In [59]:
```
from sklearn.cluster import KMeans
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
import matplotlib.pyplot as plt
%matplotlib inline
```

In [60]:
```
df = pd.read_csv('INCOME.csv')
df
```

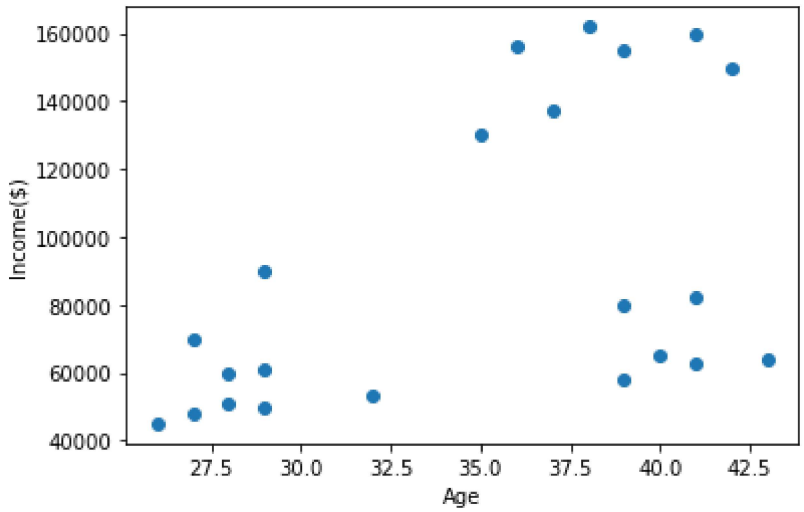
Out[60]:

	NAME	AGE	Income($)
0	Rob	27	70000
1	Michael	29	90000
2	Mohan	29	61000
3	Ismail	28	60000
4	Kory	42	150000
5	Gautam	39	155000
6	David	41	160000
7	Andrea	38	162000
8	Brad	36	156000
9	Angelina	35	130000
10	Donald	37	137000
11	Tom	26	45000
12	Arnold	27	48000
13	Jared	28	51000
14	Stark	29	49500
15	Ranbir	32	53000
16	Dipika	40	65000
17	Priyanka	41	63000
18	Nick	43	64000
19	Alia	39	80000
20	Sid	41	82000
21	Abdul	39	58000

In [61]:
```
plt.scatter(df.AGE , df['Income($)'])

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

Out[61]:
```
Text(0, 0.5, 'Income($)')
```



In [62]:
```
km = KMeans(n_clusters = 3)
y_predicted = km.fit_predict(df[['AGE','Income($)']])
y_predicted
```

Out[62]:
```
array([0, 0, 2, 2, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 0, 0, 2])
```

In [63]:
```
df['clusters'] = y_predicted
df
```

Out[63]:

	NAME	AGE	Income($)	clusters
0	Rob	27	70000	0
1	Michael	29	90000	0
2	Mohan	29	61000	2
3	Ismail	28	60000	2
4	Kory	42	150000	1
5	Gautam	39	155000	1
6	David	41	160000	1
7	Andrea	38	162000	1
8	Brad	36	156000	1
9	Angelina	35	130000	1
10	Donald	37	137000	1
11	Tom	26	45000	2
12	Arnold	27	48000	2
13	Jared	28	51000	2
14	Stark	29	49500	2
15	Ranbir	32	53000	2
16	Dipika	40	65000	2
17	Priyanka	41	63000	2
18	Nick	43	64000	2
19	Alia	39	80000	0
20	Sid	41	82000	0
21	Abdul	39	58000	2

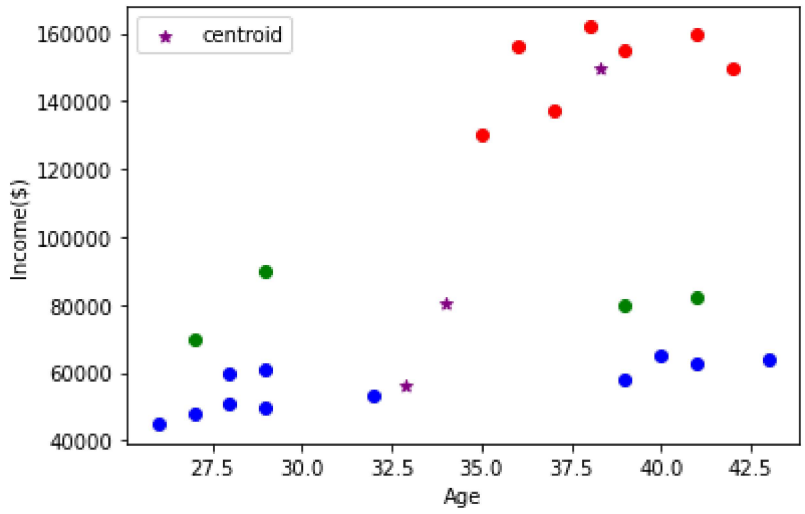
In [64]:
```
km.cluster_centers_
```

Out[64]:
```
array([[3.40000000e+01, 8.05000000e+04],
       [3.82857143e+01, 1.50000000e+05],
       [3.29090909e+01, 5.61363636e+04]])
```

In [65]:
```
df1 = df[df.clusters == 0]
df2 = df[df.clusters == 1]
df3 = df[df.clusters == 2]
```

In [66]:
```
plt.scatter(df1.AGE, df1['Income($)'], color = 'green')
plt.scatter(df2.AGE, df2['Income($)'], color = 'red')
plt.scatter(df3.AGE, df3['Income($)'], color = 'blue')
plt.scatter(km.cluster_centers_[ :, 0], km.cluster_centers_[:, 1] , color = 'purp
plt.xlabel('Age')
plt.ylabel('Income($)')
plt.legend()
```

Out[66]:
```
<matplotlib.legend.Legend at 0x23624a9e8b0>
```



In [67]:
```
scaler = MinMaxScaler()
```

In [68]:
```
scaler.fit(df[['Income($)']])
df['Income($)'] = scaler.transform(df[['Income($)']])
scaler.fit(df[['AGE']])
df['AGE'] = scaler.transform(df[['AGE']])
df.head()
```

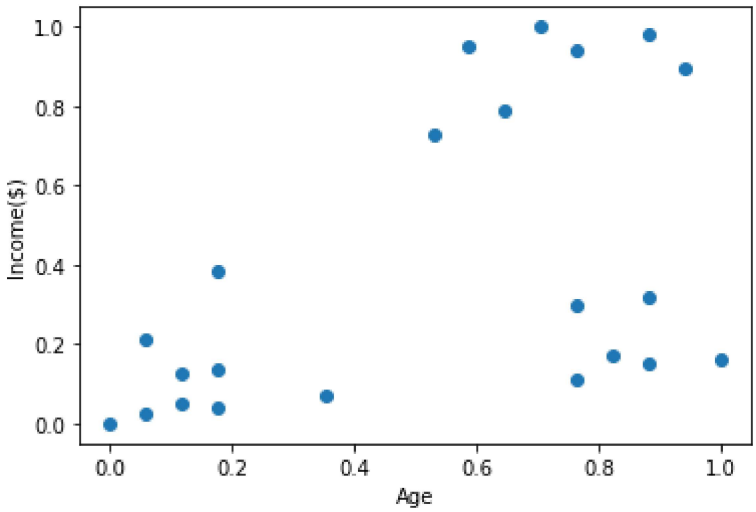
Out[68]:

	NAME	AGE	Income($)	clusters
0	Rob	0.058824	0.213675	0
1	Michael	0.176471	0.384615	0
2	Mohan	0.176471	0.136752	2
3	Ismail	0.117647	0.128205	2
4	Kory	0.941176	0.897436	1

In [69]:
```
plt.scatter(df.AGE , df['Income($)'])

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

Out[69]:
```
Text(0, 0.5, 'Income($)')
```



In [70]:
```
km = KMeans(n_clusters = 3)
y_predicted = km.fit_predict(df[['AGE','Income($)']])
y_predicted
```

Out[70]:
```
array([0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 2, 2, 2, 2, 2, 2])
```

In [71]:
```
df['clusters'] = y_predicted
df.head()
```

Out[71]:

	NAME	AGE	Income($)	clusters
0	Rob	0.058824	0.213675	0
1	Michael	0.176471	0.384615	0
2	Mohan	0.176471	0.136752	0
3	Ismail	0.117647	0.128205	0
4	Kory	0.941176	0.897436	1

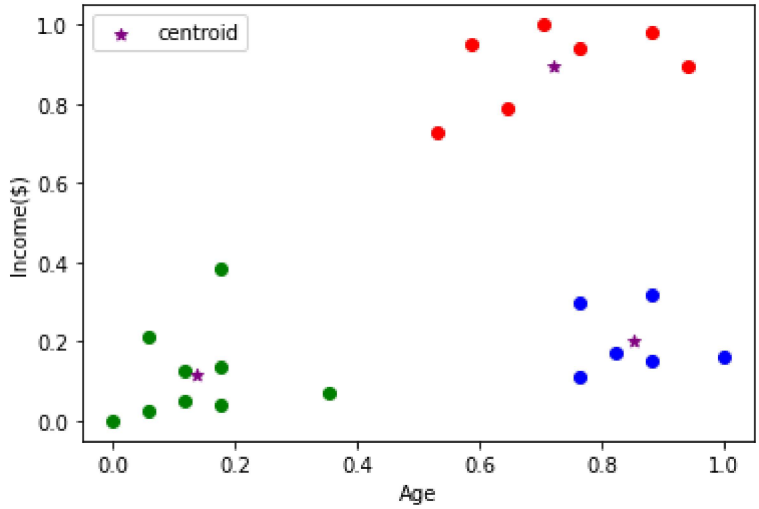
In [75]:
```
df1 = df[df.clusters == 0]
df2 = df[df.clusters == 1]
df3 = df[df.clusters == 2]
```

In [76]:
```
km.cluster_centers_
```

Out[76]:
```
array([[0.1372549 , 0.11633428],
       [0.72268908, 0.8974359 ],
       [0.85294118, 0.2022792 ]])
```

In [77]:
```
plt.scatter(df1.AGE, df1['Income($)'], color = 'green')
plt.scatter(df2.AGE, df2['Income($)'], color = 'red')
plt.scatter(df3.AGE, df3['Income($)'], color = 'blue')
plt.scatter(km.cluster_centers_[ :,0], km.cluster_centers_[:,1] , color = 'purple
plt.xlabel('Age')
plt.ylabel('Income($)')
plt.legend()
```

Out[77]:
```
<matplotlib.legend.Legend at 0x23624c770d0>
```



In [89]:
```
sse = []
k_rng = range(1,11)
for k in range(1, 11):
    km = KMeans(n_clusters = k)
    km.fit(df[['AGE','Income($)']])
    sse.append(km.inertia_)
```

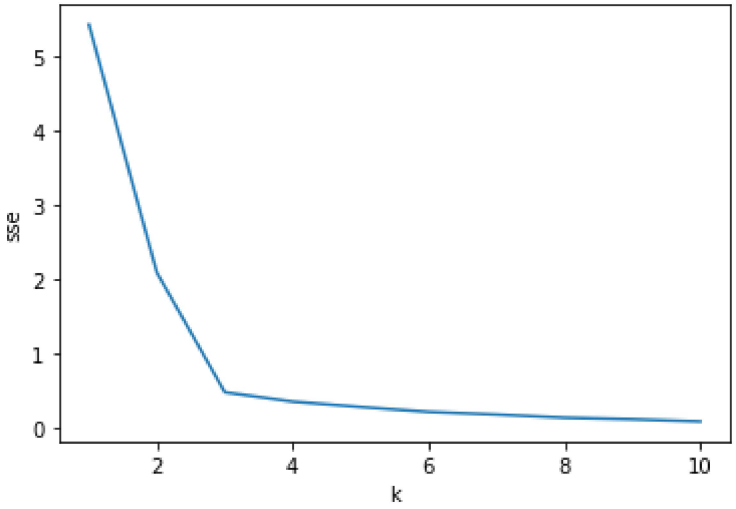
```
D:\ashwa\ana\lib\site-packages\sklearn\cluster\_kmeans.py:881: UserWarning: KMe
ans is known to have a memory leak on Windows with MKL, when there are less chu
nks than available threads. You can avoid it by setting the environment variabl
e OMP_NUM_THREADS=1.
  warnings.warn(
```

In [90]:
```
sse
```

Out[90]:
```
[5.434011511988179,
 2.091136388699078,
 0.4750783498553097,
 0.3491047094419566,
 0.2766936276300279,
 0.21055478995472496,
 0.17462386586687895,
 0.13265419827245162,
 0.11073569527418643,
 0.08139933135681812]
```

In [92]:
```
plt.xlabel('k')
plt.ylabel('sse')
plt.plot(k_rng,sse)
```

Out[92]:
```
[<matplotlib.lines.Line2D at 0x23624fb3640>]
```



In [ ]:

In [ ]: