AMCAT Data Analysis

In [12]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")
from scipy import stats as st
```

In [13]:
```
#Importing the Dataset
df=pd.read_excel(r'C:\\Amcat Project\\iml_data.xlsx')
```

Description of Data

In [15]:
```
df.drop("Unnamed: 0",axis=1,inplace=True)
df.head()
```

Out[15]:

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

5 rows × 38 columns

In [16]:
```
df.tail()
```

Out[16]:

	ID	Salary	DOJ	DOL	Designation	JobCity	Gend
3993	47916	280000	2011-10-01	2012-10-01 00:00:00	software engineer	New Delhi	
3994	752781	100000	2013-07-01	2013-07-01 00:00:00	technical writer	Hyderabad	
3995	355888	320000	2013-07-01	present	associate software engineer	Bangalore	
3996	947111	200000	2014-07-01	2015-01-01 00:00:00	software developer	Asifabadbanglore	
3997	324966	400000	2013-02-01	present	senior systems engineer	Chennai	

5 rows × 38 columns

In [17]:
```
df.shape
```

Out[17]: (3998, 38)

In [18]:
```
# Major description of data
df.describe()
```

Out[18]:

	ID	Salary	DOJ	DOB
count	3.998000e+03	3.998000e+03	3998	3998
mean	6.637945e+05	3.076998e+05	2013-07-02 11:04:10.325162496	1990-12-06 06:01:15.637819008
min	1.124400e+04	3.500000e+04	1991-06-01 00:00:00	1977-10-30 00:00:00
25%	3.342842e+05	1.800000e+05	2012-10-01 00:00:00	1989-11-16 06:00:00
50%	6.396000e+05	3.000000e+05	2013-11-01 00:00:00	1991-03-07 12:00:00
75%	9.904800e+05	3.700000e+05	2014-07-01 00:00:00	1992-03-13 18:00:00
max	1.298275e+06	4.000000e+06	2015-12-01 00:00:00	1997-05-27 00:00:00
std	3.632182e+05	2.127375e+05	NaN	NaN

8 rows × 29 columns

In [19]:
```
#Displaying the column names
df.columns
```

Out[19]:
```
Index(['ID', 'Salary', 'DOJ', 'DOL', 'Designation', 'JobCity', 'Gender',
'DOB',
       '10percentage', '10board', '12graduation', '12percentage', '12boa
rd',
       'CollegeID', 'CollegeTier', 'Degree', 'Specialization', 'collegeG
PA',
       'CollegeCityID', 'CollegeCityTier', 'CollegeState', 'GraduationYe
ar',
       'English', 'Logical', 'Quant', 'Domain', 'ComputerProgramming',
       'ElectronicsAndSemicon', 'ComputerScience', 'MechanicalEngg',
       'ElectricalEngg', 'TelecomEngg', 'CivilEngg', 'conscientiousnes
s',
       'agreeableness', 'extraversion', 'nueroticism',
       'openess_to_experience'],
      dtype='object')
```

In [20]:
```
df.info()
```

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

1. Exploratory Data Analysis

To get Insights from data-Missing Values/Duplicated Values,Outliers,Distributions

In [22]:
```
#Checking missing values
df.isna().sum()
```

Out[22]:
```
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 [23]:
```
#Check for presence of any Duplicated Values
df.duplicated().sum()
```

Out[23]:
```
0
```

In [24]:
```
#To check for  outliers
def count_outliers_iqr(df):
    for col in df.select_dtypes(include=['float64','int64']).columns:
        Q1=df[col].quantile(0.25)
        Q3=df[col].quantile(0.75)
        IQR=Q3-Q1

        lower_bound=Q1-1.5*IQR
        upper_bound=Q3+1.5*IQR

        outliers=df[(df[col]<lower_bound) | (df[col] > upper_bound)]
        num_outliers=outliers[col].count()
        print(f"'{col}':{num_outliers}")
```

```
count_outliers_iqr(df)
```

```
'ID':0
'Salary':109
'10percentage':30
'12graduation':45
'12percentage':1
'CollegeID':0
'CollegeTier':297
'collegeGPA':38
'CollegeCityID':0
'CollegeCityTier':0
'GraduationYear':2
'English':15
'Logical':18
'Quant':25
'Domain':246
'ComputerProgramming':2
'ElectronicsAndSemicon':2
'ComputerScience':902
'MechanicalEngg':235
'ElectricalEngg':161
'TelecomEngg':374
'CivilEngg':42
'conscientiousness':39
'agreeableness':123
'extraversion':40
'nueroticism':15
'openess_to_experience':95
```

In [25]:

```
#To remove outliers
def remove_outliers_iqr(df):
    for col in df.select_dtypes(include=['float64','int64']).columns:
        Q1=df[col].quantile(0.25)
        Q3=df[col].quantile(0.75)
        IQR=Q3-Q1

        lower_bound=Q1-1.5*IQR
        upper_bound=Q3+1.5*IQR

        outliers=df[(df[col]<lower_bound) | (df[col] > upper_bound)]
    return df
df_cleaned=remove_outliers_iqr(df)
print("Dataframe after removing outliers")
print(df_cleaned)
```

```
Dataframe after removing outliers
          ID   Salary        DOJ                  DOL  \
0     203097   420000 2012-06-01              present   
1     579905   500000 2013-09-01              present   
2     810601   325000 2014-06-01              present   
3     267447  1100000 2011-07-01              present   
4     343523   200000 2014-03-01  2015-03-01 00:00:00   
...      ...      ...        ...                  ...   
3993   47916   280000 2011-10-01  2012-10-01 00:00:00   
3994  752781   100000 2013-07-01  2013-07-01 00:00:00   
3995  355888   320000 2013-07-01              present   
3996  947111   200000 2014-07-01  2015-01-01 00:00:00   
3997  324966   400000 2013-02-01              present   

                       Designation          JobCity Gender        DOB  \
0         senior quality engineer        Bangalore      f 1990-02-19   
1               assistant manager           Indore      m 1989-10-04   
2                systems engineer          Chennai      f 1992-08-03   
3        senior software engineer          Gurgaon      m 1989-12-05   
4                              get          Manesar      m 1991-02-27   
...                            ...              ...    ...        ...   
3993            software engineer        New Delhi      m 1987-04-15   
3994             technical writer        Hyderabad      f 1992-08-27   
3995  associate software engineer        Bangalore      m 1991-07-03   
3996           software developer  Asifabadbanglore      f 1992-03-20   
3997      senior systems engineer          Chennai      f 1991-02-26   

      10percentage                          10board  ...  ComputerScience 
\
0            84.30  board ofsecondary education,ap  ...               -1   
1            85.40                             cbse  ...               -1   
2            85.00                             cbse  ...               -1   
3            85.60                             cbse  ...               -1   
4            78.00                             cbse  ...               -1   
...            ...                              ...  ...              ...   
3993         52.09                             cbse  ...               -1   
3994         90.00                      state board  ...               -1   
3995         81.86                       bse,odisha  ...               -1   
3996         78.72                      state board  ...              438   
3997         70.60                             cbse  ...               -1   

      MechanicalEngg ElectricalEngg  TelecomEngg  CivilEngg conscientiousn
ess  \
0                 -1             -1           -1         -1            0.9
737   
1                 -1             -1           -1         -1           -0.7
335   
2                 -1             -1           -1         -1            0.2
718   
3                 -1             -1           -1         -1            0.0
464   
4                 -1             -1           -1         -1           -0.8
810   
...              ...            ...          ...        ...               
...   
3993              -1             -1           -1         -1           -0.1
082   
3994              -1             -1           -1         -1           -0.3
027   
3995              -1             -1           -1         -1           -1.5
```

```
765
3996                -1              -1            -1          -1            -0.1
590
3997                -1              -1            -1          -1            -1.1
128

     agreeableness  extraversion  nueroticism  openess_to_experience
0           0.8128        0.5269      1.35490                -0.4455
1           0.3789        1.2396     -0.10760                 0.8637
2           1.7109        0.1637     -0.86820                 0.6721
3           0.3448       -0.3440     -0.40780                -0.9194
4          -0.2793       -1.0697      0.09163                -0.1295
...            ...           ...          ...                    ...
3993        0.3448        0.2366      0.64980                -0.9194
3994        0.8784        0.9322      0.77980                -0.0943
3995       -1.5273       -1.5051     -1.31840                -0.7615
3996        0.0459       -0.4511     -0.36120                -0.0943
3997       -0.2793       -0.6343      1.32553                -0.6035

[3998 rows x 38 columns]
```

Univariate Analysis on Numerical Data(Non-Visualization)

In [27]:
```python
numerical_df = df.select_dtypes(include=['int64', 'float64'])
def num_univariate_analysis(numerical_data):
    for column in numerical_data:
        print("*" *8, column, "*" * 8)
        # Basic statistics
        print(numerical_data[column].agg(['min', 'max', 'mean', 'median',

        print("Skewness:", numerical_data[column].skew())

        print("Kurtosis:", numerical_data[column].kurt())
        print()

num_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
Name: ID, dtype: float64
Skewness: 0.05477046850906638
Kurtosis: -1.2226938327845243

******** Salary ********
min       3.500000e+04
max       4.000000e+06
mean      3.076998e+05
median    3.000000e+05
std       2.127375e+05
Name: Salary, dtype: float64
Skewness: 6.451081166224832
Kurtosis: 80.92999627162538

******** 10percentage ********
min       43.000000
max       97.760000
mean      77.925443
median    79.150000
std        9.850162
Name: 10percentage, dtype: float64
Skewness: -0.5910185081648047
Kurtosis: -0.1102843100198605

******** 12graduation ********
min       1995.000000
max       2013.000000
mean      2008.087544
median    2008.000000
std          1.653599
Name: 12graduation, dtype: float64
Skewness: -0.9640901430967733
Kurtosis: 1.9511644059905469

******** 12percentage ********
min       40.000000
max       98.700000
mean      74.466366
median    74.400000
std       10.999933
Name: 12percentage, dtype: float64
Skewness: -0.03260741437482245
Kurtosis: -0.6307374665885321

******** CollegeID ********
min           2.000000
max       18409.000000
mean       5156.851426
median     3879.000000
std        4802.261482
Name: CollegeID, dtype: float64
Skewness: 0.649176333927607
Kurtosis: -0.7674413638286568
```

```
******** CollegeTier ********
min       1.000000
max       2.000000
mean      1.925713
median    2.000000
std       0.262270
Name: CollegeTier, dtype: float64
Skewness: -3.2479906747351404
Kurtosis: 8.553722173976427

******** collegeGPA ********
min        6.450000
max       99.930000
mean      71.486171
median    71.720000
std        8.167338
Name: collegeGPA, dtype: float64
Skewness: -1.2492091640381637
Kurtosis: 10.234244459804753

******** CollegeCityID ********
min           2.000000
max       18409.000000
mean       5156.851426
median     3879.000000
std        4802.261482
Name: CollegeCityID, dtype: float64
Skewness: 0.649176333927607
Kurtosis: -0.7674413638286568

******** CollegeCityTier ********
min       0.000000
max       1.000000
mean      0.300400
median    0.000000
std       0.458489
Name: CollegeCityTier, dtype: float64
Skewness: 0.8711203104937956
Kurtosis: -1.2417708510593095

******** GraduationYear ********
min          0.000000
max       2017.000000
mean      2012.105803
median    2013.000000
std         31.857271
Name: GraduationYear, dtype: float64
Skewness: -63.06806402522399
Kurtosis: 3984.3696957519783

******** English ********
min       180.000000
max       875.000000
mean      501.649075
median    500.000000
std       104.940021
Name: English, dtype: float64
Skewness: 0.1919970174188361
Kurtosis: -0.2541325252956774
```

```
******** Logical ********
min       195.000000
max       795.000000
mean      501.598799
median    505.000000
std        86.783297
Name: Logical, dtype: float64
Skewness: -0.21660181091305136
Kurtosis: -0.2247605173210978

******** Quant ********
min       120.000000
max       900.000000
mean      513.378189
median    515.000000
std       122.302332
Name: Quant, dtype: float64
Skewness: -0.01939903459277611
Kurtosis: -0.10247207606308217

******** Domain ********
min      -1.000000
max       0.999910
mean      0.510490
median    0.622643
std       0.468671
Name: Domain, dtype: float64
Skewness: -1.9221455634359381
Kurtosis: 3.8959505721059204

******** ComputerProgramming ********
min        -1.000000
max       840.000000
mean      353.102801
median    415.000000
std       205.355519
Name: ComputerProgramming, dtype: float64
Skewness: -0.7781056485649357
Kurtosis: -0.6663518344809041

******** ElectronicsAndSemicon ********
min        -1.000000
max       612.000000
mean       95.328414
median     -1.000000
std       158.241218
Name: ElectronicsAndSemicon, dtype: float64
Skewness: 1.1959748726431938
Kurtosis: -0.21037436823728006

******** ComputerScience ********
min        -1.000000
max       715.000000
mean       90.742371
median     -1.000000
std       175.273083
Name: ComputerScience, dtype: float64
Skewness: 1.529520866328104
Kurtosis: 0.6926409046511801
```

```
******** MechanicalEngg ********
min        -1.000000
max       623.000000
mean       22.974737
median     -1.000000
std        98.123311
Name: MechanicalEngg, dtype: float64
Skewness: 4.029563440339185
Kurtosis: 15.018956772540893

******** ElectricalEngg ********
min        -1.000000
max       676.000000
mean       16.478739
median     -1.000000
std        87.585634
Name: ElectricalEngg, dtype: float64
Skewness: 5.060407240676985
Kurtosis: 24.878193736299266

******** TelecomEngg ********
min        -1.000000
max       548.000000
mean       31.851176
median     -1.000000
std       104.852845
Name: TelecomEngg, dtype: float64
Skewness: 3.041260613001428
Kurtosis: 7.810221301546891

******** CivilEngg ********
min        -1.000000
max       516.000000
mean        2.683842
median     -1.000000
std        36.658505
Name: CivilEngg, dtype: float64
Skewness: 10.315681229498226
Kurtosis: 109.04134879243459

******** conscientiousness ********
min      -4.126700
max       1.995300
mean     -0.037831
median    0.046400
std       1.028666
Name: conscientiousness, dtype: float64
Skewness: -0.5270033403119503
Kurtosis: 0.12259561914392192

******** agreeableness ********
min      -5.781600
max       1.904800
mean      0.146496
median    0.212400
std       0.941782
Name: agreeableness, dtype: float64
Skewness: -1.2049152493551414
Kurtosis: 3.391242301790775
```

```
******** extraversion ********
min      -4.600900
max       2.535400
mean      0.002763
median    0.091400
std       0.951471
Name: extraversion, dtype: float64
Skewness: -0.5232667810368843
Kurtosis: 0.643968724144869

******** nueroticism ********
min      -2.643000
max       3.352500
mean     -0.169033
median   -0.234400
std       1.007580
Name: nueroticism, dtype: float64
Skewness: 0.16570968491563792
Kurtosis: -0.1915388018144335

******** openess_to_experience ********
min      -7.375700
max       1.822400
mean     -0.138110
median   -0.094300
std       1.008075
Name: openess_to_experience, dtype: float64
Skewness: -1.5069620137292778
Kurtosis: 5.788327241231794
```

Univariate Analysis on Numerical Data(Visualization)

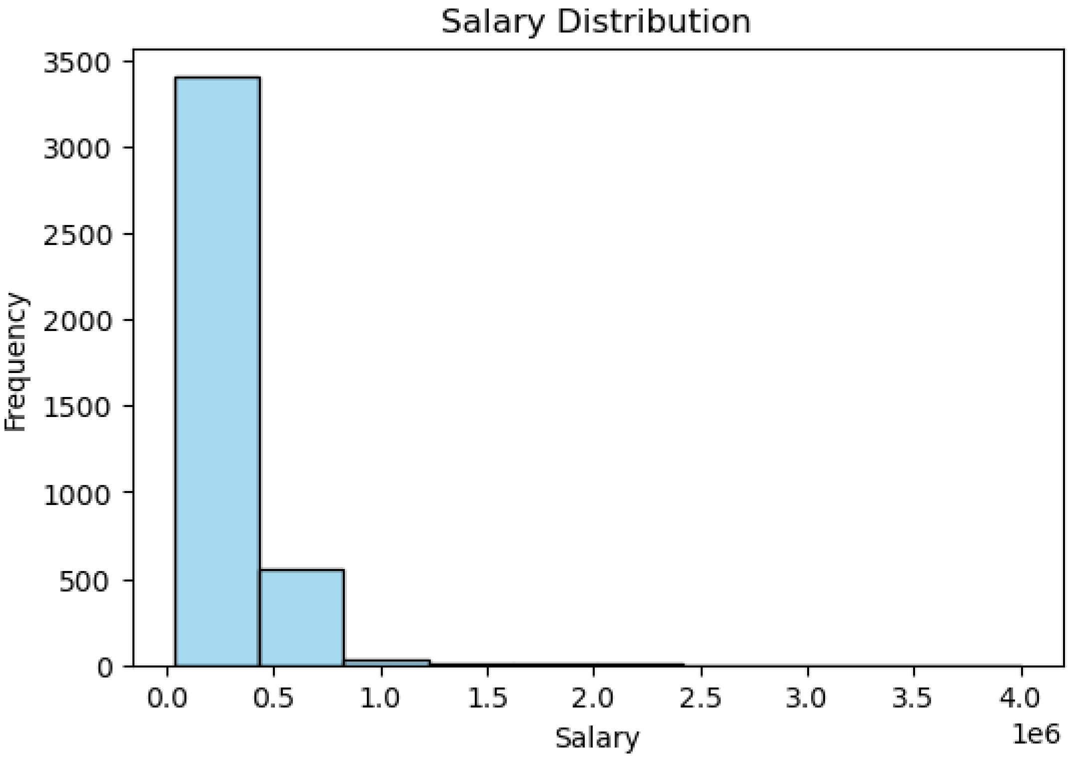
Analysis of the data using single feature/variable

In [29]:
```python
pd.DataFrame(df["Salary"].describe())
```

Out[29]:

	Salary
count	3.998000e+03
mean	3.076998e+05
std	2.127375e+05
min	3.500000e+04
25%	1.800000e+05
50%	3.000000e+05
75%	3.700000e+05
max	4.000000e+06

In [30]:
```python
#Histogram for Salary
plt.figure(figsize=(6, 4))
sns.histplot(df['Salary'], bins=10, kde=False, color='skyblue')
plt.title('Salary Distribution')
plt.xlabel('Salary')
plt.ylabel('Frequency')
plt.show()
```



1.The distribution appears like it is **right-skewed**

2.Most of the people earn a salary ranging from 0 to 100000

3.There are less people who earn more than 2.5 lakhs

What is the average 12th percentage of students?

In [33]:
```
df["12percentage"].mean()
```

Out[33]: 74.46636568284141

What are the counts of different College Tier?

In [35]:
```
pd.DataFrame(df["CollegeTier"].value_counts())
```

Out[35]:

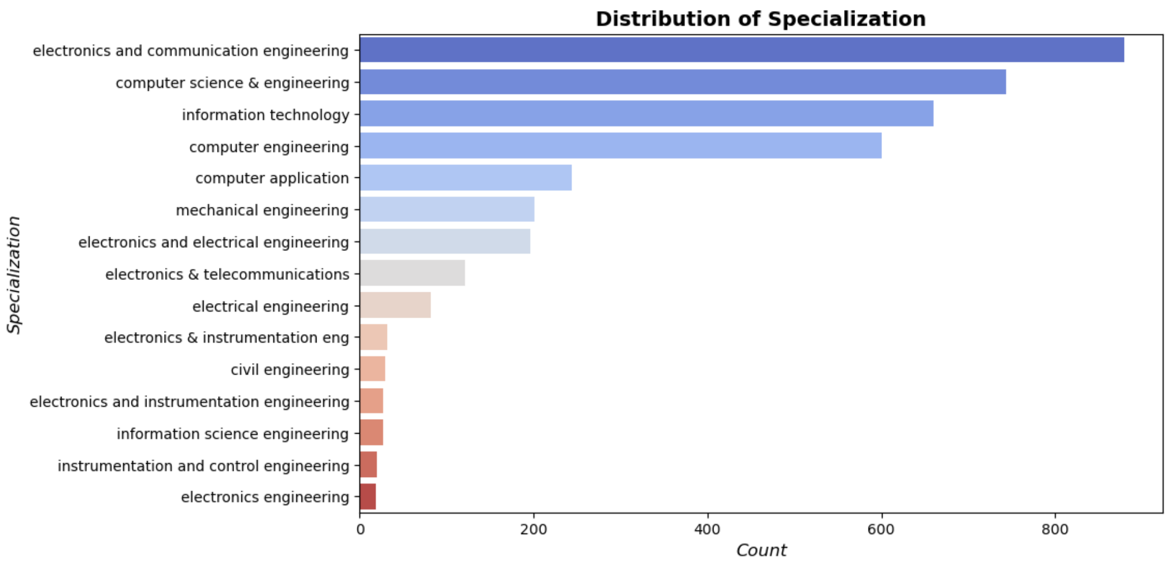
	count
CollegeTier	
2	3701
1	297

Which Specialization is most common among the Students?

In [37]:
```
df["Specialization"].value_counts().head(15)
```

Out[37]:
```
Specialization
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
Name: count, dtype: int64
```

In [38]:
```
#Countplot for Specialization
top_15_specializations = df["Specialization"].value_counts().head(15)
plt.figure(figsize=(10,6))

sns.barplot(x=top_15_specializations.values,
            y=top_15_specializations.index,
            palette="coolwarm")

plt.title('Distribution of Specialization', fontsize=14,fontweight='bold'
plt.ylabel('Specialization', fontsize=12,fontstyle='italic')
plt.xlabel('Count', fontsize=12,fontstyle='italic')
plt.show()
```

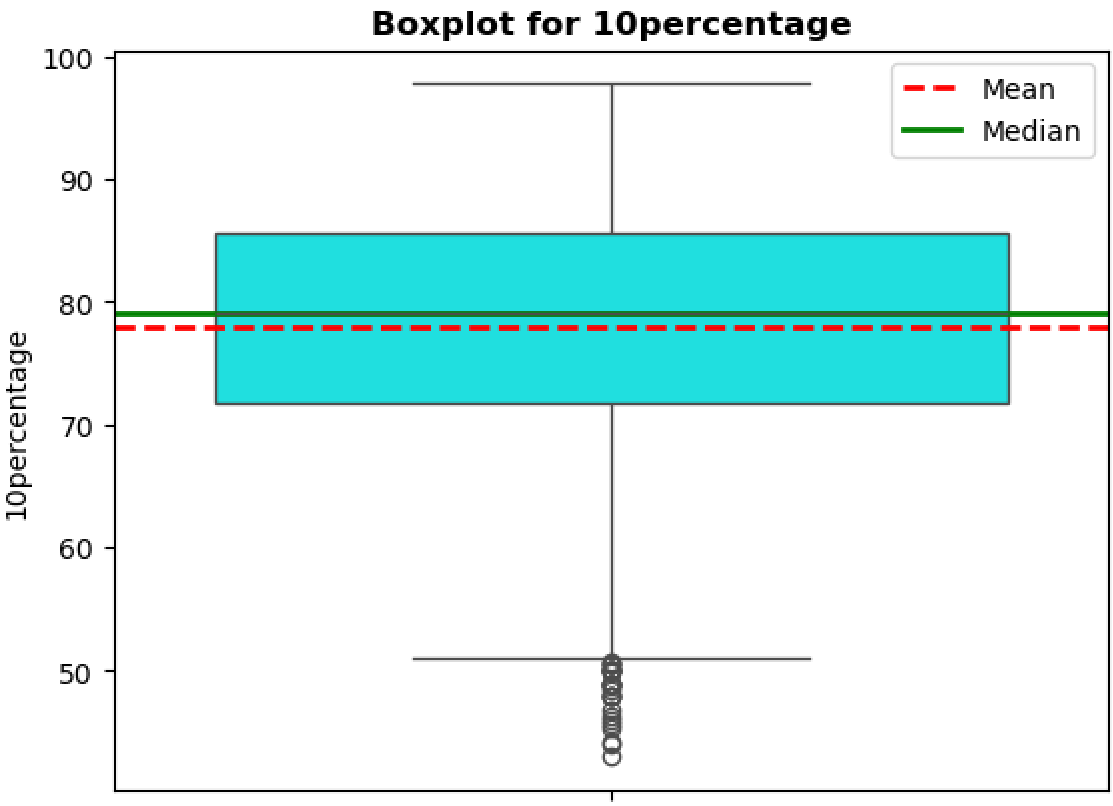


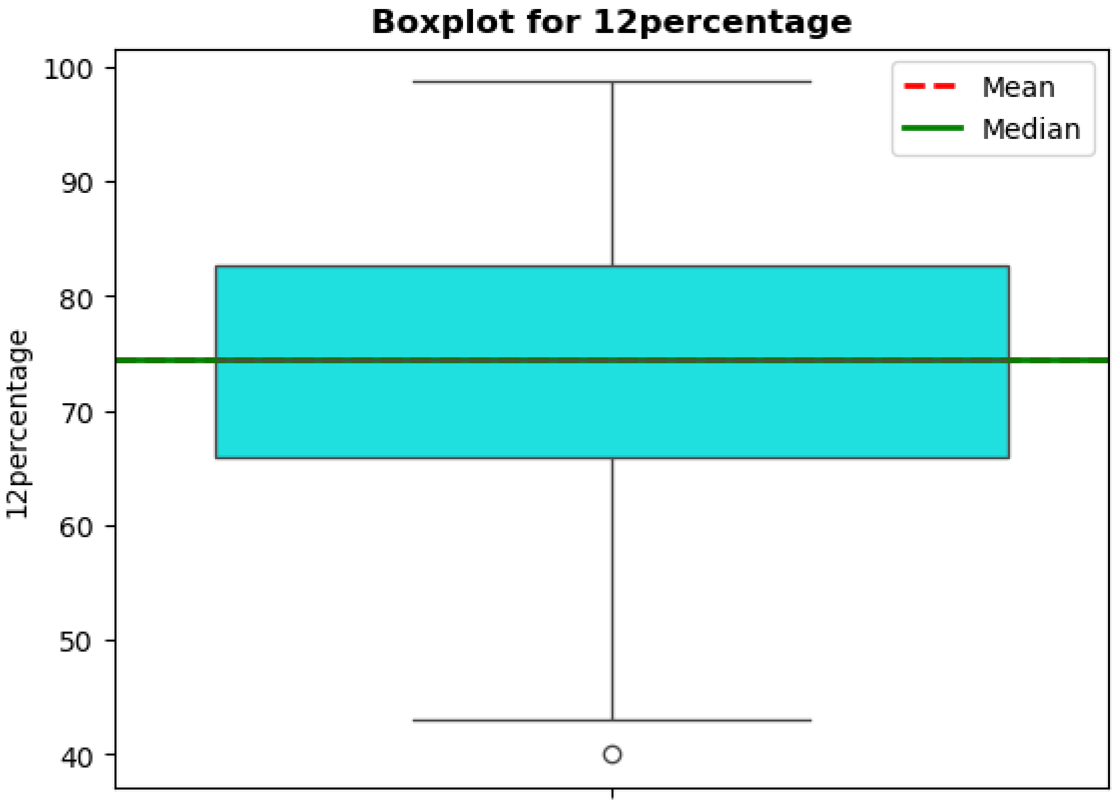
1.From the above plot,there are more electronics and communication engineers,followed by computer science&engineersless electronics

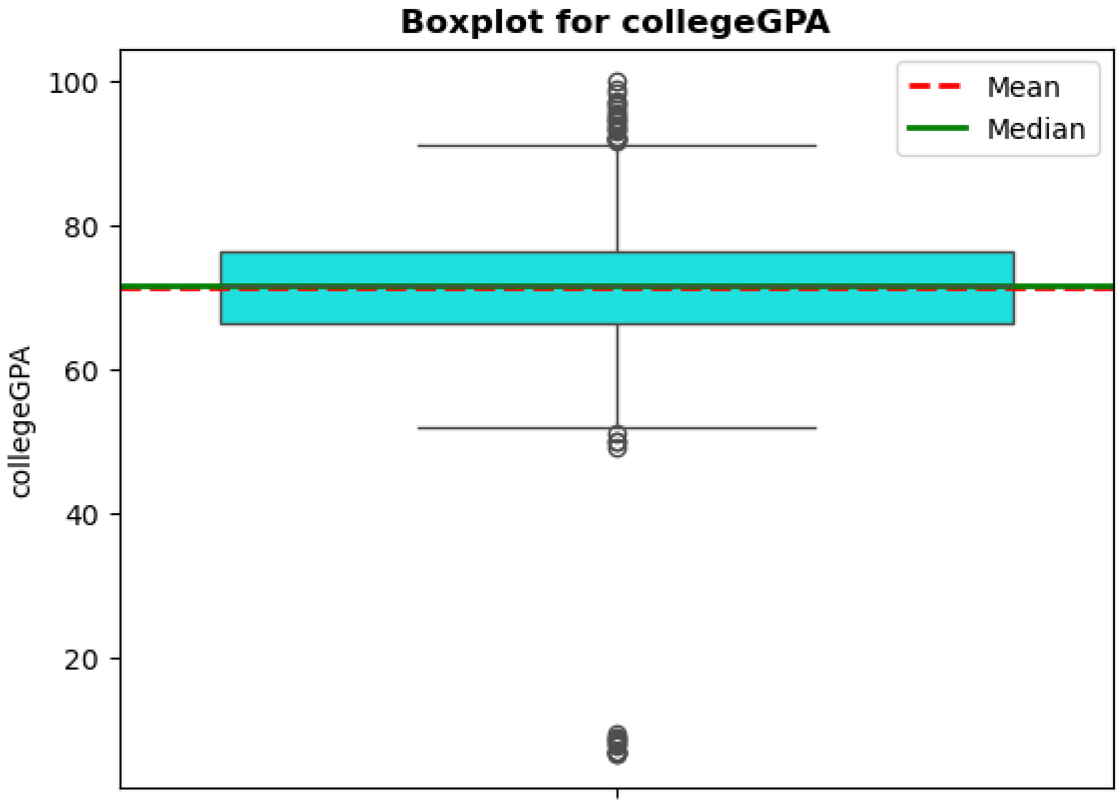
2.There are less electronics engineering engineers

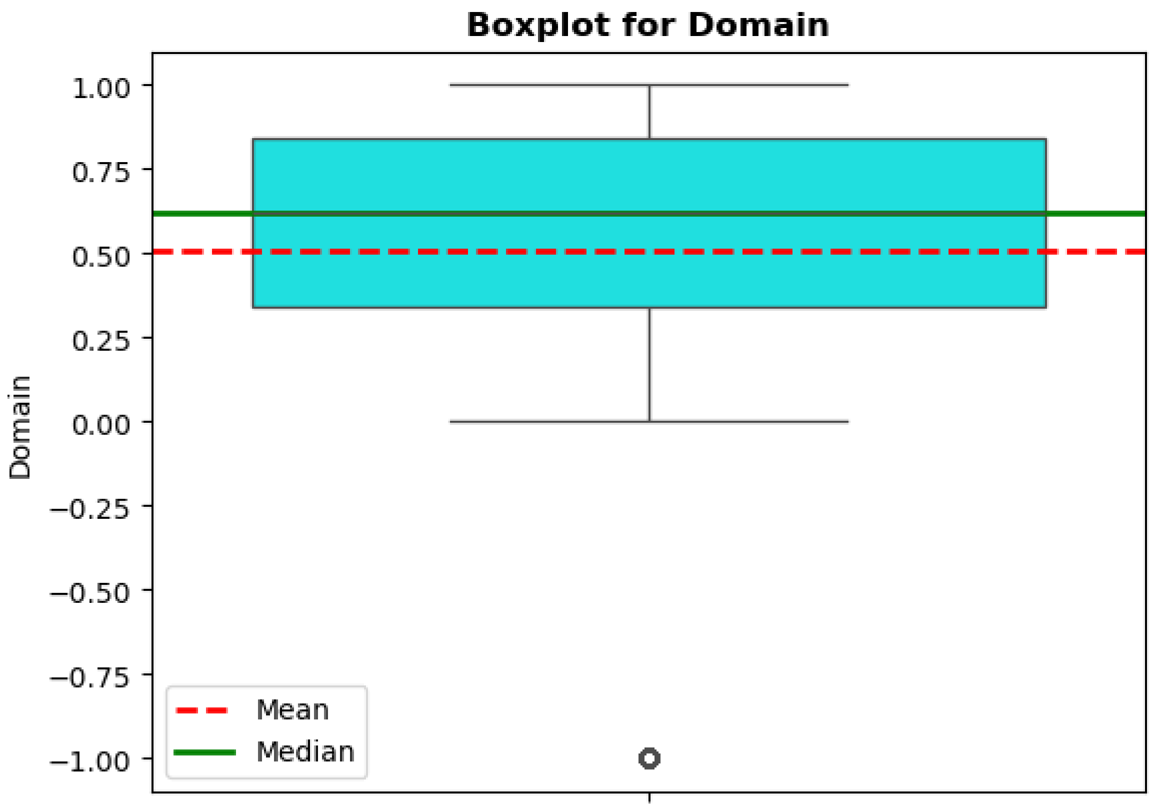
In [40]:
```
for i in df.columns:
    if df[i].dtype == "int" or df[i].dtype == "float":
        sns.boxplot(y=df[i], color='cyan')
        mean_value = df[i].mean()
        median_value = df[i].median()
```

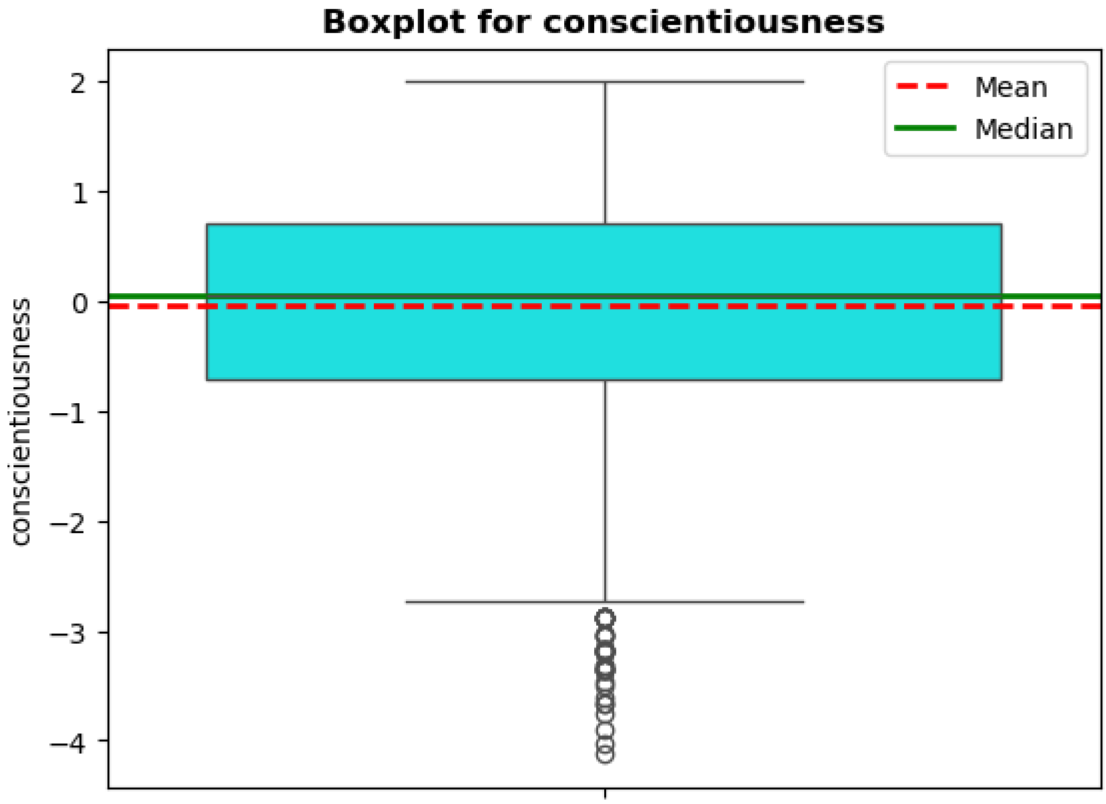
```
    plt.axhline(mean_value, color='red', linestyle='--', label='Mean'
    plt.axhline(median_value, color='green', linestyle='-', label='Me
    plt.title(f"Boxplot for {i}", fontweight='bold')
    plt.legend()
    plt.show()
```

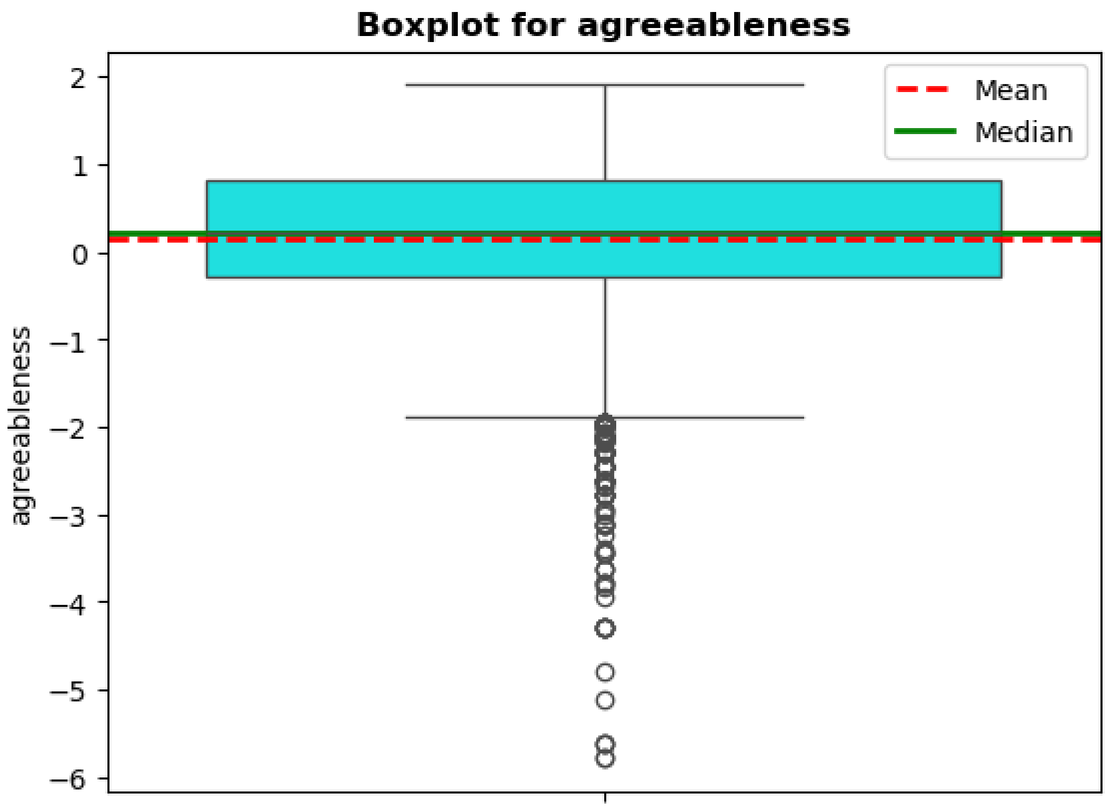


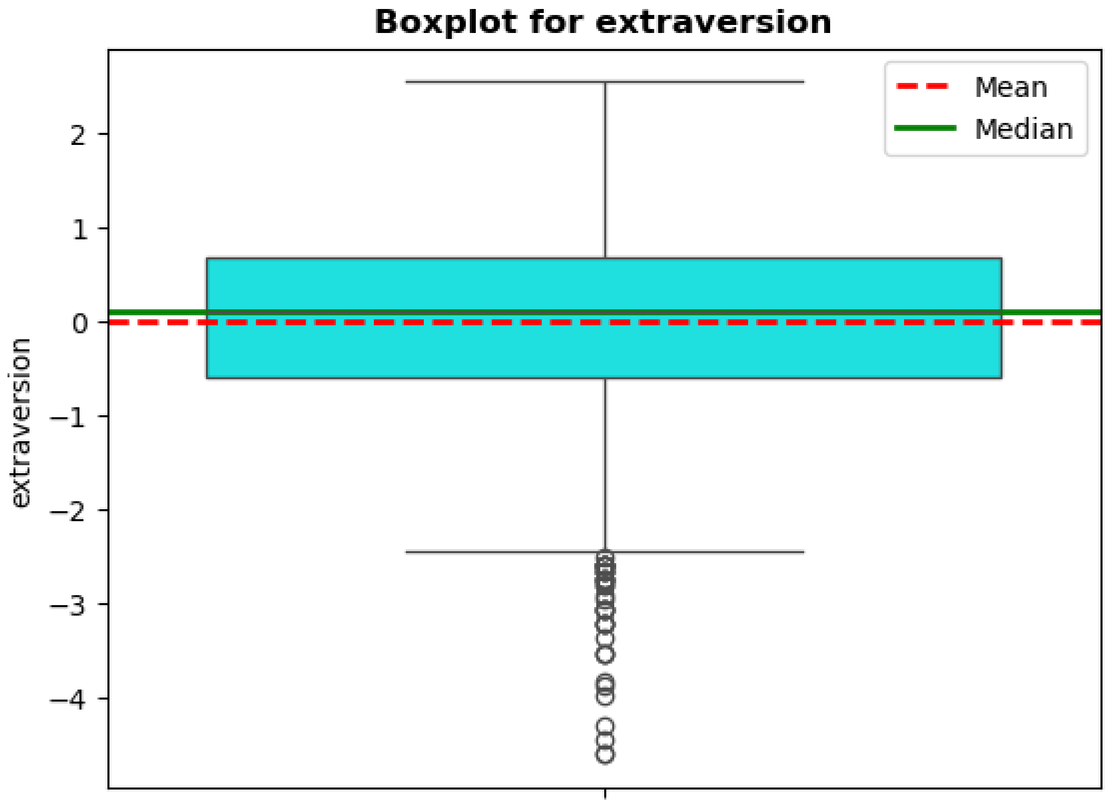


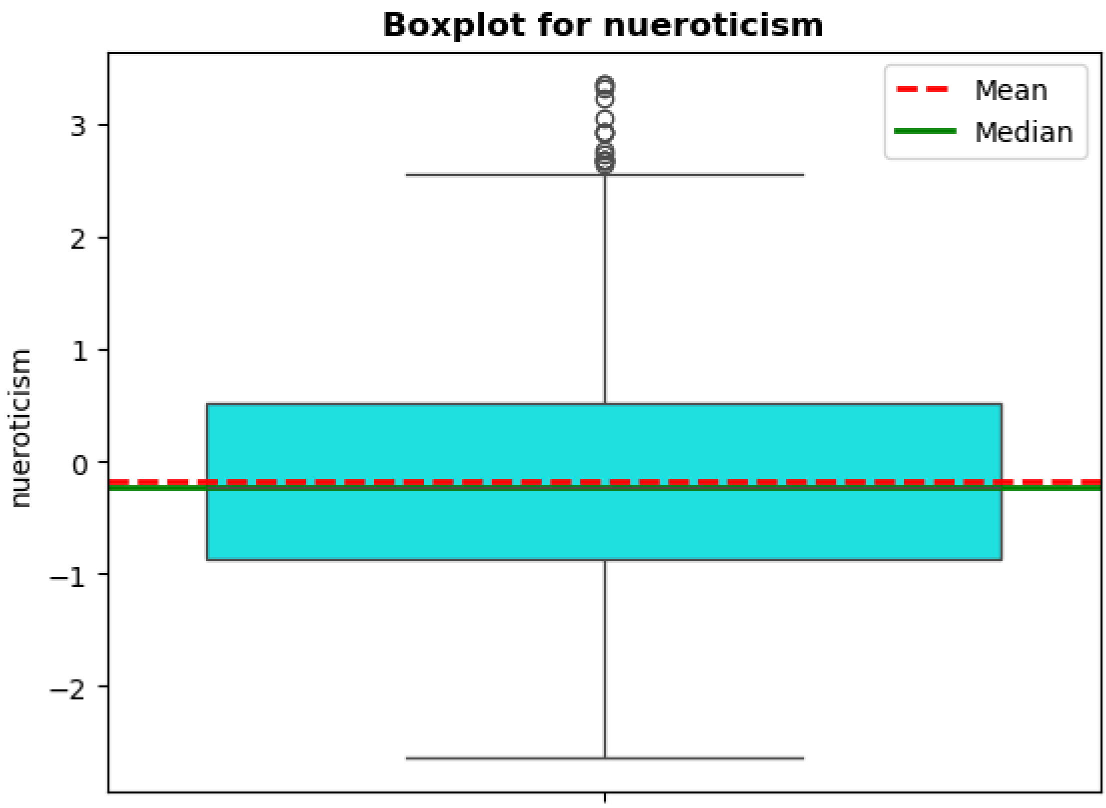

Boxplot for collegeGPA
collegeGPA
100
80
60
40
20
Mean
Median

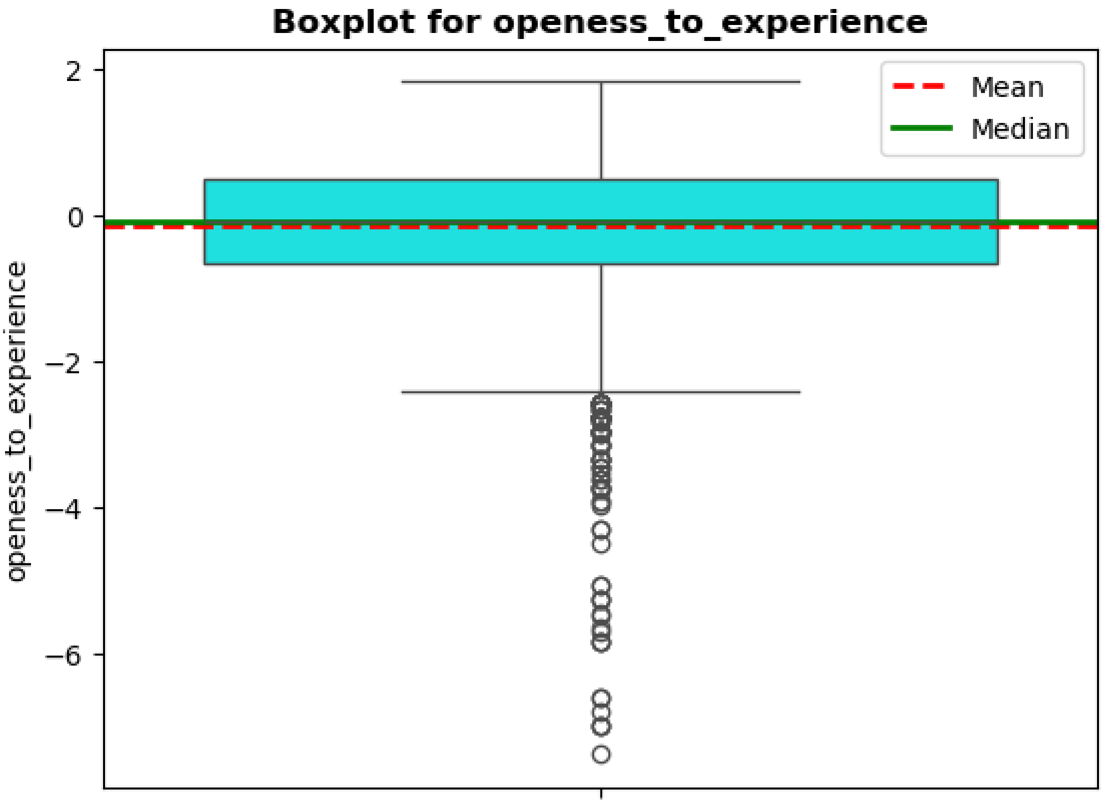

Boxplot for Domain
Domain
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
Mean
Median


Boxplot for conscientiousness
Mean
Median
conscientiousness
2
1
0
−1
−2
−3
−4


Boxplot for agreeableness
Mean
Median
agreeableness
2
1
0
−1
−2
−3
−4
−5
−6

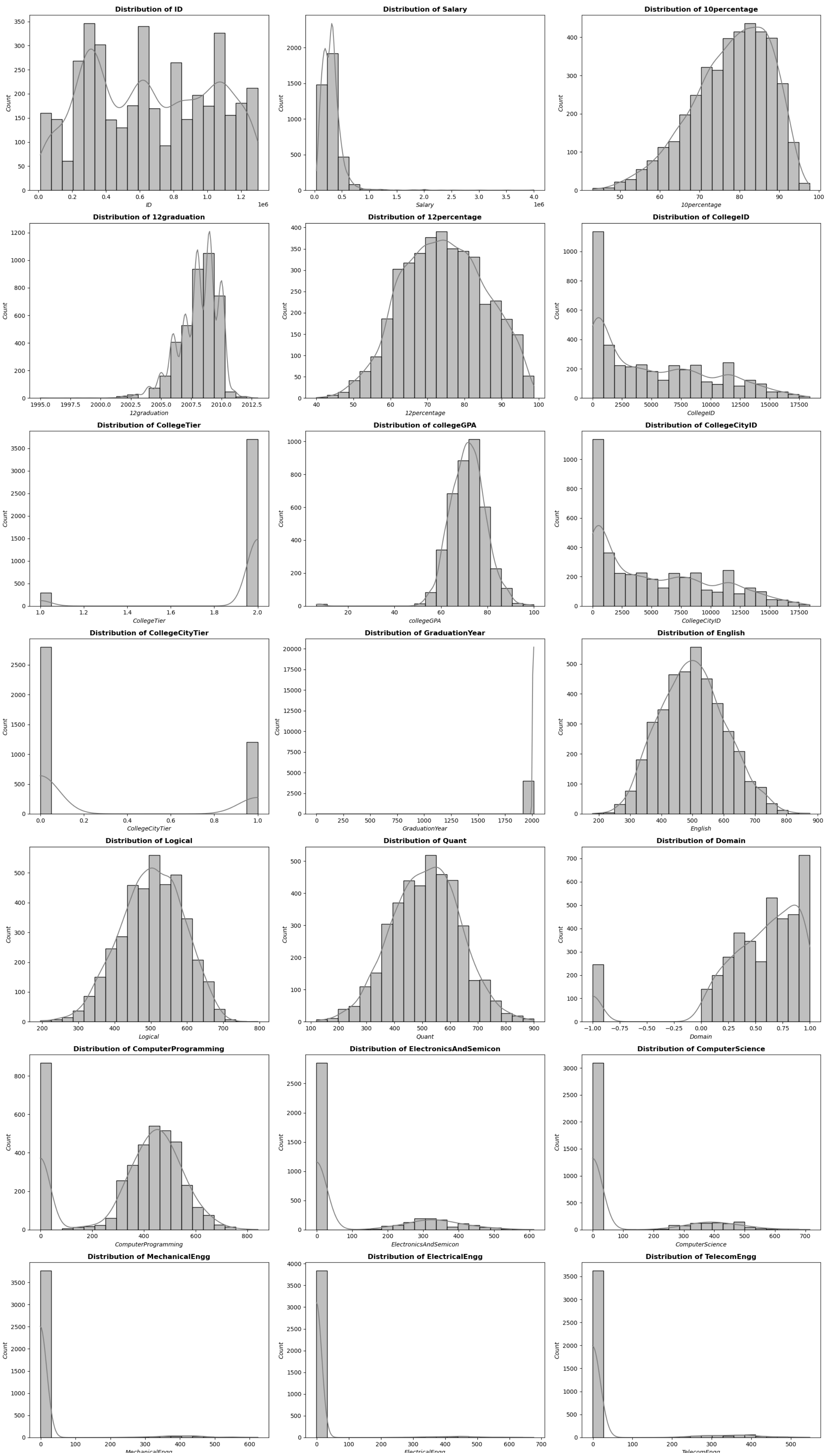

Boxplot for extraversion
Mean
Median
extraversion
2
1
0
−1
−2
−3
−4


Boxplot for nueroticism
Mean
Median
nueroticism
3
2
1
0
−1
−2

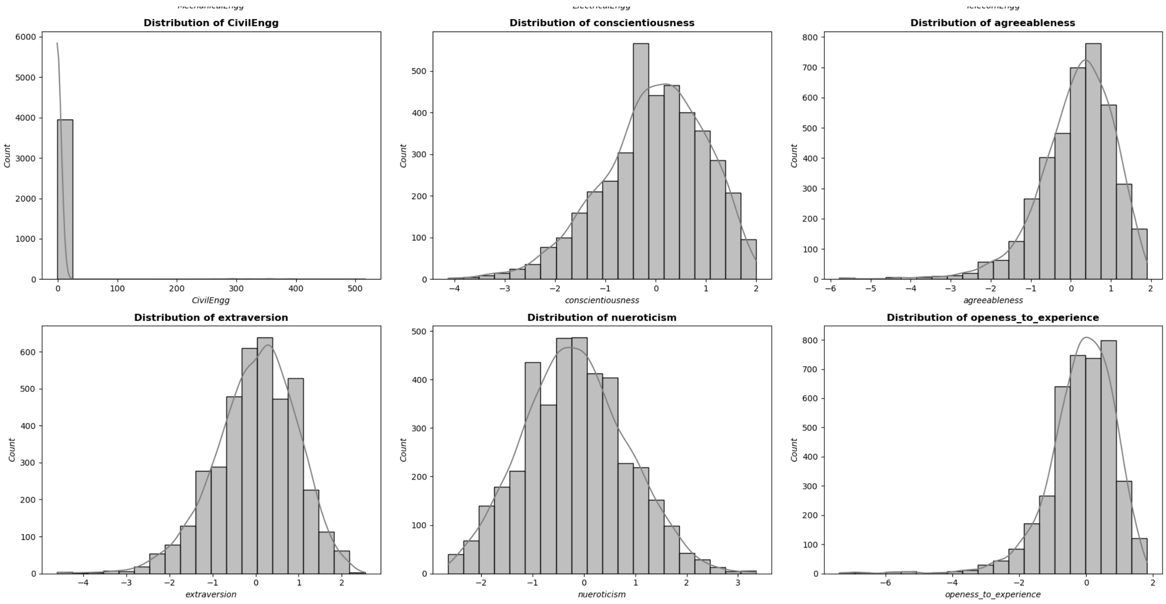


In [41]:

```
import math
numerical_columns = df.select_dtypes(include=['int64', 'float64']).column

num_cols = len(numerical_columns)
rows = math.ceil(num_cols / 3)

plt.figure(figsize=(20, rows * 5))

for i, col in enumerate(numerical_columns, 1):
    plt.subplot(rows, 3, i)
    sns.histplot(df[col].dropna(), bins=20, edgecolor='black', color='gra

    plt.title(f'Distribution of {col}', fontweight='bold')
    plt.xlabel(col, fontstyle='italic')
    plt.ylabel('Count', fontstyle='italic')

plt.tight_layout()
plt.show()
```



Distribution of ID
ID
Count
1e6
Distribution of Salary
Salary
Count
1e6
Distribution of 10percentage
10percentage
Count
Distribution of 12graduation
12graduation
Count
Distribution of 12percentage
12percentage
Count
Distribution of CollegeID
CollegeID
Count
Distribution of CollegeTier
CollegeTier
Count
Distribution of collegeGPA
collegeGPA
Count
Distribution of CollegeCityID
CollegeCityID
Count
Distribution of CollegeCityTier
CollegeCityTier
Count
Distribution of GraduationYear
GraduationYear
Count
Distribution of English
English
Count
Distribution of Logical
Logical
Count
Distribution of Quant
Quant
Count
Distribution of Domain
Domain
Count
Distribution of ComputerProgramming
ComputerProgramming
Count
Distribution of ElectronicsAndSemicon
ElectronicsAndSemicon
Count
Distribution of ComputerScience
ComputerScience
Count
Distribution of MechanicalEngg
MechanicalEngg
Count
Distribution of ElectricalEngg
ElectricalEngg
Count
Distribution of TelecomEngg
TelecomEngg
Count



Univariate Analysis on Categorical Data(Non-Visualization)

In [43]:
```
def cat_univariate_analysis(data):
    for column in data:
        print("*" * 5, column, "*" * 5)
        print("Mode of data is:",data[column].mode())
        print("Unique values of column are:",data[column].value_counts())
cat_univariate_analysis(df[['Designation','Specialization']])
```

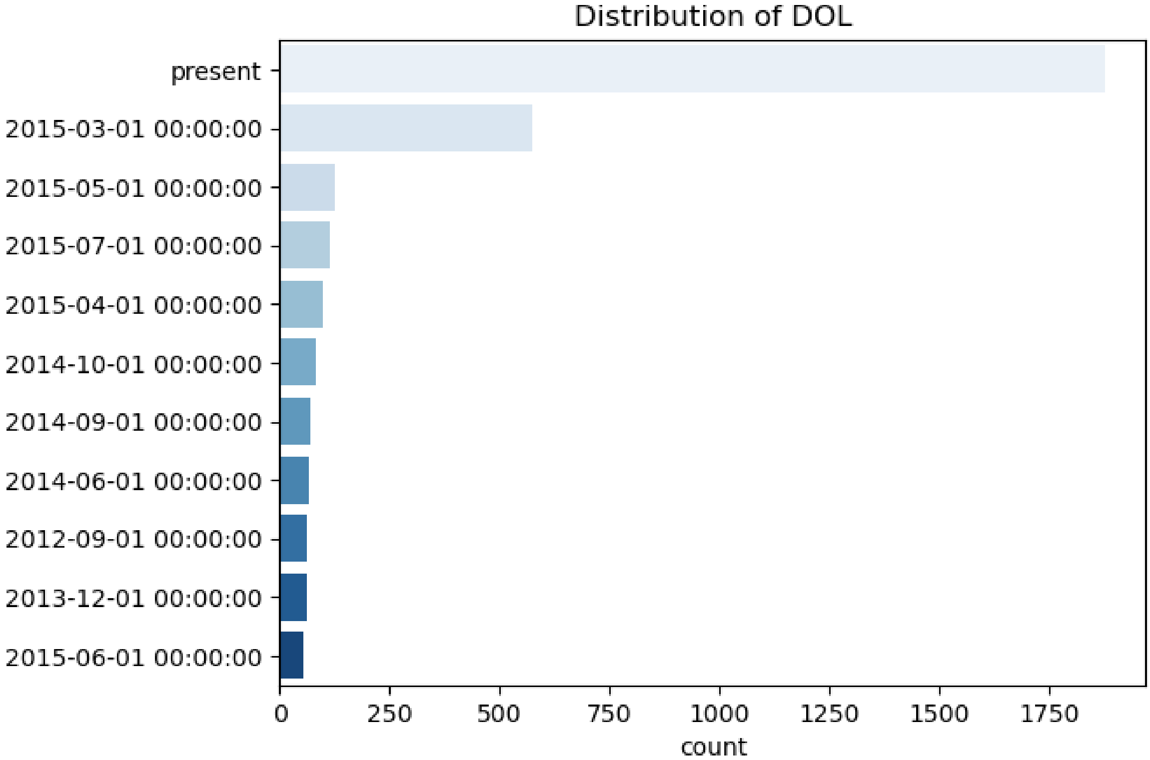
```
***** Designation *****
Mode of data is: 0    software engineer
Name: Designation, dtype: object
Unique values of column are: Designation
software engineer                  539
software developer                 265
system engineer                    205
programmer analyst                 139
systems engineer                   118
                                  ...
cad drafter                          1
noc engineer                         1
human resources intern               1
senior quality assurance engineer    1
jr. software developer               1
Name: count, Length: 419, dtype: int64
***** Specialization *****
Mode of data is: 0    electronics and communication engineering
Name: Specialization, dtype: object
Unique values of column are: Specialization
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
applied electronics and instrumentation         9
chemical engineering                            9
computer science and technology                 6
telecommunication engineering                   6
mechanical and automation                       5
automobile/automotive engineering               5
instrumentation engineering                     4
mechatronics                                    4
aeronautical engineering                        3
electronics and computer engineering            3
electrical and power engineering                2
biomedical engineering                          2
information & communication technology          2
industrial engineering                          2
computer science                                2
metallurgical engineering                       2
power systems and automation                    1
control and instrumentation engineering         1
mechanical & production engineering             1
embedded systems technology                     1
polymer technology                              1
computer and communication engineering          1
```

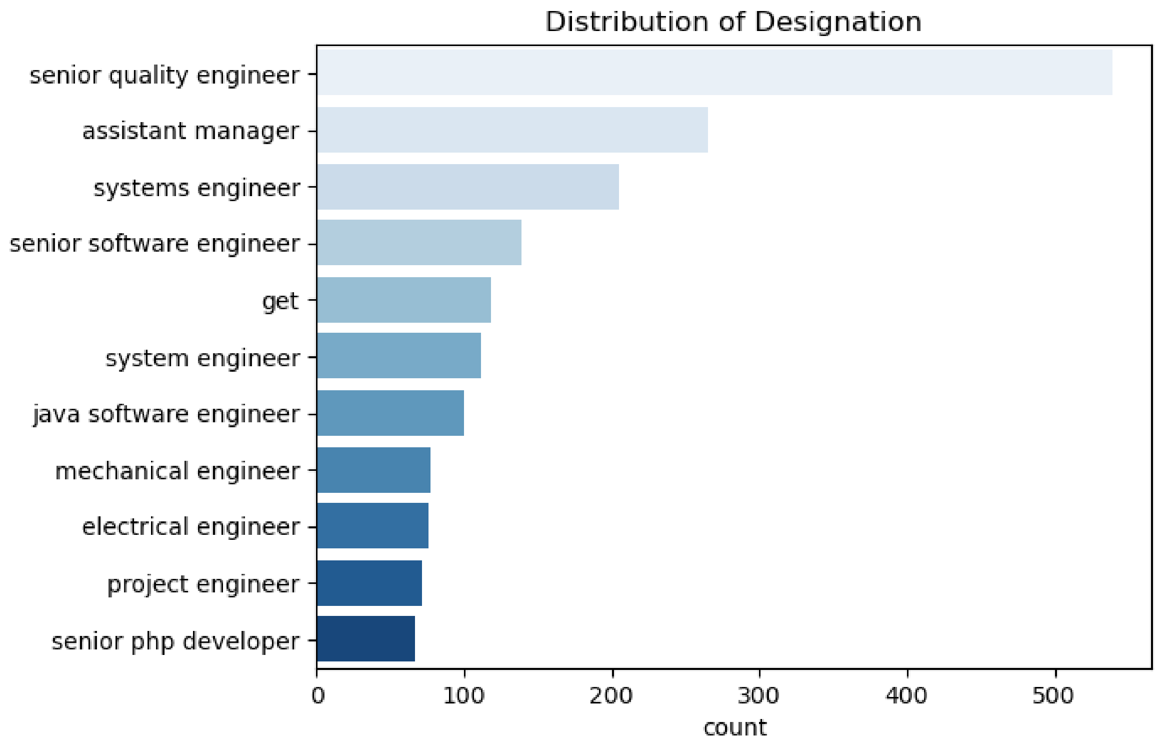
```
information science                              1
internal combustion engine                       1
computer networking                              1
ceramic engineering                              1
electronics                                      1
industrial & management engineering              1
Name: count, dtype: int64
```

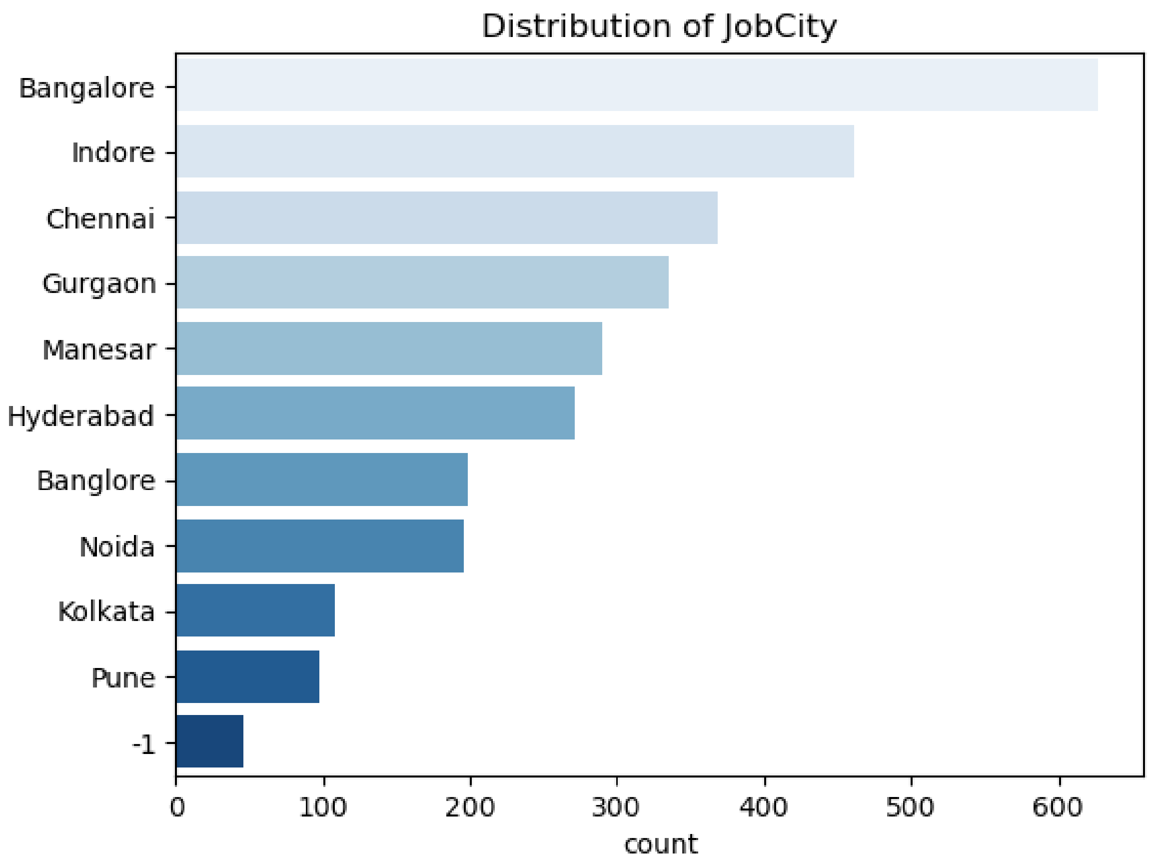
Univariate Analysis on Categorical Data(Visualization)

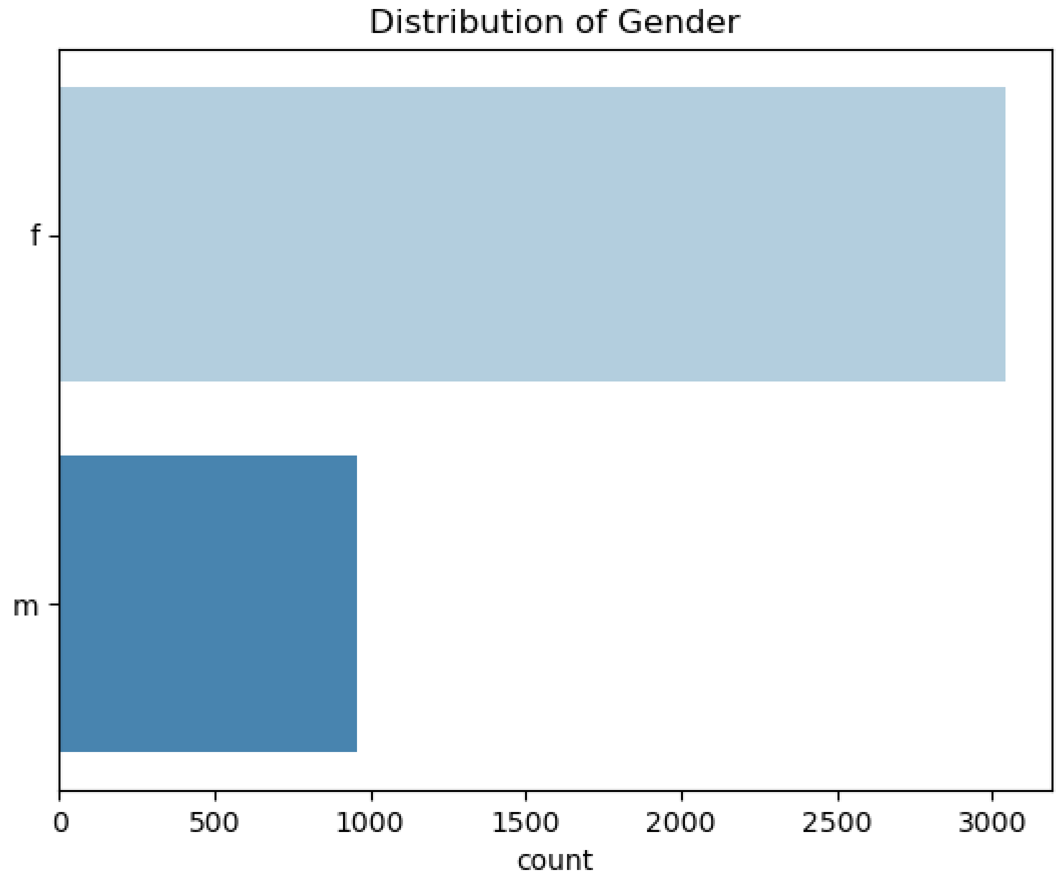
In [45]:

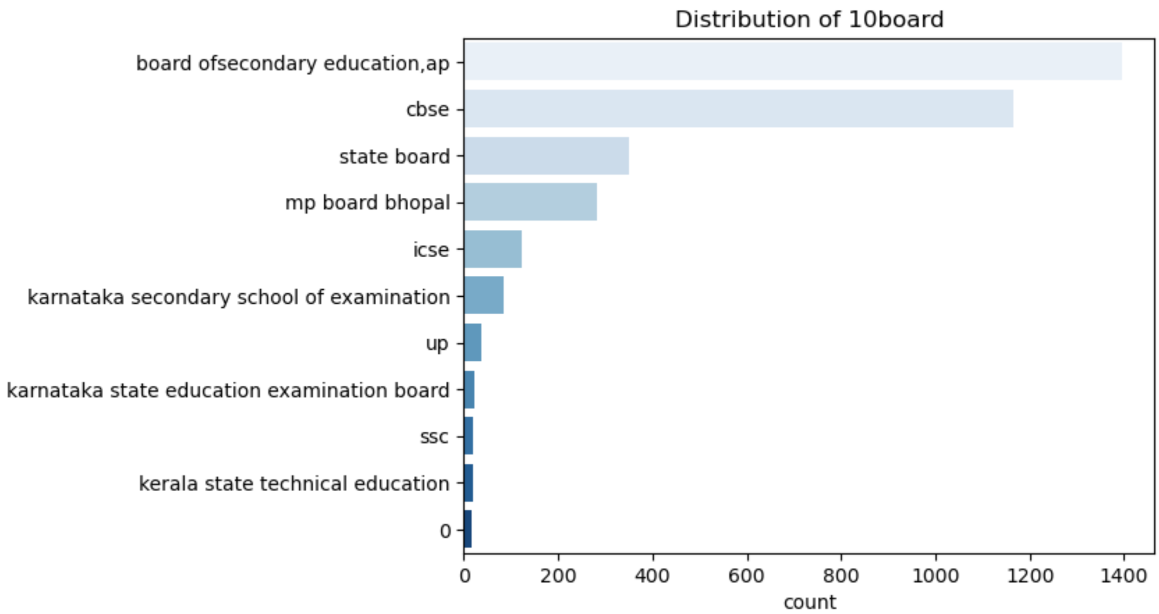
```
for i in df.columns:
    if df[i].dtype == "object":
        sns.barplot(x=df[i].value_counts()[:11],
            y=df[i].unique()[:11], palette="Blues")
        plt.title("Distribution of {}".format(i))
        plt.show()
```

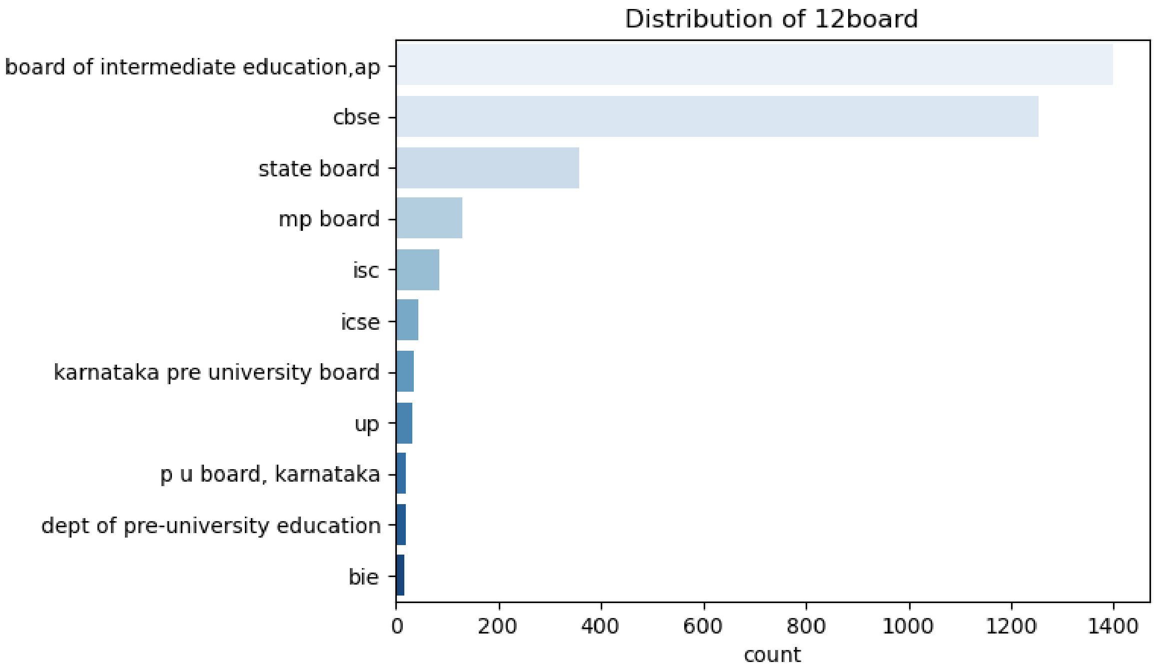


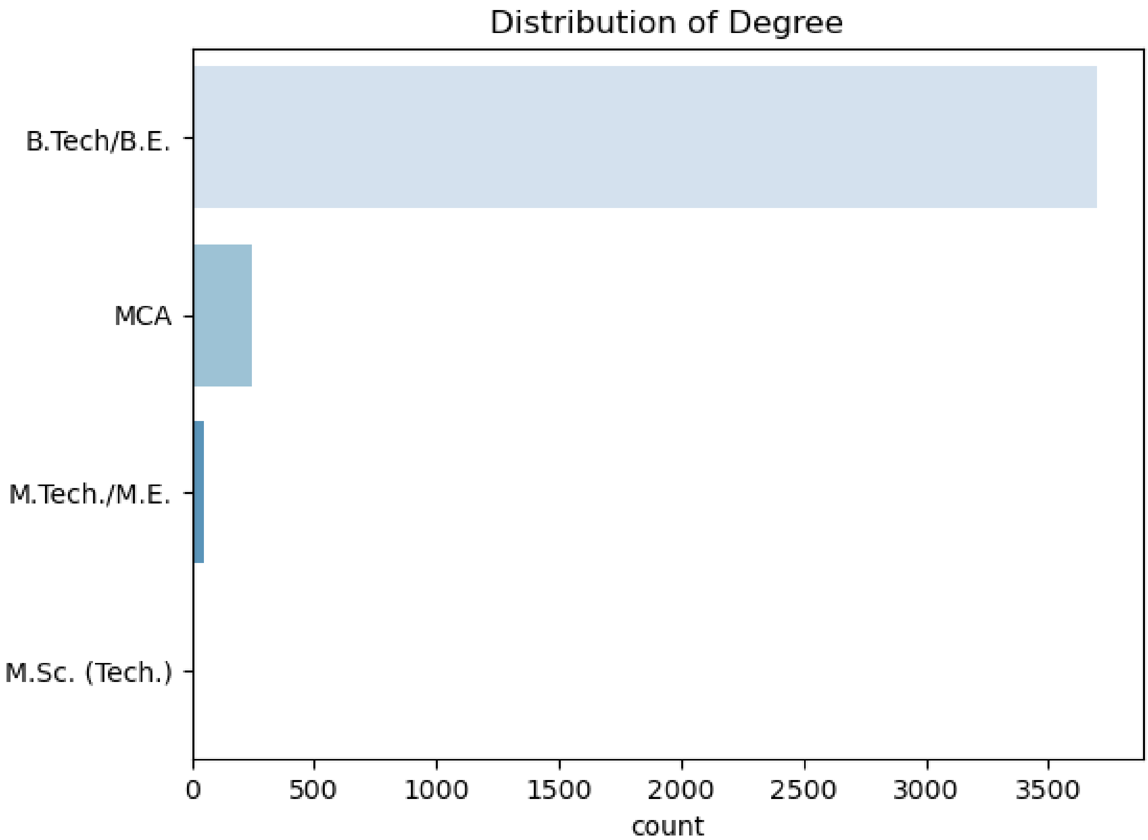

Distribution of Designation
senior quality engineer
assistant manager
systems engineer
senior software engineer
get
system engineer
java software engineer
mechanical engineer
electrical engineer
project engineer
senior php developer
0
100
200
300
400
500
count

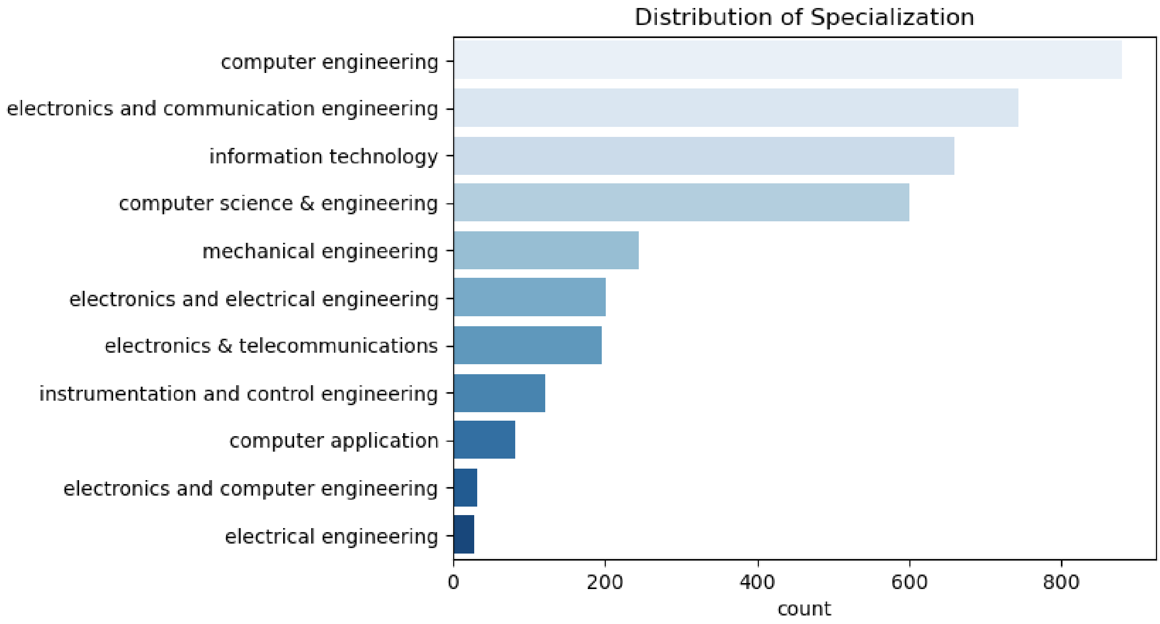

Distribution of JobCity
Bangalore
Indore
Chennai
Gurgaon
Manesar
Hyderabad
Banglore
Noida
Kolkata
Pune
-1
0
100
200
300
400
500
600
count

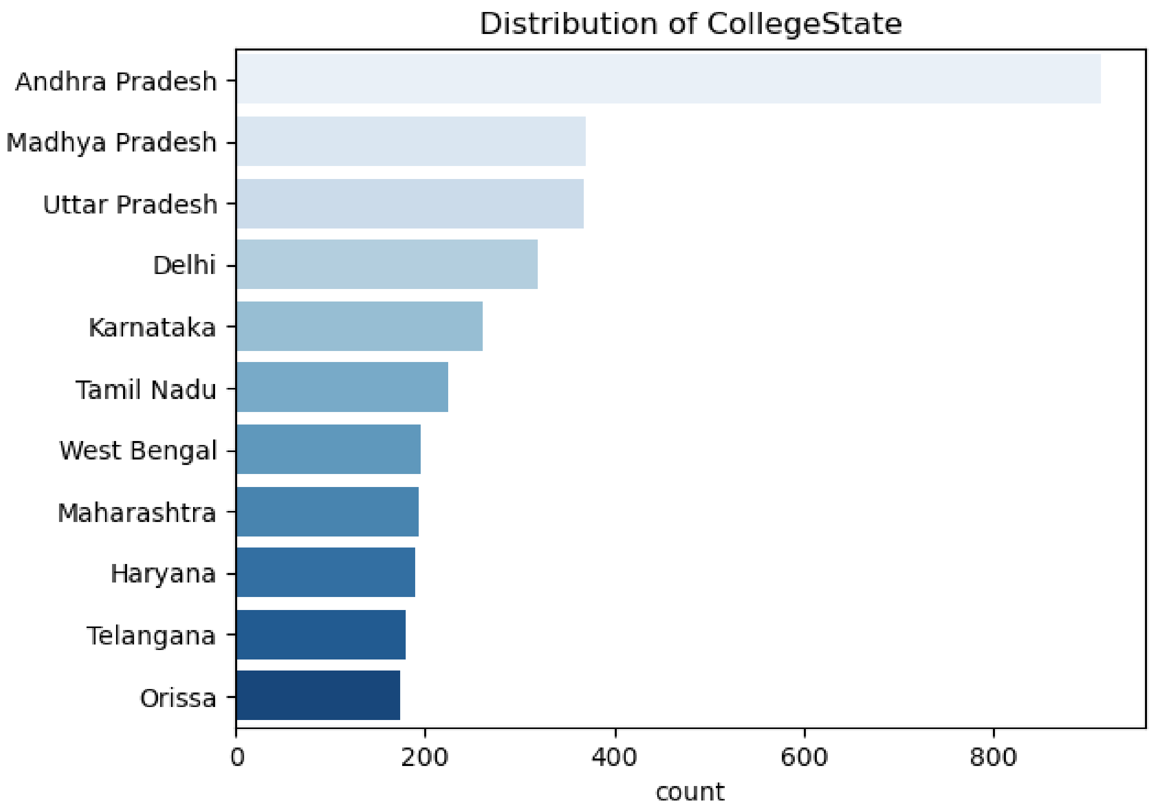

Distribution of Gender
f
m
0
500
1000
1500
2000
2500
3000
count


Distribution of 10board
board ofsecondary education,ap
cbse
state board
mp board bhopal
icse
karnataka secondary school of examination
up
karnataka state education examination board
ssc
kerala state technical education
0
0
200
400
600
800
1000
1200
1400
count


Distribution of 12board
board of intermediate education,ap
cbse
state board
mp board
isc
icse
karnataka pre university board
up
p u board, karnataka
dept of pre-university education
bie
0
200
400
600
800
1000
1200
1400
count


Distribution of Degree
B.Tech/B.E.
MCA
M.Tech./M.E.
M.Sc. (Tech.)
0
500
1000
1500
2000
2500
3000
3500
count





Bivariate Analysis

Analysing the data using 2 features/Relationship between 2 variables

Categorical vs Categorical(Non-Visualization)

Cross tabulation (crosstab) is a useful analysis tool commonly used to compare the results for one or more variables with the results of another variable

In [48]:
```
pd.crosstab(df["Specialization"],df["Gender"],margins=True)
```

Out[48]:

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

Gender	f	m	All
Specialization			
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

This cross tab shows the number of female,male present for a paricular specialization

Numerical vs Categorical(Non-Visualization)

Group-by aggregation is a data manipulation technique that consists of two steps. First, we group the data based on the values of specific columns. Second, we perform some aggregation operations (e.g., sum, average, median, count unique) on top of the grouped data

In [51]:
```
df.groupby(["Specialization"])["Salary"].sum().sort_values(ascending=Fals
```

```
Out[51]: Specialization
electronics and communication engineering      261195000
computer engineering                           224460000
computer science & engineering                 206415000
information technology                         203605000
computer application                            68415000
mechanical engineering                          63809000
electronics and electrical engineering          56235000
electronics & telecommunications                35520000
electrical engineering                          24090000
electronics & instrumentation eng               11665000
civil engineering                               11055000
electronics and instrumentation engineering      8840000
instrumentation and control engineering          7880000
information science engineering                  7460000
electronics engineering                          5310000
industrial & production engineering              3845000
biotechnology                                    3815000
other                                            3465000
chemical engineering                             3330000
applied electronics and instrumentation          3135000
telecommunication engineering                    2055000
mechanical and automation                        1545000
computer science and technology                  1475000
automobile/automotive engineering                1110000
mechatronics                                     1015000
instrumentation engineering                       960000
information & communication technology            775000
industrial engineering                            740000
polymer technology                                700000
metallurgical engineering                         675000
electronics and computer engineering              660000
biomedical engineering                            580000
computer science                                  580000
computer networking                               565000
information science                               460000
aeronautical engineering                          445000
electrical and power engineering                  420000
internal combustion engine                        360000
ceramic engineering                               335000
industrial & management engineering               320000
control and instrumentation engineering           305000
embedded systems technology                       200000
computer and communication engineering            120000
mechanical & production engineering               100000
power systems and automation                      100000
electronics                                        40000
Name: Salary, dtype: int64
```

The above groupby tells the sum of salaries by their specialization. electronics and communication engineering has the highest sum of salaries whereas electronics(specialization)has the lowest sum of salaries

Numerical vs Numerical(Non-Visualization)

Correlation Coefficient **(Pearson's r)**

The value ranges between -1 and 1

1.If it is 1 then perfect positive linear relationship

2.If it is -1 then perfect negative linear relationship

3.If it is 0 then no linear relationship

In [54]:
```
correlation=df["Salary"].corr(df["12percentage"])
print(f'Pearson Coorelation Coefficient:{correlation}')
```

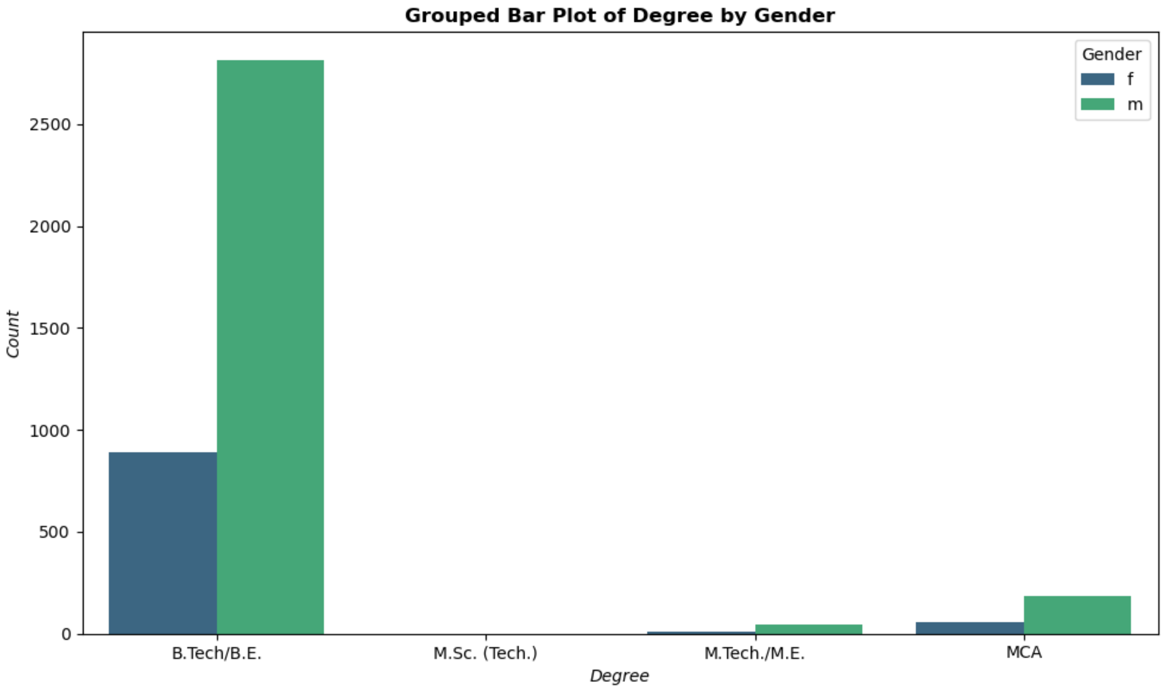
```
Pearson Coorelation Coefficient:0.17025447790246095
```

The correlation tells that it is positively correlated but not perfect positive linear relationship.

Categorical vs Categorical(Visualization)

Grouped Bar plot is a type of bar chart where bars representing different categories are grouped together based on a common target variable.

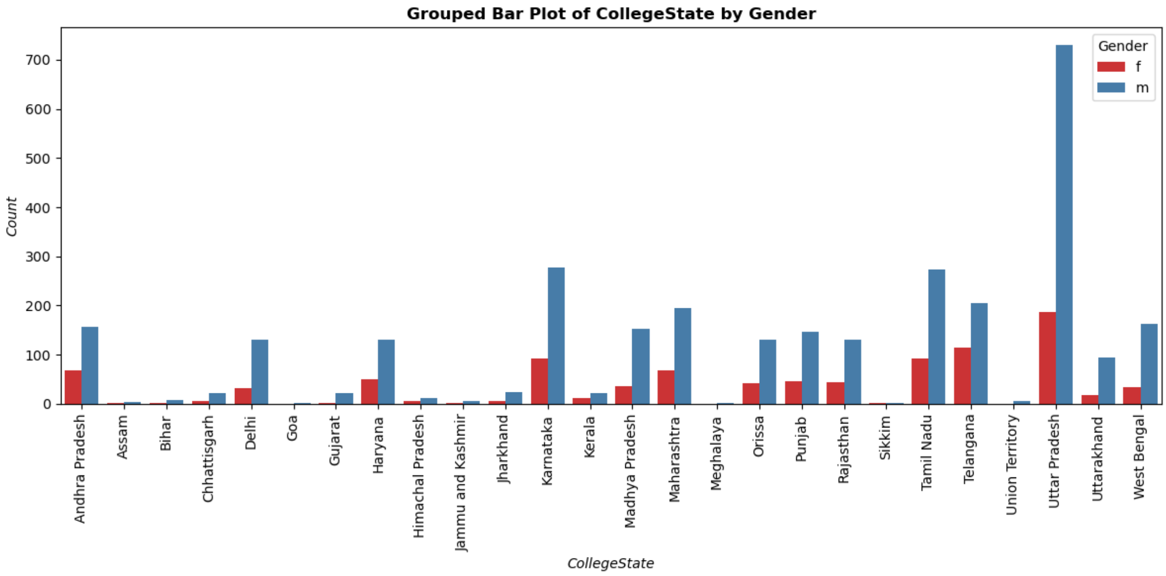
In [57]:
```
grouped_1 = df.groupby(['Degree', 'Gender']).size().reset_index(name='Cou

plt.figure(figsize=(10, 6))
sns.barplot(x='Degree', y='Count', hue='Gender', data=grouped_1, palette=

plt.title('Grouped Bar Plot of Degree by Gender', fontweight='bold')
plt.xlabel('Degree', fontstyle='italic')
plt.ylabel('Count', fontstyle='italic')
plt.tight_layout()
plt.show()
```



B.Tech/B.E. is choosen by many people(male and female)compared with other degrees

In [59]:
```
grouped_2 = df.groupby(['CollegeState', 'Gender']).size().reset_index(nam

plt.figure(figsize=(12, 6))
```

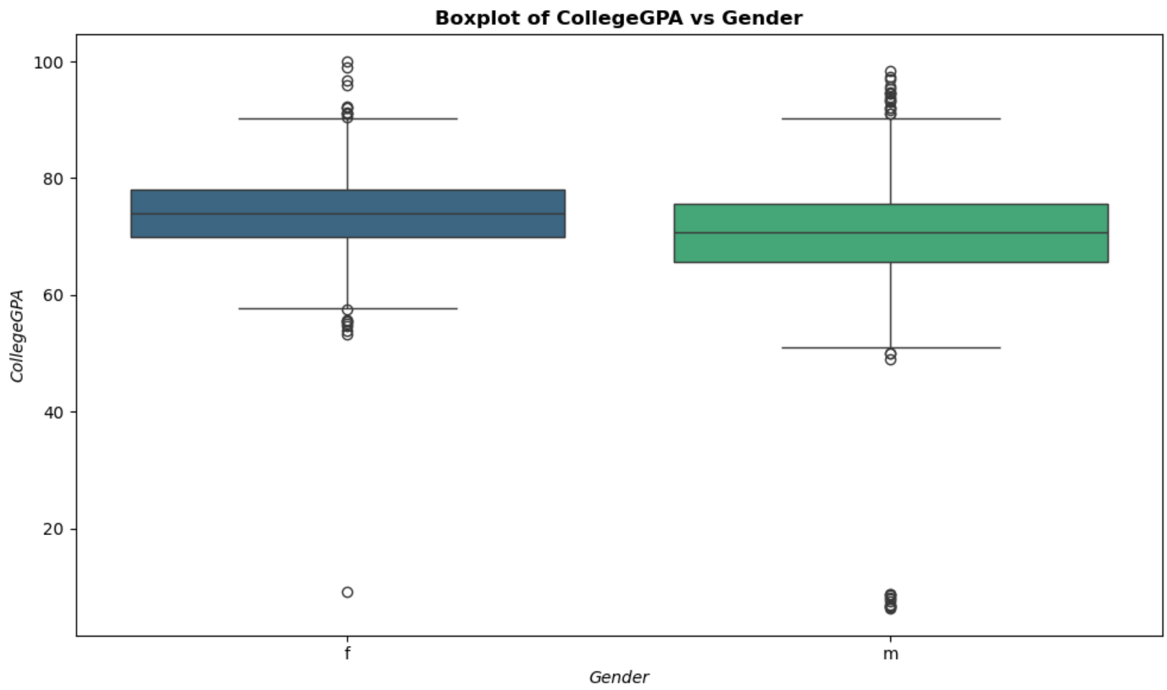
```
sns.barplot(x='CollegeState', y='Count', hue='Gender', data=grouped_2, pa
plt.title('Grouped Bar Plot of CollegeState by Gender', fontweight='bold'
plt.xlabel('CollegeState', fontstyle='italic')
plt.ylabel('Count', fontstyle='italic')
plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



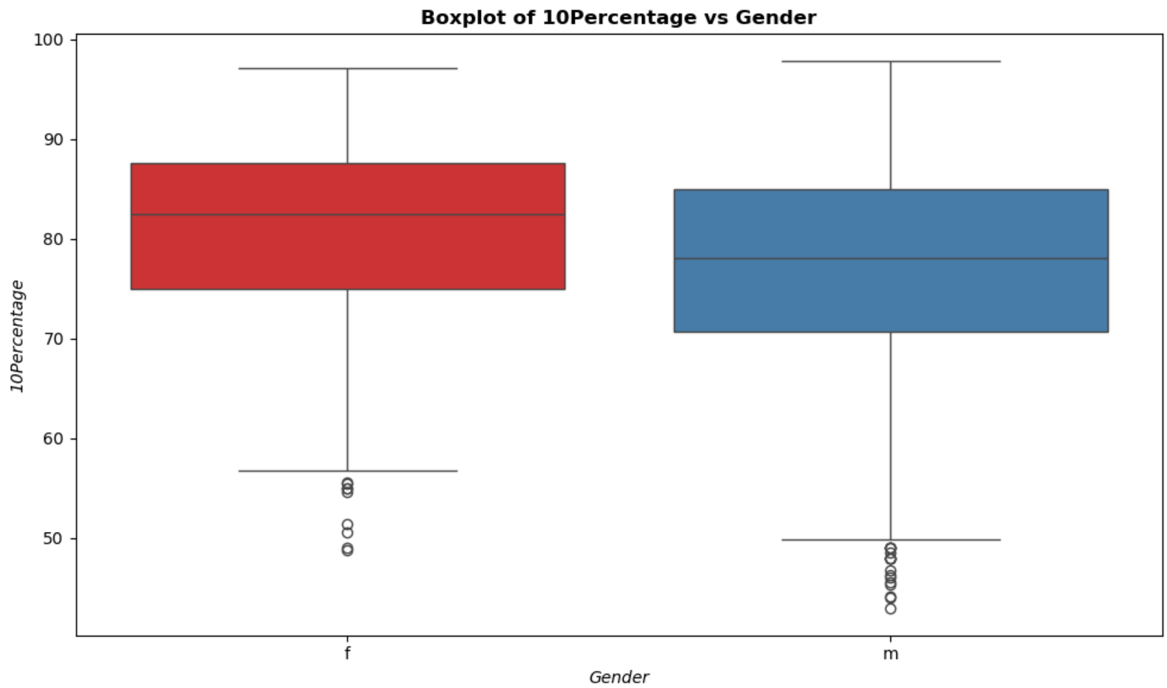
Uttar Pradesh has highest working professionals both male and female And Meghalaya almost doesn't has any female working professionals.

Numerical vs Categorical(Visualization)

In [62]:
```
plt.figure(figsize=(10, 6))
sns.boxplot(x='Gender', y='collegeGPA', data=df, palette='viridis')
plt.title('Boxplot of CollegeGPA vs Gender', fontweight='bold')
plt.xlabel('Gender', fontstyle='italic')
plt.ylabel('CollegeGPA', fontstyle='italic')
plt.tight_layout()
plt.show()
```



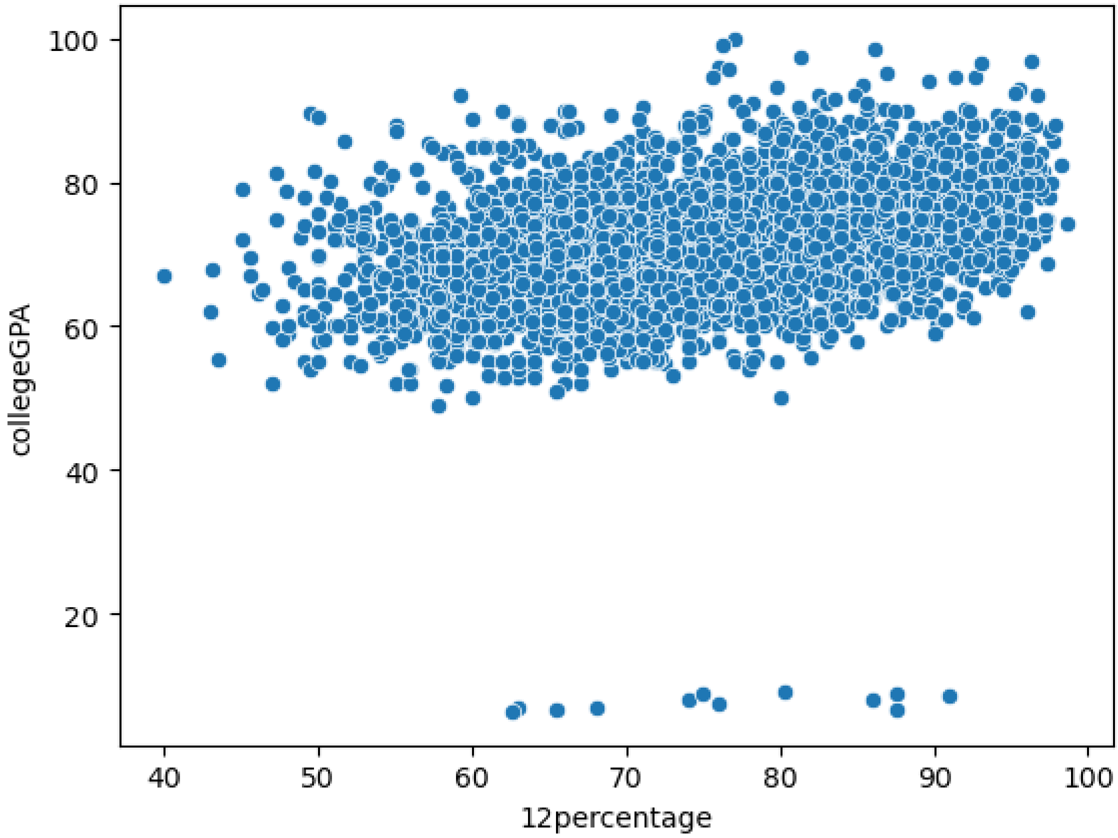
In [63]:
```
plt.figure(figsize=(10, 6))
sns.boxplot(x='Gender', y='10percentage', data=df, palette='Set1')
plt.title('Boxplot of 10Percentage vs Gender', fontweight='bold')
plt.xlabel('Gender', fontstyle='italic')
plt.ylabel('10Percentage', fontstyle='italic')
plt.tight_layout()
plt.show()
```



Numerical vs Numerical(Visualization)

In [65]:
```
sns.scatterplot(data=df,x="12percentage",y="collegeGPA")
```

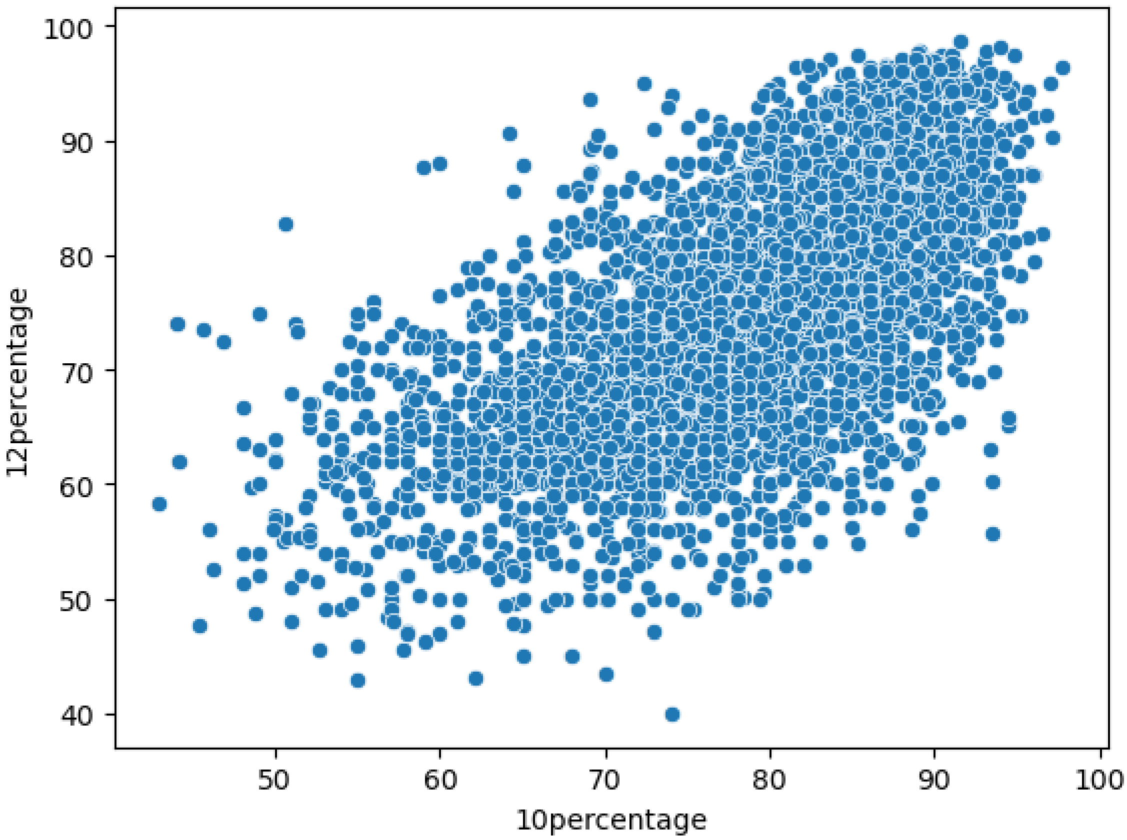
Out[65]: <Axes: xlabel='12percentage', ylabel='collegeGPA'>



1. Positive Correlation-Higher 12th-grade percentages are generally associated with higher college GPAs.
2. Most students have 12th-grade percentages between 60-90 and GPAs between 60-80.
3. Some students have low percentages but high GPAs, and vice versa.

In [67]:
```
sns.scatterplot(data=df,x="10percentage",y="12percentage")
```

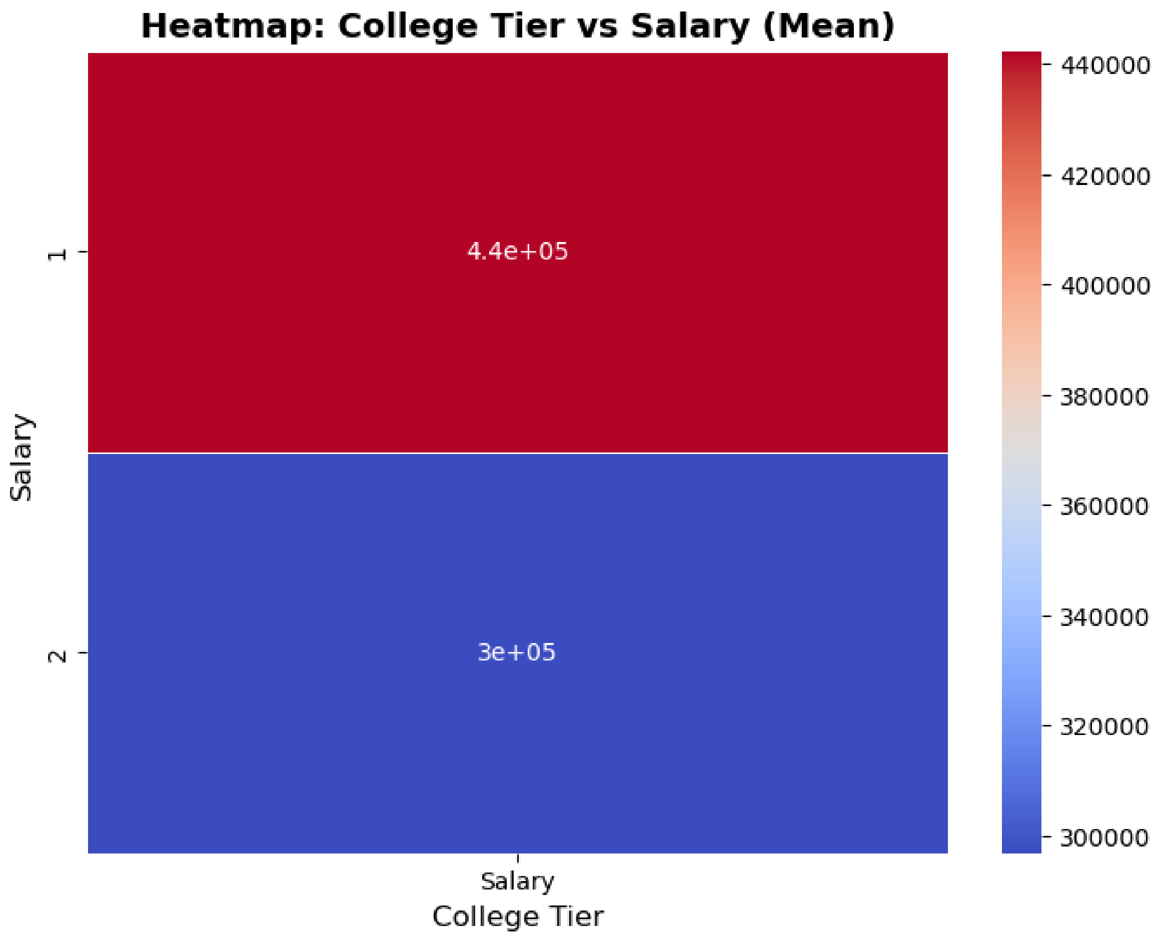
Out[67]: <Axes: xlabel='10percentage', ylabel='12percentage'>



1. Positive Correlation- Higher 10th-grade percentages are strongly linked to higher 12th-grade percentages.
2. Most students have 10th-grade percentages between 60-90 and similar 12th-grade percentages.
3. A few students have high 10th percentages but relatively lower 12th percentages, and vice versa.

In [69]:

```
pivot_table = df.pivot_table(values='Salary', index='CollegeTier', aggfun

plt.figure(figsize=(8, 6))
sns.heatmap(pivot_table, annot=True, cmap='coolwarm', linewidths=0.5)

plt.title('Heatmap: College Tier vs Salary (Mean)', fontsize=14, fontweig
plt.xlabel('College Tier', fontsize=12)
plt.ylabel('Salary', fontsize=12)

plt.show()
```



College Tier 1 has a significantly higher average salary (440,000) compared to College Tier 2 (300,000).

This shows that students or professionals coming from College Tier 1 tend to secure jobs with higher salaries compared to those from College Tier 2.

Research Questions

Is there a relationship between gender and specilaization?(Does the presence of Specialization depend on Gender?)

In [73]:
```
df[["Gender","Specialization"]].head()
```

Out[73]:

	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 [74]:
```
df["Gender"].value_counts()
```

Out[74]:
```
Gender
m    3041
f     957
Name: count, dtype: int64
```

In [75]:
```
grouped_3 = df.groupby(['Specialization', 'Gender']).size().unstack(fill_
grouped_3
```

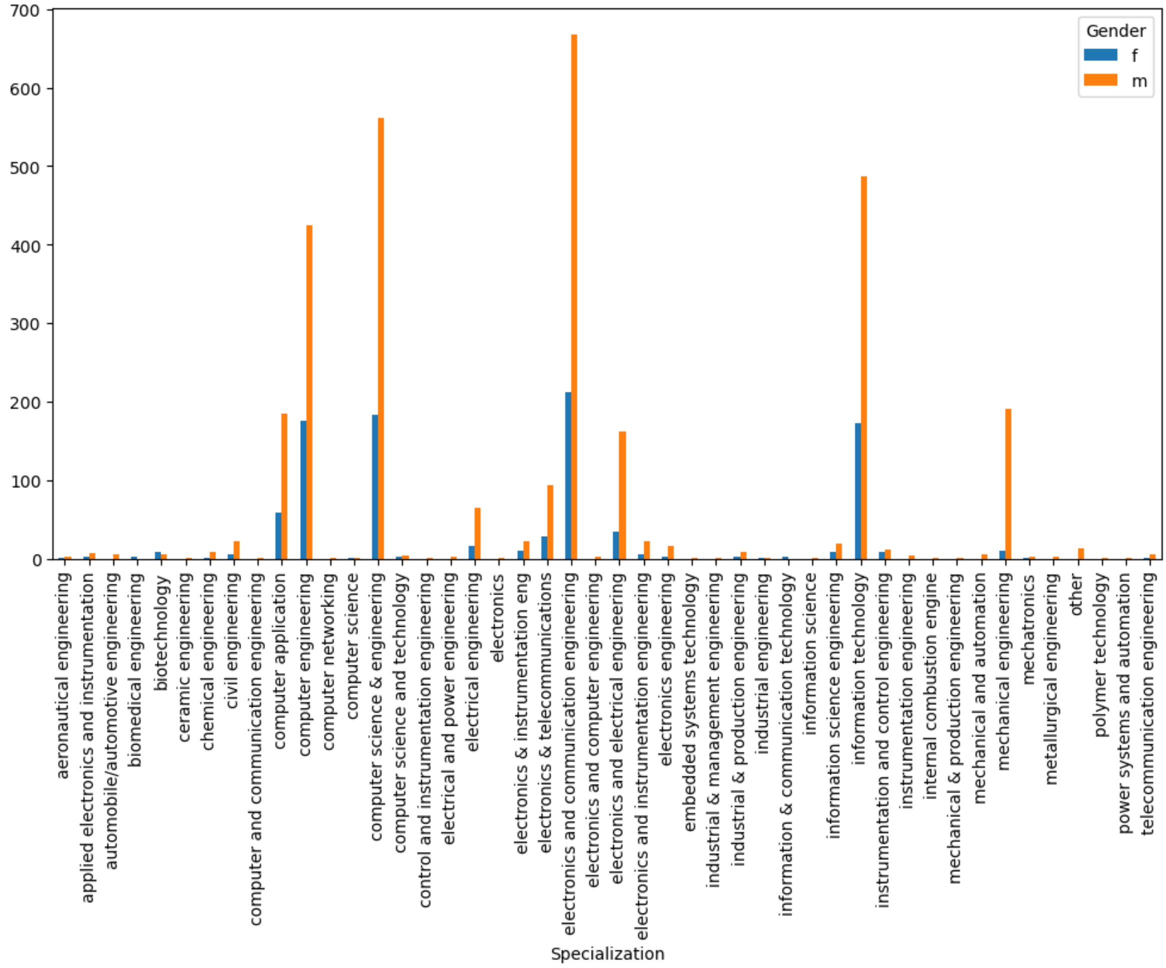
Out[75]:

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

Gender	f	m
Specialization		
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 [76]:
```
grouped_3.plot(kind="bar",figsize=(12,6))
```

Out[76]: <Axes: xlabel='Specialization'>

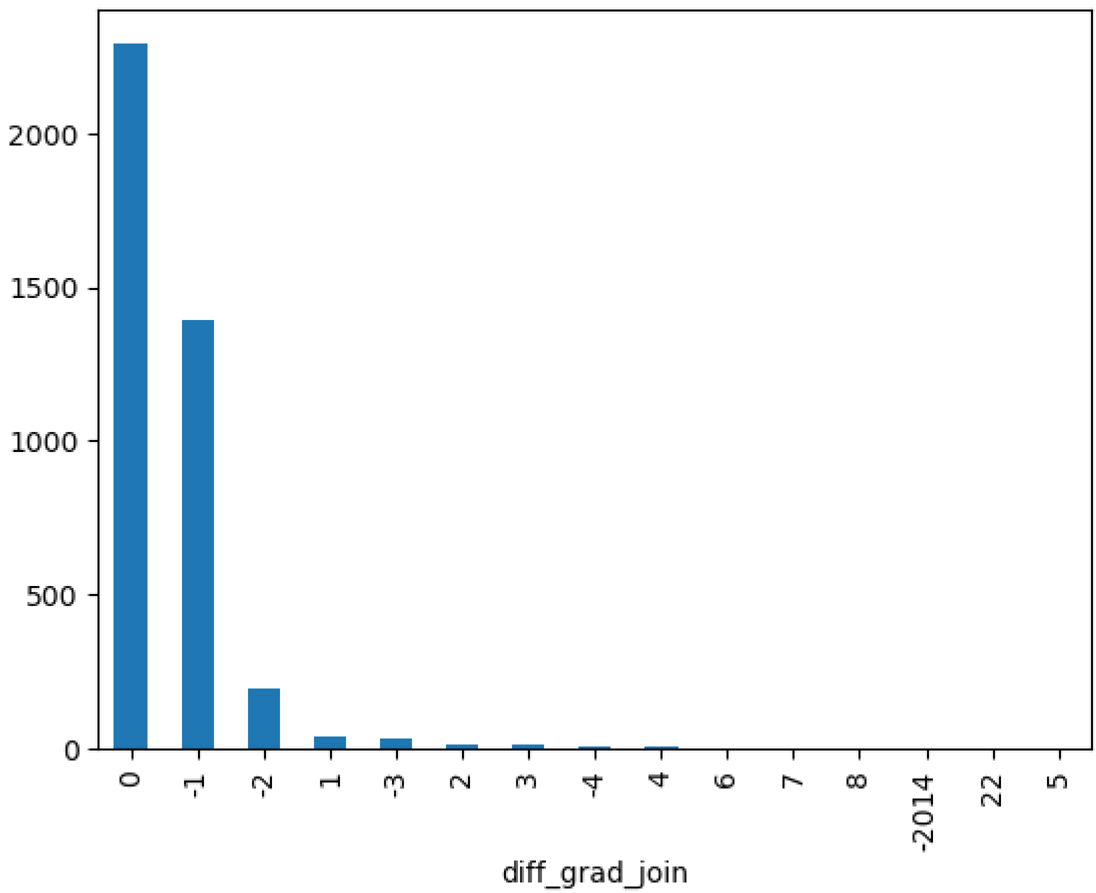


Yes,every Specialization have more male engineers compared to female engineers.In ceramic engineering,polymer technology,information science and power systems and automation there are no female engineers.Overall the percentage of female engineers is less comparitively.

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 Engineer you can earn up to 2.5-3 lakhs as a fresh graduate.” Test this claim with the data given to you.

In [79]:
```
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[79]: <Axes: xlabel='diff_grad_join'>



In [80]:
```
df_modified=df[["Designation","Specialization","Salary"]]
print(df_modified)
```

```
                                Designation                             Specializati
on  \
0          senior quality engineer                          computer engineeri
ng
1                assistant manager  electronics and communication engineeri
ng
2                 systems engineer                        information technolo
gy
3         senior software engineer                          computer engineeri
ng
4                              get  electronics and communication engineeri
ng
...                            ...                                            
...
3993             software engineer                        information technolo
gy
3994              technical writer  electronics and communication engineeri
ng
3995  associate software engineer                          computer engineeri
ng
3996            software developer              computer science & engineeri
ng
3997       senior systems engineer                        information technolo
gy

       Salary
0      420000
1      500000
2      325000
3     1100000
4      200000
...       ...
3993   280000
3994   100000
3995   320000
3996   200000
3997   400000

[3998 rows x 3 columns]
```

In [110…

```
df1 = df[df['Designation'].isin(['programmer analyst', 'software engineer
df1
```

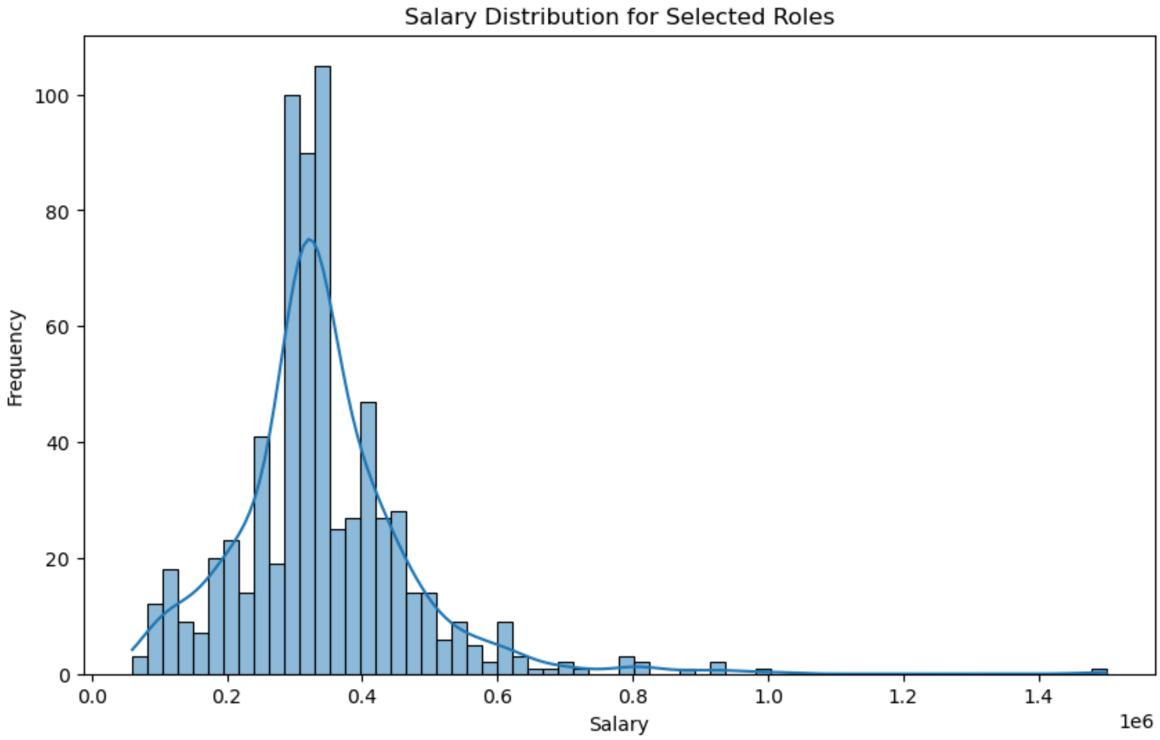
Out[110…

	ID	Salary	DOJ	DOL	Designation	JobCity	Gender	I
19	466888	325000	2014-09-01	present	software engineer	Pune	f	1 1
20	140069	320000	2010-11-01	2012-09-01 00:00:00	software engineer	Bangalore	f	1 0
21	339689	200000	2012-08-01	2013-12-01 00:00:00	software engineer	-1	f	1 0
24	963123	335000	2014-06-01	2015-06-01 00:00:00	programmer analyst	Hyderabad	m	1 0
31	1094324	340000	2014-08-01	2015-04-01 00:00:00	software engineer	Bangalore	m	1 1
...	...	...	...	...	...	...	...	
3979	212055	550000	2013-07-01	2014-04-01 00:00:00	software engineer	Bangalore	m	1 0
3981	1077872	220000	2014-09-01	present	software engineer	Gurgaon	m	1 1
3984	305041	480000	2011-12-01	present	software engineer	Gurgaon	f	1 0
3989	1204604	300000	2014-09-01	present	software engineer	Bangalore	m	1 1
3993	47916	280000	2011-10-01	2012-10-01 00:00:00	software engineer	New Delhi	m	1 0

692 rows × 39 columns

In [112…

```
plt.figure(figsize=(10,6))
sns.histplot(data=df1, x="Salary", kde=True)
plt.title('Salary Distribution for Selected Roles')
plt.xlabel('Salary')
plt.ylabel('Frequency')
plt.show()
```



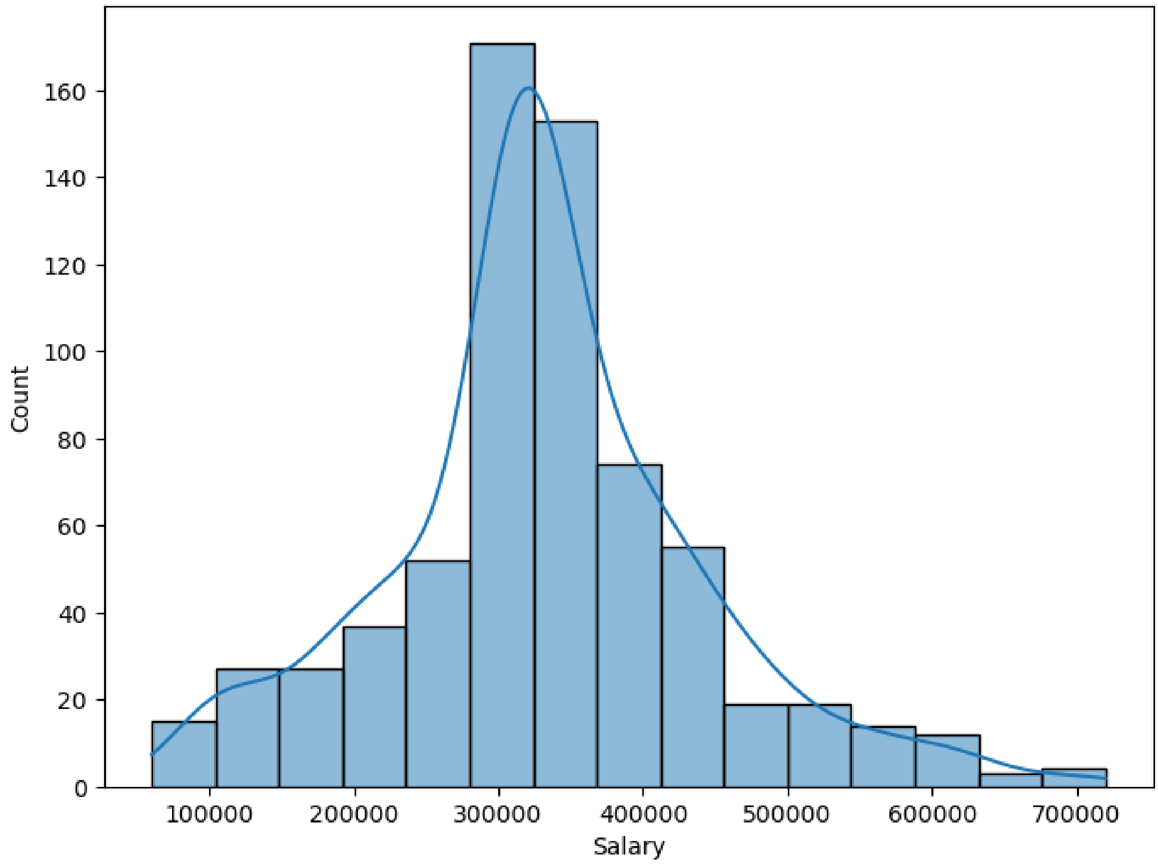
In [114…
```
average_salary=df1['Salary'].mean()
print('Average Salary:',average_salary)
```

```
Average Salary: 339790.4624277457
```

In [116…
```
max_salary = df1['Salary'].max()
if max_salary >= 250000 and max_salary <= 300000:
    print("The claim that fresh graduates can earn up to 2.5-3 lakhs is s
else:
    print("The claim that fresh graduates can earn up to 2.5-3 lakhs is n
```

```
The claim that fresh graduates can earn up to 2.5-3 lakhs is not supported
by the data.
```

In [118…
```
from scipy.stats import zscore

df2 = df1[zscore(df1['Salary']) < 3]
fig, ax = plt.subplots(figsize=(8, 6))
sns.histplot(df2['Salary'], bins=15,kde=True)
plt.show()
```



In [89]:
```
df_modified_filter3=df_modified[(df_modified["Designation"]=="hardware en
print(df_modified_filter3.head(5))
```

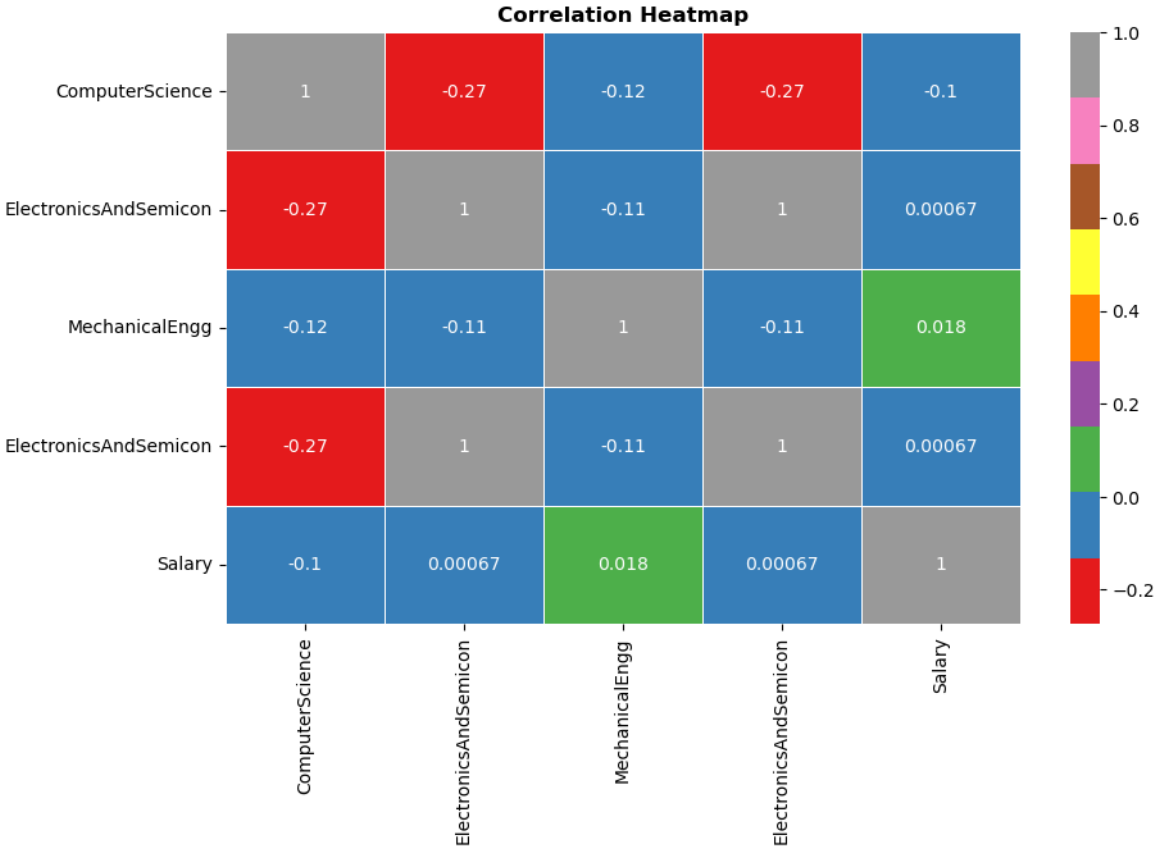
```
Empty DataFrame
Columns: [Designation, Specialization, Salary]
Index: []
```

Programming Analyst, Software Engineer and Associate Engineer can earn up to 2.5-3 lakhs as a fresh graduate is not supported by the data. The statistics does not show any students who is computer science & engineering working as hardware engineer

Additional Research

How does different engineering specializations(Computer Science&Engineering,Electronics and Communication engineering,MechanicalEngg,Electronics & Instrumentation Eng)contribute to Salary??

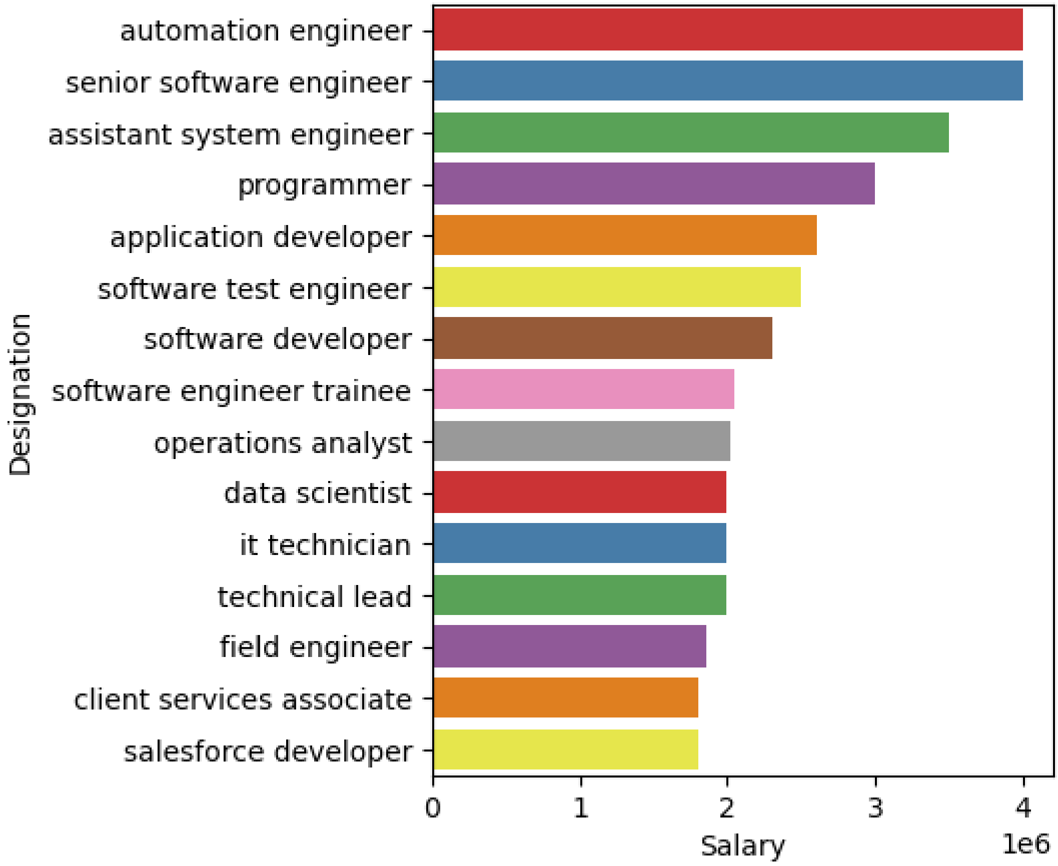
In [93]:
```
sc= ['ComputerScience', 'ElectronicsAndSemicon', 'MechanicalEngg', 'Elect

corr_matrix = df[sc].corr()

plt.figure(figsize=(10, 6))
sns.heatmap(corr_matrix, annot=True, cmap="Set1", linewidths=0.5)
plt.title('Correlation Heatmap', fontweight='bold')
plt.show()
```



Which top 50 jobs Designation has more salary in IT companies ?

In [103…
```
df_destignation = df.groupby('Designation')['Salary'].max().sort_values(a
```

In [107…
```
plt.figure(figsize=(4,5))
sns.barplot(y='Designation', x='Salary', data=df_destignation,palette="Se
plt.show()
```



High-Paying Roles: Analysis reveals that job titles such as Automation Engineer,senior software engineer,application developer, and Technology Lead are among the top 15 positions commanding higher salaries within IT firms.

CONCLUSION

The actual average salaries for roles such as Programming Analyst, Software Engineer and Associate Engineer align closely with the salary range (2.5-3 lakhs) mentioned in the Times of India article but not in the given range. Graduates specializing in Computer Science and IT-related fields tend to receive higher salaries, highlighting the increasing demand for tech skills in the industry.There is an uneven gender distribution in various job roles, with male graduates dominating certain specializations, suggesting possible gender biases in hiring practices. The tech industry continues to drive salary increases, particularly for roles requiring programming and software engineering skills, underscoring the importance of tech expertise in today's job markete accurate.

In [ ]:

```

```