In [28]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [29]:
```
PATH = "C:/Users/DELL/Downloads/data.xlsx - Sheet1.csv"

df = pd.read_csv(PATH)

df.head()
```

Out[29]:

	Unnamed: 0	ID	Salary	DOJ	DOL	Designation	JobCity	Gender	DOB	10perce
0	train	203097	420000.0	6/1/12 0:00	present	senior quality engineer	Bangalore	f	2/19/90 0:00	
1	train	579905	500000.0	9/1/13 0:00	present	assistant manager	Indore	m	10/4/89 0:00	
2	train	810601	325000.0	6/1/14 0:00	present	systems engineer	Chennai	f	8/3/92 0:00	
3	train	267447	1100000.0	7/1/11 0:00	present	senior software engineer	Gurgaon	m	12/5/89 0:00	
4	train	343523	200000.0	3/1/14 0:00	3/1/15 0:00	get	Manesar	m	2/27/91 0:00	

5 rows × 39 columns

About of Data

In [30]:
```
df.shape
```

Out[30]:
```
(3998, 39)
```

In [31]:
```
df.columns
```

Out[31]:
```
Index(['Unnamed: 0', 'ID', 'Salary', 'DOJ', 'DOL', 'Designation', 'JobCity',
       'Gender', 'DOB', '10percentage', '10board', '12graduation',
       '12percentage', '12board', 'CollegeID', 'CollegeTier', 'Degree',
       'Specialization', 'collegeGPA', 'CollegeCityID', 'CollegeCityTier',
       'CollegeState', 'GraduationYear', 'English', 'Logical', 'Quant',
       'Domain', 'ComputerProgramming', 'ElectronicsAndSemicon',
       'ComputerScience', 'MechanicalEngg', 'ElectricalEngg', 'TelecomEngg',
       'CivilEngg', 'conscientiousness', 'agreeableness', 'extraversion',
       'nueroticism', 'openess_to_experience'],
      dtype='object')
```

In [32]:
```
df.describe()
```

Out[32]:

	ID	Salary	10percentage	12graduation	12percentage	CollegeID	Col
count	3.998000e+03	3.998000e+03	3998.000000	3998.000000	3998.000000	3998.000000	3998
mean	6.637945e+05	3.076998e+05	77.925443	2008.087544	74.466366	5156.851426	1
std	3.632182e+05	2.127375e+05	9.850162	1.653599	10.999933	4802.261482	C
min	1.124400e+04	3.500000e+04	43.000000	1995.000000	40.000000	2.000000	1
25%	3.342842e+05	1.800000e+05	71.680000	2007.000000	66.000000	494.000000	2
50%	6.396000e+05	3.000000e+05	79.150000	2008.000000	74.400000	3879.000000	2
75%	9.904800e+05	3.700000e+05	85.670000	2009.000000	82.600000	8818.000000	2
max	1.298275e+06	4.000000e+06	97.760000	2013.000000	98.700000	18409.000000	2

8 rows × 27 columns

In [33]:
```
df.head()
```

Out[33]:

	Unnamed: 0	ID	Salary	DOJ	DOL	Designation	JobCity	Gender	DOB	10perce
0	train	203097	420000.0	6/1/12 0:00	present	senior quality engineer	Bangalore	f	2/19/90 0:00	
1	train	579905	500000.0	9/1/13 0:00	present	assistant manager	Indore	m	10/4/89 0:00	
2	train	810601	325000.0	6/1/14 0:00	present	systems engineer	Chennai	f	8/3/92 0:00	
3	train	267447	1100000.0	7/1/11 0:00	present	senior software engineer	Gurgaon	m	12/5/89 0:00	
4	train	343523	200000.0	3/1/14 0:00	3/1/15 0:00	get	Manesar	m	2/27/91 0:00	

5 rows × 39 columns

In [34]:
```
df.tail()
```

Out[34]:

	Unnamed: 0	ID	Salary	DOJ	DOL	Designation	JobCity	Gender	DOE
3993	train	47916	280000.0	10/1/11 0:00	10/1/12 0:00	software engineer	New Delhi	m	4/15/87 0:0(
3994	train	752781	100000.0	7/1/13 0:00	7/1/13 0:00	technical writer	Hyderabad	f	8/27/92 0:0(
3995	train	355888	320000.0	7/1/13 0:00	present	associate software engineer	Bangalore	m	7/3/91 0:0(
3996	train	947111	200000.0	7/1/14 0:00	1/1/15 0:00	software developer	Asifabadbanglore	f	3/20/92 0:0(
3997	train	324966	400000.0	2/1/13 0:00	present	senior systems engineer	Chennai	f	2/26/91 0:0(

5 rows × 39 columns

In [37]:
```
df.duplicated().sum()
```

Out[37]: 0

In [35]:
```
df.isnull().sum()
```

Out[35]:
```
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
```

In [36]:
```
print(df.count())
```

```
Unnamed: 0               3998
ID                       3998
Salary                   3998
DOJ                      3998
DOL                      3998
Designation              3998
JobCity                  3998
Gender                   3998
DOB                      3998
10percentage             3998
10board                  3998
12graduation             3998
12percentage             3998
12board                  3998
CollegeID                3998
CollegeTier              3998
Degree                   3998
Specialization           3998
collegeGPA               3998
CollegeCityID            3998
CollegeCityTier          3998
CollegeState             3998
GraduationYear           3998
English                  3998
Logical                  3998
Quant                    3998
Domain                   3998
ComputerProgramming      3998
ElectronicsAndSemicon    3998
ComputerScience          3998
MechanicalEngg           3998
ElectricalEngg           3998
TelecomEngg              3998
CivilEngg                3998
conscientiousness        3998
agreeableness            3998
extraversion             3998
nueroticism              3998
openess_to_experience    3998
dtype: int64
```

In [10]:
```
df.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   float64
 3   DOJ                    3998 non-null   object 
 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   object 
 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: float64(10), int64(17), object(12)
memory usage: 1.2+ MB
```

In [11]:
```
df.dtypes
```

Out[11]:
```
Unnamed: 0                 object
ID                          int64
Salary                    float64
DOJ                        object
DOL                        object
Designation                object
JobCity                    object
Gender                     object
DOB                        object
10percentage              float64
10board                    object
12graduation                int64
12percentage              float64
12board                    object
CollegeID                   int64
CollegeTier                 int64
Degree                     object
Specialization             object
collegeGPA                float64
CollegeCityID               int64
CollegeCityTier             int64
CollegeState               object
GraduationYear              int64
English                     int64
Logical                     int64
Quant                       int64
Domain                    float64
ComputerProgramming         int64
ElectronicsAndSemicon       int64
ComputerScience             int64
MechanicalEngg              int64
ElectricalEngg              int64
TelecomEngg                 int64
CivilEngg                   int64
conscientiousness         float64
agreeableness             float64
extraversion              float64
nueroticism               float64
openess_to_experience     float64
dtype: object
```

Univariate Analysis on Numerical Data

In [16]:
```
numerical_df = df.select_dtypes(include=['int64', 'float64'])
```

In [40]:
```
def univariate_analysis(numerical_data):
    for col_name in numerical_data:
        print("*"*10, col_name, "*"*10)
        print(numerical_data[col_name].agg(['min', 'max', 'mean', 'median','std
        print("-------------------------------------------------------------
```

In [41]:
```
univariate_analysis(numerical_df)
```

```
********** ID **********
min       1.124400e+04
max       1.298275e+06
mean      6.637945e+05
median    6.396000e+05
std       3.632182e+05
skew      5.477047e-02
kurt     -1.222694e+00
Name: ID, dtype: float64
------------------------------------------------------------------------------------------------------
********** Salary **********
min       3.500000e+04
max       4.000000e+06
mean      3.076998e+05
median    3.000000e+05
std       2.127375e+05
skew      6.451081e+00
kurt      8.093000e+01
Name: Salary, dtype: float64
------------------------------------------------------------------------------------------------------
********** 10percentage **********
min       43.000000
max       97.760000
mean      77.925443
median    79.150000
std        9.850162
skew      -0.591019
kurt      -0.110284
Name: 10percentage, dtype: float64
------------------------------------------------------------------------------------------------------
********** 12graduation **********
min       1995.000000
max       2013.000000
mean      2008.087544
median    2008.000000
std          1.653599
skew        -0.964090
kurt         1.951164
Name: 12graduation, dtype: float64
------------------------------------------------------------------------------------------------------
********** 12percentage **********
min       40.000000
max       98.700000
mean      74.466366
median    74.400000
std       10.999933
skew      -0.032607
kurt      -0.630737
Name: 12percentage, dtype: float64
------------------------------------------------------------------------------------------------------
********** CollegeID **********
min           2.000000
max       18409.000000
mean       5156.851426
median     3879.000000
std        4802.261482
skew          0.649176
```

```
kurt         -0.767441
Name: CollegeID, dtype: float64
--------------------------------------------------------------------------------
--------------------
********** CollegeTier **********
min       1.000000
max       2.000000
mean      1.925713
median    2.000000
std       0.262270
skew     -3.247991
kurt      8.553722
Name: CollegeTier, dtype: float64
--------------------------------------------------------------------------------
--------------------
********** collegeGPA **********
min        6.450000
max       99.930000
mean      71.486171
median    71.720000
std        8.167338
skew      -1.249209
kurt      10.234244
Name: collegeGPA, dtype: float64
--------------------------------------------------------------------------------
--------------------
********** CollegeCityID **********
min           2.000000
max       18409.000000
mean       5156.851426
median     3879.000000
std        4802.261482
skew          0.649176
kurt         -0.767441
Name: CollegeCityID, dtype: float64
--------------------------------------------------------------------------------
--------------------
********** CollegeCityTier **********
min       0.000000
max       1.000000
mean      0.300400
median    0.000000
std       0.458489
skew      0.871120
kurt     -1.241771
Name: CollegeCityTier, dtype: float64
--------------------------------------------------------------------------------
--------------------
********** GraduationYear **********
min          0.000000
max       2017.000000
mean      2012.105803
median    2013.000000
std         31.857271
skew       -63.068064
kurt      3984.369696
Name: GraduationYear, dtype: float64
--------------------------------------------------------------------------------
--------------------
********** English **********
min       180.000000
max       875.000000
```

```
mean      501.649075
median    500.000000
std       104.940021
skew        0.191997
kurt       -0.254133
Name: English, dtype: float64
------------------------------------------------------------------------------
--------------------
********** Logical **********
min       195.000000
max       795.000000
mean      501.598799
median    505.000000
std        86.783297
skew       -0.216602
kurt       -0.224761
Name: Logical, dtype: float64
------------------------------------------------------------------------------
--------------------
********** Quant **********
min       120.000000
max       900.000000
mean      513.378189
median    515.000000
std       122.302332
skew       -0.019399
kurt       -0.102472
Name: Quant, dtype: float64
------------------------------------------------------------------------------
--------------------
********** Domain **********
min      -1.000000
max       0.999910
mean      0.510490
median    0.622643
std       0.468671
skew     -1.922146
kurt      3.895951
Name: Domain, dtype: float64
------------------------------------------------------------------------------
--------------------
********** ComputerProgramming **********
min        -1.000000
max       840.000000
mean      353.102801
median    415.000000
std       205.355519
skew       -0.778106
kurt       -0.666352
Name: ComputerProgramming, dtype: float64
------------------------------------------------------------------------------
--------------------
********** ElectronicsAndSemicon **********
min        -1.000000
max       612.000000
mean       95.328414
median     -1.000000
std       158.241218
skew        1.195975
kurt       -0.210374
Name: ElectronicsAndSemicon, dtype: float64
------------------------------------------------------------------------------
```

```
--------------------
********** ComputerScience **********
min        -1.000000
max       715.000000
mean       90.742371
median     -1.000000
std       175.273083
skew        1.529521
kurt        0.692641
Name: ComputerScience, dtype: float64
------------------------------------------------------------------------------
--------------------
********** MechanicalEngg **********
min        -1.000000
max       623.000000
mean       22.974737
median     -1.000000
std        98.123311
skew        4.029563
kurt       15.018957
Name: MechanicalEngg, dtype: float64
------------------------------------------------------------------------------
--------------------
********** ElectricalEngg **********
min        -1.000000
max       676.000000
mean       16.478739
median     -1.000000
std        87.585634
skew        5.060407
kurt       24.878194
Name: ElectricalEngg, dtype: float64
------------------------------------------------------------------------------
--------------------
********** TelecomEngg **********
min        -1.000000
max       548.000000
mean       31.851176
median     -1.000000
std       104.852845
skew        3.041261
kurt        7.810221
Name: TelecomEngg, dtype: float64
------------------------------------------------------------------------------
--------------------
********** CivilEngg **********
min        -1.000000
max       516.000000
mean        2.683842
median     -1.000000
std        36.658505
skew       10.315681
kurt      109.041349
Name: CivilEngg, dtype: float64
------------------------------------------------------------------------------
--------------------
********** conscientiousness **********
min      -4.126700
max       1.995300
mean     -0.037831
median    0.046400
std       1.028666
```

```
skew     -0.527003
kurt      0.122596
Name: conscientiousness, dtype: float64
------------------------------------------------------------------------------
--------------------
********** agreeableness **********
min      -5.781600
max       1.904800
mean      0.146496
median    0.212400
std       0.941782
skew     -1.204915
kurt      3.391242
Name: agreeableness, dtype: float64
------------------------------------------------------------------------------
--------------------
********** extraversion **********
min      -4.600900
max       2.535400
mean      0.002763
median    0.091400
std       0.951471
skew     -0.523267
kurt      0.643969
Name: extraversion, dtype: float64
------------------------------------------------------------------------------
--------------------
********** nueroticism **********
min      -2.643000
max       3.352500
mean     -0.169033
median   -0.234400
std       1.007580
skew      0.165710
kurt     -0.191539
Name: nueroticism, dtype: float64
------------------------------------------------------------------------------
--------------------
********** openess_to_experience **********
min      -7.375700
max       1.822400
mean     -0.138110
median   -0.094300
std       1.008075
skew     -1.506962
kurt      5.788327
Name: openess_to_experience, dtype: float64
------------------------------------------------------------------------------
--------------------
```

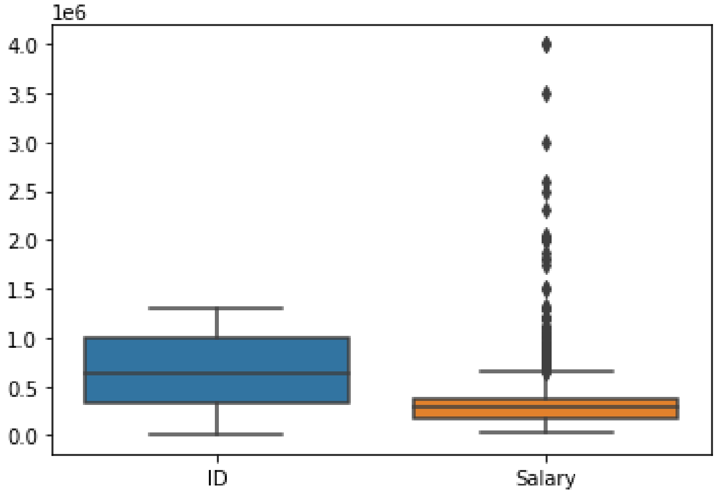
In [43]:
```
numerical_df.shape
```

Out[43]:
```
(3998, 27)
```

Univariate Analysis on Numerical Data (Visualization)

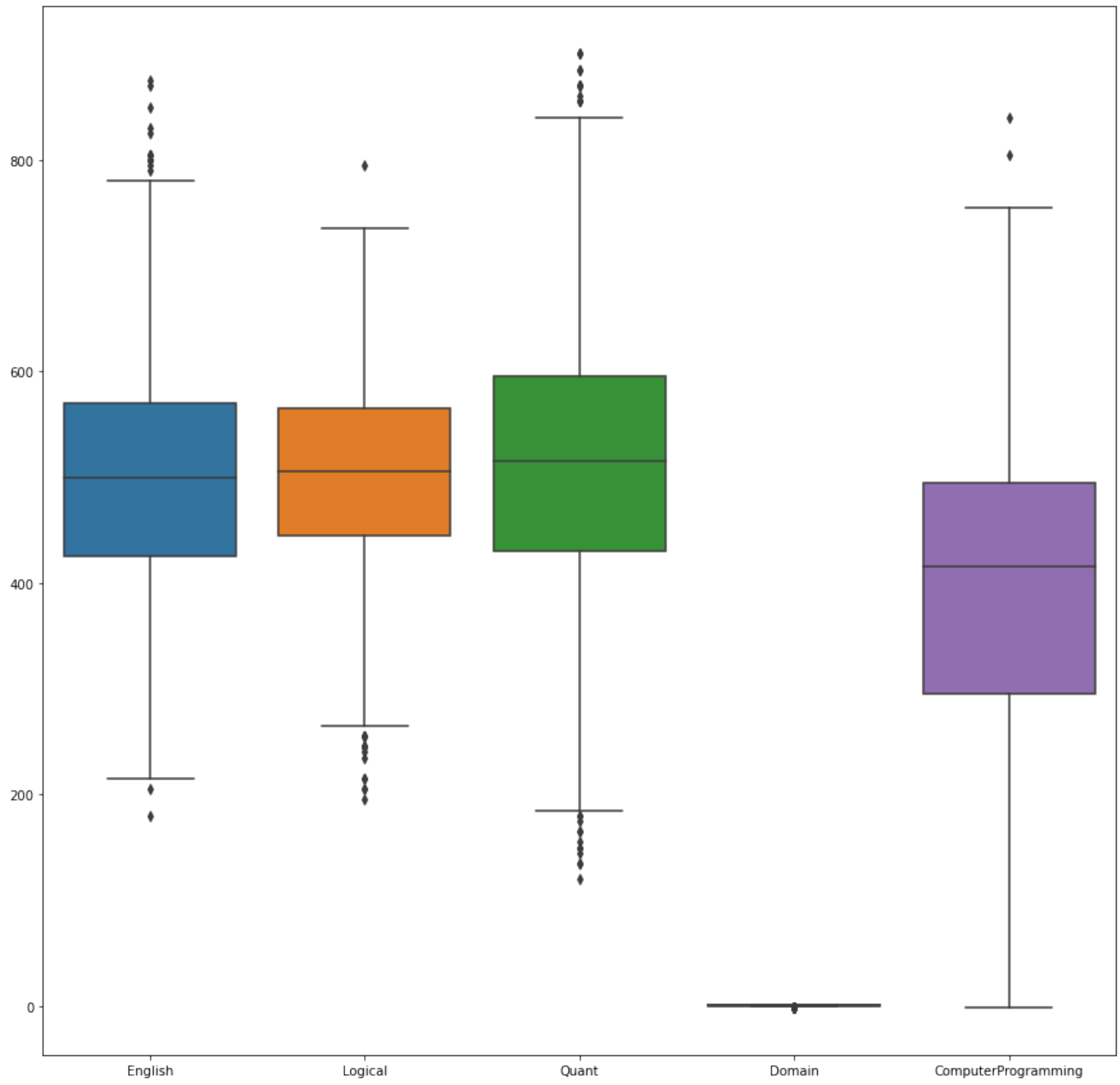
In [95]:
```
sns.boxplot(data=numerical_df.iloc[:,:2])
```

Out[95]: <AxesSubplot:>



In [96]:
```
plt.figure(figsize=(15,15))
sns.boxplot(data=numerical_df.iloc[:,11:16])
```

Out[96]: <AxesSubplot:>



In [99]:
```
def univariate_cat(data):
    for column in data:
        print("*"*6,column,"*"*5)
        print("mode of data is",data[column].mode())
        print("unique values of columns are",data[column].value_counts())
univariate_cat(df[["JobCity","Specialization"]])
```

```
****** JobCity *****
mode of data is 0    Bangalore
dtype: object
unique values of columns are Bangalore    627
-1           461
Noida        368
Hyderabad    335
Pune         290
            ...
Bathinda       1
Patiala        1
Dausa          1
Bhilai         1
Banglore       1
Name: JobCity, Length: 339, dtype: int64
****** Specialization *****
mode of data is 0    electronics and communication engineering
dtype: object
unique values of columns are electronics and communication engineering      88
0
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
information science engineering                 27
electronics and instrumentation 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
telecommunication engineering                    6
computer science and technology                  6
mechanical and automation                        5
automobile/automotive engineering                5
instrumentation engineering                      4
mechatronics                                     4
electronics and computer engineering             3
aeronautical engineering                         3
information & communication technology           2
metallurgical engineering                        2
biomedical engineering                           2
computer science                                 2
electrical and power engineering                 2
industrial engineering                           2
embedded systems technology                      1
computer networking                              1
industrial & management engineering             1
internal combustion engine                       1
control and instrumentation engineering          1
mechanical & production engineering              1
polymer technology                               1
power systems and automation                     1
electronics                                      1
```

```
information science                          1
ceramic engineering                          1
computer and communication engineering       1
Name: Specialization, dtype: int64
```

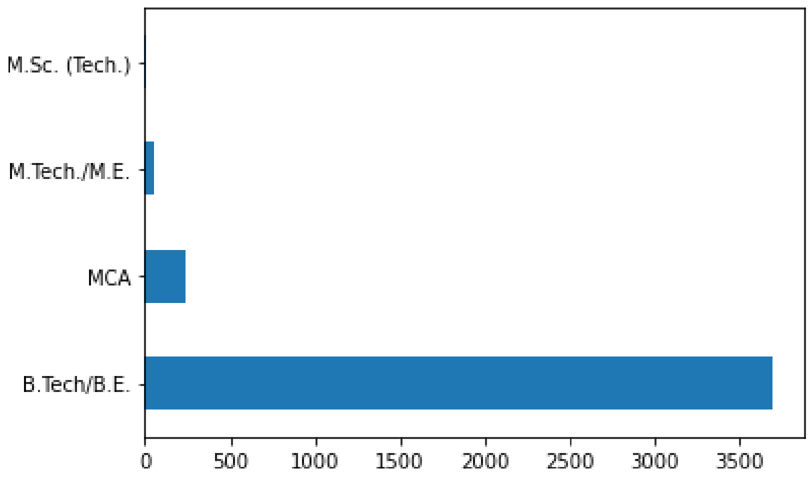
Two methods that helps performing Univariate Analysis:

Mode and Value_counts

Univariate Analysis on Categorical Data (Non-Visualize)

In [122]:
```
def visualize_cat(d):
    for column in d:
        print("*"*8,column,"*"*8)
        d[column].value_counts().plot(kind="barh")
visualize_cat(df[["Degree"]])
```

```
******** Degree ********
```



Bivariate Analysis

Categorical vs Categorical

In [117]:
```
pd.crosstab(df["Gender"],df["Specialization"],margins=True)
```

Out[117]:

Specialization	aeronautical engineering	applied electronics and instrumentation	automobile/automotive engineering	biomedical engineering	biotechnology	e
Gender						
f	1	2	0	2	9	
m	2	7	5	0	6	
All	3	9	5	2	15	

3 rows × 47 columns

In [118]:
```
pd.crosstab(df["Specialization"],df["Gender"],margins=True)
```

Out[118]:

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
instrumentation and control engineering	9	11	20
instrumentation engineering	0	4	4
internal combustion engine	0	1	1
mechanical & production engineering	0	1	1

Gender	f	m	All
Specialization			
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

Numerical vs Numerical

In [125]:
```
correlation=df["collegeGPA"].corr(df['Salary'])
print("correlation between Salary and collegeGPA is ",correlation)
```

```
correlation between Salary and collegeGPA is  0.1301025190711256
```

Numerical vs Categorical

In [133]:
```
df.groupby(["Specialization"])["Salary"].sum().sort_values(ascending=False)
```

Out[133]:
```
Specialization
electronics and communication engineering      261195000.0
computer engineering                           224460000.0
computer science & engineering                 206415000.0
information technology                         203605000.0
computer application                            68415000.0
mechanical engineering                          63809000.0
electronics and electrical engineering          56235000.0
electronics & telecommunications                35520000.0
electrical engineering                          24090000.0
electronics & instrumentation eng               11665000.0
civil engineering                               11055000.0
electronics and instrumentation engineering      8840000.0
instrumentation and control engineering          7880000.0
information science engineering                  7460000.0
electronics engineering                          5310000.0
industrial & production engineering              3845000.0
biotechnology                                    3815000.0
other                                            3465000.0
chemical engineering                             3330000.0
applied electronics and instrumentation          3135000.0
telecommunication engineering                    2055000.0
mechanical and automation                        1545000.0
computer science and technology                  1475000.0
automobile/automotive engineering                1110000.0
mechatronics                                     1015000.0
instrumentation engineering                       960000.0
information & communication technology            775000.0
industrial engineering                            740000.0
polymer technology                                700000.0
metallurgical engineering                         675000.0
electronics and computer engineering              660000.0
biomedical engineering                            580000.0
computer science                                  580000.0
computer networking                               565000.0
information science                               460000.0
aeronautical engineering                          445000.0
electrical and power engineering                  420000.0
internal combustion engine                        360000.0
ceramic engineering                               335000.0
industrial & management engineering              320000.0
control and instrumentation engineering           305000.0
embedded systems technology                       200000.0
computer and communication engineering            120000.0
mechanical & production engineering               100000.0
power systems and automation                      100000.0
electronics                                        40000.0
Name: Salary, dtype: float64
```

In [135]:
```
group = df.groupby('Specialization')

group['Salary'].agg(['min', 'max', 'mean', 'median'])
```

Out[135]:

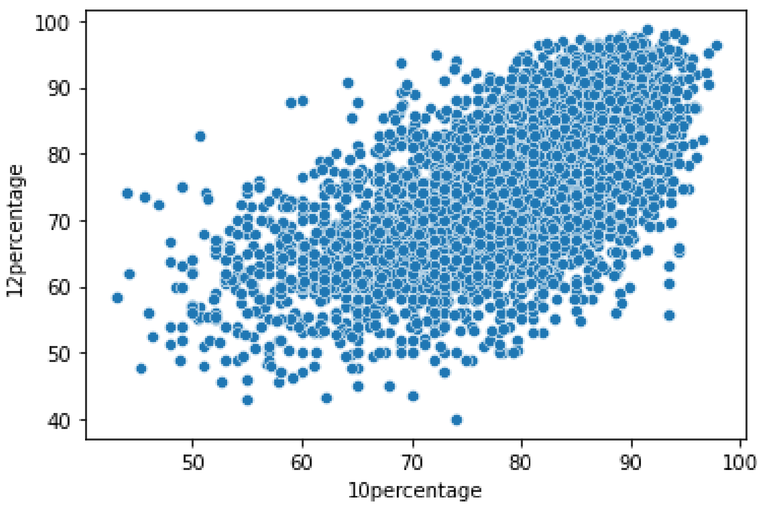
Specialization	min	max	mean	median
aeronautical engineering	120000.0	180000.0	148333.333333	145000.0
applied electronics and instrumentation	175000.0	950000.0	348333.333333	300000.0
automobile/automotive engineering	100000.0	400000.0	222000.000000	130000.0
biomedical engineering	145000.0	435000.0	290000.000000	290000.0
biotechnology	100000.0	450000.0	254333.333333	235000.0
ceramic engineering	335000.0	335000.0	335000.000000	335000.0
chemical engineering	100000.0	730000.0	370000.000000	375000.0
civil engineering	110000.0	800000.0	381206.896552	320000.0
computer and communication engineering	120000.0	120000.0	120000.000000	120000.0
computer application	50000.0	4000000.0	280389.344262	217500.0
computer engineering	35000.0	4000000.0	374100.000000	350000.0
computer networking	565000.0	565000.0	565000.000000	565000.0
computer science	180000.0	400000.0	290000.000000	290000.0
computer science & engineering	35000.0	2050000.0	277439.516129	280000.0
computer science and technology	100000.0	360000.0	245833.333333	250000.0
control and instrumentation engineering	305000.0	305000.0	305000.000000	305000.0
electrical and power engineering	180000.0	240000.0	210000.000000	210000.0
electrical engineering	40000.0	1860000.0	293780.487805	300000.0
electronics	40000.0	40000.0	40000.000000	40000.0
electronics & instrumentation eng	100000.0	2300000.0	364531.250000	310000.0
electronics & telecommunications	45000.0	630000.0	293553.719008	300000.0
electronics and communication engineering	45000.0	3000000.0	296812.500000	300000.0
electronics and computer engineering	120000.0	300000.0	220000.000000	240000.0
electronics and electrical engineering	45000.0	2500000.0	286913.265306	280000.0
electronics and instrumentation engineering	50000.0	1745000.0	327407.407407	300000.0
electronics engineering	110000.0	410000.0	279473.684211	300000.0
embedded systems technology	200000.0	200000.0	200000.000000	200000.0
industrial & management engineering	320000.0	320000.0	320000.000000	320000.0
industrial & production engineering	170000.0	660000.0	384500.000000	382500.0
industrial engineering	350000.0	390000.0	370000.000000	370000.0
information & communication technology	325000.0	450000.0	387500.000000	387500.0
information science	460000.0	460000.0	460000.000000	460000.0
information science engineering	100000.0	570000.0	276296.296296	245000.0
information technology	35000.0	2000000.0	308492.424242	300000.0
instrumentation and control engineering	150000.0	1300000.0	394000.000000	312500.0
instrumentation engineering	200000.0	260000.0	240000.000000	250000.0
internal combustion engine	360000.0	360000.0	360000.000000	360000.0
mechanical & production engineering	100000.0	100000.0	100000.000000	100000.0

Specialization	min	max	mean	median
mechanical and automation	180000.0	500000.0	309000.000000	300000.0
mechanical engineering	60000.0	1300000.0	317457.711443	275000.0
mechatronics	100000.0	350000.0	253750.000000	282500.0
metallurgical engineering	300000.0	375000.0	337500.000000	337500.0
other	110000.0	600000.0	266538.461538	240000.0
polymer technology	700000.0	700000.0	700000.000000	700000.0
power systems and automation	100000.0	100000.0	100000.000000	100000.0
telecommunication engineering	275000.0	400000.0	342500.000000	350000.0

Numerical vs Numerical

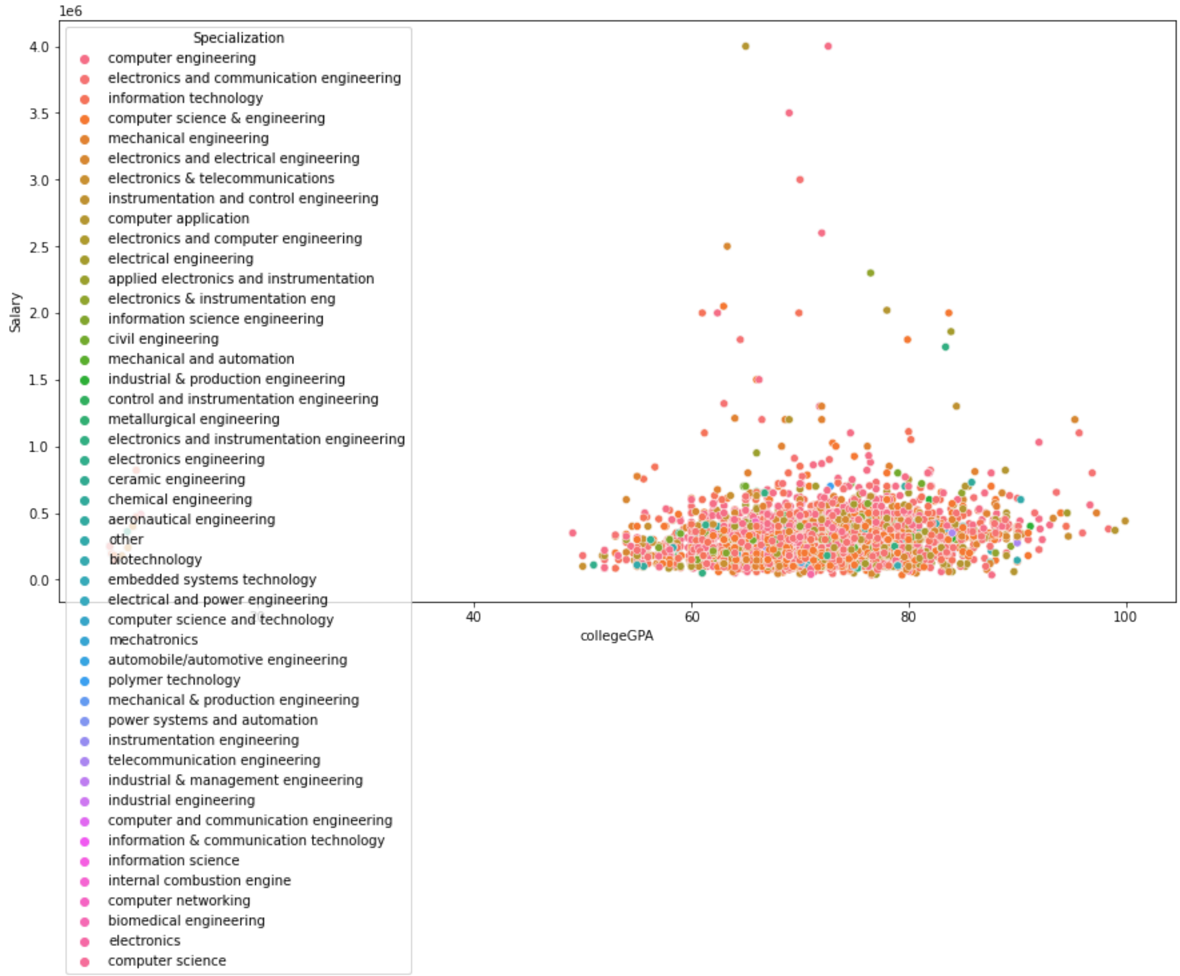
In [136]:
```
sns.scatterplot(data=df,x="10percentage",y="12percentage")
```

Out[136]:
```
<AxesSubplot:xlabel='10percentage', ylabel='12percentage'>
```



In [144]:
```
plt.figure(figsize=(15,8))
sns.scatterplot(data=df, x='collegeGPA',y='Salary',hue='Specialization')
```

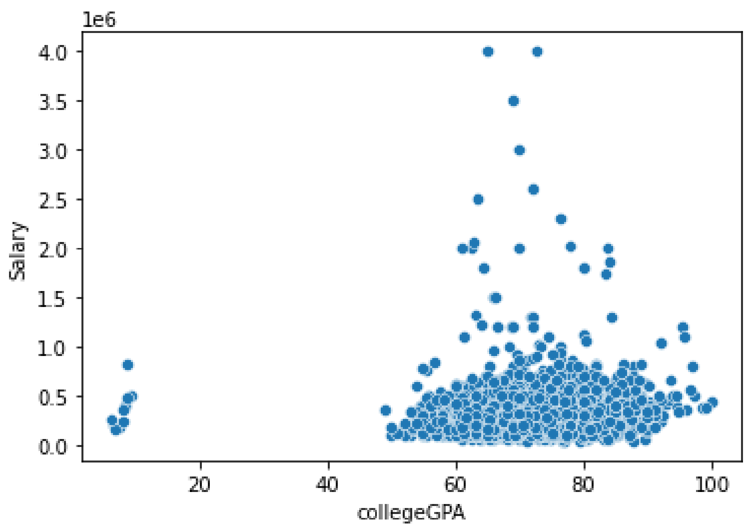
Out[144]: <AxesSubplot:xlabel='collegeGPA', ylabel='Salary'>



The Scatter plot shows the Strong positive Correlation

In [137]:
```
sns.scatterplot(data=df,x="collegeGPA",y="Salary")
```

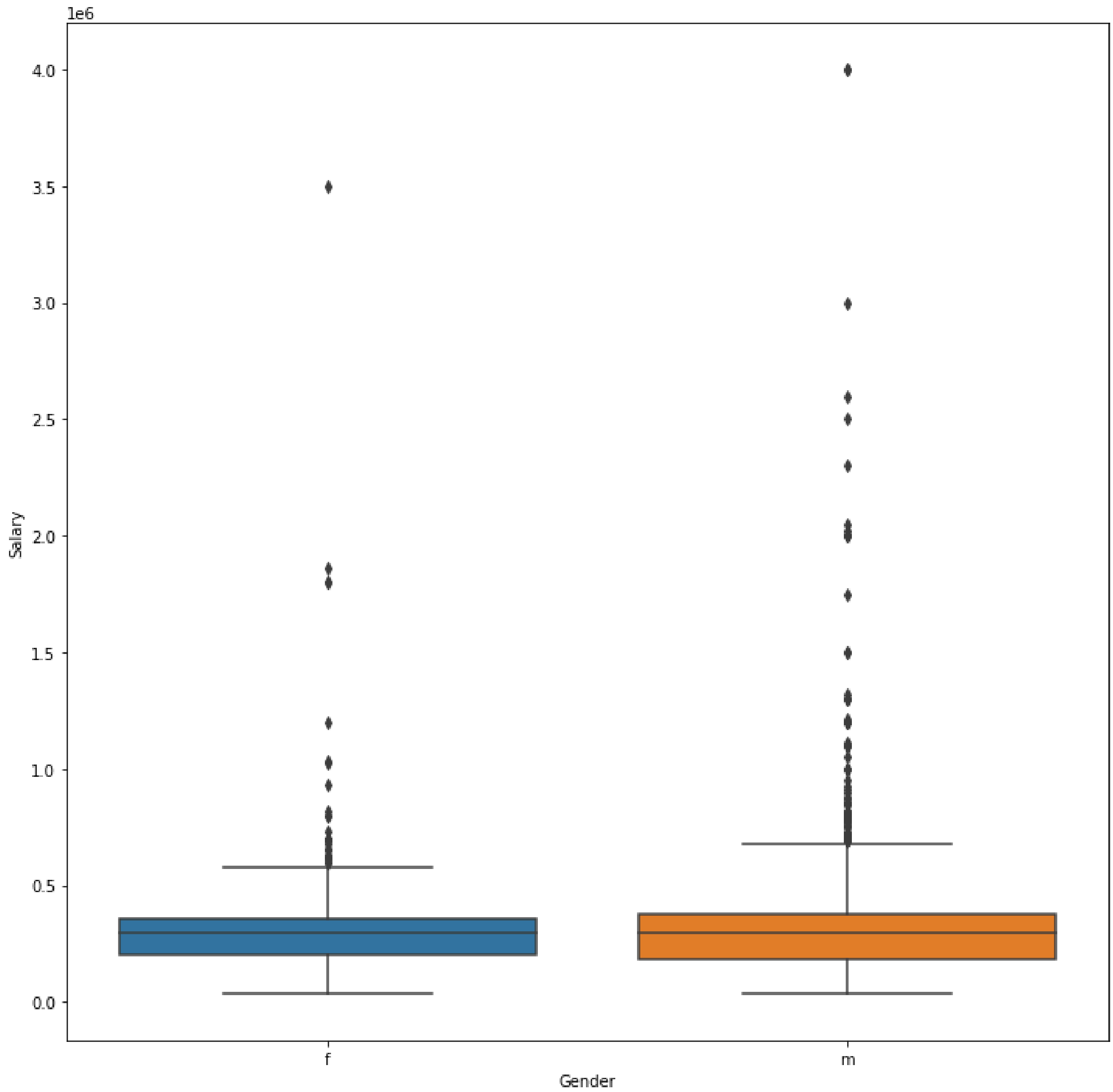
Out[137]: <AxesSubplot:xlabel='collegeGPA', ylabel='Salary'>



Categorical vs Numerical (Visualize)

In [139]:
```
plt.figure(figsize=(12,12))
sns.boxplot(data=df,y="Salary",x="Gender")
```

Out[139]:
```
<AxesSubplot:xlabel='Gender', ylabel='Salary'>
```



The Median Salary for Male Endineers is High compare to the Salary of Female Engineers

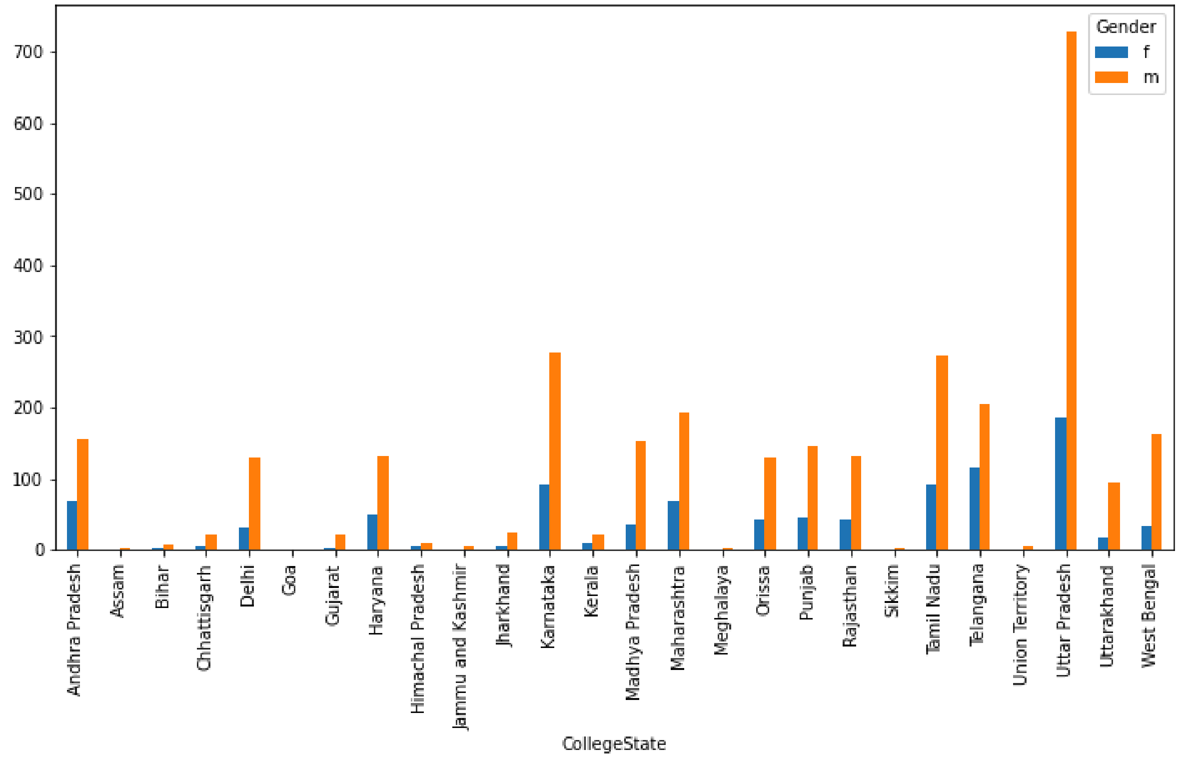
In [146]:
```
grouped=df.groupby(["CollegeState","Gender"]).size().unstack(fill_value=0)
grouped
```

Out[146]:

Gender	f	m
CollegeState		
Andhra Pradesh	69	156
Assam	1	4
Bihar	2	8
Chhattisgarh	6	21
Delhi	32	130
Goa	0	1
Gujarat	2	22
Haryana	49	131
Himachal Pradesh	5	11
Jammu and Kashmir	1	6
Jharkhand	5	23
Karnataka	93	277
Kerala	11	22
Madhya Pradesh	36	153
Maharashtra	68	194
Meghalaya	0	2
Orissa	42	130
Punjab	46	147
Rajasthan	43	131
Sikkim	1	2
Tamil Nadu	93	274
Telangana	115	204
Union Territory	0	5
Uttar Pradesh	186	729
Uttarakhand	18	95
West Bengal	33	163

In [147]:
```
grouped.plot(kind="bar",figsize=(12,6))
```

Out[147]: <AxesSubplot:xlabel='CollegeState'>

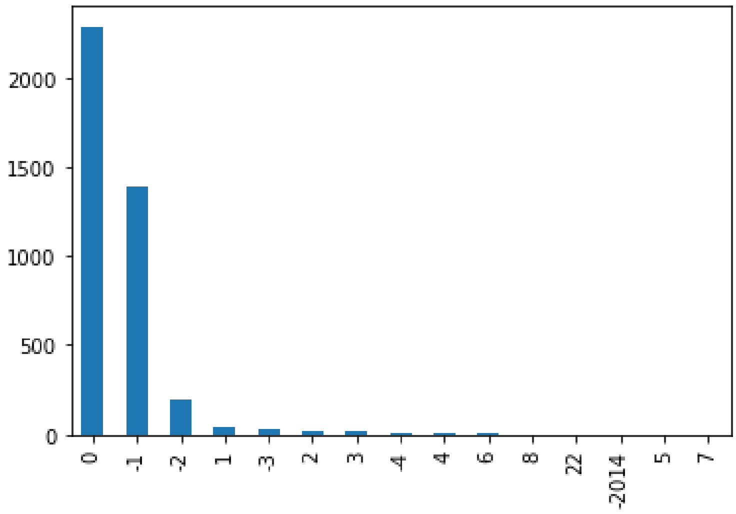


UP has highest working proffessionals

Times of India article 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 Enginner You can earn up to 2.5-3 lakhs as a fresh graduate." Test this claim with the data giben to you.

In [168]:
```
df["DOJ"]=pd.to_datetime(df["DOJ"])
df["diff_grad_join"]=df["GraduationYear"]-df["DOJ"].dt.year
df["diff_grad_join"].value_counts().plot(kind="bar")
```

Out[168]: <AxesSubplot:>



In [159]:
```
df1=df[["Designation","Specialization","Salary"]]
```

In [176]:
```
# Assuming df1 is the original DataFrame
df1_filter = df1[(df1["Designation"] == "programmer analyst") &
                 (df1["Specialization"] == "computer science & engineering")]

# Display the first 5 rows of the filtered DataFrame
print(df1_filter.head())
```

```
             Designation                  Specialization    Salary
24   programmer analyst  computer science & engineering  335000.0
473  programmer analyst  computer science & engineering  335000.0
530  programmer analyst  computer science & engineering  345000.0
595  programmer analyst  computer science & engineering  180000.0
767  programmer analyst  computer science & engineering  340000.0
```

In [181]:
```
df1_filter.count()
```

Out[181]:
```
Designation       26
Specialization    26
Salary            26
dtype: int64
```

In [177]:
```
df1_filter.plot(kind="box")
```

Out[177]: <AxesSubplot:>

In [183]:
```
df1_filter1 = df1[(df1["Designation"] == "software engineer") &
                  (df1["Specialization"] == "computer science & engineering")]
df1_filter1.head()
```

Out[183]:

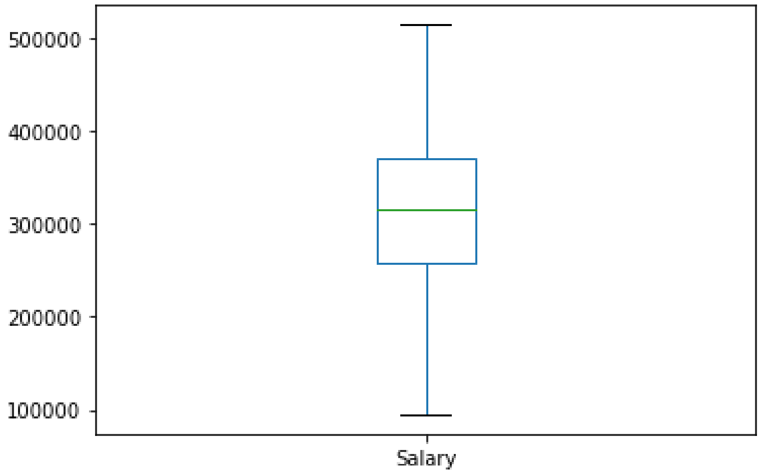
	Designation	Specialization	Salary
31	software engineer	computer science & engineering	340000.0
48	software engineer	computer science & engineering	390000.0
52	software engineer	computer science & engineering	400000.0
55	software engineer	computer science & engineering	250000.0
113	software engineer	computer science & engineering	340000.0

In [186]:
```
print(df1_filter1["Salary"].max())
print(df1_filter1["Salary"].min())
print(df1_filter1["Salary"].count())
```

```
1000000.0
85000.0
139
```

In [187]:
```
df1_filter1["Salary"].plot(kind="box",showfliers=False)
```

Out[187]: <AxesSubplot:>



In [189]:
```
df1_filter1 = df1[(df1["Designation"] == "associate engineer") &
                  (df1["Specialization"] == "computer science & engineering")]
df1_filter1.head()
```

Out[189]:

	Designation	Specialization	Salary
819	associate engineer	computer science & engineering	350000.0
3134	associate engineer	computer science & engineering	315000.0

Programming Analyst,Software Engineer and Associate Engineer can earn up to 2.5-3 lakhs as a fresher graduate.

In [190]:
```
df[["Gender","Specialization"]].head()
```

Out[190]:

	Gender	Specialization
0	f	computer engineering
1	m	electronics and communication engineering
2	f	information technology
3	m	computer engineering
4	m	electronics and communication engineering

In [191]:
```
df["Gender"].value_counts()
```

Out[191]:
```
m    3041
f     957
Name: Gender, dtype: int64
```

In [195]:
```
print("Percentage of Females")
print((957/3998)*100)
```

```
Percentage of Females
23.936968484242122
```

In [196]:
```
grouped=df.groupby(["Specialization","Gender"]).size().unstack(fill_value=0)
grouped
```

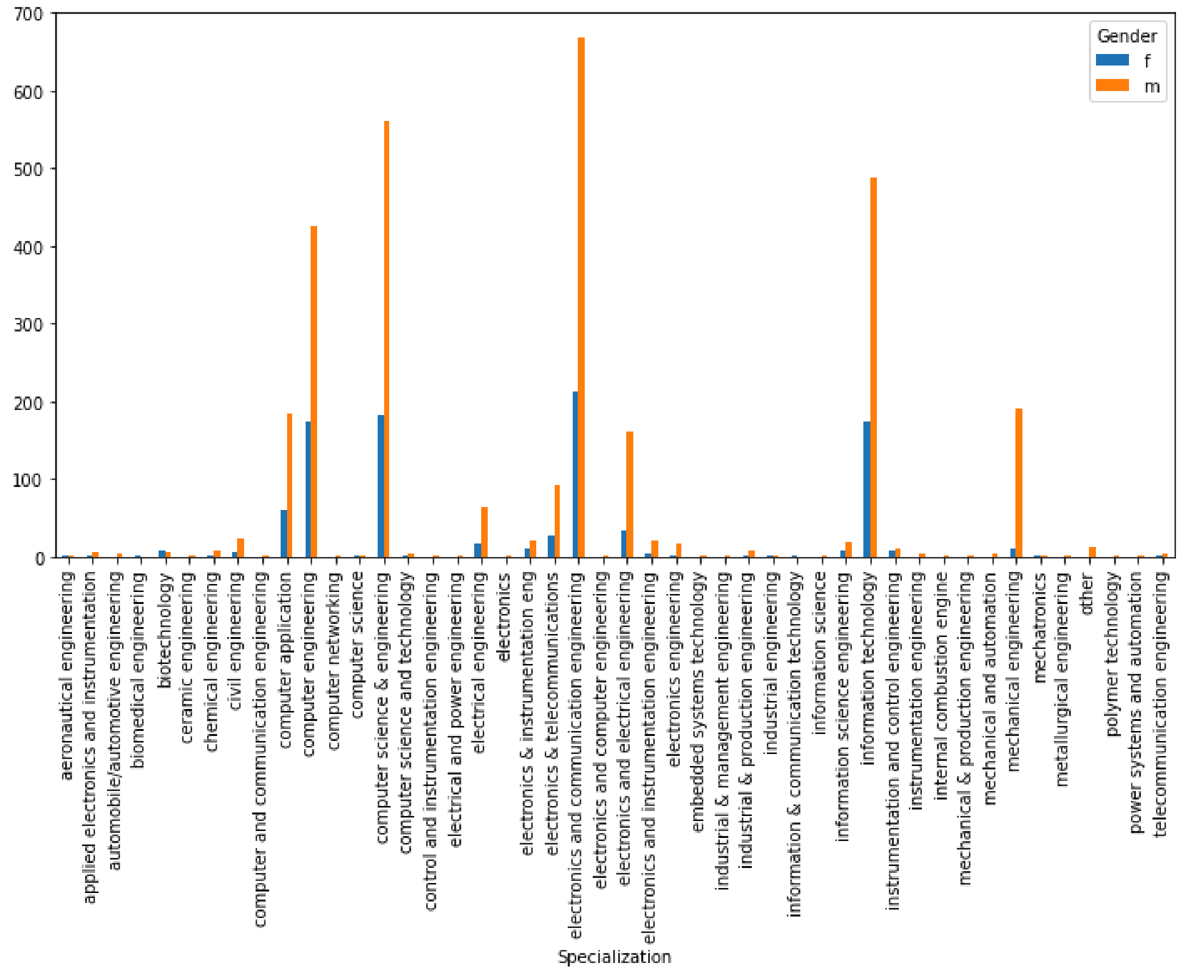
Out[196]:

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
instrumentation and control engineering	9	11
instrumentation engineering	0	4
internal combustion engine	0	1
mechanical & production engineering	0	1

Gender	f	m
Specialization		
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 [197]:
```
grouped.plot(kind="bar",figsize=(12,6))
```

Out[197]: <AxesSubplot:xlabel='Specialization'>



OBJECTIVE OF THE ANALYSIS

```
**KEY INSIGHTS:** This Analysis provided the significant insights into
distribution of Salaries and the Factors influencing them.

**Skill Impact**: Examined the relationship between cognitive, technical, and
personality skills with salary outcomes, revealing significant predictors.

**GENDER & SPECIALIZATION:** The Preferences for specialization appears to
have the some correlation with Gender.
```

Explore Trends: Look into how salaries vary by the factors such as location,gender and job roles to identify the potential inequalities in the job market.

Overall this anaysis aim to provide valuable insights for Engineering Graduates.

In [ ]: