

## Import required libraries

```
In [ ]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

## Read the data

```
In [3]: df=pd.read_csv("D:/project/Placement_Data_Full_Class.csv")
```

```
In [4]: # show the top 5 records
```

```
In [5]: df.head()
```

```
Out[5]:
```

|   | sl_no | gender | ssc_p | ssc_b   | hsc_p | hsc_b   | hsc_s    | degree_p | degree_t  | workex | etest_p |
|---|-------|--------|-------|---------|-------|---------|----------|----------|-----------|--------|---------|
| 0 | 1     | M      | 67.00 | Others  | 91.00 | Others  | Commerce | 58.00    | Sci&Tech  | No     | 55.     |
| 1 | 2     | M      | 79.33 | Central | 78.33 | Others  | Science  | 77.48    | Sci&Tech  | Yes    | 86.     |
| 2 | 3     | M      | 65.00 | Central | 68.00 | Central | Arts     | 64.00    | Comm&Mgmt | No     | 75.     |
| 3 | 4     | M      | 56.00 | Central | 52.00 | Central | Science  | 52.00    | Sci&Tech  | No     | 66.     |
| 4 | 5     | M      | 85.80 | Central | 73.60 | Central | Commerce | 73.30    | Comm&Mgmt | No     | 96.     |

```
In [6]: # show Last 3 records
```

```
In [7]: df.tail(3)
```

```
Out[7]:
```

|     | sl_no | gender | ssc_p | ssc_b   | hsc_p | hsc_b  | hsc_s    | degree_p | degree_t  | workex | etest_p |
|-----|-------|--------|-------|---------|-------|--------|----------|----------|-----------|--------|---------|
| 212 | 213   | M      | 67.0  | Others  | 67.0  | Others | Commerce | 73.0     | Comm&Mgmt | Yes    | !       |
| 213 | 214   | F      | 74.0  | Others  | 66.0  | Others | Commerce | 58.0     | Comm&Mgmt | No     | !       |
| 214 | 215   | M      | 62.0  | Central | 58.0  | Others | Science  | 53.0     | Comm&Mgmt | No     | !       |

```
In [8]: # show columns
```

```
In [9]: df.columns
```

```
Out[9]: Index(['sl_no', 'gender', 'ssc_p', 'ssc_b', 'hsc_p', 'hsc_b', 'hsc_s',
              'degree_p', 'degree_t', 'workex', 'etest_p', 'specialisation', 'mba_p',
              'status', 'salary'],
              dtype='object')
```

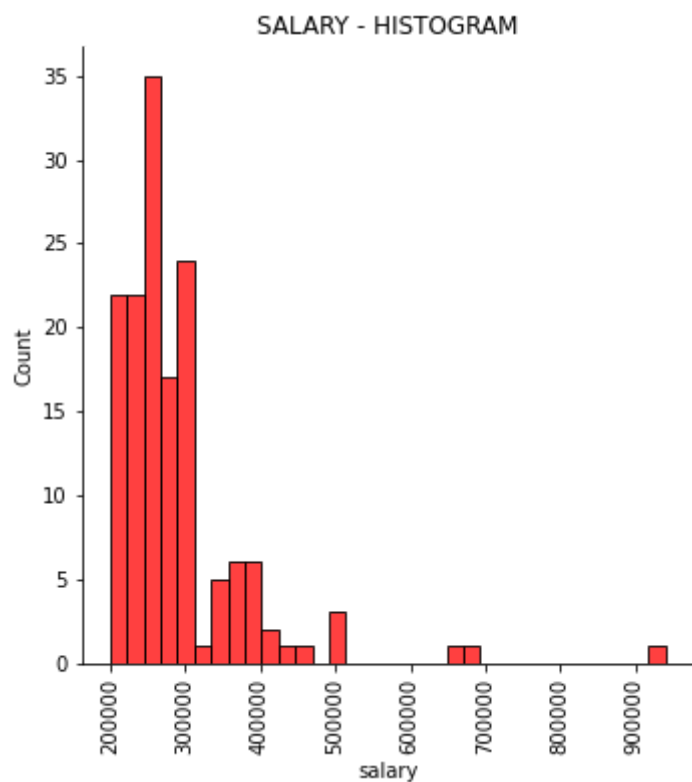
```
In [10]: # check for null data
```

```
In [12]: df.isna().sum()
```

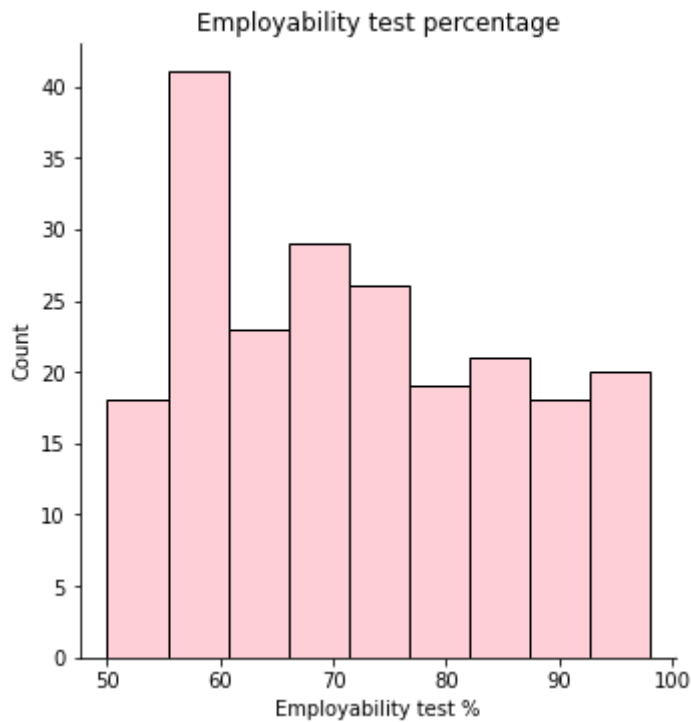
```
Out[12]: sl_no      0
gender      0
ssc_p      0
ssc_b      0
```

```
hsc_p      0
hsc_b      0
hsc_s      0
degree_p   0
degree_t   0
workex     0
etest_p    0
specialisation 0
mba_p      0
status     0
salary     67
dtype: int64
```

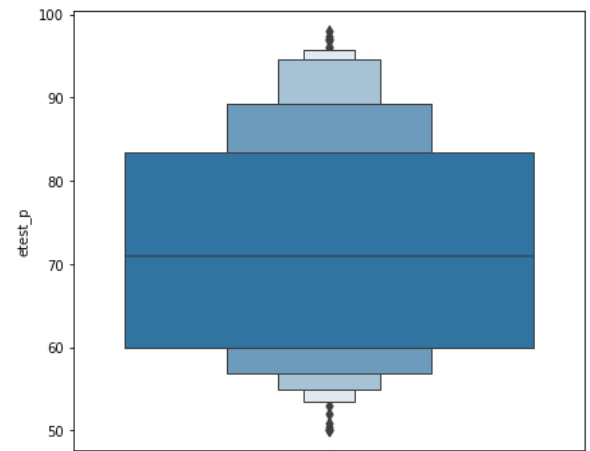
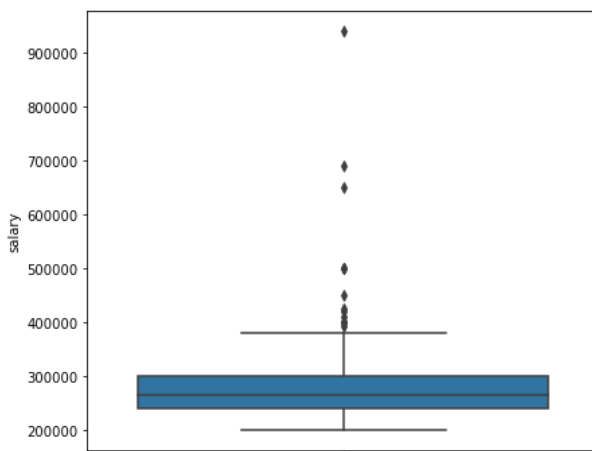
```
In [22]: sns.displot(df.salary,color='red')
plt.title('SALARY - HISTOGRAM')
plt.xticks(rotation='vertical')
plt.show()
```



```
In [25]: sns.displot(df.etest_p,color='pink')
plt.title('Employability test percentage')
plt.xlabel('Employability test %')
#plt.xticks(rotation='vertical')
plt.show()
```



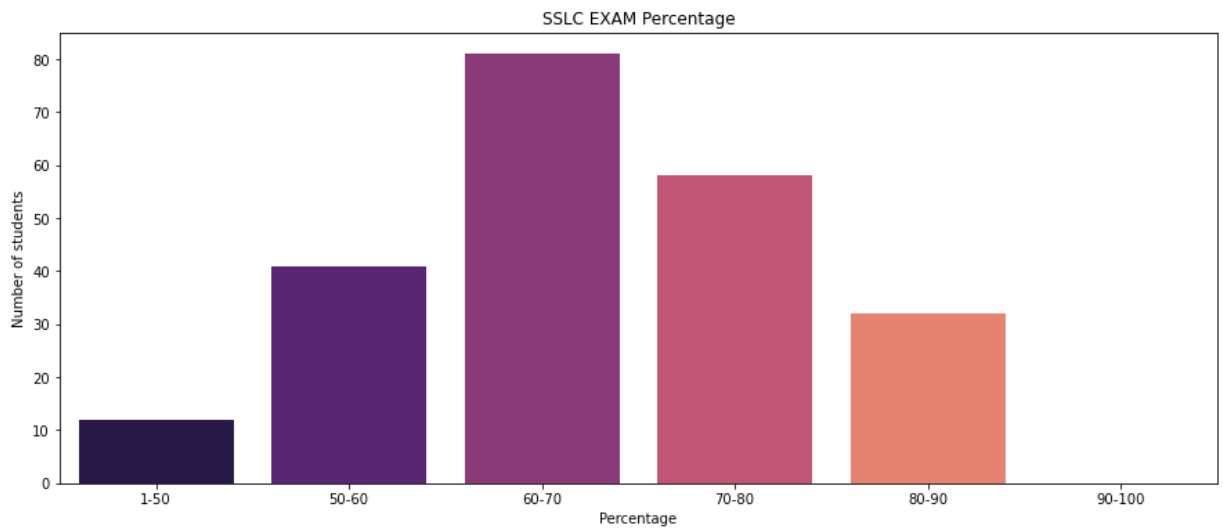
```
In [26]: plt.figure(figsize=(15,6))
plt.subplot(1,2,1)
sns.boxplot(y=df.salary)
plt.subplot(1,2,2)
sns.boxenplot(y=df.etest_p)
plt.show()
```



```
In [56]: sslc_50 = df.ssc_p[(df.ssc_p <=50)&(df.ssc_p >0)]
sslc50_60 = df.ssc_p[(df.ssc_p <=60)&(df.ssc_p >=50)]
sslc60_70 = df.ssc_p[(df.ssc_p <=70)&(df.ssc_p >=60)]
sslc70_80 = df.ssc_p[(df.ssc_p <=80)&(df.ssc_p >=70)]
sslc80_90 = df.ssc_p[(df.ssc_p <=90)&(df.ssc_p >=80)]
sslc90_100 = df.ssc_p[(df.ssc_p <=100)&(df.ssc_p >=90)]

sslc_x=['1-50', '50-60', '60-70', '70-80', '80-90', '90-100']
sslc_y=[len(sslc_50.values),len(sslc50_60.values),len(sslc60_70.values),len(sslc70_80_90.values),len(sslc80_90.values),len(sslc90_100.values)]

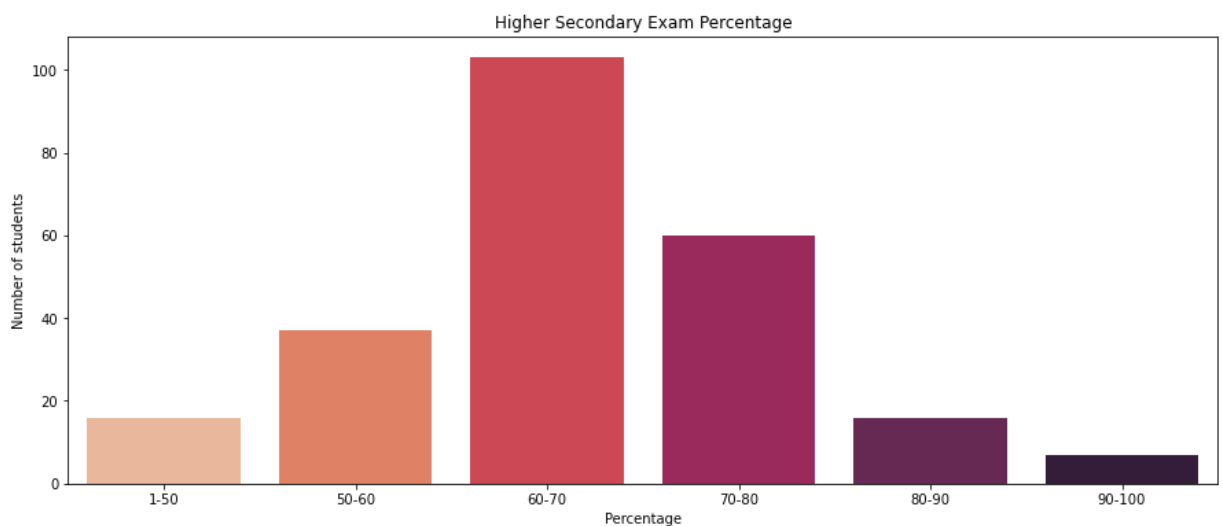
plt.figure(figsize=(15,6))
sns.barplot(x=sslc_x,y=sslc_y,palette='magma')
plt.title('SSLC EXAM Percentage')
plt.xlabel('Percentage')
plt.ylabel('Number of students')
plt.show()
```



```
In [30]: hsc_50 = df.hsc_p[(df.hsc_p <=50)&(df.hsc_p >0)]
hsc50_60 = df.hsc_p[(df.hsc_p <=60)&(df.hsc_p >=50)]
hsc60_70 = df.hsc_p[(df.hsc_p <=70)&(df.hsc_p >=60)]
hsc70_80 = df.hsc_p[(df.hsc_p <=80)&(df.hsc_p >=70)]
hsc80_90 = df.hsc_p[(df.hsc_p <=90)&(df.hsc_p >=80)]
hsc90_100 = df.hsc_p[(df.hsc_p <=100)&(df.hsc_p >=90)]

hsc_x=['1-50', '50-60', '60-70', '70-80', '80-90', '90-100']
hsc_y=[len(hsc_50.values),len(hsc50_60.values),len(hsc60_70.values),len(hsc70_80),len(hsc80_90),len(hsc90_100)]

plt.figure(figsize=(15,6))
sns.barplot(x=hsc_x,y=hsc_y,palette='rocket_r')
plt.title('Higher Secondary Exam Percentage')
plt.xlabel('Percentage')
plt.ylabel('Number of students')
plt.show()
```

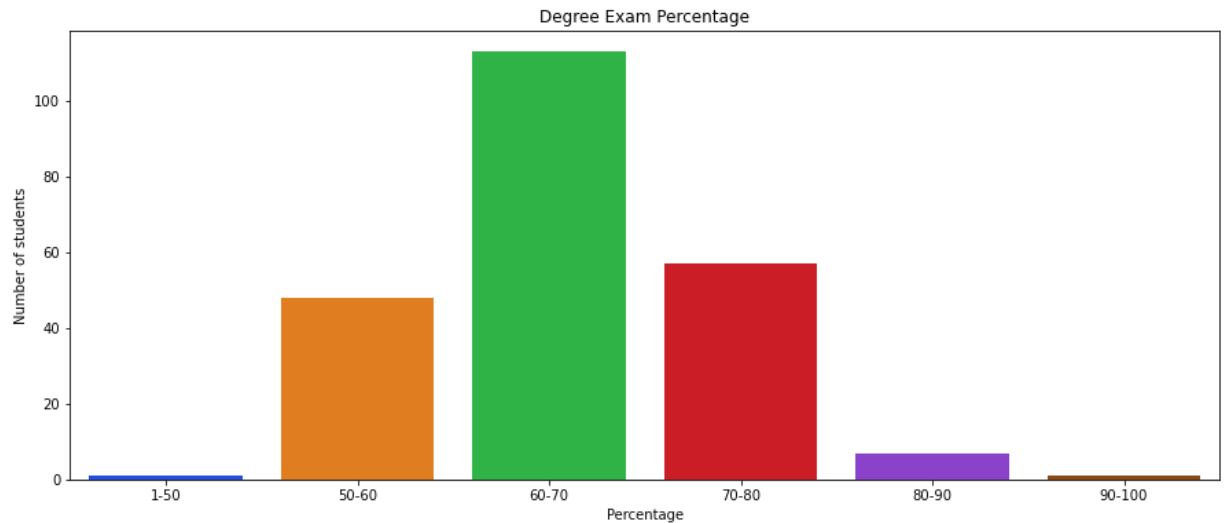


```
In [32]: degree_50 = df.degree_p[(df.degree_p <=50)&(df.degree_p >0)]
degree50_60 = df.degree_p[(df.degree_p <=60)&(df.degree_p >=50)]
degree60_70 = df.degree_p[(df.degree_p <=70)&(df.degree_p >=60)]
degree70_80 = df.degree_p[(df.degree_p <=80)&(df.degree_p >=70)]
degree80_90 = df.degree_p[(df.degree_p <=90)&(df.degree_p >=80)]
degree90_100 = df.degree_p[(df.degree_p <=100)&(df.degree_p >=90)]

degree_x=['1-50', '50-60', '60-70', '70-80', '80-90', '90-100']
degree_y=[len(degree_50.values),len(degree50_60.values),len(degree60_70.values),len(degree70_80.values),len(degree80_90.values),len(degree90_100.values)]

plt.figure(figsize=(15,6))
sns.barplot(x=degree_x,y=degree_y,palette='bright')
```

```
plt.title('Degree Exam Percentage')
plt.xlabel('Percentage')
plt.ylabel('Number of students')
plt.show()
```

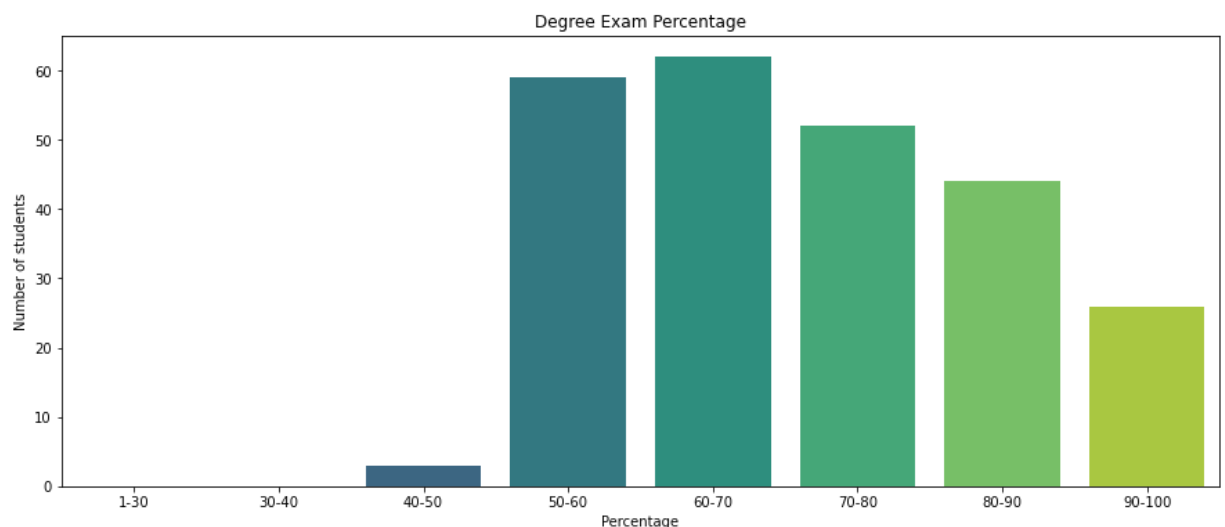


In [37]:

```
etest30 = df.etest_p[(df.etest_p <=30)&(df.etest_p >0)]
etest30_40 = df.etest_p[(df.etest_p <=40)&(df.etest_p >30)]
etest40_50 = df.etest_p[(df.etest_p <=50)&(df.etest_p >40)]
etest50_60 = df.etest_p[(df.etest_p <=60)&(df.etest_p >=50)]
etest60_70 = df.etest_p[(df.etest_p <=70)&(df.etest_p >=60)]
etest70_80 = df.etest_p[(df.etest_p <=80)&(df.etest_p >=70)]
etest80_90 = df.etest_p[(df.etest_p <=90)&(df.etest_p >=80)]
etest90_100 = df.etest_p[(df.etest_p <=100)&(df.etest_p >=90)]

etest_x=['1-30', '30-40', '40-50', '50-60', '60-70', '70-80', '80-90', '90-100']
etest_y=[len(etest30.values), len(etest30_40.values), len(etest40_50.values), len(etest50_60.values), len(etest60_70.values), len(etest70_80.values), len(etest80_90.values), len(etest90_100.values)]

plt.figure(figsize=(15,6))
sns.barplot(x=etest_x,y=etest_y,palette='viridis')
plt.title('Degree Exam Percentage')
plt.xlabel('Percentage')
plt.ylabel('Number of students')
plt.show()
```

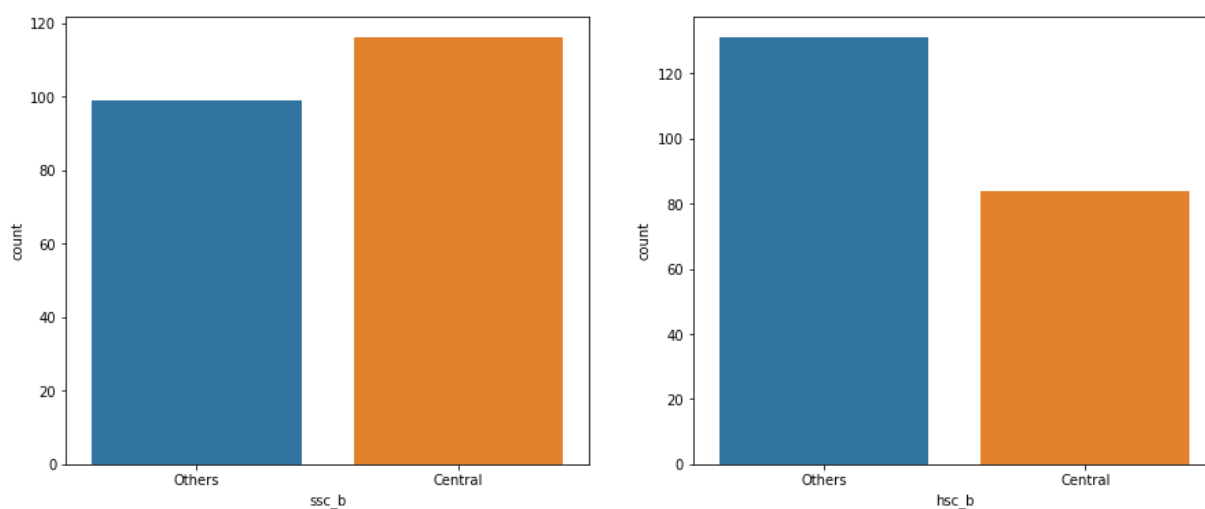


In [43]:

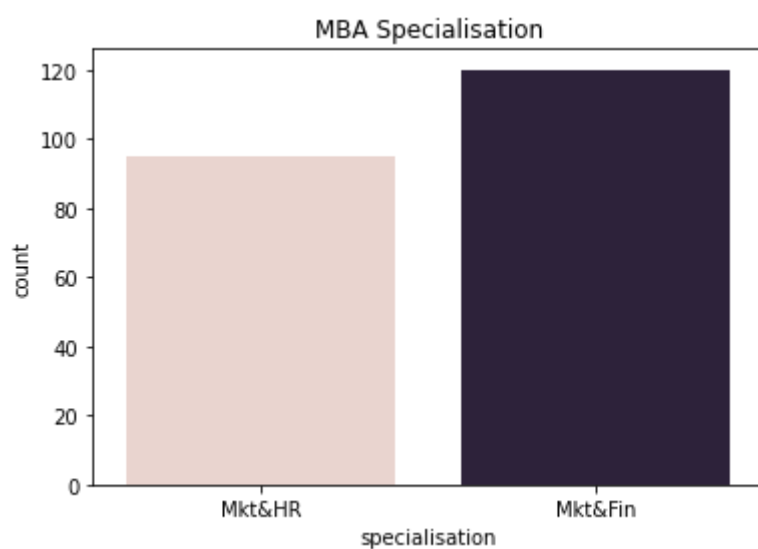
```
plt.figure(figsize=(15,6))
plt.subplot(1,2,1)
sns.countplot(x=df.ssc_b)

plt.subplot(1,2,2)
sns.countplot(x=df.hsc_b)
```

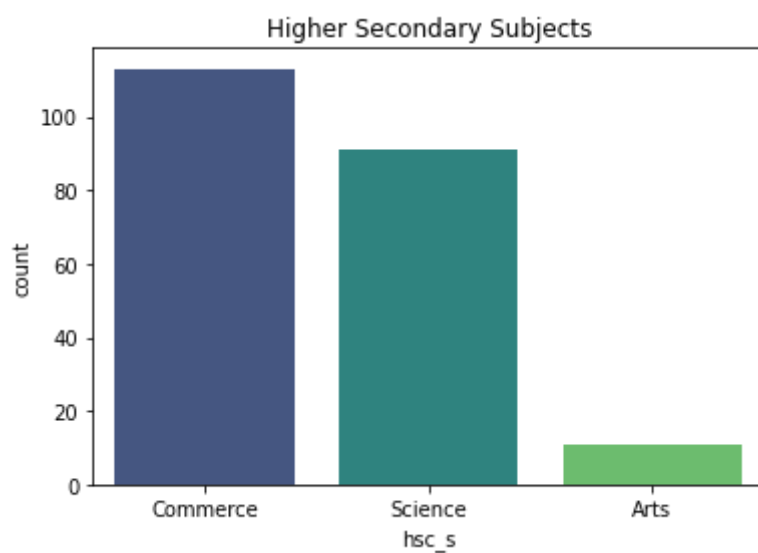
```
plt.show()
```



```
In [60]: sns.countplot(x=df['specialisation'],palette='ch:.0015')  
plt.title("MBA Specialisation")  
plt.show()
```



```
In [52]: sns.countplot(x=df.hsc_s,palette='viridis')  
plt.title('Higher Secondary Subjects')  
plt.show()
```



```
In [20]: df.columns
```

```
Out[20]: Index(['sl_no', 'gender', 'ssc_p', 'ssc_b', 'hsc_p', 'hsc_b', 'hsc_s',
               'degree_p', 'degree_t', 'workex', 'etest_p', 'specialisation', 'mba_p',
               'status', 'salary'],
              dtype='object')
```

## 3D clustering using KMeans

```
In [67]: df1=df[['gender','ssc_p','hsc_p','degree_p']]
```

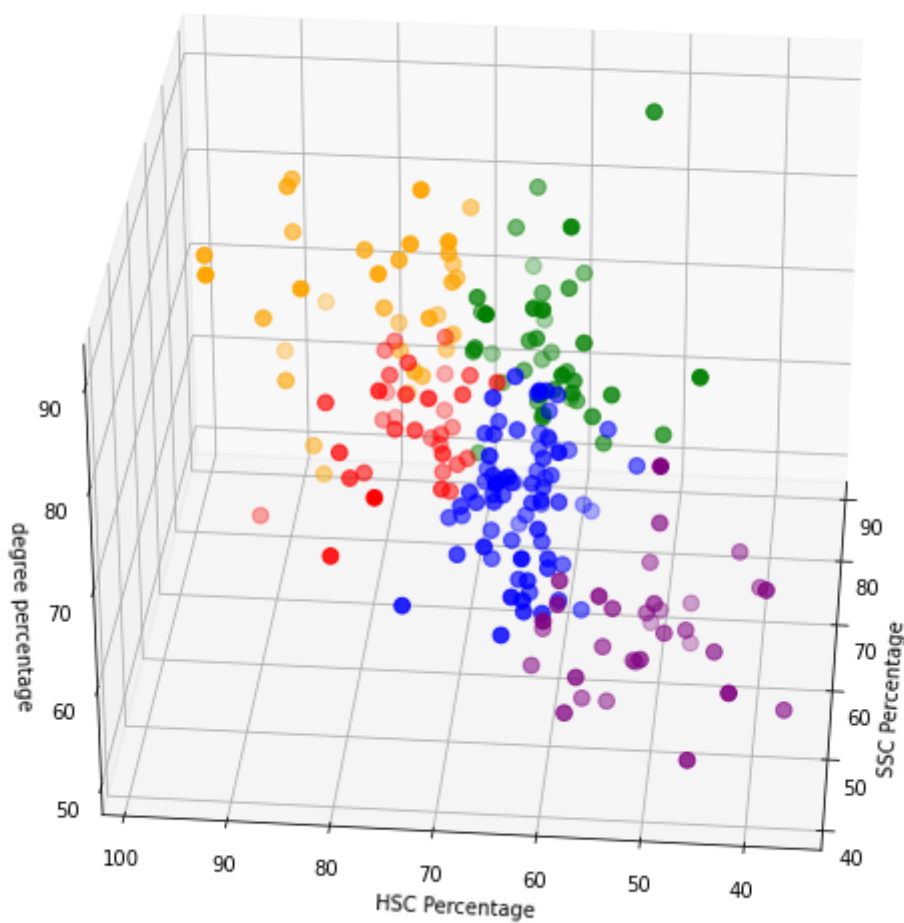
```
In [68]: from sklearn.cluster import KMeans
km = KMeans(n_clusters=5)
clusters = km.fit_predict(df1.iloc[:,1:])
df1['label'] = clusters

from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

fig = plt.figure(figsize=(20,10))
ax = fig.add_subplot(111, projection='3d')
ax.scatter(df1.ssc_p[df1.label == 0], df1['hsc_p'][df1.label ==0], df1['degree_p'][d
ax.scatter(df1.ssc_p[df1.label == 1], df1['hsc_p'][df1.label ==1], df1['degree_p'][d
ax.scatter(df1.ssc_p[df1.label == 2], df1['hsc_p'][df1.label ==2], df1['degree_p'][d
ax.scatter(df1.ssc_p[df1.label == 3], df1['hsc_p'][df1.label ==3], df1['degree_p'][d
ax.scatter(df1.ssc_p[df1.label == 4], df1['hsc_p'][df1.label ==4], df1['degree_p'][d
ax.view_init(30,185)
plt.xlabel('SSC Percentage')
plt.ylabel('HSC Percentage')
ax.set_zlabel('degree percentage')
plt.show()
```

<ipython-input-68-c2ed926f681c>:4: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)  
df1['label'] = clusters



```
In [21]: df.head(1)
```

Out[21]:

|   | sl_no | gender | ssc_p | ssc_b  | hsc_p | hsc_b  | hsc_s    | degree_p | degree_t | workex | etest_p | sp |
|---|-------|--------|-------|--------|-------|--------|----------|----------|----------|--------|---------|----|
| 0 | 1     | M      | 67.0  | Others | 91.0  | Others | Commerce | 58.0     | Sci&Tech | No     | 55.0    |    |

```
In [ ]:
```