CIA Component 3

Prepare a case study report on specific business problem and solution.

The used dataset is about the campus recruitment of students of a B-school.

AIM : To analyse the dataset and find the factors affecting the placement and salary of recruited students.

Importing the required libraries

In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Importing the dataset

In [3]:
```
data=pd.read_csv(r"Placement_Data_Full_Class.csv")
```

In [4]:
```
data.head()
```

Out[4]:

	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
0	1	M	67.00	Others	91.00	Others	Commerce	58.00	Sci&Tech	No	55.0	Mkt&HR	58.80	Placed	270000.0
1	2	M	79.33	Central	78.33	Others	Science	77.48	Sci&Tech	Yes	86.5	Mkt&Fin	66.28	Placed	200000.0
2	3	M	65.00	Central	68.00	Central	Arts	64.00	Comm&Mgmt	No	75.0	Mkt&Fin	57.80	Placed	250000.0
3	4	M	56.00	Central	52.00	Central	Science	52.00	Sci&Tech	No	66.0	Mkt&HR	59.43	Not Placed	NaN
4	5	M	85.80	Central	73.60	Central	Commerce	73.30	Comm&Mgmt	No	96.8	Mkt&Fin	55.50	Placed	425000.0

The abbreviation in the dataset are as follows:

sl_n0 = Serial Number
ssc_p = Senior school percentage
ssc_b = Senior school board
hsc_p = High school percentage
hsc_b = High school board
degree_p = Degree(UG) percentage
degree_t = Degree type
workex = Work experience
etest_p = employability test percentage
mba_p =MBA(Post Graduation) percentage

***Pre processing of data ***

In [5]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 215 entries, 0 to 214
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   sl_no           215 non-null    int64  
 1   gender          215 non-null    object 
 2   ssc_p           215 non-null    float64
 3   ssc_b           215 non-null    object 
 4   hsc_p           215 non-null    float64
 5   hsc_b           215 non-null    object 
 6   hsc_s           215 non-null    object 
 7   degree_p        215 non-null    float64
 8   degree_t        215 non-null    object 
 9   workex          215 non-null    object 
 10  etest_p         215 non-null    float64
 11  specialisation  215 non-null    object 
 12  mba_p           215 non-null    float64
 13  status          215 non-null    object 
 14  salary          148 non-null    float64
dtypes: float64(6), int64(1), object(8)
memory usage: 25.3+ KB
```

In [7]:
```
data.isna().sum()
```

Out[7]:
```
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
```

We find that salary column has 67 null values which correspond to students who are not placed yet

So we fill the null values with 0.

In [8]:
```
data.fillna(0)
```

Out[8]:

	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
0	1	M	67.00	Others	91.00	Others	Commerce	58.00	Sci&Tech	No	55.0	Mkt&HR	58.80	Placed	270000.0
1	2	M	79.33	Central	78.33	Others	Science	77.48	Sci&Tech	Yes	86.5	Mkt&Fin	66.28	Placed	200000.0
2	3	M	65.00	Central	68.00	Central	Arts	64.00	Comm&Mgmt	No	75.0	Mkt&Fin	57.80	Placed	250000.0
3	4	M	56.00	Central	52.00	Central	Science	52.00	Sci&Tech	No	66.0	Mkt&HR	59.43	Not Placed	0.0
4	5	M	85.80	Central	73.60	Central	Commerce	73.30	Comm&Mgmt	No	96.8	Mkt&Fin	55.50	Placed	425000.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
210	211	M	80.60	Others	82.00	Others	Commerce	77.60	Comm&Mgmt	No	91.0	Mkt&Fin	74.49	Placed	400000.0
211	212	M	58.00	Others	60.00	Others	Science	72.00	Sci&Tech	No	74.0	Mkt&Fin	53.62	Placed	275000.0
212	213	M	67.00	Others	67.00	Others	Commerce	73.00	Comm&Mgmt	Yes	59.0	Mkt&Fin	69.72	Placed	295000.0
213	214	F	74.00	Others	66.00	Others	Commerce	58.00	Comm&Mgmt	No	70.0	Mkt&HR	60.23	Placed	204000.0
214	215	M	62.00	Central	58.00	Others	Science	53.00	Comm&Mgmt	No	89.0	Mkt&HR	60.22	Not Placed	0.0

215 rows × 15 columns

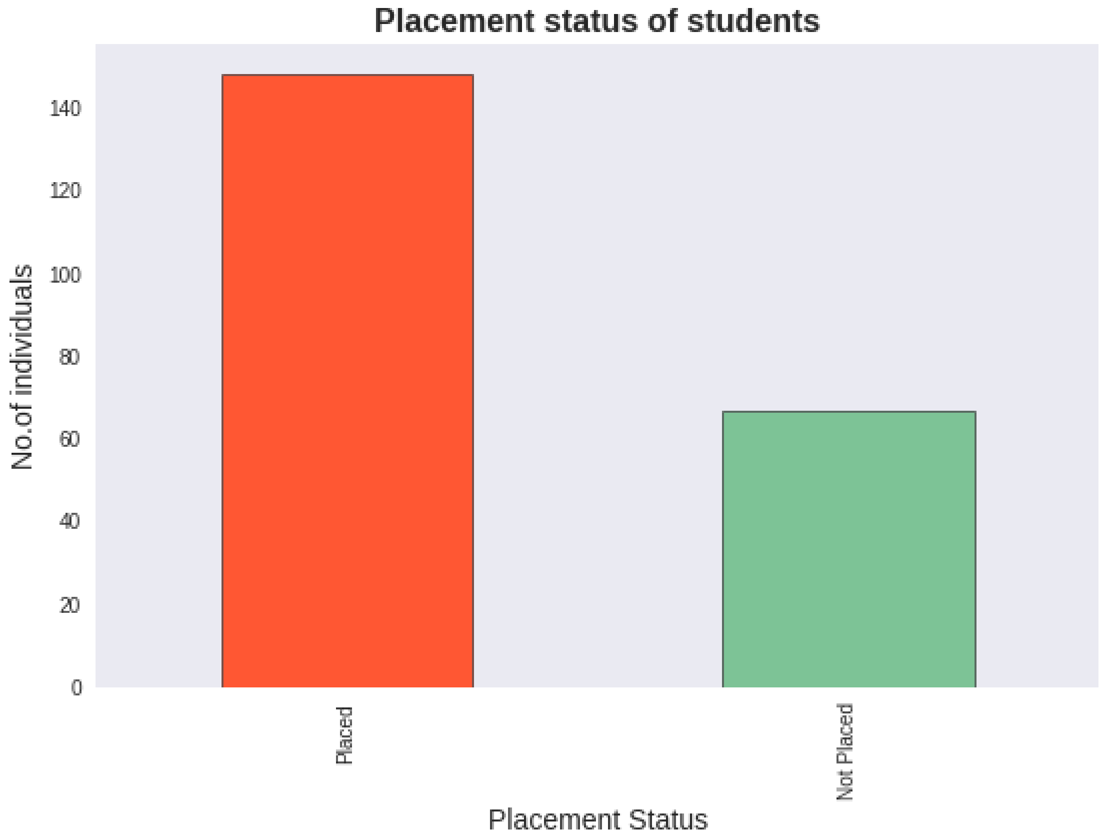
Analysis

In [ ]:
```
data['status'].value_counts()*100/len(data)
```

Out[143]:
```
Placed        68.837209
Not Placed    31.162791
Name: status, dtype: float64
```

In [ ]:
```
data['status'].value_counts().plot(kind='bar',color=('#FF5733','#7DC396'),ec='k')
plt.xlabel('Placement Status',size=14)
plt.ylabel('No.of individuals',size=14)
plt.title('Placement status of students',size=16,fontweight='bold')
```

Out[144]:
```
Text(0.5, 1.0, 'Placement status of students')
```



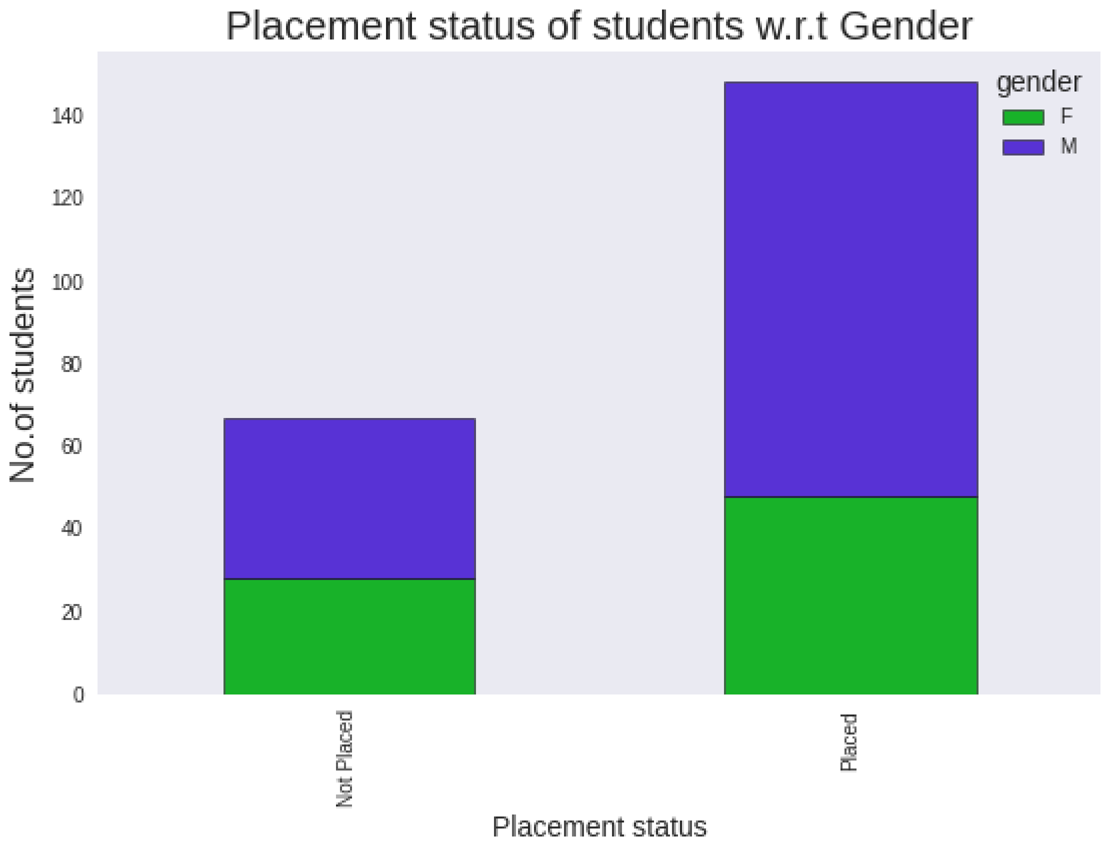
***Placement of students w.r.t Gender ***

In [ ]:
```
data['gender'].groupby(by=data['status']).value_counts()*100/len(data)
```

Out[146]:
```
status      gender
Not Placed  M         18.139535
            F         13.023256
Placed      M         46.511628
            F         22.325581
Name: gender, dtype: float64
```

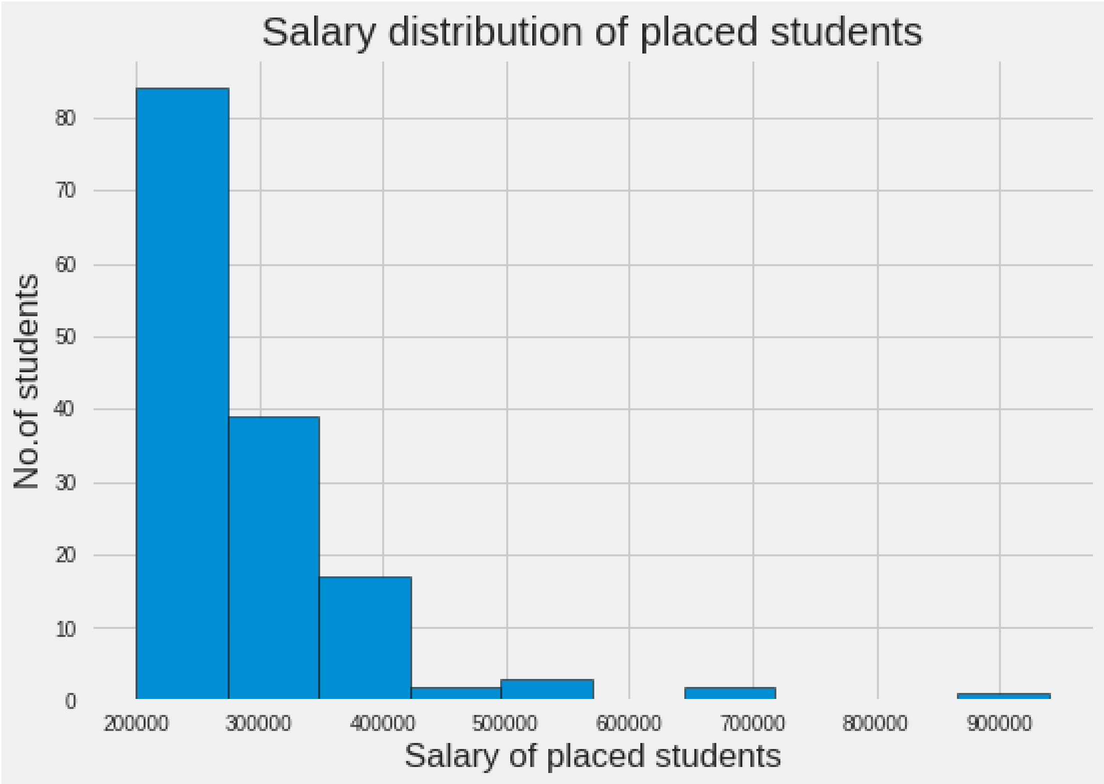
In [ ]:
```
data['gender'].groupby(by=data['status']).value_counts().unstack().plot(kind='bar',stacked=True,color=('#18B229','#5832D!
plt.xlabel('Placement status',size=14)
plt.ylabel('No.of students')
plt.title('Placement status of students w.r.t Gender')
```

Out[147]:
```
Text(0.5, 1.0, 'Placement status of students w.r.t Gender')
```



Let's have a look at the salaries of the placed students.

In [ ]:
```
plt.style.use('fivethirtyeight')
plt.hist(data['salary'],bins=10,ec='k')
plt.xlabel('Salary of placed students')
plt.ylabel('No.of students')
plt.title('Salary distribution of placed students')
plt.show()
```

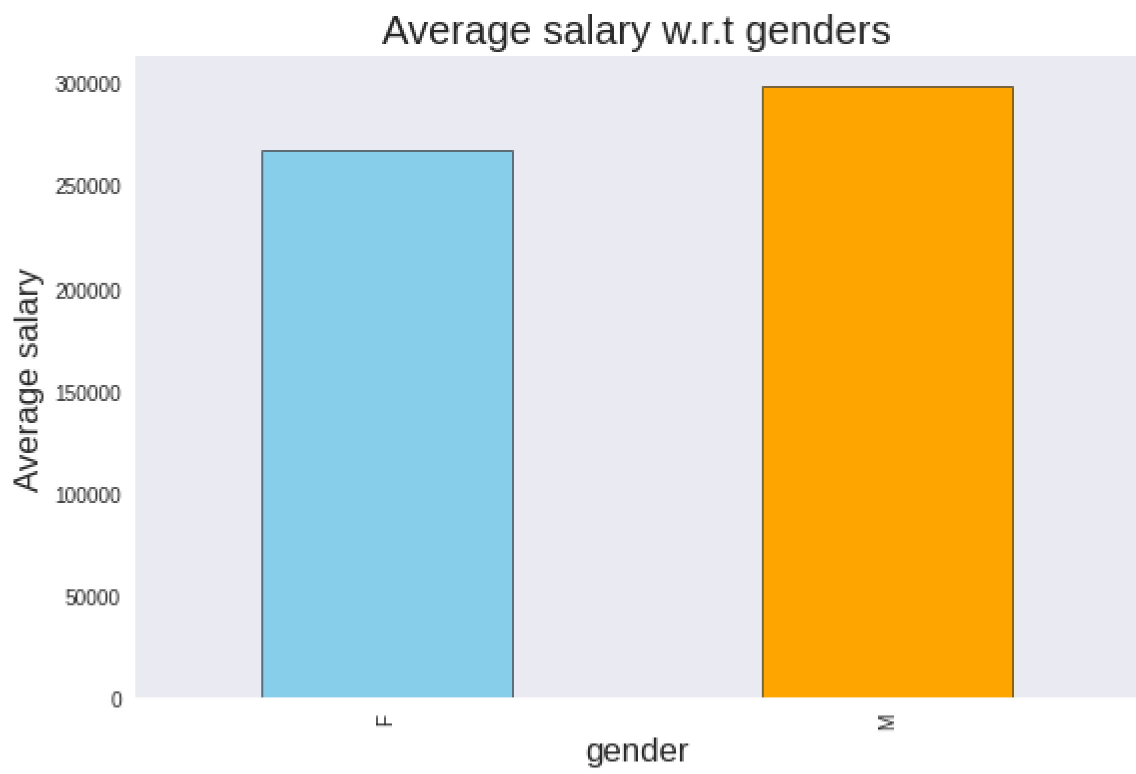


We find that the salary of majority of the placed students lie in the range of 200000-40000.

Almost 40% of the students have their in the range of 20000-30000

In [ ]:
```
data['salary'].groupby(by=data['gender']).mean().plot(kind='bar',color=('skyblue','orange'),ec='k')
plt.ylabel('Average salary')
plt.title ('Average salary w.r.t genders')
```

Out[150]:
```
Text(0.5, 1.0, 'Average salary w.r.t genders')
```

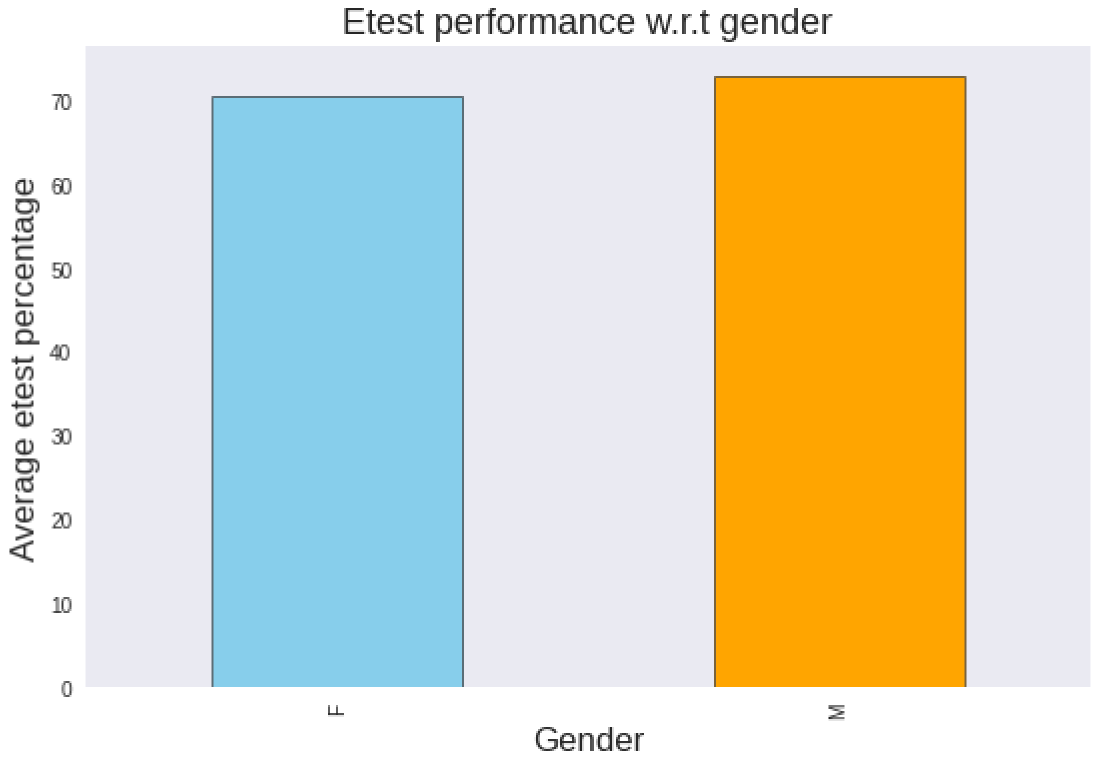


The highest salary received by a male student is around 95k whereas highest pay for female student is just above 60k.

We also find that the average salary for a male student is more than that of the female student.

In [ ]:
```
data['etest_p'].groupby(by=data['gender']).mean().plot(kind='bar',color=('skyblue','orange'),ec='k')
plt.xlabel('Gender')
plt.ylabel('Average etest percentage')
plt.title('Etest performance w.r.t gender',size=18)
```

Out[154]: `Text(0.5, 1.0, 'Etest performance w.r.t gender')`



We find that the average percentage of female students is higher in MBA examination

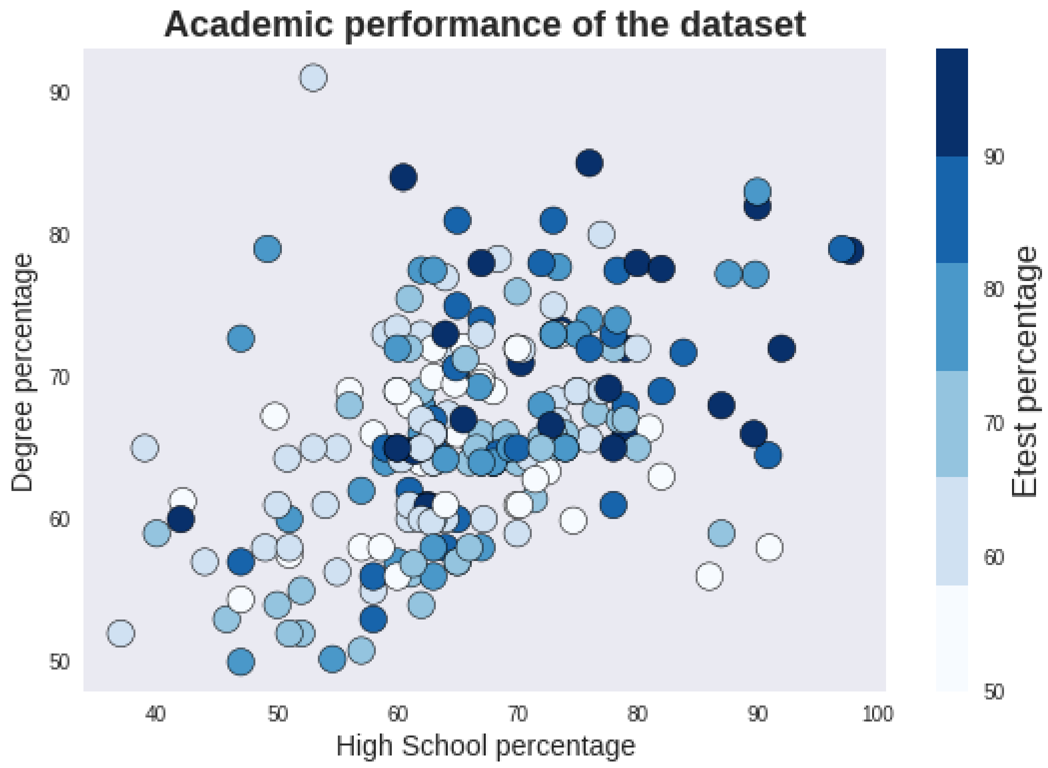
Whereas male students have a higher average percentage in employability test

This could be a reason for higher pay

***Academic performance of students at various levels ***

Lets have a look at the relationship b/w the percentage acquired from higher secondary to the employability test

In [ ]:
```
plt.scatter(data['hsc_p'],data['degree_p'],s=180,ec='k',c=data['etest_p'],cmap=plt.cm.get_cmap('Blues', 6))
color_bar=plt.colorbar()
plt.xlabel('High School percentage',size=14)
plt.ylabel('Degree percentage',size=14)
plt.title('Academic performance of the dataset',size=18,fontweight='bold')
color_bar.set_label('Etest percentage')
plt.show()
```



In [11]:
```
data.drop(['ssc_b','hsc_b','sl_no','salary'],axis=1,inplace=True)
```

We find that the grades acquired at higher secondary level and degree examination have an almost linear relationship

While the etest percentage varies, with students with low hsc and degree percentages scoring well in the employability test

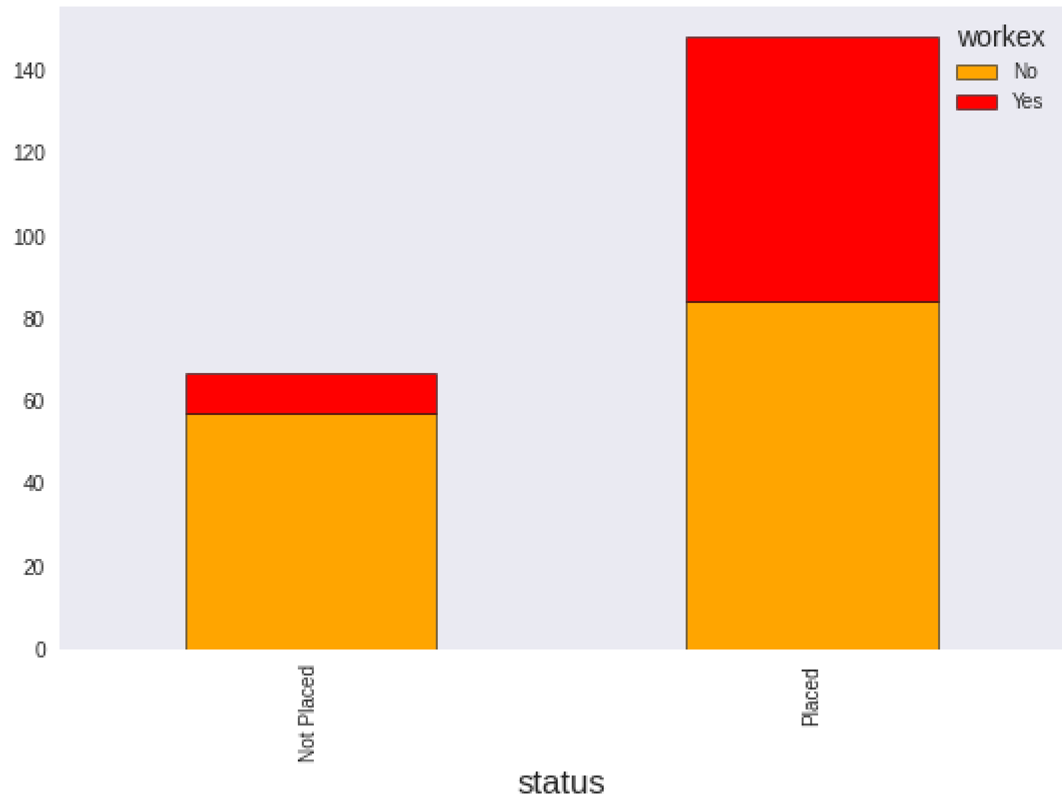
Lets have a look at the impact of work experience on placement and salary

In [ ]:
```
data['workex'].groupby(by=data['status']).value_counts()
```

Out[157]:
```
status      workex
Not Placed  No        57
            Yes       10
Placed      No        84
            Yes       64
Name: workex, dtype: int64
```

In [ ]:
```
data['workex'].groupby(by=data['status']).value_counts().unstack().plot(kind='bar',stacked=True,color=('orange','red'),e
```

Out[158]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f9f02c82810>`



In [ ]:
```
plt.style.use('seaborn-dark')
data['salary'].groupby(by=data['workex']).mean().plot(kind='bar',color=('#D7EB2D','#2DEB4F'),ec='k')
plt.xlabel('Work experience')
plt.ylabel('Average Salary')
plt.title('Average salary w.r.t Work experience')
```

Out[159]: `Text(0.5, 1.0, 'Average salary w.r.t Work experience')`

We find that students with prior work experience had higher probability getting placed and a higher salary with respect to students without experience.

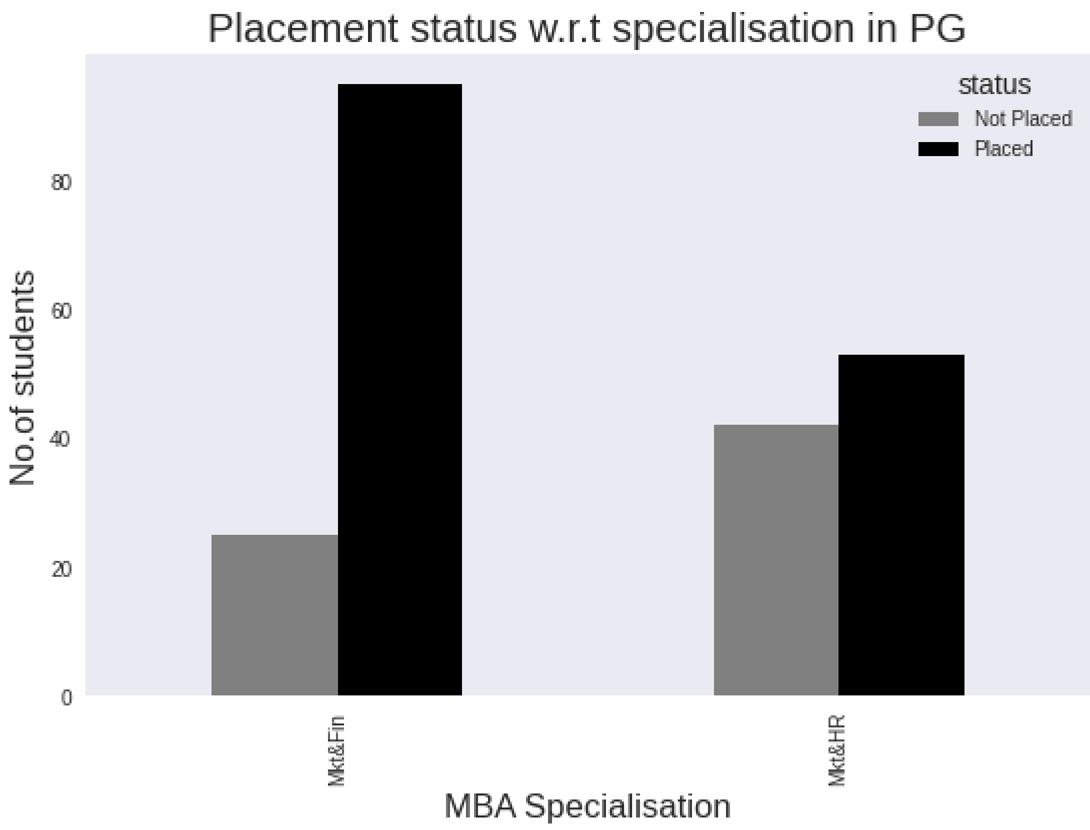
MBA specialisation and its impact

In [ ]:
```
data['specialisation'].groupby(by=data['status']).value_counts()
```

Out[122]:
```
status      specialisation
Not Placed  Mkt&HR            42
            Mkt&Fin           25
Placed      Mkt&Fin           95
            Mkt&HR            53
Name: specialisation, dtype: int64
```

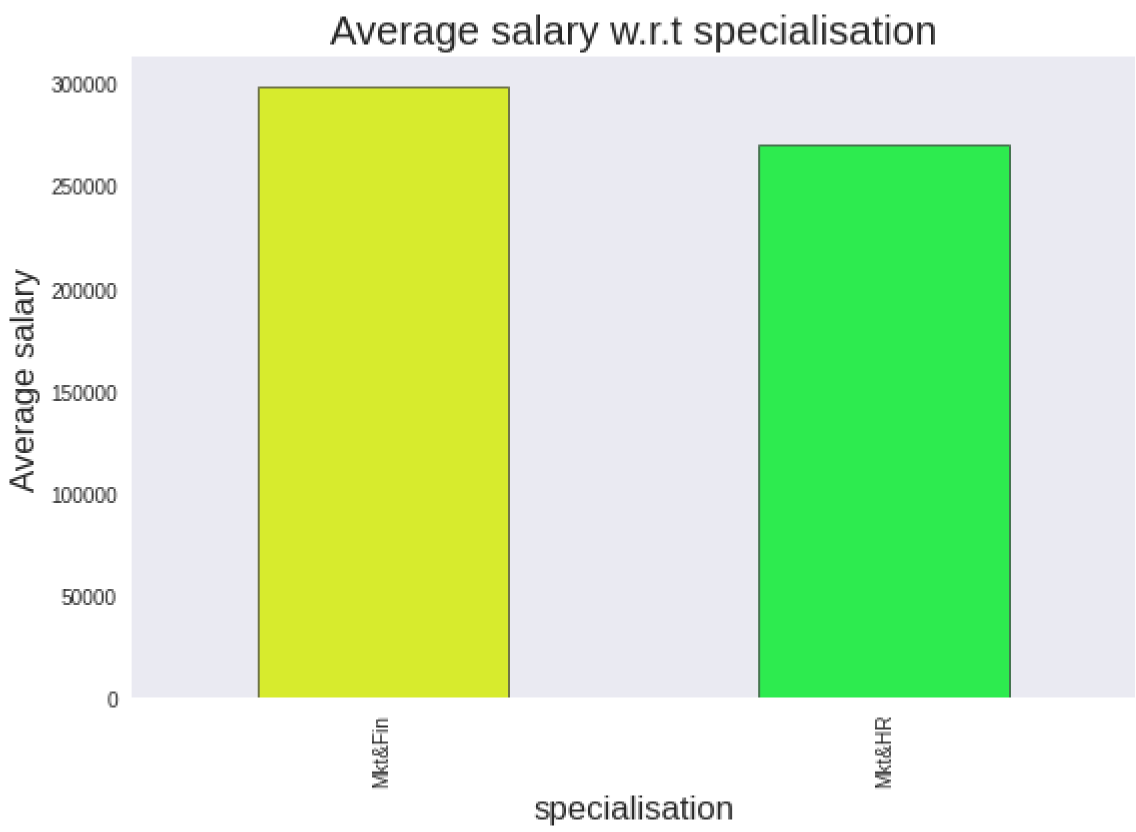
In [ ]:
```
data['status'].groupby(by=data['specialisation']).value_counts().unstack().plot(kind='bar',color=('grey','k'))
plt.xlabel('MBA Specialisation')
plt.ylabel('No.of students')
plt.title('Placement status w.r.t specialisation in PG')
```

Out[160]:
```
Text(0.5, 1.0, 'Placement status w.r.t specialisation in PG')
```



In [ ]:
```
data['salary'].groupby(by=data['specialisation']).mean().plot(kind='bar',color=('#D7EB2D','#2DEB4F'),ec='k')
plt.xlabel('specialisation')
plt.ylabel('Average salary')
plt.title('Average salary w.r.t specialisation')
```

Out[161]:
```
Text(0.5, 1.0, 'Average salary w.r.t specialisation')
```



We find that a specialisation in Marketing and finance improves chances of getting recruited and offers a bigger paycheck.

Model building

In [10]:
```
from sklearn.preprocessing import LabelEncoder
```

In [ ]:
```
# Label Encoding to convert categorical to numerical values
```

In [12]:
```
labelencoder=LabelEncoder()
data['specialisation']=labelencoder.fit_transform(data['specialisation'])
data['workex']=labelencoder.fit_transform(data['workex'])
data['gender']=labelencoder.fit_transform(data['gender'])
data['hsc_s']=labelencoder.fit_transform(data['hsc_s'])
data['degree_t']=labelencoder.fit_transform(data['degree_t'])
data['status']=labelencoder.fit_transform(data['status'])
```

In [ ]:
```
#Seperating the target variable
```

In [13]:
```
target=data['status']
inputs=data.drop(['status'],axis=1)
```

Splitting the dataset into train and test datasets

In [14]:
```
from sklearn.model_selection import train_test_split
inputs_train, inputs_test, target_train, target_test = train_test_split(inputs, target, test_size = 0.30, random_state =
```

In [15]:
```
from sklearn.svm import SVC
```

Fitting the dataset to the model

In [21]:
```
support_vector_classifier = SVC(kernel='rbf')
support_vector_classifier.fit(inputs_train,target_train)
target_pred_svc = support_vector_classifier.predict(inputs_test)
```

Confusion metrics for the model

In [22]:
```
from sklearn.metrics import confusion_matrix
cm_support_vector_classifier = confusion_matrix(target_test,target_pred_svc)
print(cm_support_vector_classifier,end='\n\n')
```

```
[[ 8 11]
 [ 3 43]]
```

In [23]:
```
numerator = cm_support_vector_classifier[0][0] + cm_support_vector_classifier[1][1]
denominator = sum(cm_support_vector_classifier[0]) + sum(cm_support_vector_classifier[1])
acc_svc = (numerator/denominator) * 100
print("Accuracy : ",round(acc_svc,2),"%")
```

```
Accuracy :  78.46 %
```

In [24]:
```
from sklearn.model_selection import cross_val_score
cross_val_svc = cross_val_score(estimator = SVC(kernel = 'rbf'), X = inputs_train, y = target_train, cv = 10, n_jobs = -
print("Cross Validation Accuracy : ",round(cross_val_svc.mean() * 100 , 2),"%")
```

```
Cross Validation Accuracy :  86.67 %
```

The accuracy and cross validation score for the model is high so the model can be validated

*Conclusion *

The recruitement and a higher salary is positively impacted by the following:

- A specialisation in Marketing and Finance
- Prior work experience
- Higher Employability test and post graduatuion percentage

There is a gender bias in recruitment and salary package which needs to be managed.

Type *Markdown* and LaTeX: α^2