.isna().sum()
```

```
Out[16]: Person ID      0  
Gender      0  
Age         0  
Occupation  0  
Sleep Duration  0  
Quality of Sleep  0  
Physical Activity Level  0  
BMI Category  0  
Blood Pressure  0  
Heart Rate   0  
Daily Steps  0  
Sleep Disorder  0  
dtype: int64
```

```
In [22]: y=df1['Age']
x=df1.drop(['Gender','BMI Category','Sleep Disorder','Occupation','Blood Pressure'])
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
print(x_train)
```

	Person ID	Age	Sleep Duration	Quality of Sleep	\
336	337	54	8.4	9	
10	11	29	6.1	6	
131	132	38	7.3	8	
228	229	44	6.6	7	
344	345	57	8.2	9	
..	
152	153	39	7.2	8	
351	352	57	8.1	9	
339	340	55	8.1	9	
119	120	37	7.2	8	
187	188	43	6.3	6	

	Physical Activity Level	Heart Rate	Daily Steps
336	30	65	5000
10	30	70	8000
131	60	68	8000
228	45	65	6000
344	75	68	7000
..
152	60	68	8000
351	75	68	7000
339	75	72	5000
119	60	68	7000
187	45	72	6000

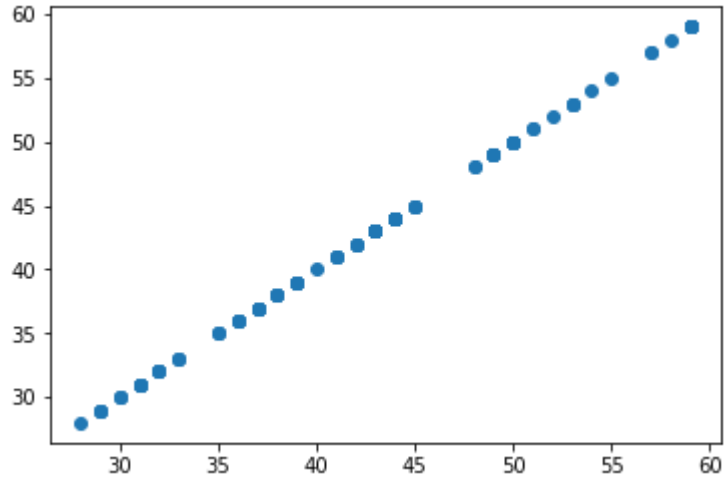
[261 rows x 7 columns]

```
In [23]: model=LinearRegression()
model.fit(x_train,y_train)
model.intercept_
```

Out[23]: -2.0605739337042905e-13

```
In [24]: prediction=model.predict(x_test)
plt.scatter(y_test,prediction)
```

```
Out[24]: <matplotlib.collections.PathCollection at 0x1a81d9e5550>
```



```
In [25]: model.score(x_test,y_test)
```

```
Out[25]: 1.0
```

```
In [26]: from sklearn.linear_model import Ridge,Lasso
```

```
In [27]: rr=Ridge(alpha=10)
rr.fit(x_train,y_train)
```

```
Out[27]: Ridge(alpha=10)
```

```
In [28]: rr.score(x_test,y_test)
```

```
Out[28]: 0.9999788606864037
```

```
In [29]: la =Lasso(alpha=10)
la.fit(x_train,y_train)
```

```
Out[29]: Lasso(alpha=10)
```

```
In [30]: la.score(x_test,y_test)
```

```
Out[30]: 0.9793081412531147
```

```
In [ ]:
```