In [24]:
```python
#import files
from sklearn.cluster import KMeans
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from matplotlib import pyplot as plt
get_ipython().run_line_magic('matplotlib', 'inline')
```

In [25]:
```python
#read file
df=pd.read_csv("excleofDataSet.csv")
df.head()
```

Out[25]:

	Unnamed: 0	sl_no	University_iD	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p
0	0	1.0	0	M	67.00	Others	67.00	Others	Commerce	58.00
1	1	2.0	12346	M	79.33	Central	79.33	Others	Science	77.48
2	2	3.0	0	Other	65.00	NaN	65.00	Central	NaN	0.00
3	3	4.0	12348	M	56.00	Central	56.00	Central	Science	52.00
4	4	5.0	12349	M	85.80	Central	85.80	Central	Commerce	73.30

In [6]:
```python
df1 = df[['ssc_p', 'salary']]
print(df1)
```

```
     ssc_p  salary
0    67.00       0
1    79.33  200000
2    65.00       0
3    56.00       0
4    85.80  425000
..     ...     ...
213  80.60  400000
214  58.00  275000
215  67.00  295000
216  74.00  204000
217  62.00       0

[218 rows x 2 columns]
```

In [7]:

```
df1_norm = (df1-df1.min())/(df1.max()-df1.min())
print("Scaled Dataset Using Pandas")
df1_norm.head()
```

```
Scaled Dataset Using Pandas
```

Out[7]:

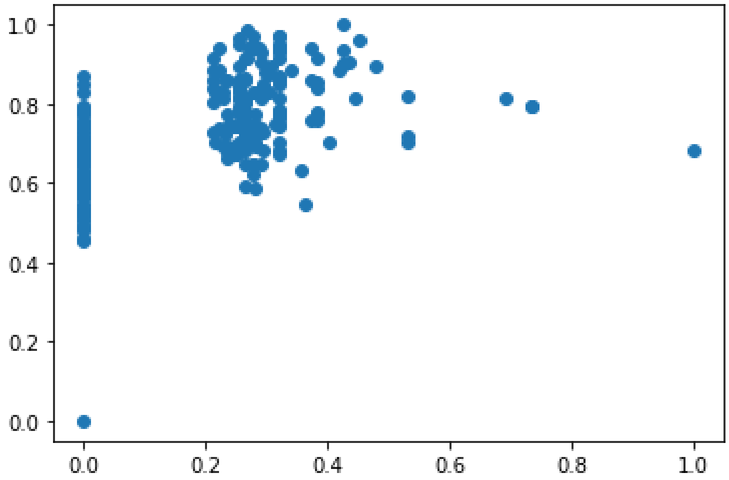
	ssc_p	salary
0	0.749441	0.000000
1	0.887360	0.212766
2	0.727069	0.000000
3	0.626398	0.000000
4	0.959732	0.452128

In [8]:

```
#Scatter Plot
plt.scatter(df1_norm['salary'],df1_norm['ssc_p'])
```

Out[8]:

```
<matplotlib.collections.PathCollection at 0x1ca912d4100>
```



km=KMeans(n_clusters=3) km

In [35]:

```
# Choose K
km=KMeans(n_clusters=2)
km
#convert all in array /group
y_predicted = km.fit_predict(df1_norm[['salary','ssc_p']])
y_predicted
```

Out[35]:

```
array([0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 1,
       1, 1, 1, 0, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 1,
       1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0,
       1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0,
       1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0,
       1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1,
       1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1,
       1, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 1, 1,
       0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0,
       1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0])
```

In [36]:

```
#dataframe vS/group
df1_norm['cluster']=y_predicted
df1_norm.head()
```

Out[36]:

	ssc_p	salary	cluster
0	0.749441	0.000000	0
1	0.887360	0.212766	1
2	0.727069	0.000000	0
3	0.626398	0.000000	0
4	0.959732	0.452128	1

In [37]:

```
#Centroids
km.cluster_centers_
```

Out[37]:

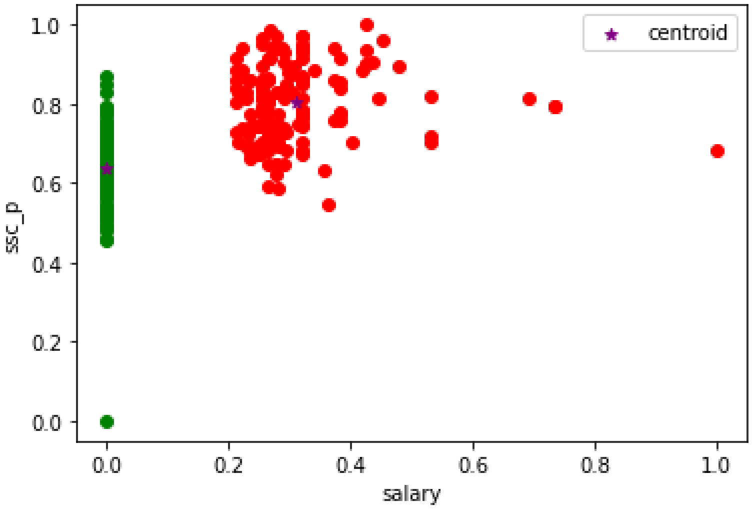
```
array([[5.55111512e-17, 6.35134071e-01],
       [3.10681719e-01, 8.02460850e-01]])
```

In [39]:

```
#datafram to three group and ploat Scatter plot
df = df1_norm[df1_norm.cluster==0]
df2 = df1_norm[df1_norm.cluster==1]
plt.scatter(df.salary ,df['ssc_p'],color='green')
plt.scatter(df2.salary ,df2['ssc_p'],color='red')
#ploatling centroids
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color='purple',mark
plt.xlabel('salary')
plt.ylabel('ssc_p')
plt.legend()
```

Out[39]:

```
<matplotlib.legend.Legend at 0x1ca91d5b2b0>
```



In [40]:

```
df
```

Out[40]:

	ssc_p	salary	cluster
0	0.749441	0.0	0
2	0.727069	0.0	0
3	0.626398	0.0	0
5	0.615213	0.0	0
6	0.514541	0.0	0
...	...	...	...
201	0.749441	0.0	0
204	0.606264	0.0	0
209	0.458613	0.0	0
211	0.480984	0.0	0
217	0.693512	0.0	0

71 rows × 3 columns

In [44]:
```
df
```

Out[44]:

	ssc_p	salary	cluster
0	0.749441	0.0	0
2	0.727069	0.0	0
3	0.626398	0.0	0
5	0.615213	0.0	0
6	0.514541	0.0	0
...	...	...	...
201	0.749441	0.0	0
204	0.606264	0.0	0
209	0.458613	0.0	0
211	0.480984	0.0	0
217	0.693512	0.0	0

71 rows × 3 columns

In [43]:
```
df2
```

Out[43]:

	ssc_p	salary	cluster
1	0.887360	0.212766	1
4	0.959732	0.452128	1
7	0.917226	0.268085	1
10	0.648770	0.276596	1
11	0.778523	0.265957	1
...	...	...	...
212	0.693512	0.229787	1
213	0.901566	0.425532	1
214	0.648770	0.292553	1
215	0.749441	0.313830	1
216	0.827740	0.217021	1

147 rows × 3 columns

In [ ]:
```

```