In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:

```
df=pd.read_csv(r"C:\Users\yoshitha lakshmi\OneDrive\Desktop\python\Income.csv")
df
```

Out[2]:

	Gender	Age	Income($)
0	Male	19	15
1	Male	21	15
2	Female	20	16
3	Female	23	16
4	Female	31	17
...	...	...	...
195	Female	35	120
196	Female	45	126
197	Male	32	126
198	Male	32	137
199	Male	30	137

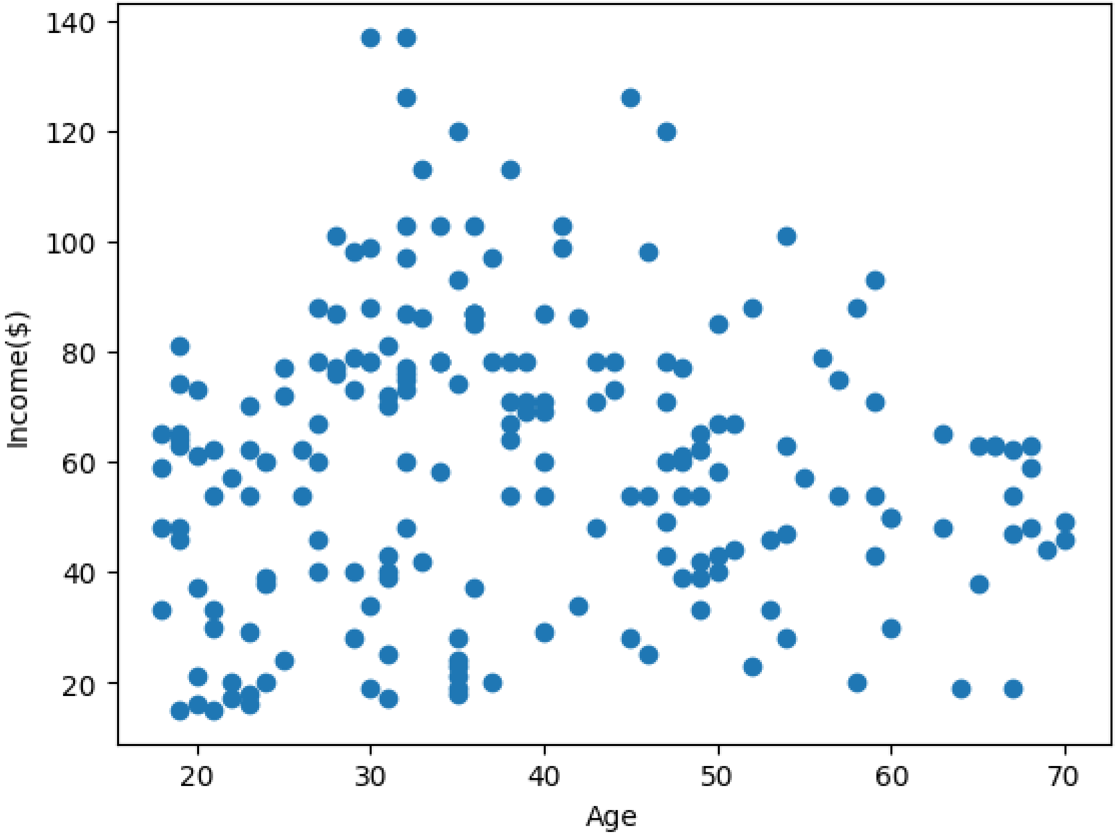
200 rows × 3 columns

In [3]:

```
plt.scatter(df["Age"],df["Income($)"])
plt.xlabel("Age")
plt.ylabel("Income($)")
```

Out[3]:

```
Text(0, 0.5, 'Income($)')
```



In [4]:

```
from sklearn.cluster import KMeans
```

In [5]:

```
km=KMeans()
km
```

Out[5]:

```
▾ KMeans

KMeans()
```

In [6]:

```
y_predicted=km.fit_predict(df[["Age","Income($)"]])
y_predicted
```

```
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
```

Out[6]:

```
array([5, 5, 5, 5, 5, 5, 5, 5, 4, 5, 4, 5, 4, 5, 5, 5, 5, 5, 4, 5, 5, 5,
       4, 5, 4, 5, 4, 5, 4, 5, 4, 5, 4, 1, 4, 1, 4, 1, 1, 1, 4, 1, 4, 1,
       4, 1, 4, 1, 1, 1, 4, 1, 1, 4, 4, 4, 4, 2, 1, 4, 2, 1, 2, 4, 2, 1,
       4, 2, 1, 1, 2, 4, 2, 2, 2, 1, 3, 3, 1, 3, 2, 3, 2, 3, 1, 3, 2, 6,
       3, 3, 2, 6, 3, 3, 6, 6, 3, 6, 3, 6, 6, 3, 2, 6, 3, 6, 2, 3, 2, 2,
       2, 6, 3, 6, 6, 6, 2, 3, 3, 3, 6, 3, 3, 3, 6, 6, 3, 3, 3, 3, 3, 3,
       6, 6, 6, 6, 3, 6, 6, 6, 3, 6, 6, 6, 6, 6, 3, 6, 6, 6, 3, 3, 3, 6,
       3, 6, 6, 6, 6, 6, 3, 6, 6, 6, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 7, 7, 7, 7, 7, 7,
       7, 7])
```

In [7]:

```
df["cluster"]=y_predicted
df.head()
```

Out[7]:

```
   Gender  Age  Income($)  cluster
0    Male   19         15        5
1    Male   21         15        5
2  Female   20         16        5
3  Female   23         16        5
4  Female   31         17        5
```

In [14]:

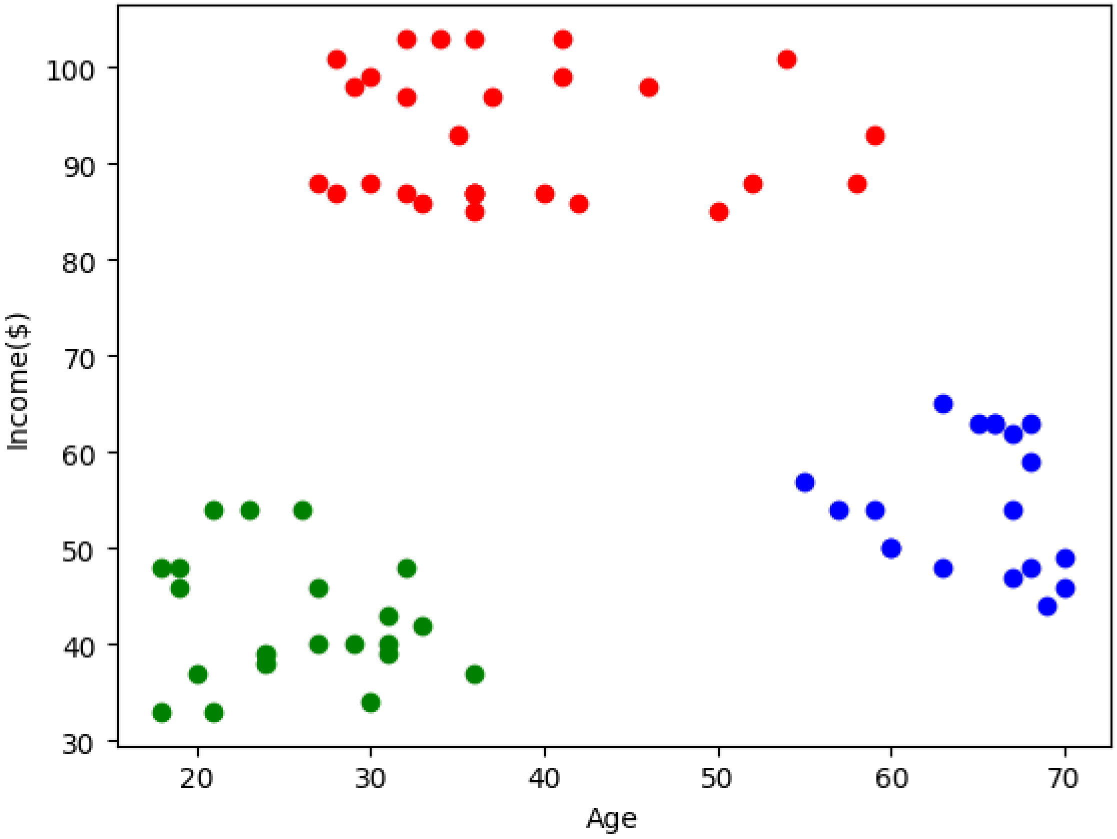
```
df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]

plt.scatter(df1["Age"],df1["Income($)"],color="red")
plt.scatter(df2["Age"],df2["Income($)"],color="green")
plt.scatter(df3["Age"],df3["Income($)"],color="blue")

plt.xlabel("Age")
plt.ylabel("Income($)")
```

Out[14]:

```
Text(0, 0.5, 'Income($)')
```



In [15]:

```python
from sklearn.preprocessing import MinMaxScaler
```

In [16]:

```python
scaler=MinMaxScaler()
```

In [17]:

```python
scaler.fit(df[["Income($)"]])
df["Income($)"]=scaler.transform(df[["Income($)"]])
df.head()
```

Out[17]:

	Gender	Age	Income($)	cluster
0	Male	19	0.000000	5
1	Male	21	0.000000	5
2	Female	20	0.008197	5
3	Female	23	0.008197	5
4	Female	31	0.016393	5

In [18]:

```python
scaler.fit(df[["Age"]])
df["Age"]=scaler.transform(df[["Age"]])
df.head()
```

Out[18]:

	Gender	Age	Income($)	cluster
0	Male	0.019231	0.000000	5
1	Male	0.057692	0.000000	5
2	Female	0.038462	0.008197	5
3	Female	0.096154	0.008197	5
4	Female	0.250000	0.016393	5

In [27]:
```
km=KMeans()
```

In [28]:
```
y_predicted=km.fit_predict(df[["Age","Income($)"]])
y_predicted
```

```
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
```

Out[28]:
```
array([4, 4, 4, 4, 6, 4, 6, 4, 1, 6, 1, 6, 7, 4, 6, 4, 6, 4, 7, 6, 6, 4,
       7, 6, 7, 6, 7, 6, 6, 4, 1, 4, 7, 4, 7, 4, 7, 6, 6, 4, 1, 4, 7, 6,
       7, 4, 7, 6, 6, 6, 7, 6, 6, 1, 7, 7, 7, 1, 2, 7, 1, 2, 1, 7, 1, 2,
       7, 1, 2, 6, 1, 7, 1, 1, 1, 2, 7, 7, 2, 7, 1, 0, 1, 7, 2, 7, 3, 2,
       0, 3, 1, 2, 3, 0, 0, 2, 3, 2, 3, 2, 2, 3, 1, 2, 3, 2, 1, 3, 1, 1,
       1, 2, 0, 2, 2, 2, 1, 3, 3, 3, 2, 0, 0, 0, 2, 0, 3, 0, 3, 0, 3, 0,
       2, 0, 2, 0, 3, 0, 2, 0, 3, 0, 0, 0, 2, 0, 3, 0, 0, 0, 3, 0, 3, 0,
       3, 0, 0, 0, 0, 0, 3, 0, 2, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 0,
       3, 0, 3, 5, 5, 5, 5, 5, 5, 5, 3, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5,
       5, 5])
```

In [29]:
```
df["New cluster"]=y_predicted
df.head()
```

Out[29]:

	Gender	Age	Income($)	cluster	New cluster
0	Male	0.019231	0.000000	5	4
1	Male	0.057692	0.000000	5	4
2	Female	0.038462	0.008197	5	4
3	Female	0.096154	0.008197	5	4
4	Female	0.250000	0.016393	5	6

In [30]:

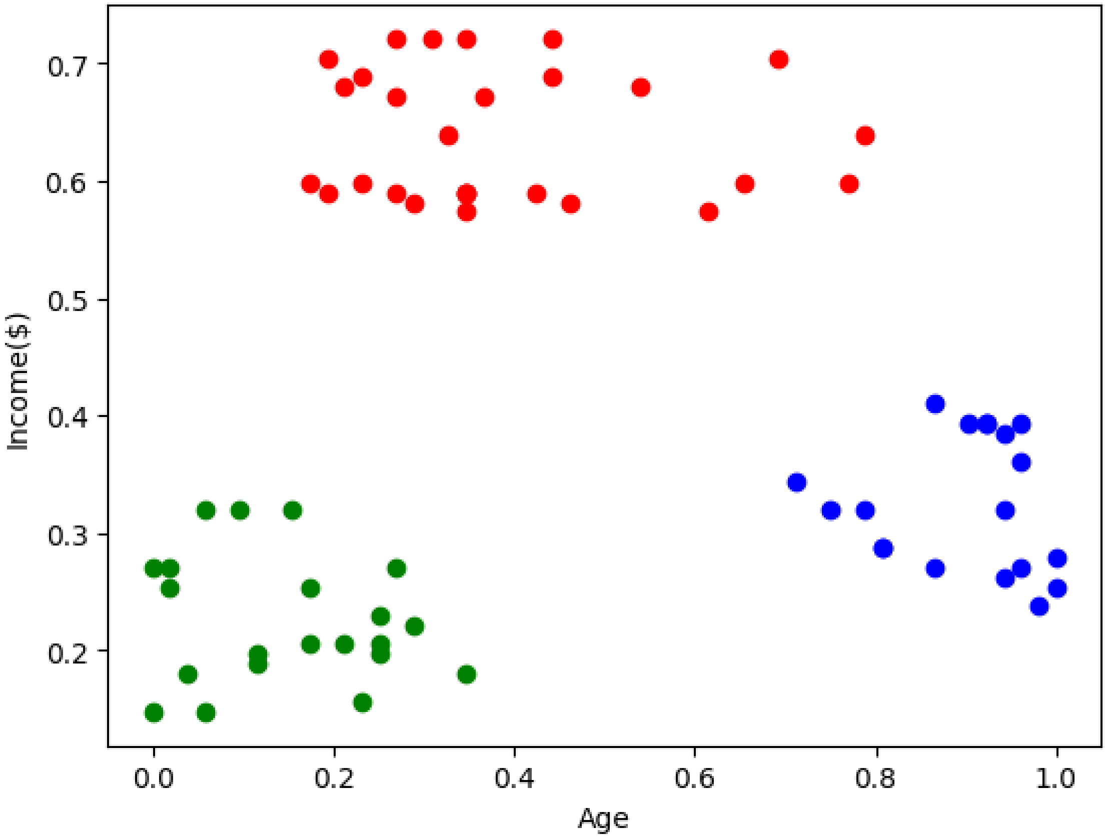
```
df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]

plt.scatter(df1["Age"],df1["Income($)"],color="red")
plt.scatter(df2["Age"],df2["Income($)"],color="green")
plt.scatter(df3["Age"],df3["Income($)"],color="blue")

plt.xlabel("Age")
plt.ylabel("Income($)")
```

Out[30]:

```
Text(0, 0.5, 'Income($)')
```



In [32]:

```
km.cluster_centers_
```

Out[32]:

```
array([[0.30903399, 0.50114373],
       [0.89799331, 0.28011404],
       [0.07322485, 0.38272383],
       [0.62352071, 0.47225725],
       [0.07239819, 0.08003857],
       [0.33942308, 0.77295082],
       [0.28388278, 0.1245121 ],
       [0.58974359, 0.20969945]])
```

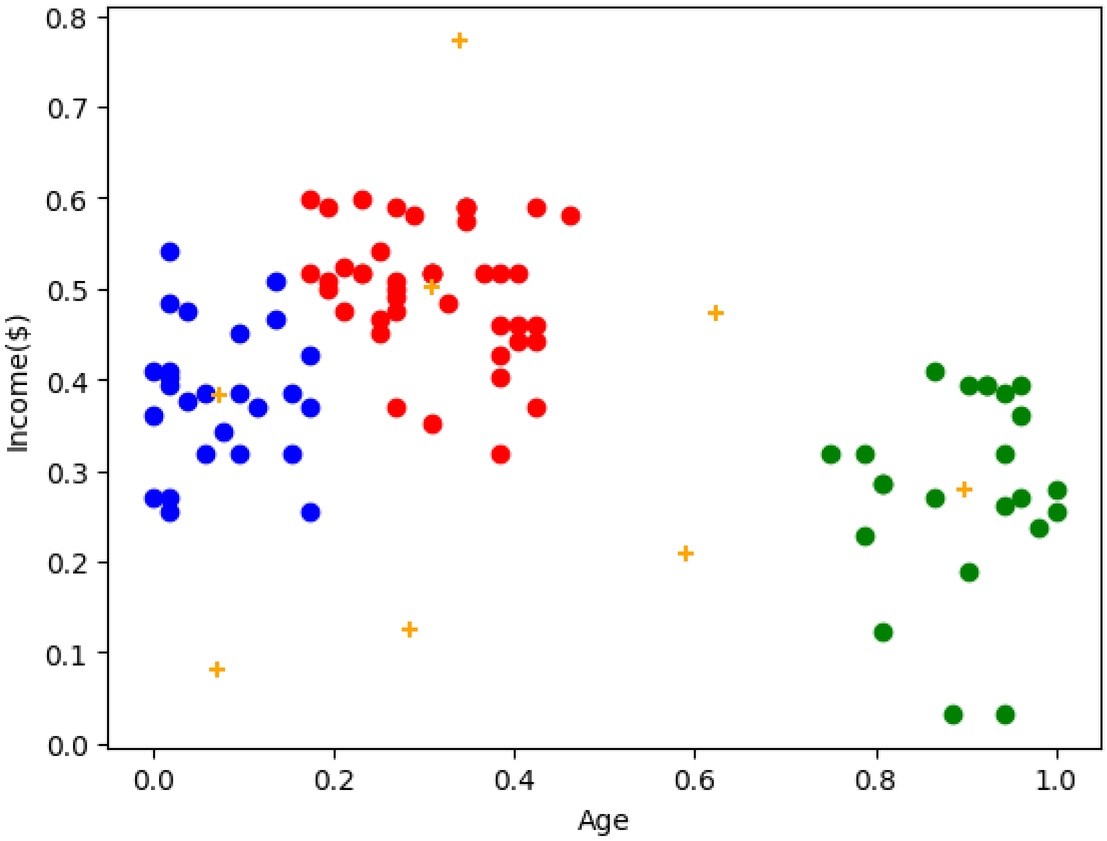
In [36]:

```
df1=df[df["New cluster"]==0]
df2=df[df["New cluster"]==1]
df3=df[df["New cluster"]==2]

plt.scatter(df1["Age"],df1["Income($)"],color="red")
plt.scatter(df2["Age"],df2["Income($)"],color="green")
plt.scatter(df3["Age"],df3["Income($)"],color="blue")
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color="orange",marker="+")
plt.xlabel("Age")
plt.ylabel("Income($)")
```

Out[36]:

```
Text(0, 0.5, 'Income($)')
```



In [37]:

```
k_rng=range(1,10)
sse=[]
for k in k_rng:
    km=KMeans(n_clusters=k)
    km.fit(df[["Age","Income($)"]])
    sse.append(km.inertia_)
sse
```

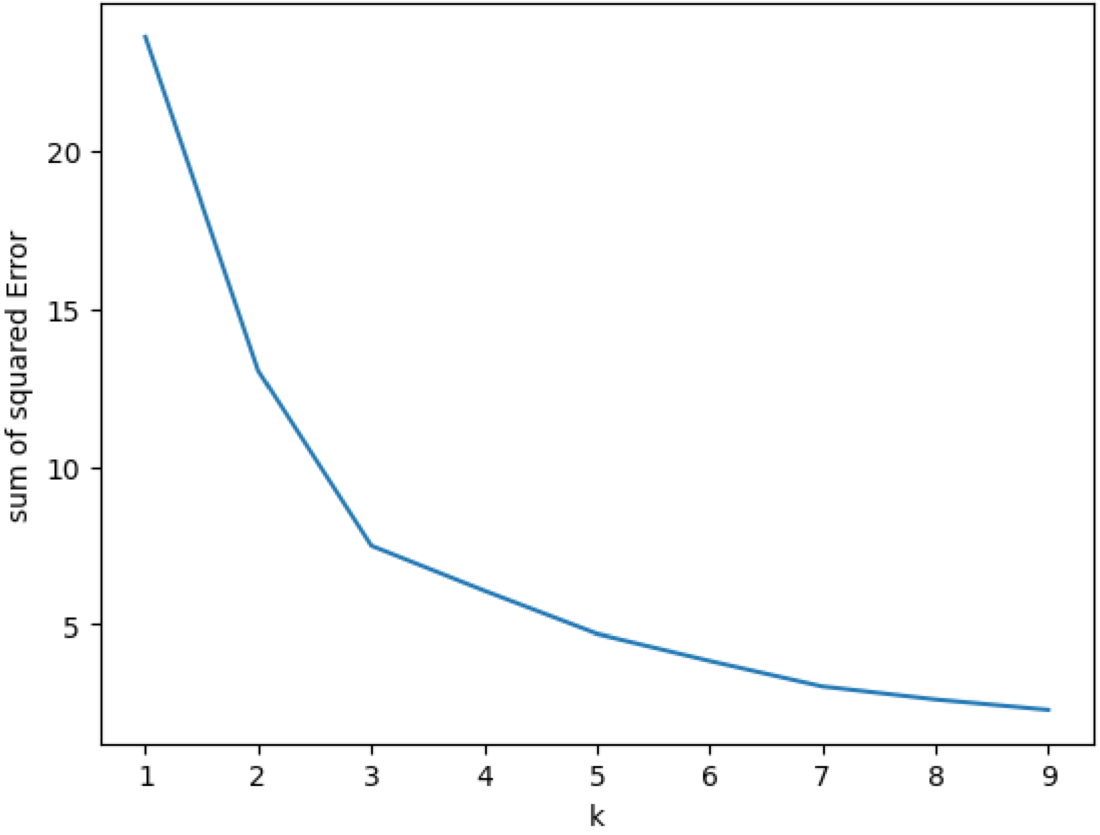
```
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
C:\Users\yoshitha lakshmi\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explic
itly to suppress the warning
  warnings.warn(
Out[37]: [23.583906150363603,
           13.028938428018286,
           7.493024843304991,
           6.07288472287425545,
           4.713416604872824,
           3.8612812134405137,
           3.054717436369358,
           2.642520343536072,
           2.3135720353543285]
```

In [38]:

```
plt.plot(k_rng,sse)
plt.xlabel("k")
plt.ylabel("sum of squared Error")
```

Out[38]:

```
Text(0, 0.5, 'sum of squared Error')
```



In [ ]:

```
```