EDA - Exploratory Data Analysis

EDA helps us to analyse the data by visulazing the graphs and plots. It tells the relationship between features of any dataset. As compared to looking at the whole data, visualizing is easy. So, we will do this on a givem dataset.

In [1]:
```
# importing all the libraries

import warnings
import random
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn import preprocessing
from scipy.stats import norm
import matplotlib.pyplot as plt
from scipy.stats import chi2
from scipy.stats import chi2_contingency

warnings.filterwarnings('ignore')
```

In [2]:
```
# read excel data

data = pd.read_excel('data.xlsx')
```

Introduction about Dataset - This dataset was released by Aspiring Minds from the Aspiring Mind Employment Outcome 2015 (AMEO). The study is primarily limited only to students with engineering disciplines.

The dataset contains the employment outcomes of engineering graduates as dependent variables (Salary, Job Titles, and Job Locations) along with the standardized scores from three different areas – cognitive skills, technical skills and personality skills.

The dataset also contains demographic features. The dataset contains around 40 independent variables and 4000 data points. The independent variables are both continuous and categorical in nature. The dataset contains a unique identifier for each candidate.

We will analyse the data by plotting different different graphs 😉.

Info about Data - Firstly we will see what our data contains, what is the size of the data and is there any missing value or not. What our data describes and many more things using pandas library which makes this work very easy 😎😎😎.

In [3]:
```
pd.set_option('display.max_columns', None)
```

In [4]:
```
data.head() # We can see top 5 rows of dataset using head function.
```

Out[4]:

	Unnamed: 0	ID	Salary	DOJ	DOL	Designation	JobCity	Gender	DOB	10percer
0	train	203097	420000	2012-06-01	present	senior quality engineer	Bangalore	f	1990-02-19	
1	train	579905	500000	2013-09-01	present	assistant manager	Indore	m	1989-10-04	
2	train	810601	325000	2014-06-01	present	systems engineer	Chennai	f	1992-08-03	
3	train	267447	1100000	2011-07-01	present	senior software engineer	Gurgaon	m	1989-12-05	
4	train	343523	200000	2014-03-01	2015-03-01 00:00:00	get	Manesar	m	1991-02-27	

In [5]:
```
data.shape # It tells us the size of the data. In this data we have 3998 rows
 and 39 features as indexing starts from 0.
```

Out[5]: (3998, 39)

In [6]:
```
data.describe() # Using describe function we can find the statistical values o
f all numerical values of a dataset.
```

Out[6]:

	ID	Salary	10percentage	12graduation	12percentage	CollegeID	C
count	3.998000e+03	3.998000e+03	3998.000000	3998.000000	3998.000000	3998.000000	39
mean	6.637945e+05	3.076998e+05	77.925443	2008.087544	74.466366	5156.851426	
std	3.632182e+05	2.127375e+05	9.850162	1.653599	10.999933	4802.261482	
min	1.124400e+04	3.500000e+04	43.000000	1995.000000	40.000000	2.000000	
25%	3.342842e+05	1.800000e+05	71.680000	2007.000000	66.000000	494.000000	
50%	6.396000e+05	3.000000e+05	79.150000	2008.000000	74.400000	3879.000000	
75%	9.904800e+05	3.700000e+05	85.670000	2009.000000	82.600000	8818.000000	
max	1.298275e+06	4.000000e+06	97.760000	2013.000000	98.700000	18409.000000	

In [7]:
```
data.describe(include='all') # It will include categorical values as well.
```

Out[7]:

	Unnamed: 0	ID	Salary	DOJ	DOL	Designation	JobCity	Gende
count	3998	3.998000e+03	3.998000e+03	3998	3998	3998	3998	399
unique	1	NaN	NaN	81	67	419	339	
top	train	NaN	NaN	2014-07-01 00:00:00	present	software engineer	Bangalore	r
freq	3998	NaN	NaN	199	1875	539	627	304
first	NaN	NaN	NaN	1991-06-01 00:00:00	NaN	NaN	NaN	Nal
last	NaN	NaN	NaN	2015-12-01 00:00:00	NaN	NaN	NaN	Nal
mean	NaN	6.637945e+05	3.076998e+05	NaN	NaN	NaN	NaN	Nal
std	NaN	3.632182e+05	2.127375e+05	NaN	NaN	NaN	NaN	Nal
min	NaN	1.124400e+04	3.500000e+04	NaN	NaN	NaN	NaN	Nal
25%	NaN	3.342842e+05	1.800000e+05	NaN	NaN	NaN	NaN	Nal
50%	NaN	6.396000e+05	3.000000e+05	NaN	NaN	NaN	NaN	Nal
75%	NaN	9.904800e+05	3.700000e+05	NaN	NaN	NaN	NaN	Nal
max	NaN	1.298275e+06	4.000000e+06	NaN	NaN	NaN	NaN	Nal

In [8]:
```
data.info() # We can see the type and features of a dataset using info.
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3998 entries, 0 to 3997
Data columns (total 39 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Unnamed: 0             3998 non-null   object
 1   ID                     3998 non-null   int64
 2   Salary                 3998 non-null   int64
 3   DOJ                    3998 non-null   datetime64[ns]
 4   DOL                    3998 non-null   object
 5   Designation            3998 non-null   object
 6   JobCity                3998 non-null   object
 7   Gender                 3998 non-null   object
 8   DOB                    3998 non-null   datetime64[ns]
 9   10percentage           3998 non-null   float64
 10  10board                3998 non-null   object
 11  12graduation           3998 non-null   int64
 12  12percentage           3998 non-null   float64
 13  12board                3998 non-null   object
 14  CollegeID              3998 non-null   int64
 15  CollegeTier            3998 non-null   int64
 16  Degree                 3998 non-null   object
 17  Specialization         3998 non-null   object
 18  collegeGPA             3998 non-null   float64
 19  CollegeCityID          3998 non-null   int64
 20  CollegeCityTier        3998 non-null   int64
 21  CollegeState           3998 non-null   object
 22  GraduationYear         3998 non-null   int64
 23  English                3998 non-null   int64
 24  Logical                3998 non-null   int64
 25  Quant                  3998 non-null   int64
 26  Domain                 3998 non-null   float64
 27  ComputerProgramming    3998 non-null   int64
 28  ElectronicsAndSemicon  3998 non-null   int64
 29  ComputerScience        3998 non-null   int64
 30  MechanicalEngg         3998 non-null   int64
 31  ElectricalEngg         3998 non-null   int64
 32  TelecomEngg            3998 non-null   int64
 33  CivilEngg              3998 non-null   int64
 34  conscientiousness      3998 non-null   float64
 35  agreeableness          3998 non-null   float64
 36  extraversion           3998 non-null   float64
 37  nueroticism            3998 non-null   float64
 38  openess_to_experience  3998 non-null   float64
dtypes: datetime64[ns](2), float64(9), int64(18), object(10)
memory usage: 1.2+ MB
```

In [9]:
```
data.isnull().sum() # To check null values in dataset.
```

Out[9]:
```
Unnamed: 0              0
ID                      0
Salary                  0
DOJ                     0
DOL                     0
Designation             0
JobCity                 0
Gender                  0
DOB                     0
10percentage            0
10board                 0
12graduation            0
12percentage            0
12board                 0
CollegeID               0
CollegeTier             0
Degree                  0
Specialization          0
collegeGPA              0
CollegeCityID           0
CollegeCityTier         0
CollegeState            0
GraduationYear          0
English                 0
Logical                 0
Quant                   0
Domain                  0
ComputerProgramming     0
ElectronicsAndSemicon   0
ComputerScience         0
MechanicalEngg          0
ElectricalEngg          0
TelecomEngg             0
CivilEngg               0
conscientiousness       0
agreeableness           0
extraversion            0
nueroticism             0
openess_to_experience   0
dtype: int64
```

We don't have any missing value in dataset so we can easily plot the graphs.

In [10]:
```
data.Degree.value_counts() # It will count the values of a categorical column.
```

Out[10]:
```
B.Tech/B.E.      3700
MCA               243
M.Tech./M.E.       53
M.Sc. (Tech.)       2
Name: Degree, dtype: int64
```

In [11]:
```
data.groupby('Degree')['Salary'].sum() # Using group by function we can group
 the data by any column.
```

Out[11]:
```
Degree
B.Tech/B.E.      1141904000
M.Sc. (Tech.)        640000
M.Tech./M.E.       19405000
MCA                68235000
Name: Salary, dtype: int64
```

In [12]:
```
new_df = data.groupby('Degree')
```

In [13]:
```
new_df1 = data.groupby('12board')
```

In [14]:
```
new_df['Specialization'].sum()
```

Out[14]:
```
Degree
B.Tech/B.E.      computer engineeringelectronics and communicat...
M.Sc. (Tech.)                  information sciencecomputer science
M.Tech./M.E.     computer science & engineeringelectrical engin...
MCA              computer applicationcomputer applicationcomput...
Name: Specialization, dtype: object
```

In [15]:
```
new_df1['12percentage'].sum()
```

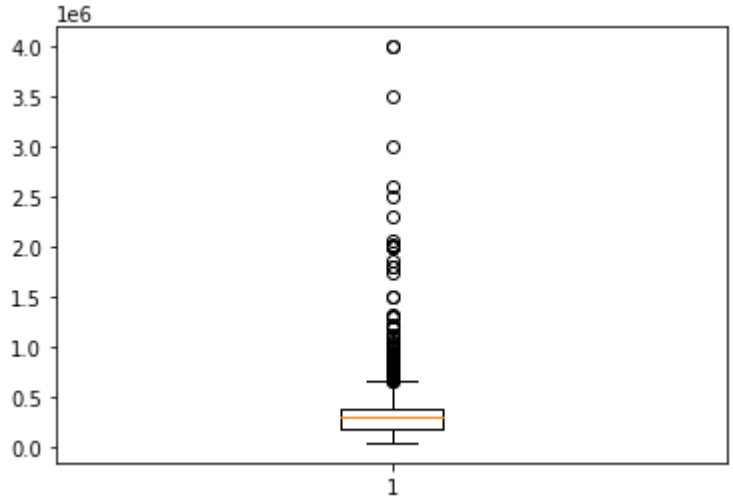
Out[15]:
```
12board
0                                                     26816.43
 board of intermediate                                  185.20
 upboard                                                 62.40
ahsec                                                    57.80
aissce                                                   66.20
                                                         ...
west bengal board of higher secondary education          63.50
west bengal council of higher secondary education       355.19
west bengal council of higher secondary eucation         81.00
west bengal council of higher secondary examination (wbchse)    75.00
west bengal state council of technical education          78.00
Name: 12percentage, Length: 340, dtype: float64
```

Univariate analysis -

Univariate means - single variable. We will plot the graphs for single variable. We have many graphs in it. Let's see -

In [16]:
```
# Let's visualize the Salary column -

print("Mean ", data['Salary'].mean())
print("Median ", data['Salary'].median())
print("Minimum value ", data['Salary'].min())
print("Maximum value ", data['Salary'].max())

plt.boxplot(data['Salary'])
plt.show()
```

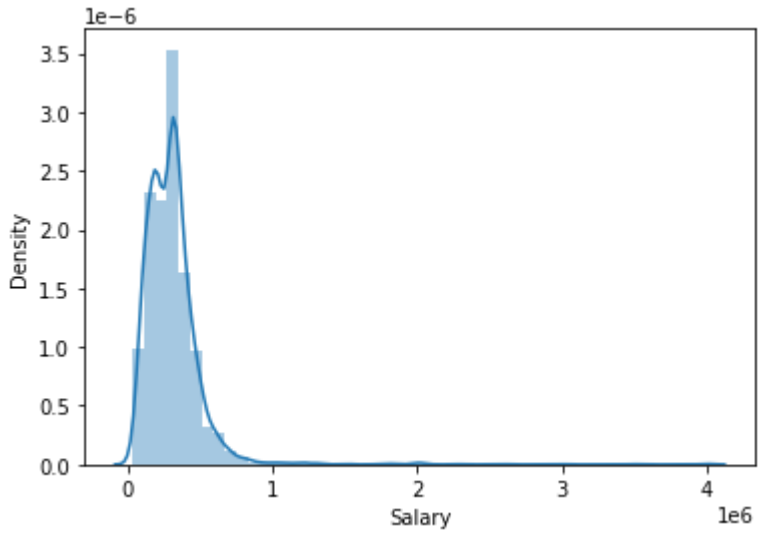
```
Mean  307699.8499249625
Median  300000.0
Minimum value  35000
Maximum value  4000000
```



In salary, we can see we have many values which are extremely high. These are called outliers. Boxplot helps to identify the outliers. And it also tells the mean, median, IQR.

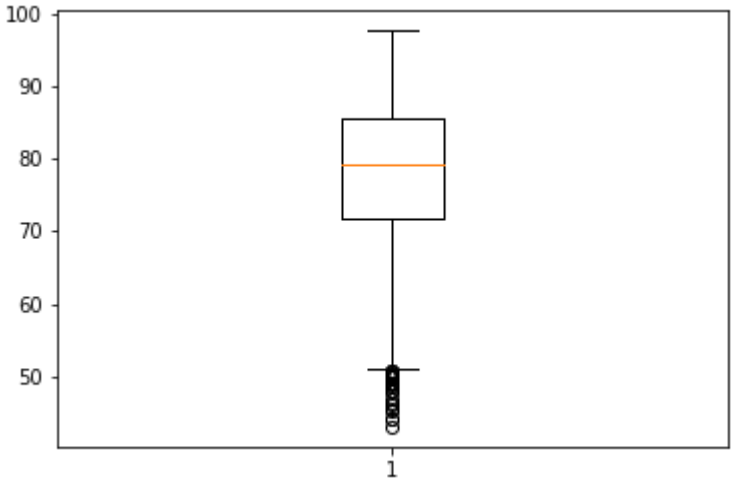
In [17]:
```
# We can see the distribution plot as well of Salary -

sns.distplot(data['Salary'])
```

Out[17]: <AxesSubplot:xlabel='Salary', ylabel='Density'>



In [18]:
```
# 10 percentage

print("Mean ", data['10percentage'].mean())
print("Median ", data['10percentage'].median())
print("Minimum value ", data['10percentage'].min())
print("Maximum value ", data['10percentage'].max())

plt.boxplot(data['10percentage'])
plt.show()
```

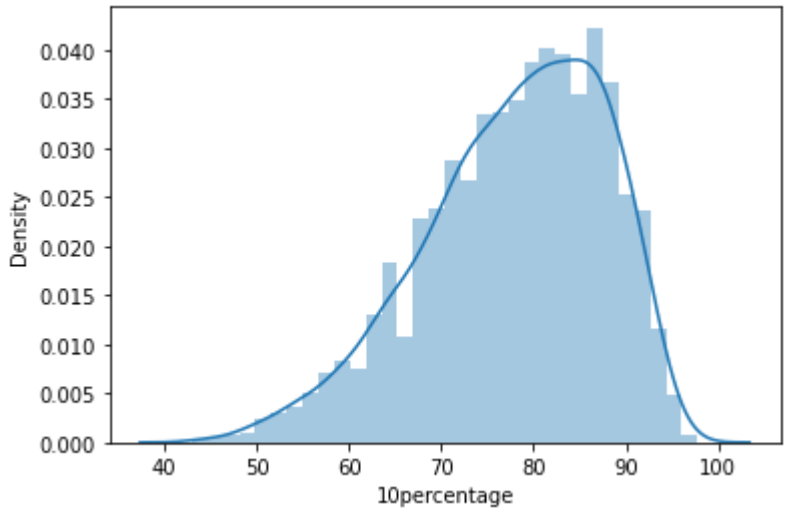
```
Mean  77.9254427213607
Median  79.15
Minimum value  43.0
Maximum value  97.76
```



In 10 percentage, we have some values which are extremely low.

In [19]:
```
sns.distplot(data['10percentage'])
```

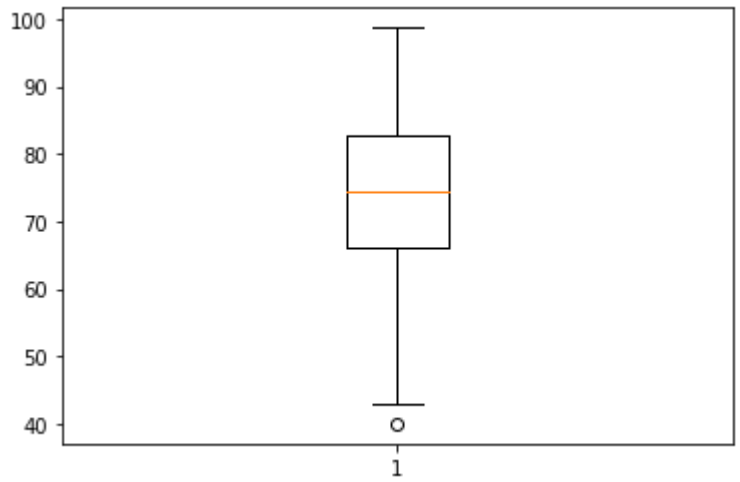
Out[19]: <AxesSubplot:xlabel='10percentage', ylabel='Density'>



The distribution looks like left-skewed because we saw we have some outliers (extremely low values) in this column.

In [20]:
```
# 12 percentage

print("Mean ", data['12percentage'].mean())
print("Median ", data['12percentage'].median())
print("Minimum value ", data['12percentage'].min())
print("Maximum value ", data['12percentage'].max())

plt.boxplot(data['12percentage'])
plt.show()
```

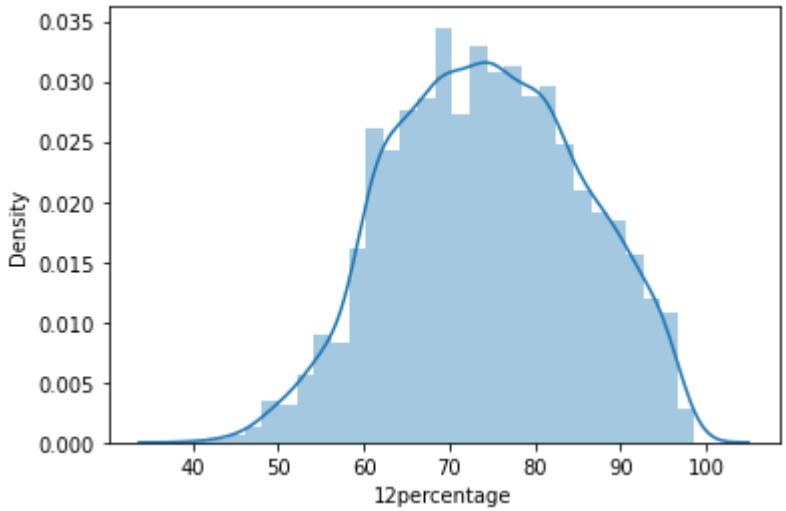
```
Mean  74.46636568284141
Median  74.4
Minimum value  40.0
Maximum value  98.7
```



In 12 percentage, we have only one single value which is extremely low.

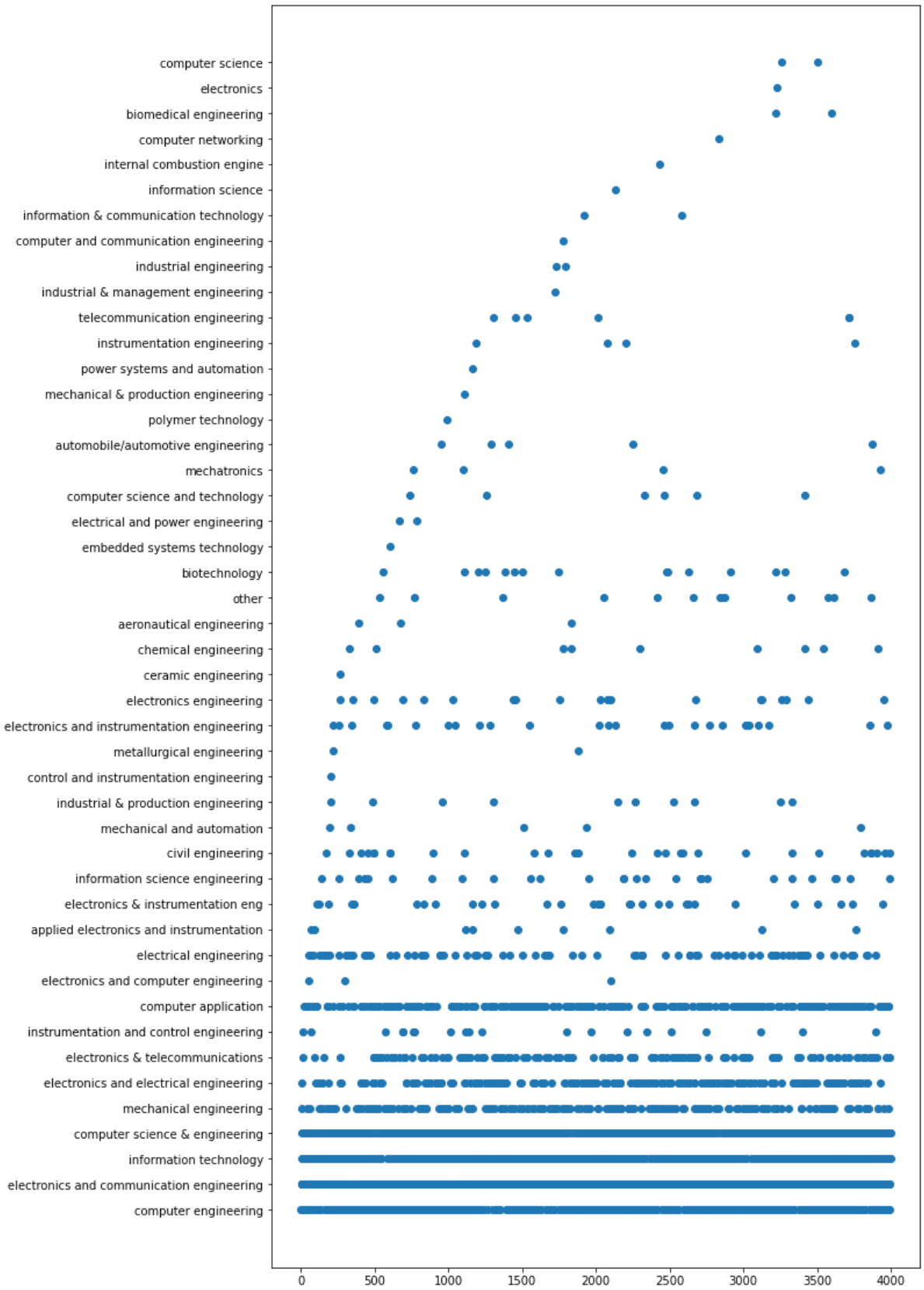
In [21]:
```
# Distribution plot of 12 percentage

sns.distplot(data['12percentage'])
```

Out[21]: <AxesSubplot:xlabel='12percentage', ylabel='Density'>



In [22]:
```
# Scatter plot of Specialization column -

plt.figure(figsize=(10,20))
plt.scatter(data.index, data['Specialization'])
```

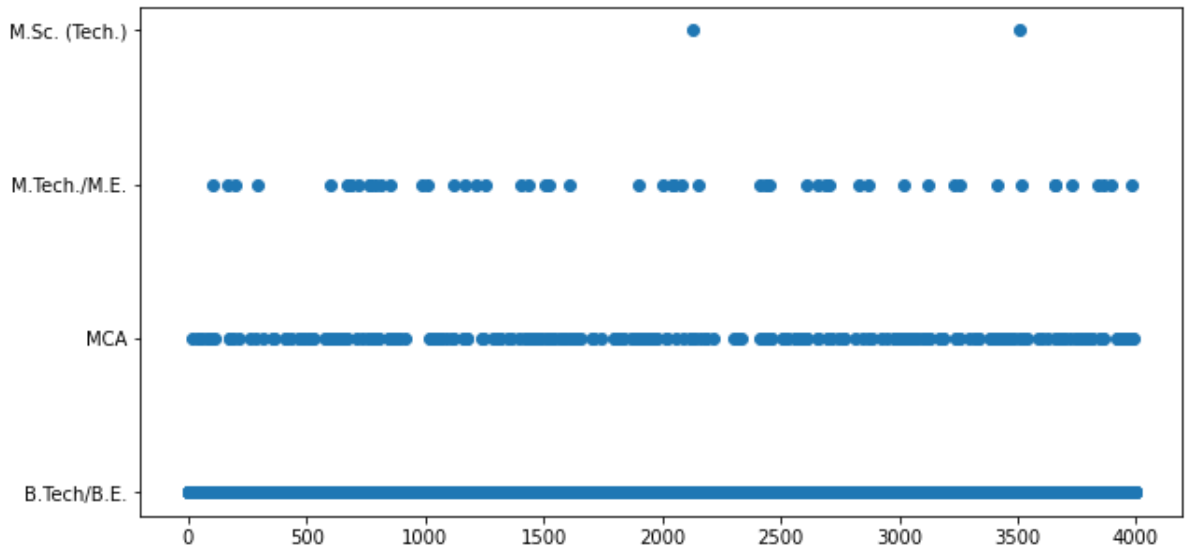
Out[22]: <matplotlib.collections.PathCollection at 0x13dfd00ed90>



We can see what specialization values we have in dataset and highly asked courses are computer engineering, electronics and communication engineering, information technology, computer science & engineering.

In [23]:
```
# Scatter plot of Degree column -

plt.figure(figsize=(10,5))
plt.scatter(data.index, data['Degree'])
```

Out[23]: <matplotlib.collections.PathCollection at 0x13dfcfd9c70>



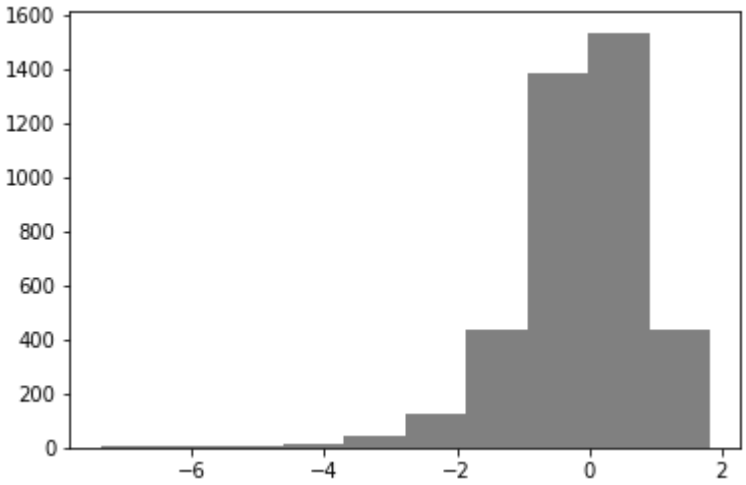
Mostly students choose B.tech/B.E. as their degree.

In [24]:
```
# This is same as Scatter plot that we already plotted for Specialization -

plt.figure(figsize=(5,10))
sns.stripplot(y=data['Specialization'])
```

Out[24]: <AxesSubplot:ylabel='Specialization'>

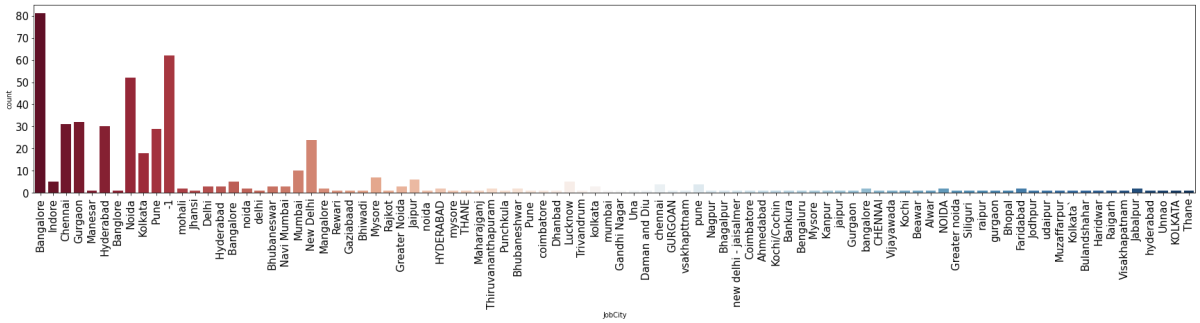
In [25]:
```
# Histogram plot for openess_to_experience column -

plt.hist(data['openess_to_experience'], color='grey')
```

Out[25]:
```
(array([   7.,    5.,    7.,   11.,   40.,  127.,  439., 1389., 1536.,
         437.]),
 array([-7.3757 , -6.45589, -5.53608, -4.61627, -3.69646, -2.77665,
        -1.85684, -0.93703, -0.01722,  0.90259,  1.8224 ]),
 <BarContainer object of 10 artists>)
```



In [26]:
```
plt.figure(figsize=(30, 5))
sns.countplot(x='JobCity', data=data[:500], palette='RdBu')
plt.xticks(fontsize=15)
plt.xticks(rotation=90)
plt.yticks(fontsize=15)
```

Out[26]:
```
(array([ 0., 10., 20., 30., 40., 50., 60., 70., 80., 90.]),
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, '')])
```



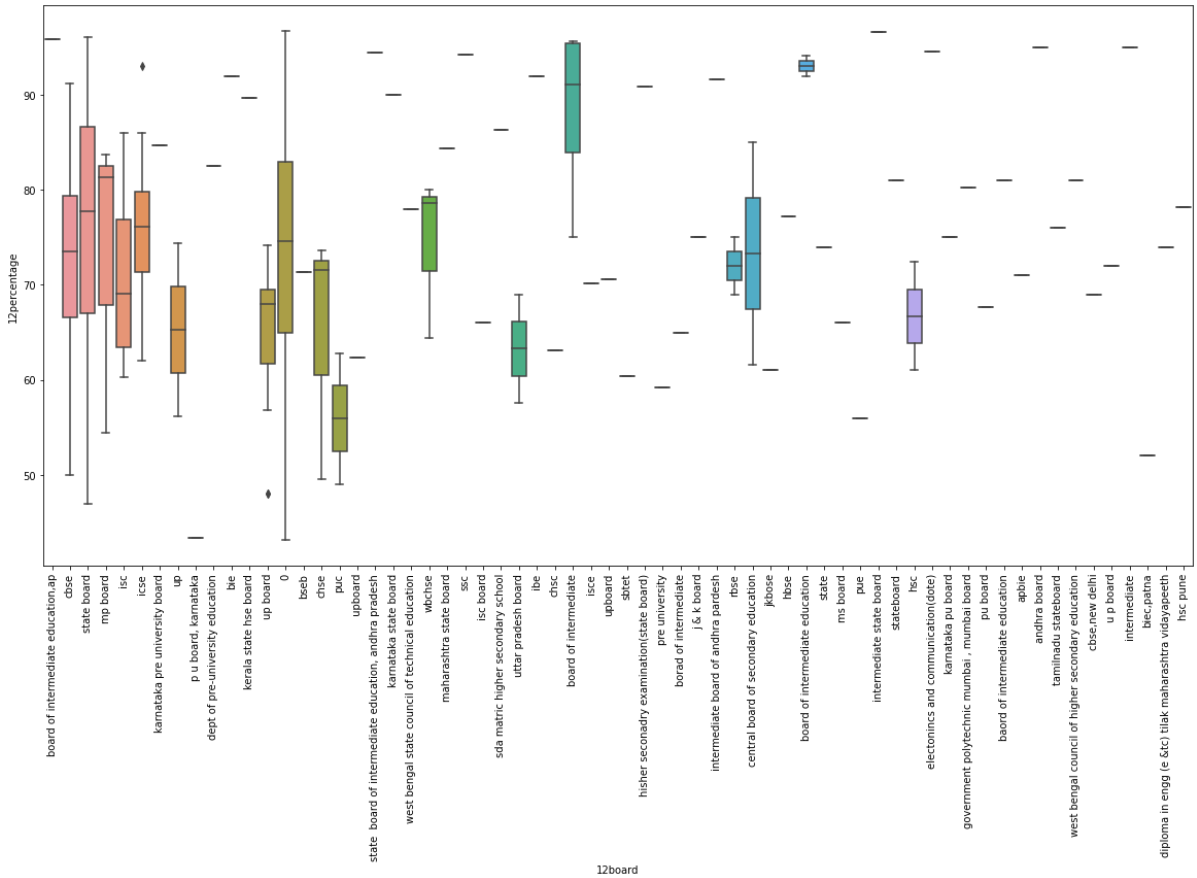
The highest jobs are available in Banglore.

Bivariate analysis -

Bivariate means - Two variables. We will plot the graphs for two variables. We have many graphs in it. Let's see -

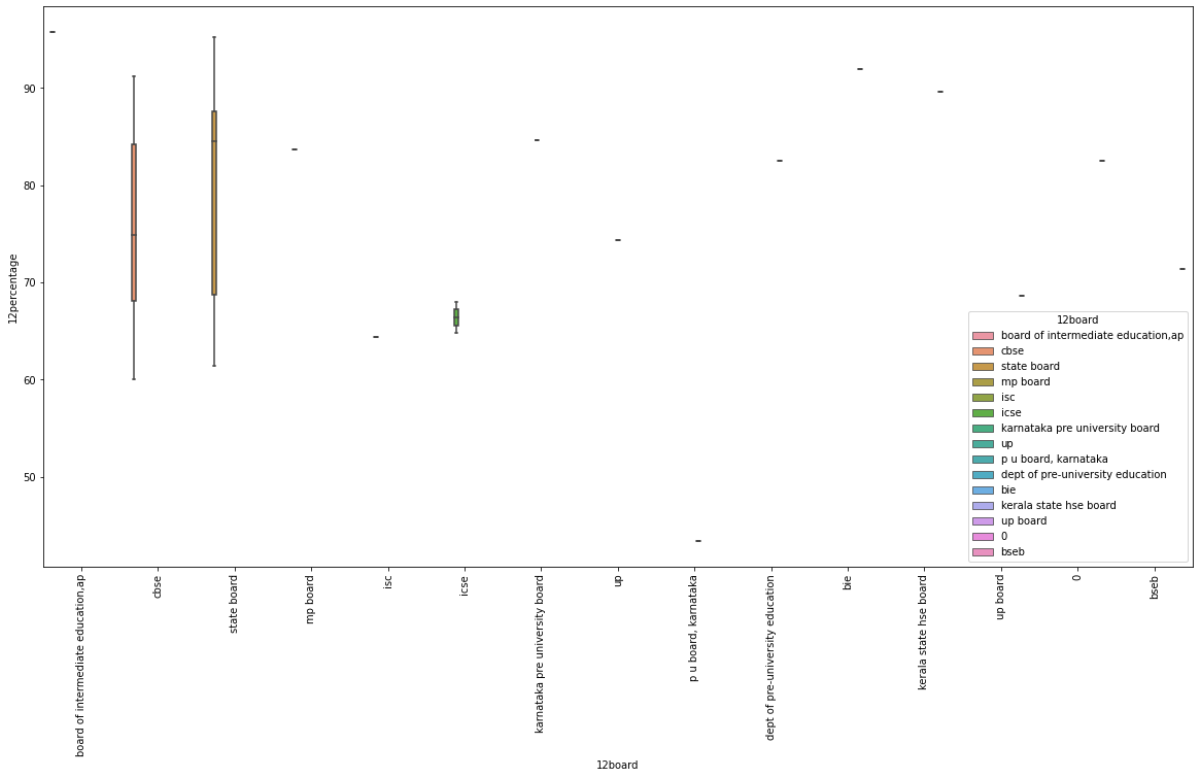
In [27]:
```
# Boxplot for two variables - One numerical, One categorical -

plt.figure(figsize=(20,10))
sns.boxplot(data = data[:500], x='12board', y='12percentage')
plt.xticks(fontsize=10)
plt.xticks(rotation=90)
plt.yticks(fontsize=10)
```

Out[27]:
```
(array([ 40.,  50.,  60.,  70.,  80.,  90., 100.]),
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, '')])
```



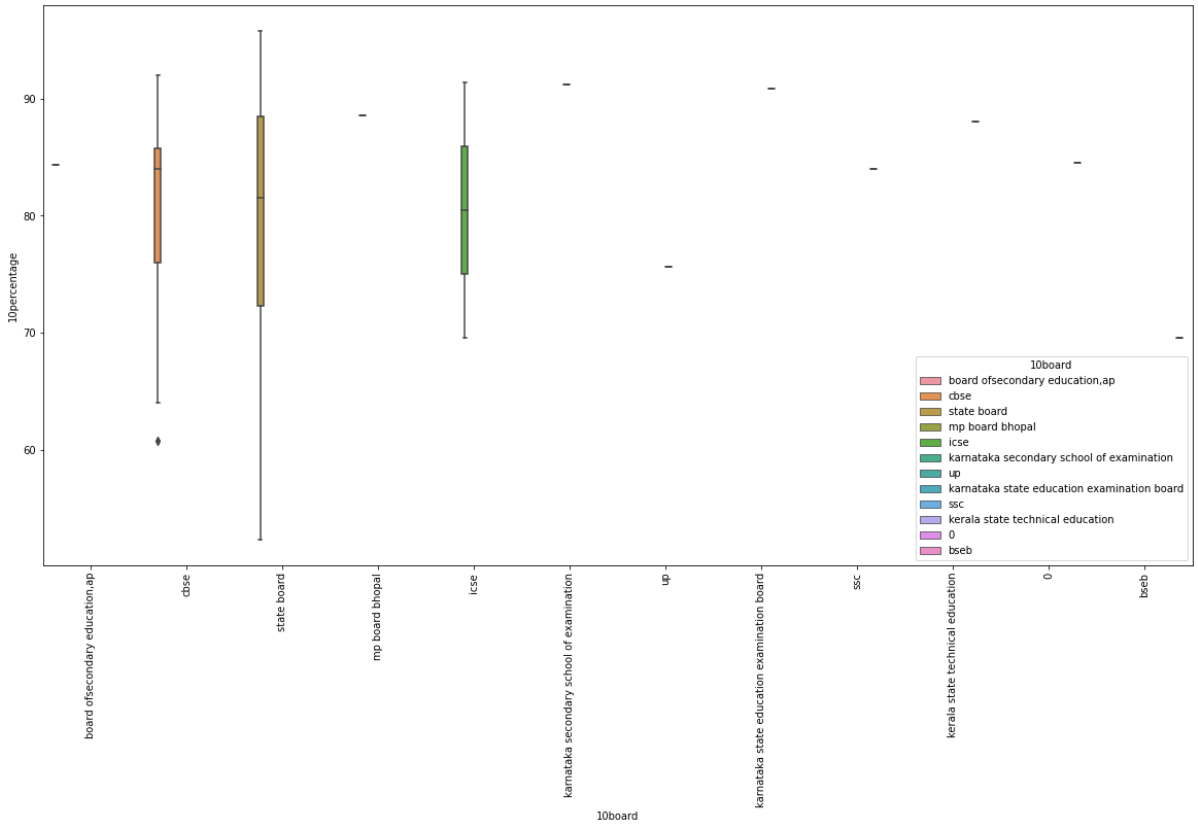
In [28]:
```
# Boxplot on 12 board and 12 percenatge with hue 12 board -

plt.figure(figsize=(20,10))
sns.boxplot(data = data[:50], x='12board', y='12percentage', hue='12board')
plt.xticks(fontsize=10)
plt.xticks(rotation=90)
plt.yticks(fontsize=10)
```

Out[28]:
```
(array([ 40.,  50.,  60.,  70.,  80.,  90., 100.]),
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, '')])
```



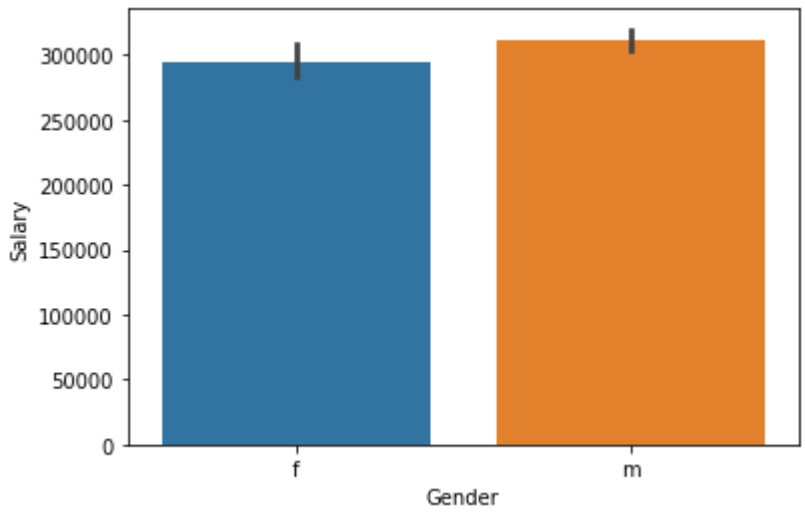
In [29]:
```
# 10 percentage -

plt.figure(figsize=(20,10))
sns.boxplot(data = data[:50], x='10board', y='10percentage', hue='10board')
plt.xticks(fontsize=10)
plt.xticks(rotation=90)
plt.yticks(fontsize=10)
```

Out[29]:
```
(array([ 50.,  60.,  70.,  80.,  90., 100.]),
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, '')])
```



In [30]:
```
sns.barplot(x='Gender', y='Salary', data=data)
```

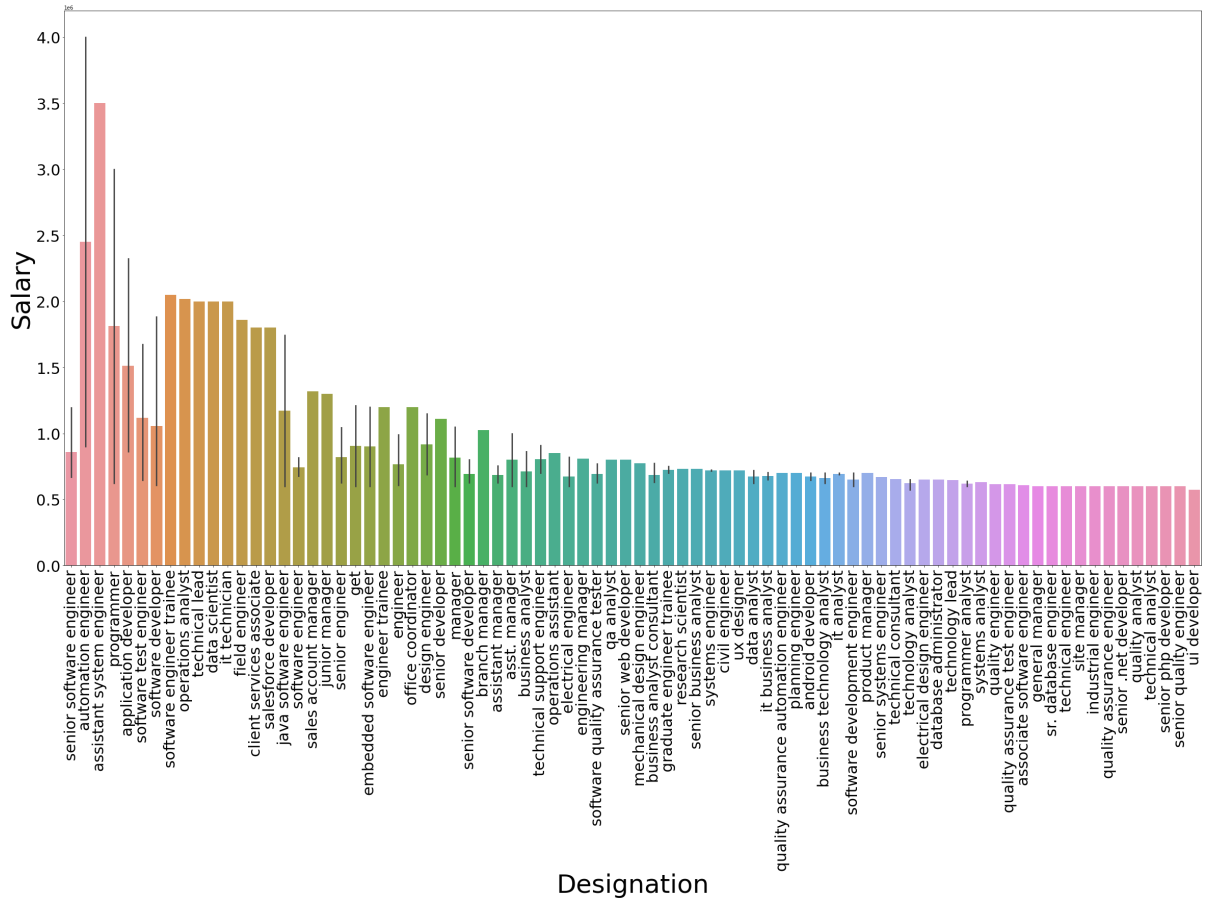
Out[30]: <AxesSubplot:xlabel='Gender', ylabel='Salary'>



After barplot of Gender and Salary we can see that males are getting highest salary.

In [31]:
```
# Designation by Salary -

plt.figure(figsize=(40, 20))
data1 = data.sort_values(by=['Salary'], ascending=False)
sns.barplot(x='Designation', y='Salary', data=data1[:200])
plt.xticks(fontsize=30)
plt.xticks(rotation=90)
plt.yticks(fontsize=30)
plt.xlabel('Designation', fontsize= 50)
plt.ylabel('Salary', fontsize = 50)
```

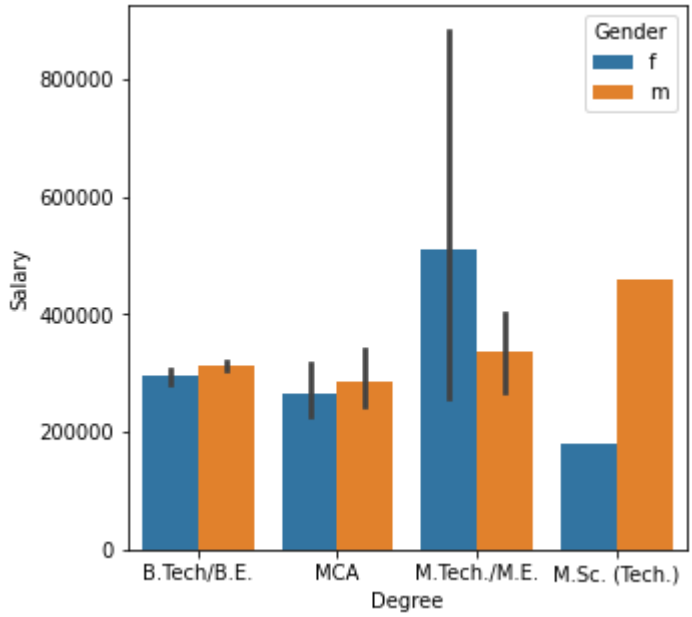
Out[31]: Text(0, 0.5, 'Salary')



The people who are at the post of *assistant system engineer* are getting high salary

In [32]:
```
plt.figure(figsize=(5,5))
sns.barplot(x='Degree', y='Salary', data=data, hue='Gender')
```

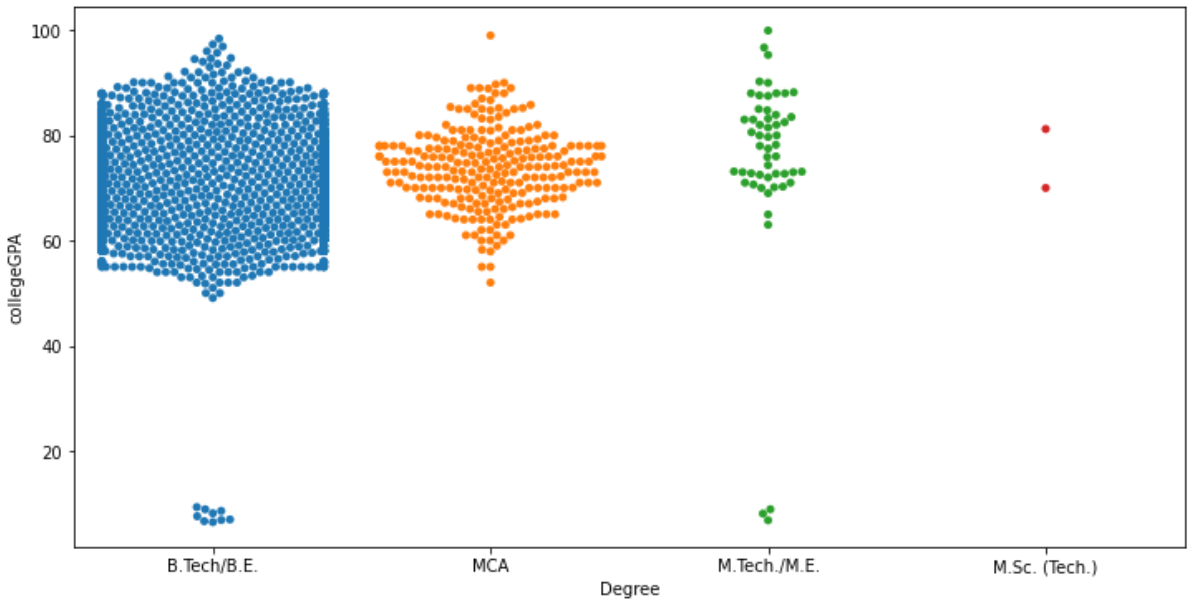
Out[32]: <AxesSubplot:xlabel='Degree', ylabel='Salary'>



The salary of females are high who did M.Tech/M.E.

In [33]:
```
plt.figure(figsize=(12,6))
sns.swarmplot(x='Degree',y='collegeGPA',data=data)
```

Out[33]: <AxesSubplot:xlabel='Degree', ylabel='collegeGPA'>



This shows the collegeGPA of students of different Degrees. And swarmplot spread the data so we can easily visualize.

In [34]:
```
data1 = data[['10percentage', '12percentage', 'collegeGPA', 'English', 'Logical', 'Quant', 'Gender', 'Salary']]
sns.pairplot(data1, hue='Gender')
```

Out[34]: <seaborn.axisgrid.PairGrid at 0x13d82b01040>

This is the pair plot between many features of dataset by gender.

Research Question -

Times of india dated Jan 18, 2019 states that "After doing your Computer Science Engineering if you take up jobs as a Programming Analyst, software Engineer, Hardware engineer and Associate Engineer you can earn upto 2.5 - 3 Lakhs as a fresher graduate." Test this claim with the given data.

We will verify this using hypothesis testing. so, in this case -

Step1 - Alternative hypothesis -

$$H_1 :< 3$$

> Null hypothesis -
>
> $$H_0 :>= 3$$

Step2 - Collect sample of size 50 and then compute mean

Step3 - Compute test statistic:

$$z = \frac{\bar{x} - \mu}{\sigma / \sqrt[2]{n}}$$

Step4 - Decide α

Step5 - Reject or accept based on Tailed test or P value

In [35]:
```
# z_score for sampling distributions

def z_score(sample_size, sample_mean, pop_mean, pop_std):
    numerator = sample_mean - pop_mean
    denomenator = pop_std / sample_size**0.5
    return numerator / denomenator
```

In [36]:
```
data['Designation'].unique()
```

Out[36]:
```
array(['senior quality engineer', 'assistant manager', 'systems engineer',
       'senior software engineer', 'get', 'system engineer',
       'java software engineer', 'mechanical engineer',
       'electrical engineer', 'project engineer', 'senior php developer',
       'senior systems engineer', 'quality assurance engineer',
       'qa analyst', 'network engineer', 'product development engineer',
       'associate software developer', 'data entry operator',
       'software engineer', 'developer', 'electrical project engineer',
       'programmer analyst', 'systems analyst', 'ase',
       'telecommunication engineer', 'application developer',
       'ios developer', 'executive assistant', 'online marketing manager',
       'documentation specialist', 'associate software engineer',
       'management trainee', 'site manager', 'software developer',
       '.net developer', 'production engineer', 'jr. software engineer',
       'trainee software developer', 'ui developer',
       'assistant system engineer', 'android developer',
       'customer service', 'test engineer', 'java developer', 'engineer',
       'recruitment coordinator', 'technical support engineer',
       'data analyst', 'assistant software engineer', 'faculty',
       'entry level management trainee',
       'customer service representative', 'software test engineer',
       'firmware engineer', 'php developer', 'research associate',
       'research analyst', 'quality engineer', 'programmer',
       'technical support executive', 'business analyst', 'web developer',
       'application engineer', 'project coordinator', 'engineer trainee',
       'sap consultant', 'quality analyst', 'marketing coordinator',
       'system administrator', 'senior engineer',
       'business development managerde', 'network administrator',
       'technical support specialist', 'business development executive',
       'junior software engineer', 'asp.net developer',
       'graduate engineer trainee', 'field engineer',
       'assistant professor', 'trainee software engineer',
       'senior software developer',
       'quality assurance automation engineer', 'design engineer',
       'telecom engineer', 'quality control engineer',
       'hardware engineer', 'hr recruiter', 'sales associate',
       'junior engineer', 'associate engineer', 'maintenance engineer',
       'sales engineer', 'human resources associate',
       'mobile application developer',
       'electronic field service engineer', 'process associate',
       'field service engineer', 'it support specialist',
       'software development engineer', 'business process analyst',
       'operation engineer', 'electrical designer', 'marketing assistant',
       'sales executive', 'admin assistant', 'senior java developer',
       'account executive', 'oracle dba', 'rf engineer',
       'embedded software engineer', 'programmer analyst trainee',
       'technical engineer', 'operations executive', 'trainee engineer',
       'recruiter', 'lecturer', '.net web developer',
       'marketing executive', 'operations assistant', 'associate manager',
       'electrical design engineer', 'systems administrator',
       'client services associate', 'it analyst', 'senior developer',
       'cad designer', 'business technology analyst', 'asst. manager',
       'service engineer', 'executive recruiter', 'planning engineer',
       'associate technical operations', 'web designer',
       'software architect', 'software quality assurance tester',
       'seo trainee', 'process engineer',
       'software quality assurance analyst', 'designer',
```

```
'business systems consultant', 'business development manager',
'junior research fellow', 'technical recruiter',
'operations analyst', 'quality assurance test engineer',
'linux systems administrator', 'software trainee',
'entry level sales and marketing', 'electrical field engineer',
'windows systems administrator', 'junior software developer',
'python developer', 'web application developer',
'assistant systems engineer', 'javascript developer',
'operation executive', 'performance engineer', 'technical writer',
'operations engineer and jetty handling', 'lead engineer',
'portfolio analyst', 'associate system engineer',
'mechanical design engineer', 'product engineer',
'network security engineer', 'operations manager',
'technical lead', 'operations', 'quality assurance tester',
'automation engineer', 'data scientist', 'quality associate',
'manual tester', 'sr. engineer', 'embedded engineer',
'service and sales engineer', 'telecom support engineer',
'engineer- customer support', 'cloud engineer', 'branch manager',
'business analyst consultant', 'technology lead',
'software trainee engineer', 'dcs engineer', 'junior manager',
'ux designer', 'clerical', 'hr generalist',
'database administrator', 'senior design engineer', 'seo',
'assistant engineer', 'marketing analyst', 'it executive',
'salesforce developer', 'software tester', 'sql dba',
'junior engineer product support', 'manager',
'senior business analyst', 'c# developer',
'implementation engineer', 'executive hr', 'executive engineer',
'sharepoint developer', 'system analyst',
'sales management trainee', 'senior project engineer',
'it recruiter', 'software engineer analyst',
'desktop support technician', 'continuous improvement engineer',
'process advisor', 'etl developer', 'sales and service engineer',
'project manager', 'training specialist', 'product manager',
'staffing recruiter', 'assistant programmer', 'quality controller',
'mis executive', 'game developer', 'digital marketing specialist',
'principal software engineer', 'software devloper',
'senior mechanical engineer', 'technical operations analyst',
'service coordinator', 'testing engineer', 'technical assistant',
'sap abap consultant', 'seo engineer', 'project assistant',
'talent acquisition specialist', 'sales account manager',
'software engineer trainee', 'customer service manager',
'help desk analyst', 'general manager', 'engineering manager',
'senior network engineer',
'field based employee relations manager', 'phone banking officer',
'support engineer', 'associate test engineer',
'technology analyst', 'network support engineer',
'it business analyst', 'junior system analyst',
'senior .net developer', 'secretary', 'research engineer',
'quality assurance auditor', 'process executive',
'lecturer & electrical maintenance', 'office coordinator',
'hr manager', 'html developer', 'sales support',
'front end web developer', 'administrative support',
'territory sales manager', 'project administrator',
'environmental engineer', 'web designer and seo',
'information security analyst',
'field business development associate', 'operational executive',
'administrative coordinator', 'senior risk consultant',
```

```
'desktop support engineer', 'cad drafter', 'noc engineer',
'industrial engineer', 'it engineer', 'human resources intern',
'senior quality assurance engineer', 'clerical assistant',
'software enginner', 'quality assurance',
'delivery software engineer', 'graphic designer',
'sales development manager', 'visiting faculty',
'business intelligence analyst', 'team lead',
'operational excellence manager', 'sales & service engineer',
'web intern', 'full stack developer', 'database developer',
'sr. database engineer', 'graduate apprentice trainee',
'software engineer associate', 'technical analyst',
'executive engg', 'it technician', 'business system analyst',
'process control engineer', 'technical consultant',
'business office manager', 'quality control inspector',
'product design engineer', 'manufacturing engineer',
'seo executive', 'sap analyst', 'software engineere',
'financial service consultant', 'co faculty', 'software analyst',
'desktop support analyst', 'graduate engineer',
'engineering technician', 'it assistant', 'marketing manager',
'human resource assistant', 'hr assistant', 'product developer',
'customer support engineer',
'quality control inspection technician', 'gis/cad engineer',
'senior web developer', 'sql developer', 'research staff member',
'sap abap associate consultant', 'associate qa',
'corporate recruiter', 'project management officer',
'business systems analyst', 'software programmer',
'help desk technician', 'sales manager', 'catalog associate',
'assistant store manager', 'software engg', 'it developer',
'apprentice', 'business consultant', 'controls engineer',
'ruby on rails developer', 'risk consultant', 'account manager',
'professor', 'assistant administrator', 'civil engineer',
'educator', 'service manager', 'teradata dba',
'full-time loss prevention associate', 'junior recruiter',
'associate developer', 'assistant electrical engineer',
'shift engineer', 'dotnet developer', 'rf/dt engineer',
'human resources analyst', 'software test engineerte',
'junior .net developer', 'java trainee', 'maintenance supervisor',
'r&d engineer', 'front end developer', 'engineer-hws',
'operations engineer', 'senior research fellow',
'web designer and joomla administrator',
'enterprise solutions developer',
'information technology specialist', 'site engineer',
'graduate trainee engineer', 'quality assurance analyst',
'cnc programmer', 'financial analyst', 'system engineer trainee',
'sap mm consultant', 'assistant system engineer trainee',
'qa trainee', 'teradata developer', 'hr executive',
'senior programmer', 'software test engineer (etl)',
'associate software engg', 'supply chain analyst', 'sales trainer',
'software executive', 'team leader',
'assistant system engineer - trainee', 'seo analyst',
'risk investigator', 'executive administrative assistant',
'program manager', 'r & d', 'sap functional consultant',
'website developer/tester', 'software designer',
'sales coordinator', 'qa engineer', 'aircraft technician',
'customer care executive', 'senior test engineer',
'program analyst trainee', 'electrical controls engineer',
'trainee decision scientist', 'editor', 'bss engineer', 'dba',
```

```
       'software eng', 'computer faculty', 'recruitment associate',
       'logistics executive', 'quality consultant',
       'senior sales executive', 'db2 dba', 'test technician',
       'it operations associate', 'software engineering associate',
       'research scientist', 'jr. software developer'], dtype=object)
```

In [37]:
```
data1 = data[(data['Designation']=='programmer analyst') | (data['Designation'
]=='software engineer') |
            (data['Designation']=='associate software engineer') | (data['Desi
gnation']=='electrical engineer')]
data2 = data1[['Salary']]
```

In [38]:
```
data2.shape[0]
```

Out[38]: 747

In [39]:
```
samples = random.sample(range(0, data2.shape[0]), 100)
sample_mean = data2.iloc[samples, 0].mean()
print(sample_mean)
```

```
359600.0
```

In [40]:
```
std = data['Salary'].std()
print(std)
```

```
212737.49995685622
```

In [41]:
```
# Left Tail - Calculating the z-critical value

confidence_level = 0.95
alpha = 1 - confidence_level
z_critical = norm.ppf(1 - alpha) # Left tailed Z score for 95% Confidence Leve
l

print(z_critical)
```

```
1.6448536269514722
```

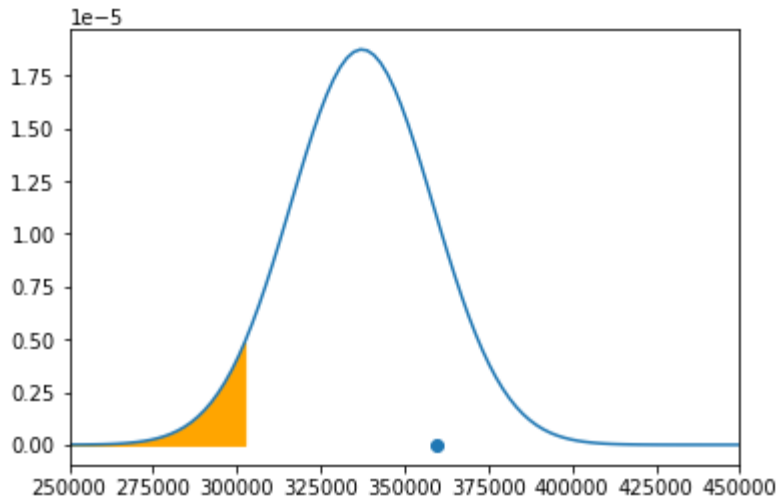
In [42]:
```
# Defining the sample and population parameters

sample_size = 100
sample_mean = sample_mean
pop_mean = data2['Salary'].mean()
pop_std = std
print(pop_mean)
```

```
337289.156626506
```

In [43]:
```
# Calculating the z-score

z = z_score(sample_size, sample_mean, pop_mean, pop_std)
print(z)
```

```
1.0487499090672163
```

In [44]:
```
# Ploting the sampling distribution with rejection regions

# Defining the x minimum and x maximum
x_min = 250000
x_max = 450000


# Defining the sampling distribution mean and sampling distribution std
mean = pop_mean
std = pop_std / sample_size**0.5


# Ploting the graph and setting the x limits
x = np.linspace(x_min, x_max, 100)
y = norm.pdf(x, mean, std)
plt.xlim(x_min, x_max)
plt.plot(x, y)


# Computing the left critical value (left tailed Test)
z_critical_left = pop_mean - (z_critical * std)


# Shading the left rejection region
x2 = np.linspace(x_min, z_critical_left, 100)
y2 = norm.pdf(x2, mean, std)
plt.fill_between(x2, y2, color='orange')


# Ploting the sample mean and concluding the results
plt.scatter(sample_mean, 0)
plt.annotate("x_bar", (sample_mean, 0.0007))
```



In [45]:
```
# Conclusion using z test

if(np.abs(z) > z_critical):
    print("Reject Null Hypothesis")
else:
    print("Fail to reject Null Hypothesis")
```

```
Fail to reject Null Hypothesis
```

In [46]:
```
# Conclusion using p test

p_value = 2 * (1.0 - norm.cdf(np.abs(z)))

print("p_value = ", p_value)

if(p_value < alpha):
    print("Reject Null Hypothesis")
else:
    print("Fail to reject Null Hypothesis")
```

```
p_value =  0.29429323713166067
Fail to reject Null Hypothesis
```

Conclusion - After hypothesis testing we can see that people who are at the post of programmer analyst or software engineer or associate software engineer or electrical engineer are getting salary around 350000.

Ques - is there a relationship between Gender and specialization ?

In [47]:
```
data.Gender.value_counts()
```

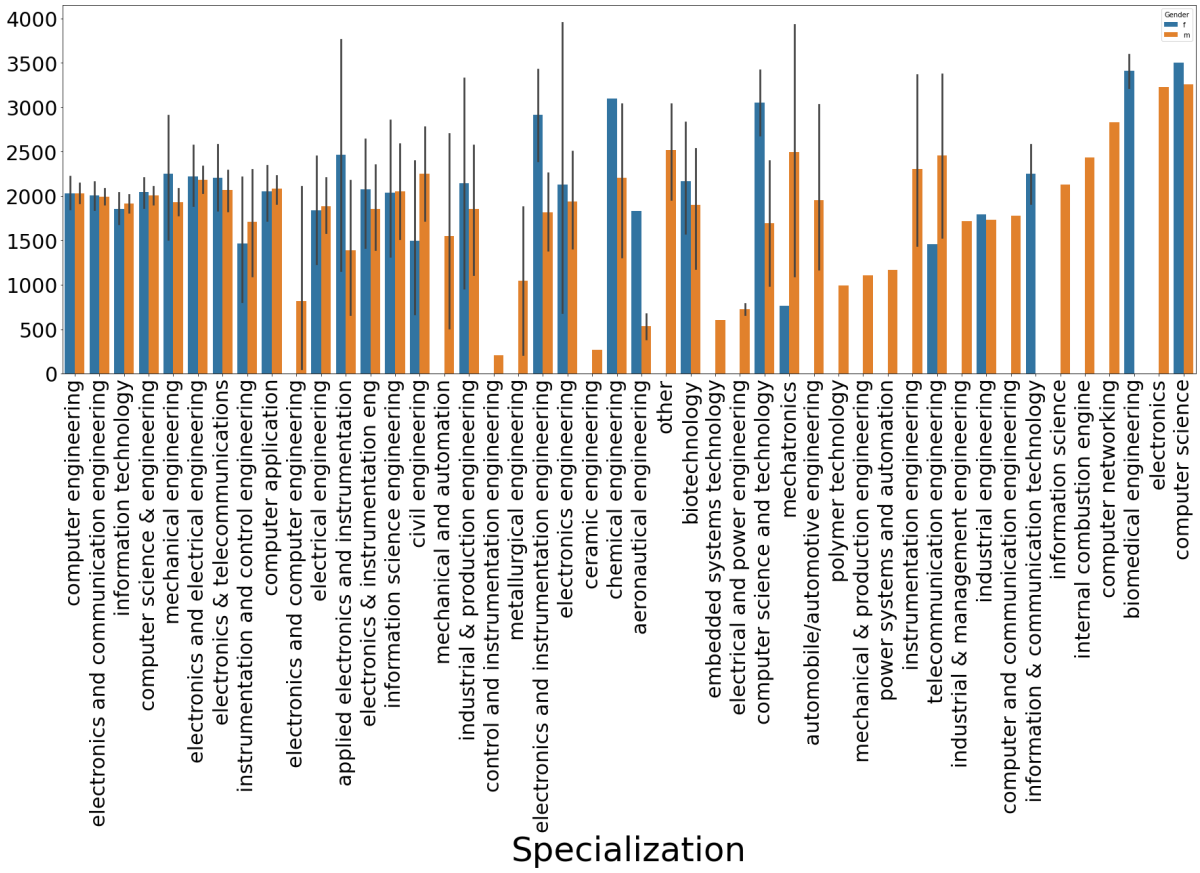
Out[47]:
```
m    3041
f     957
Name: Gender, dtype: int64
```

In [48]:
```
data.Specialization.value_counts()
```

Out[48]:
```
electronics and communication engineering      880
computer science & engineering                 744
information technology                         660
computer engineering                           600
computer application                           244
mechanical engineering                         201
electronics and electrical engineering         196
electronics & telecommunications               121
electrical engineering                          82
electronics & instrumentation eng               32
civil engineering                               29
electronics and instrumentation engineering     27
information science engineering                 27
instrumentation and control engineering         20
electronics engineering                         19
biotechnology                                   15
other                                           13
industrial & production engineering             10
chemical engineering                             9
applied electronics and instrumentation          9
computer science and technology                  6
telecommunication engineering                    6
automobile/automotive engineering                5
mechanical and automation                        5
mechatronics                                     4
instrumentation engineering                      4
aeronautical engineering                         3
electronics and computer engineering             3
computer science                                 2
metallurgical engineering                        2
electrical and power engineering                 2
information & communication technology           2
industrial engineering                           2
biomedical engineering                           2
control and instrumentation engineering          1
computer and communication engineering           1
industrial & management engineering              1
polymer technology                               1
embedded systems technology                      1
internal combustion engine                       1
ceramic engineering                              1
power systems and automation                     1
information science                              1
computer networking                              1
mechanical & production engineering              1
electronics                                      1
Name: Specialization, dtype: int64
```

In [49]:
```
plt.figure(figsize=(30, 10))
sns.barplot(x='Specialization', y=data.index, data=data, hue='Gender')
plt.xticks(fontsize=30)
plt.xticks(rotation=90)
plt.yticks(fontsize=30)
plt.xlabel('Specialization', fontsize= 50)
```

Out[49]: Text(0.5, 0, 'Specialization')



***Conclusion* - We can see that preference of specialization depend on the gender.**

Let's try to test this with chi-square test -

Understanding the Chi2 Test -

Lets make a bold Claim that **Gender** and **Specialization** are dependent.

Step1 - Alternate Hypothesis:

$$H_1 : They\ are\ Dependent$$

> Null Hypothesis:
>
> $$H_0 : They\ are\ Independent$$

Step2 -

- Collect the sample of size n
- Compute the sample frequencies

Step3 - Compute χ^2 test statistic
Now you need to check, if the difference in the observed and expected frequencies is too extreme to reject the NULL hypothesis.

- Have a look at Observed Frequencies (in the sample)
- Compute the Expected Frequencies (under null hyp assumption)

  $$Expected\ Value = \frac{row\ total * col\ total}{grand\ total}$$

  Now, test statistic can be computed using below mentioned formula:

  $$\chi^2 = \Sigma\frac{(o-e)^2}{e}$$

Step4 - Decide α and $df = (rows - 1)(cols - 1)$

Step5 - Apply decision rule

- Chi Square Test

  $$if\ \chi^2 > \chi^2_{df,\alpha} \Rightarrow RejectH_0$$

- p-value Test

  $$p\ value = (1.0 - cdf(test\ statistic))$$

  Now,

  $$if(p\ value < \alpha) \Rightarrow AcceptH_1\ or\ RejectH_0$$

In [50]:
```
# Step - 2 => Looking at the freqency distribution

pd.crosstab(data.Specialization, data.Gender, margins=True)
```

Out[50]:

Gender	f	m	All
Specialization			
aeronautical engineering	1	2	3
applied electronics and instrumentation	2	7	9
automobile/automotive engineering	0	5	5
biomedical engineering	2	0	2
biotechnology	9	6	15
ceramic engineering	0	1	1
chemical engineering	1	8	9
civil engineering	6	23	29
computer and communication engineering	0	1	1
computer application	59	185	244
computer engineering	175	425	600
computer networking	0	1	1
computer science	1	1	2
computer science & engineering	183	561	744
computer science and technology	2	4	6
control and instrumentation engineering	0	1	1
electrical and power engineering	0	2	2
electrical engineering	17	65	82
electronics	0	1	1
electronics & instrumentation eng	10	22	32
electronics & telecommunications	28	93	121
electronics and communication engineering	212	668	880
electronics and computer engineering	0	3	3
electronics and electrical engineering	34	162	196
electronics and instrumentation engineering	5	22	27
electronics engineering	3	16	19
embedded systems technology	0	1	1
industrial & management engineering	0	1	1
industrial & production engineering	2	8	10
industrial engineering	1	1	2
information & communication technology	2	0	2
information science	0	1	1
information science engineering	8	19	27
information technology	173	487	660

Gender	f	m	All
Specialization			
instrumentation and control engineering	9	11	20
instrumentation engineering	0	4	4
internal combustion engine	0	1	1
mechanical & production engineering	0	1	1
mechanical and automation	0	5	5
mechanical engineering	10	191	201
mechatronics	1	3	4
metallurgical engineering	0	2	2
other	0	13	13
polymer technology	0	1	1
power systems and automation	0	1	1
telecommunication engineering	1	5	6
All	957	3041	3998

In [51]:
```
# These are the observed frequencies

observed = pd.crosstab(data.Specialization, data.Gender)
observed
```

Out[51]:

Gender	f	m
Specialization		
aeronautical engineering	1	2
applied electronics and instrumentation	2	7
automobile/automotive engineering	0	5
biomedical engineering	2	0
biotechnology	9	6
ceramic engineering	0	1
chemical engineering	1	8
civil engineering	6	23
computer and communication engineering	0	1
computer application	59	185
computer engineering	175	425
computer networking	0	1
computer science	1	1
computer science & engineering	183	561
computer science and technology	2	4
control and instrumentation engineering	0	1
electrical and power engineering	0	2
electrical engineering	17	65
electronics	0	1
electronics & instrumentation eng	10	22
electronics & telecommunications	28	93
electronics and communication engineering	212	668
electronics and computer engineering	0	3
electronics and electrical engineering	34	162
electronics and instrumentation engineering	5	22
electronics engineering	3	16
embedded systems technology	0	1
industrial & management engineering	0	1
industrial & production engineering	2	8
industrial engineering	1	1
information & communication technology	2	0
information science	0	1
information science engineering	8	19
information technology	173	487

Gender	f	m
Specialization		
instrumentation and control engineering	9	11
instrumentation engineering	0	4
internal combustion engine	0	1
mechanical & production engineering	0	1
mechanical and automation	0	5
mechanical engineering	10	191
mechatronics	1	3
metallurgical engineering	0	2
other	0	13
polymer technology	0	1
power systems and automation	0	1
telecommunication engineering	1	5

In [52]:
```
# chi2_contigency returns chi2 test statistic, p-value, degree of freedoms, expected frequencies

chi2_contingency(observed)
```

Out[52]:
```
(104.46891913608454,
 1.2453868176977011e-06,
 45,
 array([[7.18109055e-01, 2.28189095e+00],
        [2.15432716e+00, 6.84567284e+00],
        [1.19684842e+00, 3.80315158e+00],
        [4.78739370e-01, 1.52126063e+00],
        [3.59054527e+00, 1.14094547e+01],
        [2.39369685e-01, 7.60630315e-01],
        [2.15432716e+00, 6.84567284e+00],
        [6.94172086e+00, 2.20582791e+01],
        [2.39369685e-01, 7.60630315e-01],
        [5.84062031e+01, 1.85593797e+02],
        [1.43621811e+02, 4.56378189e+02],
        [2.39369685e-01, 7.60630315e-01],
        [4.78739370e-01, 1.52126063e+00],
        [1.78091046e+02, 5.65908954e+02],
        [1.43621811e+00, 4.56378189e+00],
        [2.39369685e-01, 7.60630315e-01],
        [4.78739370e-01, 1.52126063e+00],
        [1.96283142e+01, 6.23716858e+01],
        [2.39369685e-01, 7.60630315e-01],
        [7.65982991e+00, 2.43401701e+01],
        [2.89637319e+01, 9.20362681e+01],
        [2.10645323e+02, 6.69354677e+02],
        [7.18109055e-01, 2.28189095e+00],
        [4.69164582e+01, 1.49083542e+02],
        [6.46298149e+00, 2.05370185e+01],
        [4.54802401e+00, 1.44519760e+01],
        [2.39369685e-01, 7.60630315e-01],
        [2.39369685e-01, 7.60630315e-01],
        [2.39369685e+00, 7.60630315e+00],
        [4.78739370e-01, 1.52126063e+00],
        [4.78739370e-01, 1.52126063e+00],
        [2.39369685e-01, 7.60630315e-01],
        [6.46298149e+00, 2.05370185e+01],
        [1.57983992e+02, 5.02016008e+02],
        [4.78739370e+00, 1.52126063e+01],
        [9.57478739e-01, 3.04252126e+00],
        [2.39369685e-01, 7.60630315e-01],
        [2.39369685e-01, 7.60630315e-01],
        [1.19684842e+00, 3.80315158e+00],
        [4.81133067e+01, 1.52886693e+02],
        [9.57478739e-01, 3.04252126e+00],
        [4.78739370e-01, 1.52126063e+00],
        [3.11180590e+00, 9.88819410e+00],
        [2.39369685e-01, 7.60630315e-01],
        [2.39369685e-01, 7.60630315e-01],
        [1.43621811e+00, 4.56378189e+00]]))
```

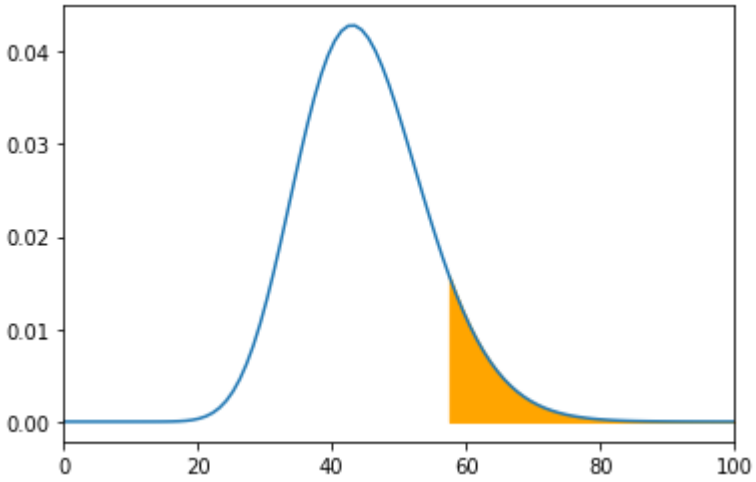
In [53]:
```
# Computing chi2 test statistic, p-value, degree of freedoms

chi2_test_stat = chi2_contingency(observed)[0]
pval = chi2_contingency(observed)[1]
df = chi2_contingency(observed)[2]
```

In [54]:
```
confidence_level = 0.90
alpha = 1 - confidence_level
chi2_critical = chi2.ppf(1 - alpha, df)

chi2_critical
```

Out[54]: 57.50530474499599

In [55]:
```
# Ploting the chi2 distribution to visualise

# Defining the x minimum and x maximum
x_min = 0
x_max = 100

# Ploting the graph and setting the x limits
x = np.linspace(x_min, x_max, 100)
y = chi2.pdf(x, df)
plt.xlim(x_min, x_max)
plt.plot(x, y)


# Setting Chi2 Critical value
chi2_critical_right = chi2_critical

# Shading the right rejection region
x1 = np.linspace(chi2_critical_right, x_max, 100)
y1 = chi2.pdf(x1, df)
plt.fill_between(x1, y1, color='orange')
```

Out[55]: <matplotlib.collections.PolyCollection at 0x13d89612250>



In [56]:
```
if(chi2_test_stat > chi2_critical):
    print("Reject Null Hypothesis")
else:
    print("Fail to Reject Null Hypothesis")
```

```
Reject Null Hypothesis
```

In [57]:
```
if(pval < alpha):
    print("Reject Null Hypothesis")
else:
    print("Fail to Reject Null Hypothesis")
```

```
Reject Null Hypothesis
```

Now it's clear that preference of specialization depend on the gender.

Column Standardization -

Standardization is used on the data values that are normally distributed. Further, by applying standardization, we tend to make the mean of the dataset as 0 and the standard deviation equivalent to 1.

For numerical features - We can do this using scale which comes under sklearn.preprocessing

In [58]:
```
standard_10percentage = preprocessing.scale(data['10percentage'])
print(standard_10percentage)
```

```
[ 0.64723345  0.75892071  0.71830716 ...  0.39949082  0.08067447
 -0.74378053]
```

In [59]:
```
standard_12percentage = preprocessing.scale(data['12percentage'])
print(standard_12percentage)
```

```
[ 1.93967569  0.95772873 -0.56974433 ... -0.81523107 -0.41699702
 -0.58792853]
```

Label Encoding -

Label encoding is used to transform categorical data into numerical data.

For categorical features - If we have 2 categories than we can convert into binary. If we have more than 2 categories we can use dummy variables.

In [60]:
```
# For two values -

encoding_gender = preprocessing.OneHotEncoder(sparse=False)
data['Gender'] = encoding_gender.fit_transform(data[['Gender']])
```

In [61]:
```
data.head()
```

Out[61]:

	Unnamed: 0	ID	Salary	DOJ	DOL	Designation	JobCity	Gender	DOB	10percer
0	train	203097	420000	2012-06-01	present	senior quality engineer	Bangalore	1.0	1990-02-19	
1	train	579905	500000	2013-09-01	present	assistant manager	Indore	0.0	1989-10-04	
2	train	810601	325000	2014-06-01	present	systems engineer	Chennai	1.0	1992-08-03	
3	train	267447	1100000	2011-07-01	present	senior software engineer	Gurgaon	0.0	1989-12-05	
4	train	343523	200000	2014-03-01	2015-03-01 00:00:00	get	Manesar	0.0	1991-02-27	

In [62]:
```
# For more than two values -

encoding_degree = preprocessing.OrdinalEncoder()
data['Degree'] = encoding_degree.fit_transform(data[['Degree']])
```

In [63]:
```
data.head()
```

Out[63]:

	Unnamed: 0	ID	Salary	DOJ	DOL	Designation	JobCity	Gender	DOB	10percer
0	train	203097	420000	2012-06-01	present	senior quality engineer	Bangalore	1.0	1990-02-19	
1	train	579905	500000	2013-09-01	present	assistant manager	Indore	0.0	1989-10-04	
2	train	810601	325000	2014-06-01	present	systems engineer	Chennai	1.0	1992-08-03	
3	train	267447	1100000	2011-07-01	present	senior software engineer	Gurgaon	0.0	1989-12-05	
4	train	343523	200000	2014-03-01	2015-03-01 00:00:00	get	Manesar	0.0	1991-02-27	

Thank you 😉😉😉