```python
In [1]:  import pandas as pd
         import numpy as np
         import seaborn as sns
         import matplotlib.pyplot as plt
         import plotly.express as px
         %matplotlib inline
```

```python
In [2]:  data = pd.read_csv("Sleep_health_and_lifestyle_dataset.csv")
```

```python
In [3]:  data
```

Out[3]:

	Person ID	Gender	Age	Occupation	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	BMI Category	Blood Pressure	Heart Rate	Daily Steps	Sleep Disorder
0	1	Male	27	Software Engineer	6.1	6	42	6	Overweight	126/83	77	4200	None
1	2	Male	28	Doctor	6.2	6	60	8	Normal	125/80	75	10000	None
2	3	Male	28	Doctor	6.2	6	60	8	Normal	125/80	75	10000	None
3	4	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90	85	3000	Sleep Apnea
4	5	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90	85	3000	Sleep Apnea
...	...	...	...	...	...	...	...	...	...	...	...	...	...
369	370	Female	59	Nurse	8.1	9	75	3	Overweight	140/95	68	7000	Sleep Apnea
370	371	Female	59	Nurse	8.0	9	75	3	Overweight	140/95	68	7000	Sleep Apnea
371	372	Female	59	Nurse	8.1	9	75	3	Overweight	140/95	68	7000	Sleep Apnea
372	373	Female	59	Nurse	8.1	9	75	3	Overweight	140/95	68	7000	Sleep Apnea
373	374	Female	59	Nurse	8.1	9	75	3	Overweight	140/95	68	7000	Sleep Apnea

374 rows × 13 columns

```python
In [4]:  data.head()
```

Out[4]:

	Person ID	Gender	Age	Occupation	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	BMI Category	Blood Pressure	Heart Rate	Daily Steps	Sleep Disorder
0	1	Male	27	Software Engineer	6.1	6	42	6	Overweight	126/83	77	4200	None
1	2	Male	28	Doctor	6.2	6	60	8	Normal	125/80	75	10000	None
2	3	Male	28	Doctor	6.2	6	60	8	Normal	125/80	75	10000	None
3	4	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90	85	3000	Sleep Apnea
4	5	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90	85	3000	Sleep Apnea

```python
In [5]:  data.shape
```

Out[5]: (374, 13)

```python
In [6]:  data.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 [7]: data['Sleep Disorder'] = data['Sleep Disorder'].fillna('None')
```

```
In [8]: data.head()
```

Out[8]:

	Person ID	Gender	Age	Occupation	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	BMI Category	Blood Pressure	Heart Rate	Daily Steps	Sleep Disorder
0	1	Male	27	Software Engineer	6.1	6	42	6	Overweight	126/83	77	4200	None
1	2	Male	28	Doctor	6.2	6	60	8	Normal	125/80	75	10000	None
2	3	Male	28	Doctor	6.2	6	60	8	Normal	125/80	75	10000	None
3	4	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90	85	3000	Sleep Apnea
4	5	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90	85	3000	Sleep Apnea

```
In [9]: data.isnull().sum()
```

```
Out[9]: Person ID                0
        Gender                   0
        Age                      0
        Occupation               0
        Sleep Duration           0
        Quality of Sleep         0
        Physical Activity Level  0
        Stress Level             0
        BMI Category             0
        Blood Pressure           0
        Heart Rate               0
        Daily Steps              0
        Sleep Disorder           0
        dtype: int64
```

```
In [10]: data.duplicated().sum()
```

```
Out[10]: 0
```

```
In [11]: data.drop_duplicates()
```

Out[11]:

	Person ID	Gender	Age	Occupation	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	BMI Category	Blood Pressure	Heart Rate	Daily Steps	Sleep Disorder
0	1	Male	27	Software Engineer	6.1	6	42	6	Overweight	126/83	77	4200	None
1	2	Male	28	Doctor	6.2	6	60	8	Normal	125/80	75	10000	None
2	3	Male	28	Doctor	6.2	6	60	8	Normal	125/80	75	10000	None
3	4	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90	85	3000	Sleep Apnea
4	5	Male	28	Sales Representative	5.9	4	30	8	Obese	140/90	85	3000	Sleep Apnea
...	...	...	...	...	...	...	...	...	...	...	...	...	...
369	370	Female	59	Nurse	8.1	9	75	3	Overweight	140/95	68	7000	Sleep Apnea
370	371	Female	59	Nurse	8.0	9	75	3	Overweight	140/95	68	7000	Sleep Apnea
371	372	Female	59	Nurse	8.1	9	75	3	Overweight	140/95	68	7000	Sleep Apnea
372	373	Female	59	Nurse	8.1	9	75	3	Overweight	140/95	68	7000	Sleep Apnea
373	374	Female	59	Nurse	8.1	9	75	3	Overweight	140/95	68	7000	Sleep Apnea

374 rows × 13 columns

```
In [12]: data.describe()
```

	Person ID	Age	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	Heart Rate	Daily Steps
count	374.000000	374.000000	374.000000	374.000000	374.000000	374.000000	374.000000	374.000000
mean	187.500000	42.184492	7.132086	7.312834	59.171123	5.385027	70.165775	6816.844920
std	108.108742	8.673133	0.795657	1.196956	20.830804	1.774526	4.135676	1617.915679
min	1.000000	27.000000	5.800000	4.000000	30.000000	3.000000	65.000000	3000.000000
25%	94.250000	35.250000	6.400000	6.000000	45.000000	4.000000	68.000000	5600.000000
50%	187.500000	43.000000	7.200000	7.000000	60.000000	5.000000	70.000000	7000.000000
75%	280.750000	50.000000	7.800000	8.000000	75.000000	7.000000	72.000000	8000.000000
max	374.000000	59.000000	8.500000	9.000000	90.000000	8.000000	86.000000	10000.000000

In [13]:
```python
data.columns
```

Out[13]:
```
Index(['Person ID', 'Gender', 'Age', 'Occupation', 'Sleep Duration',
       'Quality of Sleep', 'Physical Activity Level', 'Stress Level',
       'BMI Category', 'Blood Pressure', 'Heart Rate', 'Daily Steps',
       'Sleep Disorder'],
      dtype='object')
```

In [14]:
```python
data['Gender'].unique()
```

Out[14]:
```
array(['Male', 'Female'], dtype=object)
```

In [15]:
```python
gender_count=data['Gender'].value_counts().reset_index()
gender_count
```

Out[15]:

	index	Gender
0	Male	189
1	Female	185

In [16]:
```python
data['Age'].describe()
```

Out[16]:
```
count    374.000000
mean      42.184492
std        8.673133
min       27.000000
25%       35.250000
50%       43.000000
75%       50.000000
max       59.000000
Name: Age, dtype: float64
```

In [17]:
```python
age_count=data['Age'].value_counts().reset_index()
age_count
```

	index	Age
0	43	34
1	44	30
2	37	20
3	38	20
4	50	20
5	31	18
6	32	17
7	53	17
8	59	16
9	39	15
10	45	14
11	33	13
12	30	13
13	29	13
14	35	12
15	36	12
16	41	12
17	49	11
18	57	9
19	52	9
20	42	9
21	51	8
22	54	7
23	58	6
24	28	5
25	40	4
26	48	3
27	55	2
28	56	2
29	34	2
30	27	1

```python
data['Occupation'].unique()
```

```
array(['Software Engineer', 'Doctor', 'Sales Representative', 'Teacher',
       'Nurse', 'Engineer', 'Accountant', 'Scientist', 'Lawyer',
       'Salesperson', 'Manager'], dtype=object)
```

```python
Occupation_count=data['Occupation'].value_counts().reset_index()
Occupation_count
```

	index	Occupation
0	Nurse	73
1	Doctor	71
2	Engineer	63
3	Lawyer	47
4	Teacher	40
5	Accountant	37
6	Salesperson	32
7	Software Engineer	4
8	Scientist	4
9	Sales Representative	2
10	Manager	1

```python
data['Sleep Duration'].describe()
```

```
Out[20]: count    374.000000
         mean       7.132086
         std        0.795657
         min        5.800000
         25%        6.400000
         50%        7.200000
         75%        7.800000
         max        8.500000
         Name: Sleep Duration, dtype: float64
```

```
In [21]: Sleep_Duration_count=data['Sleep Duration'].value_counts().reset_index()
         Sleep_Duration_count
```

Out[21]:
	index	Sleep Duration
0	7.2	36
1	6.0	31
2	7.8	28
3	6.5	26
4	6.1	25
5	7.7	24
6	6.6	20
7	7.1	19
8	8.1	15
9	7.3	14
10	8.4	14
11	6.3	13
12	8.5	13
13	8.0	13
14	6.2	12
15	8.2	11
16	7.6	10
17	6.4	9
18	7.9	7
19	7.4	5
20	6.7	5
21	7.5	5
22	6.8	5
23	8.3	5
24	5.9	4
25	6.9	3
26	5.8	2

```
In [22]: data['Quality of Sleep'].unique()
```

```
Out[22]: array([6, 4, 7, 5, 8, 9], dtype=int64)
```

```
In [23]: Quality_of_Sleep_count=data['Quality of Sleep'].value_counts().reset_index()
         Quality_of_Sleep_count
```

Out[23]:
	index	Quality of Sleep
0	8	109
1	6	105
2	7	77
3	9	71
4	5	7
5	4	5

```
In [24]: data['Physical Activity Level'].describe()
```

```
Out[24]:    count     374.000000
            mean       59.171123
            std        20.830804
            min        30.000000
            25%        45.000000
            50%        60.000000
            75%        75.000000
            max        90.000000
            Name: Physical Activity Level, dtype: float64
```

In [25]:
```
Physical_Activity_Level=data['Physical Activity Level'].value_counts().reset_index()
Physical_Activity_Level
```

Out[25]:

	index	Physical Activity Level
0	60	70
1	30	68
2	45	68
3	75	67
4	90	67
5	40	6
6	55	6
7	35	4
8	50	4
9	70	3
10	42	2
11	32	2
12	80	2
13	65	2
14	85	2
15	47	1

In [26]:
```
data['Stress Level'].unique()
```

Out[26]:
```
array([6, 8, 7, 4, 3, 5], dtype=int64)
```

In [27]:
```
Stress_Level_counts=data['Stress Level'].value_counts().reset_index()
Stress_Level_counts
```

Out[27]:

	index	Stress Level
0	3	71
1	8	70
2	4	70
3	5	67
4	7	50
5	6	46

In [28]:
```
data['BMI Category'].unique()
```

Out[28]:
```
array(['Overweight', 'Normal', 'Obese', 'Normal Weight'], dtype=object)
```

In [29]:
```
data['BMI Category']=data['BMI Category'].replace({'Normal':'Normal Weight'})
```

In [30]:
```
BMI_Category_count=data['BMI Category'].value_counts().reset_index()
BMI_Category_count
```

Out[30]:

	index	BMI Category
0	Normal Weight	216
1	Overweight	148
2	Obese	10

In [31]:
```
data['Blood Pressure'].unique()
```

Out[31]:
```
array(['126/83', '125/80', '140/90', '120/80', '132/87', '130/86',
       '117/76', '118/76', '128/85', '131/86', '128/84', '115/75',
       '135/88', '129/84', '130/85', '115/78', '119/77', '121/79',
       '125/82', '135/90', '122/80', '142/92', '140/95', '139/91',
       '118/75'], dtype=object)
```

In [32]:
```
Blood Pressure count=data['Blood Pressure'].value counts().reset index()
```

```
Blood_Pressure_count=data['Blood Pressure'].value_counts().reset_index()
Blood_Pressure_count
```

Out[32]:

	index	Blood Pressure
0	130/85	99
1	140/95	65
2	125/80	65
3	120/80	45
4	115/75	32
5	135/90	27
6	140/90	4
7	125/82	4
8	132/87	3
9	128/85	3
10	126/83	2
11	115/78	2
12	139/91	2
13	142/92	2
14	119/77	2
15	135/88	2
16	129/84	2
17	128/84	2
18	131/86	2
19	117/76	2
20	130/86	2
21	118/75	2
22	121/79	1
23	122/80	1
24	118/76	1

In [33]:
```
Heart_Rate_count=data['Heart Rate'].value_counts().reset_index()
Heart_Rate_count
```

Out[33]:

	index	Heart Rate
0	68	94
1	70	76
2	72	69
3	65	67
4	75	36
5	78	5
6	85	3
7	80	3
8	84	2
9	83	2
10	73	2
11	67	2
12	74	2
13	77	2
14	81	2
15	76	2
16	69	2
17	86	2
18	82	1

In [34]:
```
data['Daily Steps'].describe()
```

```
Out[34]:  count      374.000000
          mean      6816.844920
          std       1617.915679
          min       3000.000000
          25%       5600.000000
          50%       7000.000000
          75%       8000.000000
          max      10000.000000
          Name: Daily Steps, dtype: float64
```

In [35]:
```python
Daily_Steps_count5=data['Daily Steps'].value_counts().reset_index().head()

Daily_Steps_count5
```

Out[35]:

	index	Daily Steps
0	8000	101
1	6000	68
2	5000	68
3	7000	66
4	10000	36

In [36]:
```python
data['Sleep Disorder'].unique()
```

Out[36]: array(['None', 'Sleep Apnea', 'Insomnia'], dtype=object)

In [37]:
```python
Sleep_Disorder_count=data['Sleep Disorder'].value_counts().reset_index()
Sleep_Disorder_count
```

Out[37]:

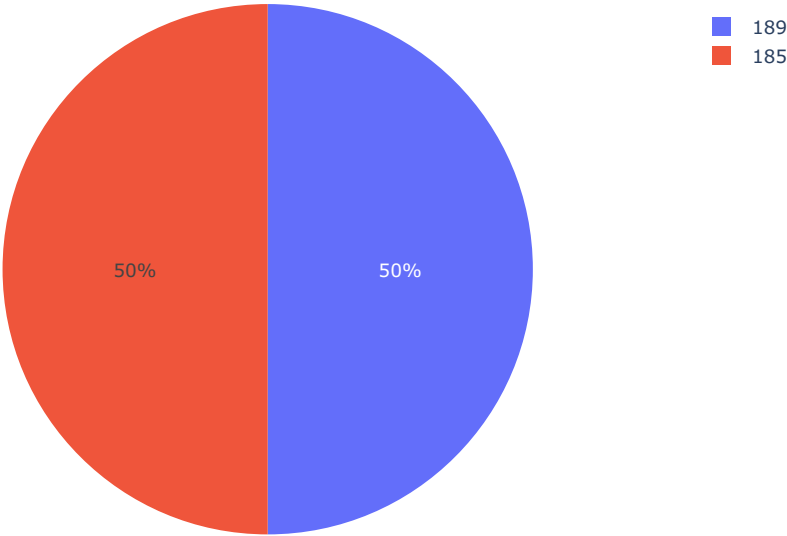|	index	Sleep Disorder
0	None	219
1	Sleep Apnea	78
2	Insomnia	77

In [ ]:

In [38]:
```python
import matplotlib.pyplot as plt
```

In [39]:
```python
fig = px.pie(gender_count, names='Gender',title='Each  Gender and it count  ')
fig.show()
```
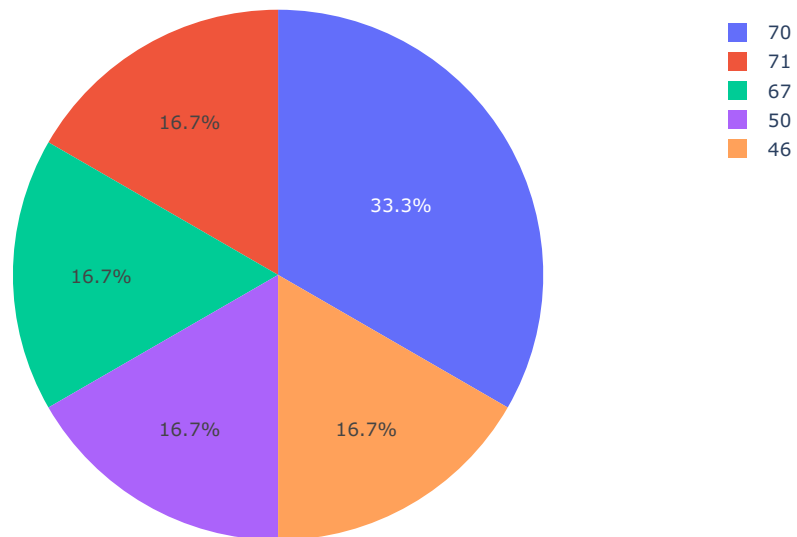
Each Gender and it count

189
185

In [40]:
```python
fig = px.pie(Stress_Level_counts ,names='Stress Level',title=" Stress Level")
fig.show()
```

Stress Level

Legend:
- 70
- 71
- 67
- 50
- 46

Pie chart values: 33.3%, 16.7%, 16.7%, 16.7%, 16.7%

In [41]:
```python
fig=px.bar(age_count,title='The Age and The Number of peapol in The same Age')

fig.show()
```

In [42]:
```python
fig=px.bar(Heart_Rate_count,title="the Heart Rate and each count")
fig.show()
```

Stress Level

```
In [43]: fig=px.bar(age_count,title='The Age and The Number of peapol in The same Age')
         fig.show()
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: