Read the file

In [2]: `import pandas as pd`

In [3]:
```
df=pd.read_excel("D:\others\Python new.xlsx")
df
```

Out[3]:

	No_of_Days	Date	Start_time	End_time	Breakfast	Office_Departure_time	Office_Departure_time_1	work	W1	Break	...	Mus
0	Day	15/2/2023	07:30:00	23:00:00	30	20.0	NaN	30.0	NaN	15.0	...	10
1	Day	16/2/2023	08:04:00	22:40:00	25	25.0	NaN	1.0	NaN	10.0	...	15
2	Day	17/2/2023	08:00:00	22:30:00	20	20.0	NaN	1.0	NaN	10.0	...	10
3	Day	18/2/2023	10:00:00	23:00:00	25	NaN	Leave	NaN	Leave	NaN	...	20
4	Day	19/2/2023	08:30:00	12:00:00	30	NaN	Leave	NaN	Leave	NaN	...	20
5	Day	20/2/2023	07:40:00	22:45:00	15	25.0	NaN	1.0	NaN	5.0	...	10
6	Day	21/2/2023	07:45:00	23:00:00	15	15.0	NaN	1.0	NaN	15.0	...	15
7	Day	22/2/2023	07:50:00	22:30:00	15	15.0	NaN	130.0	NaN	5.0	...	30
8	Day	23/2/2023	08:00:00	23:00:00	20	21.0	NaN	1.0	NaN	NaN	...	10
9	Day	24/2/2023	07:50:00	23:00:00	15	10.0	NaN	1.0	NaN	5.0	...	10
10	Day	25/2/2023	08:00:00	23:00:00	10	NaN	Outing	1.0	NaN	NaN	...	5
11	Day	26/2/2023	08:30:00	00:00:00	15	NaN	Leave	1.0	NaN	NaN	...	15
12	Day	27/2/2023	07:50:00	23:45:00	20	15.0	NaN	1.0	NaN	10.0	...	10
13	Day	28/2/2023	08:00:00	23:15:00	20	15.0	NaN	1.0	NaN	10.0	...	10
14	Day	01/03/2023	08:00:00	23:30:00	15	20.0	NaN	1.0	NaN	10.0	...	10
15	Day	02/02/2023	08:10:00	23:30:00	20	20.0	NaN	1.0	NaN	15.0	...	10
16	Day	03/03/2023	08:00:00	12:00:00	20	25.0	NaN	1.0	NaN	15.0	...	30
17	Day	04/03/2023	08:30:00	10:30 PM	30	NaN	Leave	NaN	Personal Work	NaN	...	15
18	Day	05/03/2023	08:30:00	23:30:00	1	NaN	Leave	NaN	Leave	NaN	...	Na
19	Day	06/03/2023	08:00:00	23:30:00	20	30.0	NaN	1.0	NaN	10.0	...	10
20	Day	07/03/2023	08:00:00	00:30:00	25	30.0	NaN	1.0	NaN	15.0	...	15
21	Day	08/03/2023	08:00:00	00:30:00	20	20.0	NaN	230.0	NaN	10.0	...	15
22	Day	09/03/2023	08:00:00	22:30:00	15	20.0	NaN	2.0	NaN	15.0	...	10
23	Day	10/03/2023	08:00:00	22:30:00	15	15.0	NaN	1.0	NaN	15.0	...	Na
24	Day	11/03/2023	08:00:00	22:30:00	20	NaN	Leave	NaN	Leave	NaN	...	20
25	Day	12/03/2023	07:40:00	22:30:00	20	NaN	Leave	NaN	Leave	NaN	...	20
26	Day	13/03/2023	08:00:00	23:00:00	30	15.0	NaN	2.0	NaN	15.0	...	15
27	Day	14/03/2023	08:00:00	23:00:00	30	5.0	NaN	230.0	NaN	5.0	...	10
28	Day	15/03/2023	08:00:00	23:00:00	30	15.0	NaN	2.0	NaN	15.0	...	10
29	Day	16/03/2023	07:30:00	23:30:00	5	15.0	NaN	2.0	NaN	15.0	...	5

30 rows × 37 columns

Data Cleaning

In [4]: `df.drop_duplicates()`

	No_of_Days	Date	Start_time	End_time	Breakfast	Office_Departure_time	Office_Departure_time_1	work	W1	Break	...	Mus
0	Day	15/2/2023	07:30:00	23:00:00	30	20.0	NaN	30.0	NaN	15.0	...	10
1	Day	16/2/2023	08:04:00	22:40:00	25	25.0	NaN	1.0	NaN	10.0	...	15
2	Day	17/2/2023	08:00:00	22:30:00	20	20.0	NaN	1.0	NaN	10.0	...	10
3	Day	18/2/2023	10:00:00	23:00:00	25	NaN	Leave	NaN	Leave	NaN	...	20
4	Day	19/2/2023	08:30:00	12:00:00	30	NaN	Leave	NaN	Leave	NaN	...	20
5	Day	20/2/2023	07:40:00	22:45:00	15	25.0	NaN	1.0	NaN	5.0	...	10
6	Day	21/2/2023	07:45:00	23:00:00	15	15.0	NaN	1.0	NaN	15.0	...	15
7	Day	22/2/2023	07:50:00	22:30:00	15	15.0	NaN	130.0	NaN	5.0	...	30
8	Day	23/2/2023	08:00:00	23:00:00	20	21.0	NaN	1.0	NaN	NaN	...	10
9	Day	24/2/2023	07:50:00	23:00:00	15	10.0	NaN	1.0	NaN	5.0	...	10
10	Day	25/2/2023	08:00:00	23:00:00	10	NaN	Outing	1.0	NaN	NaN	...	5
11	Day	26/2/2023	08:30:00	00:00:00	15	NaN	Leave	1.0	NaN	NaN	...	15
12	Day	27/2/2023	07:50:00	23:45:00	20	15.0	NaN	1.0	NaN	10.0	...	10
13	Day	28/2/2023	08:00:00	23:15:00	20	15.0	NaN	1.0	NaN	10.0	...	10
14	Day	01/03/2023	08:00:00	23:30:00	15	20.0	NaN	1.0	NaN	10.0	...	10
15	Day	02/02/2023	08:10:00	23:30:00	20	20.0	NaN	1.0	NaN	15.0	...	10
16	Day	03/03/2023	08:00:00	12:00:00	20	25.0	NaN	1.0	NaN	15.0	...	30
17	Day	04/03/2023	08:30:00	10:30 PM	30	NaN	Leave	NaN	Personal Work	NaN	...	15
18	Day	05/03/2023	08:30:00	23:30:00	1	NaN	Leave	NaN	Leave	NaN	...	Na
19	Day	06/03/2023	08:00:00	23:30:00	20	30.0	NaN	1.0	NaN	10.0	...	10
20	Day	07/03/2023	08:00:00	00:30:00	25	30.0	NaN	1.0	NaN	15.0	...	15
21	Day	08/03/2023	08:00:00	00:30:00	20	20.0	NaN	230.0	NaN	10.0	...	15
22	Day	09/03/2023	08:00:00	22:30:00	15	20.0	NaN	2.0	NaN	15.0	...	10
23	Day	10/03/2023	08:00:00	22:30:00	15	15.0	NaN	1.0	NaN	15.0	...	Na
24	Day	11/03/2023	08:00:00	22:30:00	20	NaN	Leave	NaN	Leave	NaN	...	20
25	Day	12/03/2023	07:40:00	22:30:00	20	NaN	Leave	NaN	Leave	NaN	...	20
26	Day	13/03/2023	08:00:00	23:00:00	30	15.0	NaN	2.0	NaN	15.0	...	15
27	Day	14/03/2023	08:00:00	23:00:00	30	5.0	NaN	230.0	NaN	5.0	...	10
28	Day	15/03/2023	08:00:00	23:00:00	30	15.0	NaN	2.0	NaN	15.0	...	10
29	Day	16/03/2023	07:30:00	23:30:00	5	15.0	NaN	2.0	NaN	15.0	...	5

30 rows × 37 columns

```python
df.fillna("0", inplace = True)
df
```

Out[5]:

	No_of_Days	Date	Start_time	End_time	Breakfast	Office_Departure_time	Office_Departure_time_1	work	W1	Break	...	Mus
0	Day	15/2/2023	07:30:00	23:00:00	30	20.0	0	30.0	0	15.0	...	10
1	Day	16/2/2023	08:04:00	22:40:00	25	25.0	0	1.0	0	10.0	...	15
2	Day	17/2/2023	08:00:00	22:30:00	20	20.0	0	1.0	0	10.0	...	10
3	Day	18/2/2023	10:00:00	23:00:00	25	0	Leave	0	Leave	0	...	20
4	Day	19/2/2023	08:30:00	12:00:00	30	0	Leave	0	Leave	0	...	20
5	Day	20/2/2023	07:40:00	22:45:00	15	25.0	0	1.0	0	5.0	...	10
6	Day	21/2/2023	07:45:00	23:00:00	15	15.0	0	1.0	0	15.0	...	15
7	Day	22/2/2023	07:50:00	22:30:00	15	15.0	0	130.0	0	5.0	...	30
8	Day	23/2/2023	08:00:00	23:00:00	20	21.0	0	1.0	0	0	...	10
9	Day	24/2/2023	07:50:00	23:00:00	15	10.0	0	1.0	0	5.0	...	10
10	Day	25/2/2023	08:00:00	23:00:00	10	0	Outing	1.0	0	0	...	5
11	Day	26/2/2023	08:30:00	00:00:00	15	0	Leave	1.0	0	0	...	15
12	Day	27/2/2023	07:50:00	23:45:00	20	15.0	0	1.0	0	10.0	...	10
13	Day	28/2/2023	08:00:00	23:15:00	20	15.0	0	1.0	0	10.0	...	10
14	Day	01/03/2023	08:00:00	23:30:00	15	20.0	0	1.0	0	10.0	...	10
15	Day	02/02/2023	08:10:00	23:30:00	20	20.0	0	1.0	0	15.0	...	10
16	Day	03/03/2023	08:00:00	12:00:00	20	25.0	0	1.0	0	15.0	...	30
17	Day	04/03/2023	08:30:00	10:30 PM	30	0	Leave	0	Personal Work	0	...	15
18	Day	05/03/2023	08:30:00	23:30:00	1	0	Leave	0	Leave	0	...	
19	Day	06/03/2023	08:00:00	23:30:00	20	30.0	0	1.0	0	10.0	...	10
20	Day	07/03/2023	08:00:00	00:30:00	25	30.0	0	1.0	0	15.0	...	15
21	Day	08/03/2023	08:00:00	00:30:00	20	20.0	0	230.0	0	10.0	...	15
22	Day	09/03/2023	08:00:00	22:30:00	15	20.0	0	2.0	0	15.0	...	10
23	Day	10/03/2023	08:00:00	22:30:00	15	15.0	0	1.0	0	15.0	...	
24	Day	11/03/2023	08:00:00	22:30:00	20	0	Leave	0	Leave	0	...	20
25	Day	12/03/2023	07:40:00	22:30:00	20	0	Leave	0	Leave	0	...	20
26	Day	13/03/2023	08:00:00	23:00:00	30	15.0	0	2.0	0	15.0	...	15
27	Day	14/03/2023	08:00:00	23:00:00	30	5.0	0	230.0	0	5.0	...	10
28	Day	15/03/2023	08:00:00	23:00:00	30	15.0	0	2.0	0	15.0	...	10
29	Day	16/03/2023	07:30:00	23:30:00	5	15.0	0	2.0	0	15.0	...	5

30 rows × 37 columns

In [6]:
```python
df["Breakfast"]=df["Breakfast"].astype("int")
df["Office_Departure_time"]=df["Office_Departure_time"].astype("int")
df["work"]=df["work"].astype("int")
df["Break"]=df["Break"].astype("int")
df["work2"]=df["work2"].astype("int")
df["lunch"]=df["lunch"].astype("int")
df["work3"]=df["work3"].astype("int")
df["break2"]=df["break2"].astype("int")
df["work4"]=df["work4"].astype("int")
df["Return_time"]=df["Return_time"].astype("int")
df["Fitness"]=df["Fitness"].astype("int")
df["Refresh"]=df["Refresh"].astype("int")
df["Music"]=df["Music"].astype("int")
df["reading "]=df["reading "].astype("int")
df["linkedin"]=df["linkedin"].astype("int")
df["study"]=df["study"].astype("int")
df["dinner"]=df["dinner"].astype("int")
df["Sleep"]=df["Sleep"].astype("int")
df
```

Out[6]:

	No_of_Days	Date	Start_time	End_time	Breakfast	Office_Departure_time	Office_Departure_time_1	work	W1	Break	...	Musi
0	Day	15/2/2023	07:30:00	23:00:00	30	20	0	30	0	15	...	1
1	Day	16/2/2023	08:04:00	22:40:00	25	25	0	1	0	10	...	1
2	Day	17/2/2023	08:00:00	22:30:00	20	20	0	1	0	10	...	1
3	Day	18/2/2023	10:00:00	23:00:00	25	0	Leave	0	Leave	0	...	2
4	Day	19/2/2023	08:30:00	12:00:00	30	0	Leave	0	Leave	0	...	2
5	Day	20/2/2023	07:40:00	22:45:00	15	25	0	1	0	5	...	1
6	Day	21/2/2023	07:45:00	23:00:00	15	15	0	1	0	15	...	1
7	Day	22/2/2023	07:50:00	22:30:00	15	15	0	130	0	5	...	3
8	Day	23/2/2023	08:00:00	23:00:00	20	21	0	1	0	0	...	1
9	Day	24/2/2023	07:50:00	23:00:00	15	10	0	1	0	5	...	1
10	Day	25/2/2023	08:00:00	23:00:00	10	0	Outing	1	0	0	...	
11	Day	26/2/2023	08:30:00	00:00:00	15	0	Leave	1	0	0	...	1
12	Day	27/2/2023	07:50:00	23:45:00	20	15	0	1	0	10	...	1
13	Day	28/2/2023	08:00:00	23:15:00	20	15	0	1	0	10	...	1
14	Day	01/03/2023	08:00:00	23:30:00	15	20	0	1	0	10	...	1
15	Day	02/02/2023	08:10:00	23:30:00	20	20	0	1	0	15	...	1
16	Day	03/03/2023	08:00:00	12:00:00	20	25	0	1	0	15	...	3
17	Day	04/03/2023	08:30:00	10:30 PM	30	0	Leave	0	Personal Work	0	...	1
18	Day	05/03/2023	08:30:00	23:30:00	1	0	Leave	0	Leave	0	...	
19	Day	06/03/2023	08:00:00	23:30:00	20	30	0	1	0	10	...	1
20	Day	07/03/2023	08:00:00	00:30:00	25	30	0	1	0	15	...	1
21	Day	08/03/2023	08:00:00	00:30:00	20	20	0	230	0	10	...	1
22	Day	09/03/2023	08:00:00	22:30:00	15	20	0	2	0	15	...	1
23	Day	10/03/2023	08:00:00	22:30:00	15	15	0	1	0	15	...	
24	Day	11/03/2023	08:00:00	22:30:00	20	0	Leave	0	Leave	0	...	2
25	Day	12/03/2023	07:40:00	22:30:00	20	0	Leave	0	Leave	0	...	2
26	Day	13/03/2023	08:00:00	23:00:00	30	15	0	2	0	15	...	1
27	Day	14/03/2023	08:00:00	23:00:00	30	5	0	230	0	5	...	1
28	Day	15/03/2023	08:00:00	23:00:00	30	15	0	2	0	15	...	1
29	Day	16/03/2023	07:30:00	23:30:00	5	15	0	2	0	15	...	1

30 rows × 37 columns

In [7]: `df.columns`

Out[7]:
```
Index(['No_of_Days', 'Date', 'Start_time', 'End_time', 'Breakfast',
       'Office_Departure_time', 'Office_Departure_time_1', 'work', 'W1',
       'Break', 'B1', 'work2', 'W2', 'lunch', 'Lunch1', 'work3', 'W3',
       'break2', 'B2', 'work4', 'W4', 'Return_time', 'Return_time1', 'Fitness',
       'Fitness1', 'Refresh', 'Refresh1', 'Music', 'Music1', 'reading ',
       'reading 1', 'linkedin', 'linkedin1', 'study', 'study1', 'dinner',
       'Sleep'],
      dtype='object')
```

In [8]: `df`

	No_of_Days	Date	Start_time	End_time	Breakfast	Office_Departure_time	Office_Departure_time_1	work	W1	Break	...	Musi
0	Day	15/2/2023	07:30:00	23:00:00	30	20	0	30	0	15	...	1
1	Day	16/2/2023	08:04:00	22:40:00	25	25	0	1	0	10	...	1
2	Day	17/2/2023	08:00:00	22:30:00	20	20	0	1	0	10	...	1
3	Day	18/2/2023	10:00:00	23:00:00	25	0	Leave	0	Leave	0	...	2
4	Day	19/2/2023	08:30:00	12:00:00	30	0	Leave	0	Leave	0	...	2
5	Day	20/2/2023	07:40:00	22:45:00	15	25	0	1	0	5	...	1
6	Day	21/2/2023	07:45:00	23:00:00	15	15	0	1	0	15	...	1
7	Day	22/2/2023	07:50:00	22:30:00	15	15	0	130	0	5	...	3
8	Day	23/2/2023	08:00:00	23:00:00	20	21	0	1	0	0	...	1
9	Day	24/2/2023	07:50:00	23:00:00	15	10	0	1	0	5	...	1
10	Day	25/2/2023	08:00:00	23:00:00	10	0	Outing	1	0	0	...	
11	Day	26/2/2023	08:30:00	00:00:00	15	0	Leave	1	0	0	...	1
12	Day	27/2/2023	07:50:00	23:45:00	20	15	0	1	0	10	...	1
13	Day	28/2/2023	08:00:00	23:15:00	20	15	0	1	0	10	...	1
14	Day	01/03/2023	08:00:00	23:30:00	15	20	0	1	0	10	...	1
15	Day	02/02/2023	08:10:00	23:30:00	20	20	0	1	0	15	...	1
16	Day	03/03/2023	08:00:00	12:00:00	20	25	0	1	0	15	...	3
17	Day	04/03/2023	08:30:00	10:30 PM	30	0	Leave	0	Personal Work	0	...	1
18	Day	05/03/2023	08:30:00	23:30:00	1	0	Leave	0	Leave	0	...	
19	Day	06/03/2023	08:00:00	23:30:00	20	30	0	1	0	10	...	1
20	Day	07/03/2023	08:00:00	00:30:00	25	30	0	1	0	15	...	1
21	Day	08/03/2023	08:00:00	00:30:00	20	20	0	230	0	10	...	1
22	Day	09/03/2023	08:00:00	22:30:00	15	20	0	2	0	15	...	1
23	Day	10/03/2023	08:00:00	22:30:00	15	15	0	1	0	15	...	
24	Day	11/03/2023	08:00:00	22:30:00	20	0	Leave	0	Leave	0	...	2
25	Day	12/03/2023	07:40:00	22:30:00	20	0	Leave	0	Leave	0	...	2
26	Day	13/03/2023	08:00:00	23:00:00	30	15	0	2	0	15	...	1
27	Day	14/03/2023	08:00:00	23:00:00	30	5	0	230	0	5	...	1
28	Day	15/03/2023	08:00:00	23:00:00	30	15	0	2	0	15	...	1
29	Day	16/03/2023	07:30:00	23:30:00	5	15	0	2	0	15	...	1

30 rows × 37 columns

Analysis

```
In [9]:   ## Copmare the how many hours I spend for work

          name=["work","work2","work3","work4"]
          df[name].count()
```

```
Out[9]:   work     30
          work2    30
          work3    30
          work4    30
          dtype: int64
```

```
In [10]:  df[name].sum()
```

```
Out[10]:  work     644
          work2     44
          work3     40
          work4   1224
          dtype: int64
```

```
In [11]:  name1=["work","work2","work3","work4"]
          df[name1].mean()
```

```
Out[11]:  work     21.466667
          work2     1.466667
          work3     1.333333
          work4    40.800000
          dtype: float64
```

```
In [12]:  name2=["work","work2","work3","work4"]
```

```
In [12]: name2=["work","work2","work3","work4"]
         df[name2].median()
```

Out[12]:
```
work     1.0
work2    2.0
work3    2.0
work4    2.0
dtype: float64
```

```
In [13]: name3=["work","work2","work3","work4"]
         df[name3].mode()
```

Out[13]:

	work	work2	work3	work4
0	1	2	2	0

```
In [14]: name2=["work","work2","work3","work4"]
         df[name2].std()
```

Out[14]:
```
work     61.534254
work2     0.899553
work3     0.922266
work4    90.638882
dtype: float64
```

```
In [15]: #Over all hours

         values=[31+44+40+45]
         print(values)
```
```
[160]
```

```
In [16]: #Average of daily hours

         V1=(160/4)
         print(V1)
```
```
40.0
```

```
In [17]: ## Copmare the how many hours I spend for study

         df["study"].count()
```
Out[17]: 30

```
In [18]: df["study"].sum()
```
Out[18]: 643

```
In [19]: ##Average of daily hours

         V2=(11.40/30)
         print(V2)
```
```
0.38
```

```
In [20]: # Correlation between the works and study

         column=["work","work2","work3","work4","study"]
         df[column].corr()
```

Out[20]:

	work	work2	work3	work4	study
work	1.000000	0.211473	0.254793	-0.103962	-0.061677
work2	0.211473	1.000000	0.886702	0.276084	0.331121
work3	0.254793	0.886702	1.000000	0.200066	0.404438
work4	-0.103962	0.276084	0.200066	1.000000	0.056918
study	-0.061677	0.331121	0.404438	0.056918	1.000000

```
In [21]: # Compare by Linkedin

         df["linkedin"].count()
```
Out[21]: 30

```
In [22]: df["linkedin"].sum()
```
Out[22]: 426

```
In [23]: ##Average of daily hours
         df["linkedin"].mean()
```
Out[23]: 14.2

```
In [24]: print(df[["linkedin","study"]].corr())
```

```
In [24]: print(df[["linkedin", "study"]].corr())
```

```
          linkedin     study
linkedin  1.000000 -0.042884
study    -0.042884  1.000000
```

```
In [25]: #Reasons for work
         df[["W1","W2","W3","W4"]]
```

Out[25]:

	W1	W2	W3	W4
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	Leave	Leave	Leave	Leave
4	Leave	Leave	Leave	Peronal work
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	Outing	Outing	Outing
11	0	Leave	Leave	Leave
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	Personal Work	peronal work	Leave	Leave
18	Leave	Leave	Shopping	Shopping
19	0	0	0	0
20	0	0	0	0
21	0	0	0	0
22	0	0	0	0
23	0	0	0	0
24	Leave	Leave	Leave	Peronal work
25	Leave	Shopping	Shopping	Shopping
26	0	0	0	0
27	0	0	0	0
28	0	0	Activities	Activities
29	0	0	0	0

```
In [26]: #Reasons for linkedin
         df["linkedin1"]
```

```
Out[26]:   0           0
           1           0
           2           0
           3           0
           4           0
           5           0
           6           0
           7           0
           8           0
           9           0
           10          0
           11          0
           12          0
           13          0
           14          0
           15          0
           16          0
           17          0
           18     Skipped
           19          0
           20          0
           21          0
           22          0
           23          0
           24          0
           25          0
           26          0
           27          0
           28          0
           29          0
           Name: linkedin1, dtype: object
```

In [27]:
```python
#Reasons for study
df["study1"]
```

```
Out[27]:   0           0
           1           0
           2           0
           3           0
           4     Skipped
           5           0
           6           0
           7           0
           8           0
           9           0
           10     Skipped
           11          0
           12          0
           13          0
           14          0
           15          0
           16     Skipped
           17          0
           18     Skipped
           19          0
           20          0
           21          0
           22          0
           23     Skipped
           24     Skipped
           25     Skipped
           26          0
           27     Skipped
           28     Skipped
           29          0
           Name: study1, dtype: object
```

In [28]:
```python
#Copmare by Fitness

df["Fitness"]
```

```
Out[28]:  0     15
          1     20
          2     15
          3     30
          4      0
          5     10
          6     15
          7      0
          8      0
          9      5
          10    10
          11     0
          12    10
          13    10
          14    10
          15     0
          16     0
          17     0
          18     0
          19    10
          20    15
          21     0
          22     0
          23    15
          24    15
          25    15
          26    15
          27    15
          28    15
          29    10
          Name: Fitness, dtype: int32
```

In [29]:
```python
df["Fitness"].sum()
```

Out[29]: 275

In [30]:
```python
#Average of daily hours
df["Fitness"].mean()
```

Out[30]: 9.166666666666666

In [31]:
```python
df["Fitness1"]
```

Out[31]:
```
0              0
1              0
2              0
3              0
4        Skipped
5              0
6              0
7        Skipped
8        Skipped
9              0
10             0
11         Leave
12             0
13             0
14             0
15       Skipped
16       Skipped
17       Skipped
18       Skipped
19             0
20             0
21       Skipped
22       Skipped
23             0
24             0
25             0
26             0
27             0
28             0
29             0
Name: Fitness1, dtype: object
```

In [32]:
```python
#Copmare to Departure time by Return time
df[["Office_Departure_time","Return_time"]]
```

	Office_Departure_time	Return_time
0	20	30
1	25	40
2	20	40
3	0	0
4	0	0
5	25	20
6	15	30
7	15	30
8	21	40
9	10	15
10	0	1
11	0	0
12	15	30
13	15	2
14	20	30
15	20	20
16	25	15
17	0	0
18	0	40
19	30	30
20	30	20
21	20	20
22	20	25
23	15	25
24	0	0
25	0	2
26	15	25
27	5	30
28	15	40
29	15	0

In [33]: `df[["Office_Departure_time","Return_time"]].count()`

Out[33]:
```
Office_Departure_time    30
Return_time              30
dtype: int64
```

In [34]: `df[["Office_Departure_time","Return_time"]].sum()`

Out[34]:
```
Office_Departure_time    411
Return_time              600
dtype: int64
```

In [35]: `df[["Office_Departure_time","Return_time"]].mean()`

Out[35]:
```
Office_Departure_time    13.7
Return_time              20.0
dtype: float64
```

In [36]: `print(df[["Office_Departure_time","Return_time"]].corr())`

```
                       Office_Departure_time  Return_time
Office_Departure_time               1.000000     0.555473
Return_time                         0.555473     1.000000
```

In [37]: `df[["Office_Departure_time_1","Return_time1"]]`

Out[37]:

	Office_Departure_time_1	Return_time1
0	0	0
1	0	0
2	0	0
3	Leave	Leave
4	Leave	Peronal work
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	Outing	0
11	Leave	Leave
12	0	0
13	0	0
14	0	0
15	0	0
16	0	0
17	Leave	Leave
18	Leave	0
19	0	0
20	0	0
21	0	0
22	0	0
23	0	0
24	Leave	Leave
25	Leave	0
26	0	0
27	0	0
28	0	0
29	0	Activities

In [38]:
```python
#Copmare to Reading by Music
df[["reading ","Music"]]
```

	reading	Music
0	10	10
1	10	15
2	10	10
3	25	20
4	0	20
5	15	10
6	5	15
7	0	30
8	0	10
9	5	10
10	0	5
11	5	15
12	5	10
13	10	10
14	5	10
15	5	10
16	0	30
17	10	15
18	0	0
19	0	10
20	10	15
21	5	15
22	10	10
23	0	0
24	0	20
25	0	20
26	10	15
27	5	10
28	5	10
29	10	5

In [39]:
```python
df[["reading ","Music"]].count()
```

Out[39]:
```
reading    30
Music      30
dtype: int64
```

In [40]:
```python
df[["reading ","Music"]].sum()
```

Out[40]:
```
reading    175
Music      385
dtype: int64
```

In [41]:
```python
df[["reading ","Music"]].mean()
```

Out[41]:
```
reading     5.833333
Music      12.833333
dtype: float64
```

In [42]:
```python
print(df[["reading ","Music"]].corr())
```

```
           reading      Music
reading   1.000000  -0.018128
Music    -0.018128   1.000000
```

In [43]:
```python
df[["reading 1","Music1"]]
```

	reading 1	Music1
0	0	0
1	0	0
2	0	0
3	0	0
4	Skipped	0
5	0	0
6	0	0
7	Skipped	0
8	Skipped	0
9	0	0
10	Skipped	0
11	0	0
12	0	0
13	0	0
14	0	0
15	0	0
16	Skipped	0
17	0	0
18	Skipped	Skipped
19	Skipped	0
20	0	0
21	0	0
22	0	0
23	Skipped	Skipped
24	Skipped	0
25	Skipped	0
26	0	0
27	0	0
28	0	0
29	0	0

Visualization

In [44]:
```python
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

In [45]:
```python
import pandas as pd
df=pd.read_excel("D:\others\Python new.xlsx")
df.head()
```

Out[45]:

	No_of_Days	Date	Start_time	End_time	Breakfast	Office_Departure_time	Office_Departure_time_1	work	W1	Break	...	Music	M
0	Day	15/2/2023	07:30:00	23:00:00	30	20.0	NaN	30.0	NaN	15.0	...	10.0	
1	Day	16/2/2023	08:04:00	22:40:00	25	25.0	NaN	1.0	NaN	10.0	...	15.0	
2	Day	17/2/2023	08:00:00	22:30:00	20	20.0	NaN	1.0	NaN	10.0	...	10.0	
3	Day	18/2/2023	10:00:00	23:00:00	25	NaN	Leave	NaN	Leave	NaN	...	20.0	
4	Day	19/2/2023	08:30:00	12:00:00	30	NaN	Leave	NaN	Leave	NaN	...	20.0	

5 rows × 37 columns

In [46]:
```python
df.drop_duplicates()
```

	No_of_Days	Date	Start_time	End_time	Breakfast	Office_Departure_time	Office_Departure_time_1	work	W1	Break	...	Mus
0	Day	15/2/2023	07:30:00	23:00:00	30	20.0	NaN	30.0	NaN	15.0	...	10
1	Day	16/2/2023	08:04:00	22:40:00	25	25.0	NaN	1.0	NaN	10.0	...	15
2	Day	17/2/2023	08:00:00	22:30:00	20	20.0	NaN	1.0	NaN	10.0	...	10
3	Day	18/2/2023	10:00:00	23:00:00	25	NaN	Leave	NaN	Leave	NaN	...	20
4	Day	19/2/2023	08:30:00	12:00:00	30	NaN	Leave	NaN	Leave	NaN	...	20
5	Day	20/2/2023	07:40:00	22:45:00	15	25.0	NaN	1.0	NaN	5.0	...	10
6	Day	21/2/2023	07:45:00	23:00:00	15	15.0	NaN	1.0	NaN	15.0	...	15
7	Day	22/2/2023	07:50:00	22:30:00	15	15.0	NaN	130.0	NaN	5.0	...	30
8	Day	23/2/2023	08:00:00	23:00:00	20	21.0	NaN	1.0	NaN	NaN	...	10
9	Day	24/2/2023	07:50:00	23:00:00	15	10.0	NaN	1.0	NaN	5.0	...	10
10	Day	25/2/2023	08:00:00	23:00:00	10	NaN	Outing	1.0	NaN	NaN	...	5
11	Day	26/2/2023	08:30:00	00:00:00	15	NaN	Leave	1.0	NaN	NaN	...	15
12	Day	27/2/2023	07:50:00	23:45:00	20	15.0	NaN	1.0	NaN	10.0	...	10
13	Day	28/2/2023	08:00:00	23:15:00	20	15.0	NaN	1.0	NaN	10.0	...	10
14	Day	01/03/2023	08:00:00	23:30:00	15	20.0	NaN	1.0	NaN	10.0	...	10
15	Day	02/02/2023	08:10:00	23:30:00	20	20.0	NaN	1.0	NaN	15.0	...	10
16	Day	03/03/2023	08:00:00	12:00:00	20	25.0	NaN	1.0	NaN	15.0	...	30
17	Day	04/03/2023	08:30:00	10:30 PM	30	NaN	Leave	NaN	Personal Work	NaN	...	15
18	Day	05/03/2023	08:30:00	23:30:00	1	NaN	Leave	NaN	Leave	NaN	...	Na
19	Day	06/03/2023	08:00:00	23:30:00	20	30.0	NaN	1.0	NaN	10.0	...	10
20	Day	07/03/2023	08:00:00	00:30:00	25	30.0	NaN	1.0	NaN	15.0	...	15
21	Day	08/03/2023	08:00:00	00:30:00	20	20.0	NaN	230.0	NaN	10.0	...	15
22	Day	09/03/2023	08:00:00	22:30:00	15	20.0	NaN	2.0	NaN	15.0	...	10
23	Day	10/03/2023	08:00:00	22:30:00	15	15.0	NaN	1.0	NaN	15.0	...	Na
24	Day	11/03/2023	08:00:00	22:30:00	20	NaN	Leave	NaN	Leave	NaN	...	20
25	Day	12/03/2023	07:40:00	22:30:00	20	NaN	Leave	NaN	Leave	NaN	...	20
26	Day	13/03/2023	08:00:00	23:00:00	30	15.0	NaN	2.0	NaN	15.0	...	15
27	Day	14/03/2023	08:00:00	23:00:00	30	5.0	NaN	230.0	NaN	5.0	...	10
28	Day	15/03/2023	08:00:00	23:00:00	30	15.0	NaN	2.0	NaN	15.0	...	10
29	Day	16/03/2023	07:30:00	23:30:00	5	15.0	NaN	2.0	NaN	15.0	...	5

30 rows × 37 columns

In [47]:
```python
df.fillna("0", inplace = True)
df
```

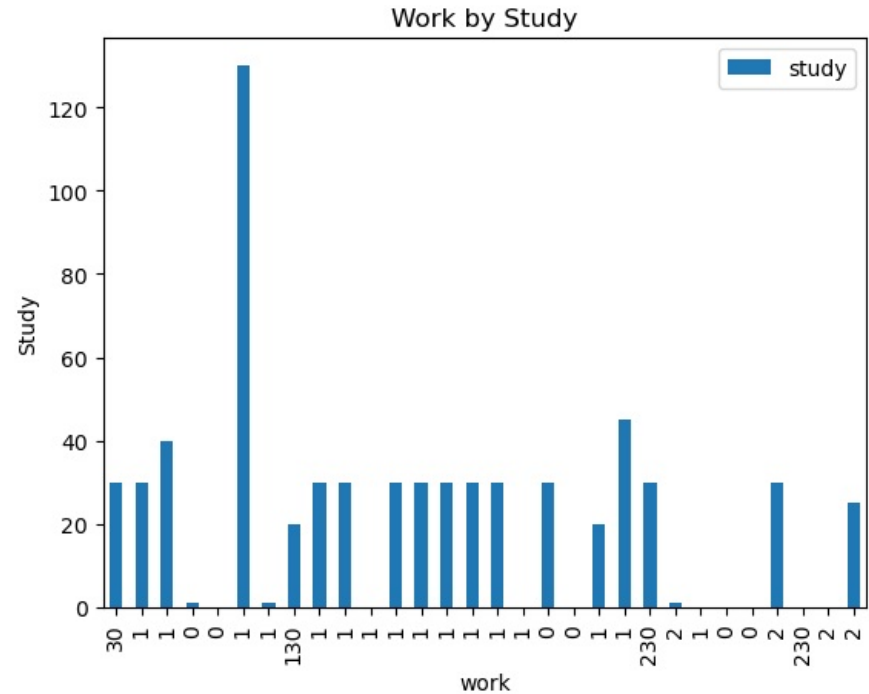|	No_of_Days	Date	Start_time	End_time	Breakfast	Office_Departure_time	Office_Departure_time_1	work	W1	Break	...	Mus
0	Day	15/2/2023	07:30:00	23:00:00	30	20.0	0	30.0	0	15.0	...	10
1	Day	16/2/2023	08:04:00	22:40:00	25	25.0	0	1.0	0	10.0	...	15
2	Day	17/2/2023	08:00:00	22:30:00	20	20.0	0	1.0	0	10.0	...	10
3	Day	18/2/2023	10:00:00	23:00:00	25	0	Leave	0	Leave	0	...	20
4	Day	19/2/2023	08:30:00	12:00:00	30	0	Leave	0	Leave	0	...	20
5	Day	20/2/2023	07:40:00	22:45:00	15	25.0	0	1.0	0	5.0	...	10
6	Day	21/2/2023	07:45:00	23:00:00	15	15.0	0	1.0	0	15.0	...	15
7	Day	22/2/2023	07:50:00	22:30:00	15	15.0	0	130.0	0	5.0	...	30
8	Day	23/2/2023	08:00:00	23:00:00	20	21.0	0	1.0	0	0	...	10
9	Day	24/2/2023	07:50:00	23:00:00	15	10.0	0	1.0	0	5.0	...	10
10	Day	25/2/2023	08:00:00	23:00:00	10	0	Outing	1.0	0	0	...	5
11	Day	26/2/2023	08:30:00	00:00:00	15	0	Leave	1.0	0	0	...	15
12	Day	27/2/2023	07:50:00	23:45:00	20	15.0	0	1.0	0	10.0	...	10
13	Day	28/2/2023	08:00:00	23:15:00	20	15.0	0	1.0	0	10.0	...	10
14	Day	01/03/2023	08:00:00	23:30:00	15	20.0	0	1.0	0	10.0	...	10
15	Day	02/02/2023	08:10:00	23:30:00	20	20.0	0	1.0	0	15.0	...	10
16	Day	03/03/2023	08:00:00	12:00:00	20	25.0	0	1.0	0	15.0	...	30
17	Day	04/03/2023	08:30:00	10:30 PM	30	0	Leave	0	Personal Work	0	...	15
18	Day	05/03/2023	08:30:00	23:30:00	1	0	Leave	0	Leave	0	...	
19	Day	06/03/2023	08:00:00	23:30:00	20	30.0	0	1.0	0	10.0	...	10
20	Day	07/03/2023	08:00:00	00:30:00	25	30.0	0	1.0	0	15.0	...	15
21	Day	08/03/2023	08:00:00	00:30:00	20	20.0	0	230.0	0	10.0	...	15
22	Day	09/03/2023	08:00:00	22:30:00	15	20.0	0	2.0	0	15.0	...	10
23	Day	10/03/2023	08:00:00	22:30:00	15	15.0	0	1.0	0	15.0	...	
24	Day	11/03/2023	08:00:00	22:30:00	20	0	Leave	0	Leave	0	...	20
25	Day	12/03/2023	07:40:00	22:30:00	20	0	Leave	0	Leave	0	...	20
26	Day	13/03/2023	08:00:00	23:00:00	30	15.0	0	2.0	0	15.0	...	15
27	Day	14/03/2023	08:00:00	23:00:00	30	5.0	0	230.0	0	5.0	...	10
28	Day	15/03/2023	08:00:00	23:00:00	30	15.0	0	2.0	0	15.0	...	10
29	Day	16/03/2023	07:30:00	23:30:00	5	15.0	0	2.0	0	15.0	...	5

30 rows × 37 columns

In [48]:
```python
df["Breakfast"]=df["Breakfast"].astype("int")
df["Office_Departure_time"]=df["Office_Departure_time"].astype("int")
df["work"]=df["work"].astype("int")
df["Break"]=df["Break"].astype("int")
df["work2"]=df["work2"].astype("int")
df["lunch"]=df["lunch"].astype("int")
df["work3"]=df["work3"].astype("int")
df["break2"]=df["break2"].astype("int")
df["work4"]=df["work4"].astype("int")
df["Return_time"]=df["Return_time"].astype("int")
df["Fitness"]=df["Fitness"].astype("int")
df["Refresh"]=df["Refresh"].astype("int")
df["Music"]=df["Music"].astype("int")
df["reading "]=df["reading "].astype("int")
df["linkedin"]=df["linkedin"].astype("int")
df["study"]=df["study"].astype("int")
df["dinner"]=df["dinner"].astype("int")
df["Sleep"]=df["Sleep"].astype("int")
df
```

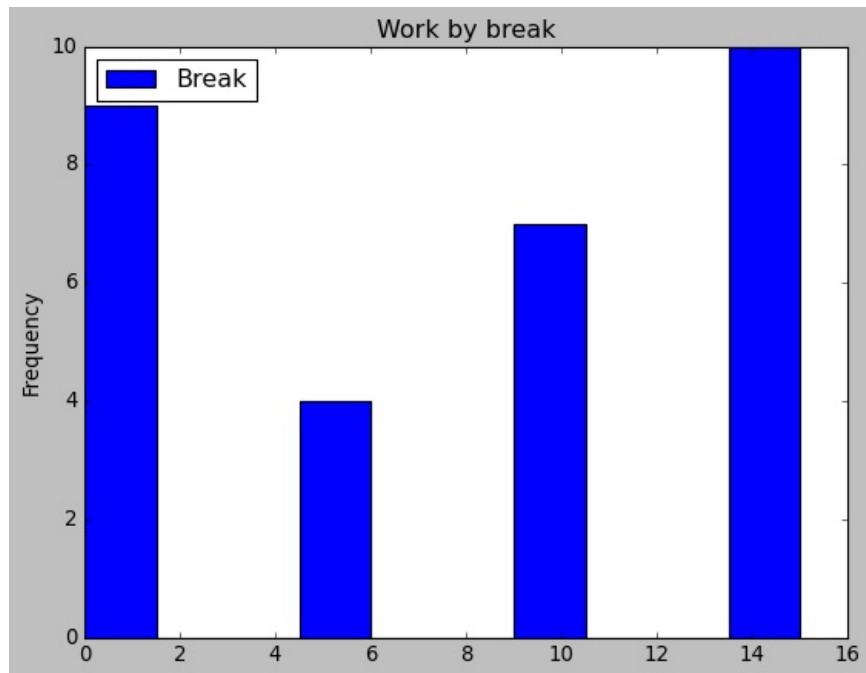|	No_of_Days	Date	Start_time	End_time	Breakfast	Office_Departure_time	Office_Departure_time_1	work	W1	Break	...	Musi
0	Day	15/2/2023	07:30:00	23:00:00	30	20	0	30	0	15	...	1
1	Day	16/2/2023	08:04:00	22:40:00	25	25	0	1	0	10	...	1
2	Day	17/2/2023	08:00:00	22:30:00	20	20	0	1	0	10	...	1
3	Day	18/2/2023	10:00:00	23:00:00	25	0	Leave	0	Leave	0	...	2
4	Day	19/2/2023	08:30:00	12:00:00	30	0	Leave	0	Leave	0	...	2
5	Day	20/2/2023	07:40:00	22:45:00	15	25	0	1	0	5	...	1
6	Day	21/2/2023	07:45:00	23:00:00	15	15	0	1	0	15	...	1
7	Day	22/2/2023	07:50:00	22:30:00	15	15	0	130	0	5	...	3
8	Day	23/2/2023	08:00:00	23:00:00	20	21	0	1	0	0	...	1
9	Day	24/2/2023	07:50:00	23:00:00	15	10	0	1	0	5	...	1
10	Day	25/2/2023	08:00:00	23:00:00	10	0	Outing	1	0	0	...	
11	Day	26/2/2023	08:30:00	00:00:00	15	0	Leave	1	0	0	...	1
12	Day	27/2/2023	07:50:00	23:45:00	20	15	0	1	0	10	...	1
13	Day	28/2/2023	08:00:00	23:15:00	20	15	0	1	0	10	...	1
14	Day	01/03/2023	08:00:00	23:30:00	15	20	0	1	0	10	...	1
15	Day	02/02/2023	08:10:00	23:30:00	20	20	0	1	0	15	...	1
16	Day	03/03/2023	08:00:00	12:00:00	20	25	0	1	0	15	...	3
17	Day	04/03/2023	08:30:00	10:30 PM	30	0	Leave	0	Personal Work	0	...	1
18	Day	05/03/2023	08:30:00	23:30:00	1	0	Leave	0	Leave	0	...	
19	Day	06/03/2023	08:00:00	23:30:00	20	30	0	1	0	10	...	1
20	Day	07/03/2023	08:00:00	00:30:00	25	30	0	1	0	15	...	1
21	Day	08/03/2023	08:00:00	00:30:00	20	20	0	230	0	10	...	1
22	Day	09/03/2023	08:00:00	22:30:00	15	20	0	2	0	15	...	1
23	Day	10/03/2023	08:00:00	22:30:00	15	15	0	1	0	15	...	
24	Day	11/03/2023	08:00:00	22:30:00	20	0	Leave	0	Leave	0	...	2
25	Day	12/03/2023	07:40:00	22:30:00	20	0	Leave	0	Leave	0	...	2
26	Day	13/03/2023	08:00:00	23:00:00	30	15	0	2	0	15	...	1
27	Day	14/03/2023	08:00:00	23:00:00	30	5	0	230	0	5	...	1
28	Day	15/03/2023	08:00:00	23:00:00	30	15	0	2	0	15	...	1
29	Day	16/03/2023	07:30:00	23:30:00	5	15	0	2	0	15	...	

30 rows × 37 columns

In [49]:
```
df.plot(kind="bar", x="work",y="study",title="Work by Study",ylabel="Study")
plt.style.use("classic")
```


Work by Study

```
In [57]: df.plot(kind="hist", x="work",y="Break",title="Work by break")
```
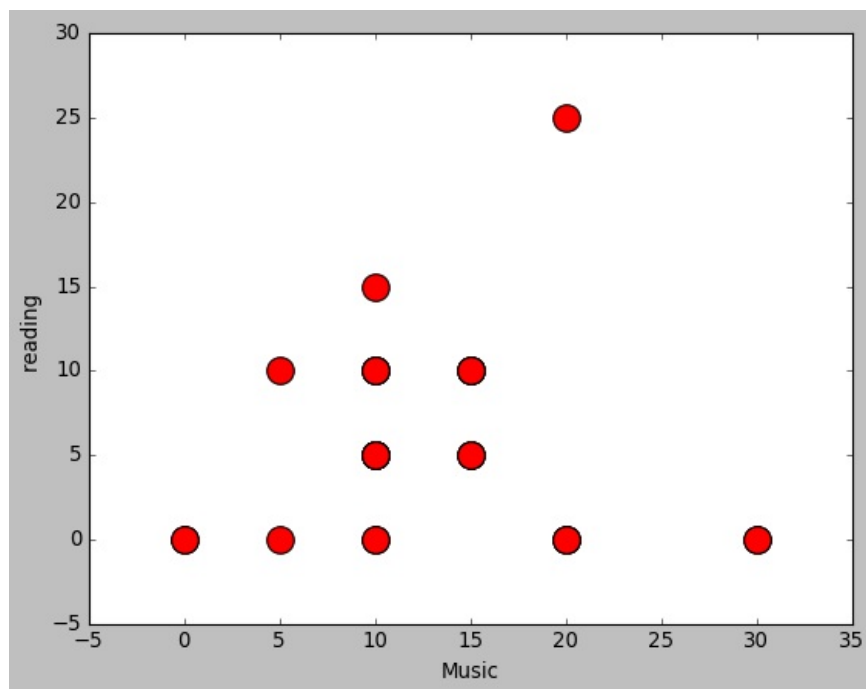
Out[57]: `<AxesSubplot:title={'center':'Work by break'}, ylabel='Frequency'>`
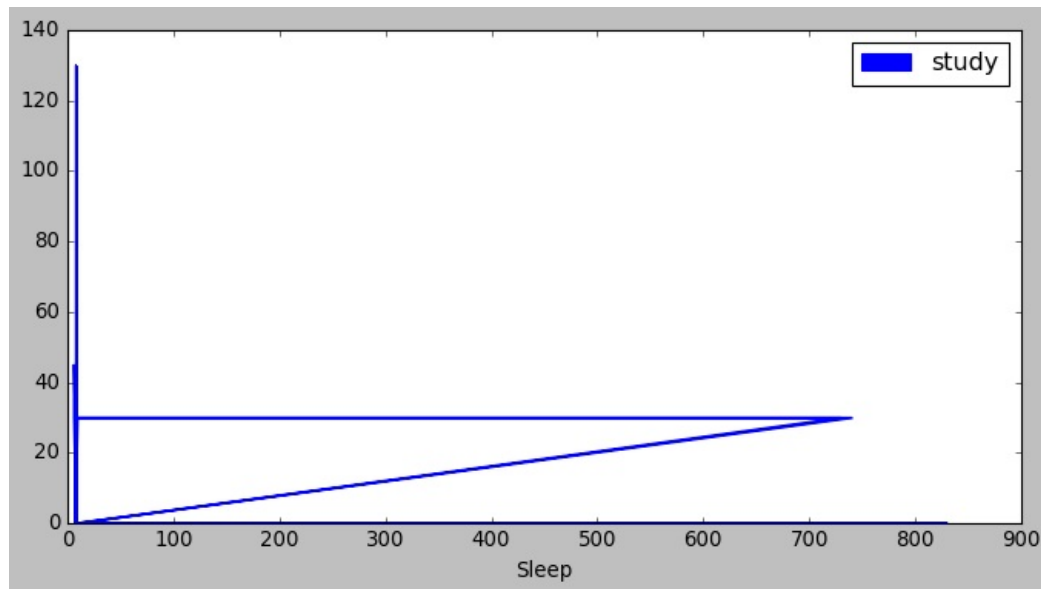


```
In [58]: df.plot.scatter(x= "Music" ,y="reading ", s=250, c= "red")
```
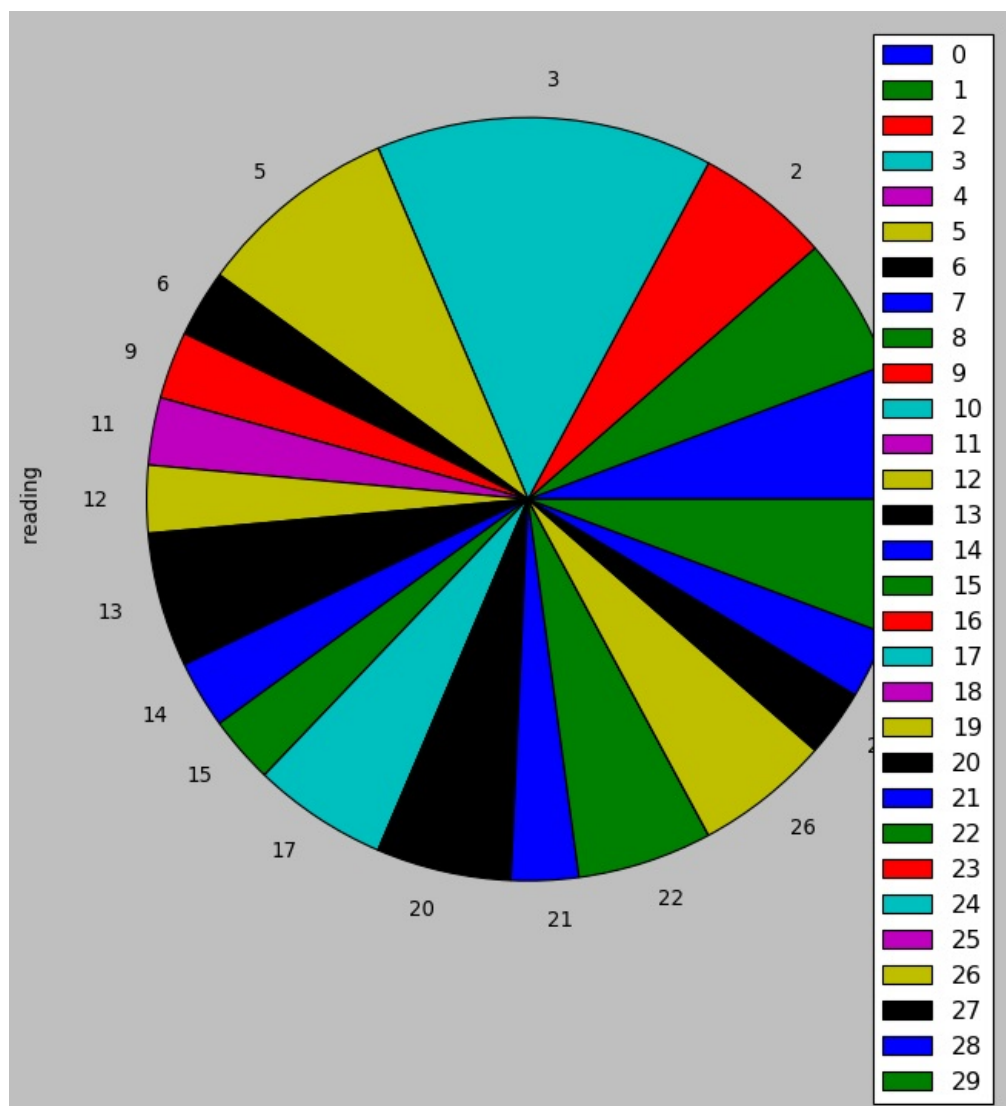
Out[58]: `<AxesSubplot:xlabel='Music', ylabel='reading '>`


```
In [59]: df.plot.area(x="Sleep", y="study",figsize=(10,5))
```

Out[59]: `<AxesSubplot:xlabel='Sleep'>`

In [60]: `df.plot.pie(y="reading ",figsize=(10,10))`

Out[60]: `<AxesSubplot:ylabel='reading '>`

`print(plt.style.available)`

['Solarize_Light2', '_classic_test_patch', '_mpl-gallery', '_mpl-gallery-nogrid', 'bmh', 'classic', 'dark_background', 'fast', 'fivethirtyeight', 'ggplot', 'grayscale', 'seaborn', 'seaborn-bright', 'seaborn-colorblind', 'seaborn-dark', 'seaborn-dark-palette', 'seaborn-darkgrid', 'seaborn-deep', 'seaborn-muted', 'seaborn-notebook', 'seaborn-paper', 'seaborn-pastel', 'seaborn-poster', 'seaborn-talk', 'seaborn-ticks', 'seaborn-white', 'seaborn-whitegrid', 'tableau-colorblind10']