In [26]:
```
#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 [27]:
```
#read file
df=pd.read_csv("excleofDataSet.csv")
df.head()
```

Out[27]:

	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	C
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	C

In [28]:
```
convert_numeric = {
    'Placed' : 1,
    'Not Placed' : 0
}
conversion = df['status'].map(convert_numeric)
print(conversion)
```

```
0      1
1      1
2      0
3      0
4      1
      ..
213    1
214    1
215    1
216    1
217    0
Name: status, Length: 218, dtype: int64
```

In [29]:
```
df1 = df.copy()
df1['status'] = conversion
print(df1)
```

```
     Unnamed: 0  sl_no  University_iD gender  ssc_p    ssc_b  hsc_p    hsc_b  \
0             0    1.0              0      M  67.00   Others  67.00   Others
1             1    2.0          12346      M  79.33  Central  79.33   Others
2             2    3.0              0  Other  65.00      NaN  65.00  Central
3             3    4.0          12348      M  56.00  Central  56.00  Central
4             4    5.0          12349      M  85.80  Central  85.80  Central
..          ...    ...            ...    ...    ...      ...    ...      ...
213          61    NaN          12555  Other  80.60   Others  80.60   Others
214          62    NaN          12556  Other  58.00   Others  58.00   Others
215          63    NaN          12557  Other  67.00   Others  67.00   Others
216          64    NaN          12558  Other  74.00   Others  74.00   Others
217          65    NaN          12559  Other  62.00  Central  62.00   Others

        hsc_s  degree_p   degree_t workex  Number of years experience test_p  \
0    Commerce     58.00   Sci&Tech      0                           0     55
1     Science     77.48   sci&Tech    Yes                           2   86.5
2         NaN      0.00  Comm&Mgmt      0                           0      0
3     Science     52.00   Sci&Tech     No                           0     66
4    Commerce     73.30  Comm&Mgmt     No                           0   96.8
..        ...       ...        ...    ...                         ...    ...
213  Commerce     77.60  Comm&Mgmt     No                          11     91
214   Science     72.00   Sci&Tech     No                           1     74
215  Commerce     73.00  Comm&Mgmt    Yes                           1     59
216  Commerce     58.00  Comm&Mgmt     No                           1     70
217   Science     53.00  Comm&Mgmt     No                           0     89

    specialisation  mba_p  status  salary  agg_school_pct        bins
0              NaN  58.80       1       0           67.00  AvrageGood
1          Mkt&Fin  66.28       1  200000           79.33  AvrageGood
2              NaN   0.00       0       0           65.00  AvrageGood
3           Mkt&HR  59.43       0       0           56.00  AvrageGood
4          Mkt&Fin  55.50       1  425000           85.80  AvrageGood
..             ...    ...     ...     ...             ...         ...
213        Mkt&Fin  74.49       1  400000           80.60  AvrageGood
214        Mkt&Fin  53.62       1  275000           58.00  AvrageGood
215        Mkt&Fin  69.72       1  295000           67.00  AvrageGood
216         Mkt&HR  60.23       1  204000           74.00  AvrageGood
217         Mkt&HR  60.22       0       0           62.00  AvrageGood

[218 rows x 20 columns]
```

In [30]:
```
df1 = df1[['status', 'agg_school_pct']]
print(df1)
```

```
     status  agg_school_pct
0         1           67.00
1         1           79.33
2         0           65.00
3         0           56.00
4         1           85.80
..      ...             ...
213       1           80.60
214       1           58.00
215       1           67.00
216       1           74.00
217       0           62.00

[218 rows x 2 columns]
```

In [17]:
```
df1_norm = (df1-df1.min())/(df1.max()-df1.min())
print("Scaled Dataset Using Pandas")
df1_norm.head()
```

```
Scaled Dataset Using Pandas
```

Out[17]:

	status	agg_school_pct
0	1.0	0.749441
1	1.0	0.887360
2	0.0	0.727069
3	0.0	0.626398
4	1.0	0.959732

In [18]:
```
km=KMeans(n_clusters=2)
km
```

Out[18]:
```
KMeans(n_clusters=2)
```

In [19]:
```
#convert all in array /group
y_predicted = km.fit_predict(df1_norm[['status','agg_school_pct']])
y_predicted
```

Out[19]:
```
array([0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0,
       0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0,
       0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1,
       0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1,
       0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1,
       0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0,
       0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0,
       0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 0,
       1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 1,
       0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1])
```

In [20]:
```
#dataframe vS/group
df1_norm['cluster']=y_predicted
df1_norm.head()
```

Out[20]:

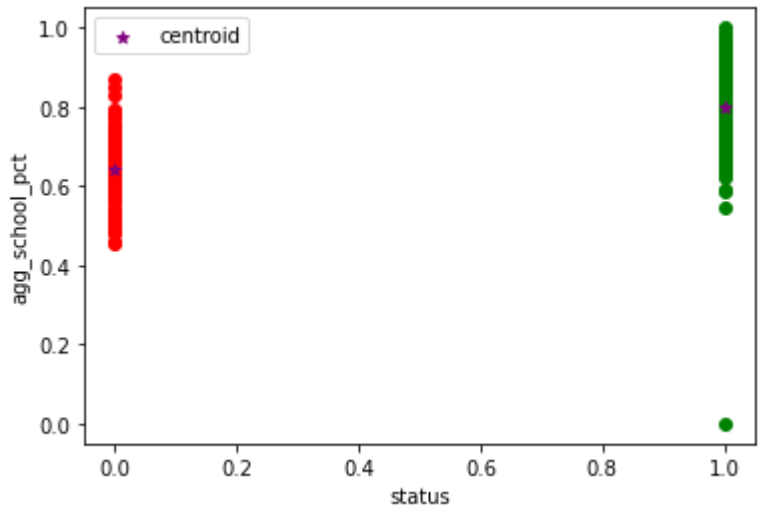
	status	agg_school_pct	cluster
0	1.0	0.749441	0
1	1.0	0.887360	0
2	0.0	0.727069	1
3	0.0	0.626398	1
4	1.0	0.959732	0

In [21]:
```
#Centroids
km.cluster_centers_
```

Out[21]:
```
array([[ 1.00000000e+00,  7.96719367e-01],
       [-9.99200722e-16,  6.42682294e-01]])
```

In [23]:
```
#datafram to two group and ploat Scatter plot
df = df1_norm[df1_norm.cluster==0]
df2 = df1_norm[df1_norm.cluster==1]
plt.scatter(df.status ,df['agg_school_pct'],color='green')
plt.scatter(df2.status ,df2['agg_school_pct'],color='red')
#ploatling centroids
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color='purple',mark
plt.xlabel('status')
plt.ylabel('agg_school_pct')
plt.legend()
```

Out[23]:
```
<matplotlib.legend.Legend at 0x20875f3af40>
```



In [24]:
```
df
```

Out[24]:

	status	agg_school_pct	cluster
0	1.0	0.749441	0
1	1.0	0.887360	0
4	1.0	0.959732	0
7	1.0	0.917226	0
8	1.0	0.000000	0
...	...	...	...
212	1.0	0.693512	0
213	1.0	0.901566	0
214	1.0	0.648770	0
215	1.0	0.749441	0
216	1.0	0.827740	0

149 rows × 3 columns

In [25]:
```
df2
```

Out[25]:

	status	agg_school_pct	cluster
2	0.0	0.727069	1
3	0.0	0.626398	1
5	0.0	0.615213	1
6	0.0	0.514541	1
9	0.0	0.648770	1
...	...	...	...
201	0.0	0.749441	1
204	0.0	0.606264	1
209	0.0	0.458613	1
211	0.0	0.480984	1
217	0.0	0.693512	1

69 rows × 3 columns

In [ ]:
```

```