Exploratory Data Analysis practise

```
In [1]: import pandas as pd
        emp = pd.read_excel(r"C:\Users\Affan\OneDrive\Desktop\FSDS Course NIT\Prakash Si
In [2]:
In [3]:
Out[3]:
                          Domain
                                            Location
            Name
                                      Age
                                                        Salary
                                                                   Exp
              Mike
                     Datascience#$ 34 years
                                                                    2+
                                             Mumbai
                                                       5^00#0
         1 Teddy^
                                     45' yr Bangalore 10%%000
                                                                    <3
                           Testing
           Uma#r Dataanalyst^^#
                                                      1$5%000
                                      NaN
                                                NaN
                                                                 4> yrs
         3
                       Ana^^lytics
                                    NaN Hyderbad
                                                       2000^0
                                                                  NaN
              Jane
           Uttam*
                          Statistics
                                     67-yr
                                                NaN
                                                        30000-
                                                                5+ year
                              NLP
                                      55yr
                                                Delhi
                                                      6000^$0
                                                                   10+
               Kim
In [4]:
        emp.isnull().sum()
Out[4]: Name
         Domain
                     2
         Age
         Location
                     2
         Salary
         Exp
         dtype: int64
In [5]: emp.info()
       <class 'pandas.core.frame.DataFrame'>
       RangeIndex: 6 entries, 0 to 5
       Data columns (total 6 columns):
        # Column
                    Non-Null Count Dtype
        0 Name 6 non-null
                                       object
        1 Domain 6 non-null
2 Age 4 non-null
3 Location 4 non-null
                                       object
                                       object
                                       object
        4 Salary 6 non-null
                                       object
                      5 non-null
                                       object
            Exp
       dtypes: object(6)
       memory usage: 420.0+ bytes
In [6]: emp.describe()
```

| Out[6]: | | N | lame | Dom | ain | Age | Loc | ation | Salaı | у Ехр | |
|---------|------|----------------|---------|------------|---------|---------|--------|-------|---------|---------|------------------|
| | c | ount | 6 | | 6 | 4 | | 4 | | 6 5 | |
| | un | nique | 6 | | 6 | 4 | | 4 | | 6 5 | |
| | | top | Mike Da | tascience | e#\$ 34 | l years | Mu | mbai | 5^00# | ŧ0 2+ | |
| | | freq | 1 | | 1 | 1 | | 1 | | 1 1 | |
| | | | | | | | | | | | |
| In [7]: | em | p.isna(|) | | | | | | | | |
| Out[7]: | | Name | Domain | Age | Locatio | on Sa | lary | Ехр | _ | | |
| | 0 | False | False | False | Fal | se F | alse | False | | | |
| | 1 | False | False | False | Fal | se F | alse | False | | | |
| | 2 | False | False | True | Tro | ue F | alse | False | | | |
| | 3 | False | False | True | Fal | se F | alse | True | | | |
| | 4 | False | False | False | Tro | ue F | alse | False | | | |
| | 5 | False | False | False | Fal | se F | alse | False | | | |
| In [8]: | ρm | p.head(|) | | | | | | | | |
| | Cili | | | | | | _ | | | _ | _ |
| Out[8]: | _ | Name | | Domain | | | | on | | | Exp 2+ |
| | 0 | Mike Teddy^ | | ience#\$ | - | | | bai | 5^00# | | <3 |
| | 2 | , | Dataana | _ | | aN | _ | | 1\$5%00 | | |
| | 3 | Jane | | ^^lytics | | | yderb | | 2000^ | | yıs IaN |
| | 4 | Uttam* | | Statistics | | -yr | • | aN | 30000 | | |
| | 7 | Ottain | ` | Jacistics | 07 | yı | 14 | aiv | 30000 | , 31 y | cai |
| In [9]: | em | p.tail(|) | | | | | | | | |
| Out[9]: | | Name | | Domain | Age | Loc | ation | : | Salary | Ехр |) |
| | 1 | Teddy^ | | Testing | 45' yr | Bang | galore | 10% | 6%000 | <3 | 3 |
| | 2 | Uma#r | Dataana | ılyst^^# | NaN | | NaN | 1\$! | 5%000 | 4> yrs | S |
| | 3 | Jane | Ana | ^^lytics | NaN | Hyd | erbad | 2 | 000^0 | NaN | 1 |
| | 4 | Uttam* | Ç | Statistics | 67-yr | | NaN | 3 | 30000- | 5+ year | r |
| | 5 | Kim | | NLP | 55yr | | Delhi | 60 | 00^\$0 | 10+ | + |
| | | | | | | | | | | | |
| n [10]: | id | (emp) | | | | | | | | | |
| ut[10]: | 18 | 80736100 | 2672 | | | | | | | | |

In [11]: len(emp)

```
Out[11]: 6
         emp.shape
In [12]:
Out[12]: (6, 6)
In [13]: emp.columns
Out[13]: Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
In [14]: len(emp.columns)
Out[14]: 6
         emp['Name']
In [15]:
Out[15]: 0
                 Mike
               Teddy^
          2
                Uma#r
          3
                 Jane
               Uttam*
          5
                  Kim
          Name: Name, dtype: object
In [16]: emp['Domain']
                Datascience#$
Out[16]: 0
          1
                      Testing
          2
               Dataanalyst^^#
          3
                  Ana^^lytics
          4
                   Statistics
                          NLP
          Name: Domain, dtype: object
In [17]:
         emp[['Name','Domain','Age','Location','Salary','Exp']]
Out[17]:
              Name
                           Domain
                                       Age
                                             Location
                                                          Salary
                                                                    Ехр
          0
                                                         5^00#0
                                                                     2+
               Mike
                      Datascience#$ 34 years
                                              Mumbai
            Teddy^
                                      45' yr Bangalore
                            Testing
                                                       10%%000
                                                                     <3
             Uma#r Dataanalyst^^#
          2
                                       NaN
                                                 NaN
                                                       1$5%000
                                                                  4> yrs
                        Ana^^lytics
                                       NaN Hyderbad
          3
               Jane
                                                         2000^0
                                                                    NaN
             Uttam*
                           Statistics
                                      67-yr
                                                 NaN
                                                         30000-
                                                                 5+ year
                Kim
                               NLP
                                       55yr
                                                 Delhi
                                                        6000^$0
                                                                    10+
```

data cleansing

```
In [18]: emp['Name']
```

```
Out[18]: 0
               Mike
         1
             Teddy^
         2
              Uma#r
         3
                Jane
         4
              Uttam*
         5
                 Kim
         Name: Name, dtype: object
In [19]: emp['Name'] = emp['Name'].str.replace(r'\W','',regex=True) #W means we can parse
In [20]: emp['Name']
Out[20]: 0
               Mike
             Teddy
         2
              Umar
         3
               Jane
         4
              Uttam
                Kim
         Name: Name, dtype: object
In [21]: emp['Domain'] = emp['Domain'].str.replace(r'\W','',regex=True)
In [22]: emp['Domain']
Out[22]: 0
              Datascience
         1
                  Testing
         2 Dataanalyst
         3
               Analytics
         4
               Statistics
         5
                      NLP
         Name: Domain, dtype: object
In [23]: emp['Age'] = emp['Age'].str.extract('(\d+)')
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        C:\Users\Affan\AppData\Local\Temp\ipykernel_10220\1884116463.py:1: SyntaxWarning:
        invalid escape sequence '\d'
          emp['Age'] = emp['Age'].str.extract('(\d+)')
In [24]: emp['Age']
Out[24]: 0
               34
         1
              45
         2
              NaN
         3
              NaN
               67
         4
         5
               55
         Name: Age, dtype: object
In [25]: emp['Location'] = emp['Location'].str.replace(r'\W','',regex=True)
In [26]: emp['Location']
```

```
Out[26]: 0
                 Mumbai
               Bangalore
                     NaN
          3
              Hyderbad
                     NaN
                   Delhi
          Name: Location, dtype: object
In [27]: emp['Salary'] = emp['Salary'].str.replace(r'\W','',regex=True)
In [28]:
         emp['Salary']
Out[28]: 0
                5000
               10000
          2
               15000
          3
               20000
          4
               30000
               60000
          Name: Salary, dtype: object
In [29]: emp['Exp'] = emp['Exp'].str.extract('(\d+)')
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        C:\Users\Affan\AppData\Local\Temp\ipykernel_10220\3836251810.py:1: SyntaxWarning:
        invalid escape sequence '\d'
          emp['Exp'] = emp['Exp'].str.extract('(\d+)')
In [30]: emp['Exp']
Out[30]: 0
                 2
          1
                 3
          2
                 4
          3
               NaN
                 5
          4
          5
                10
          Name: Exp, dtype: object
In [31]: clean_data = emp.copy()
          clean_data #raw to clean data
Out[31]:
             Name
                                Age
                                      Location Salary
                                                        Exp
                       Domain
          0
              Mike
                    Datascience
                                 34
                                       Mumbai
                                                 5000
                                                          2
             Teddy
                        Testing
                                 45
                                     Bangalore
                                                10000
                                                          3
          2
             Umar
                    Dataanalyst
                                NaN
                                          NaN
                                                15000
                                                          4
              Jane
                      Analytics
                                NaN
                                      Hyderbad
                                                20000
                                                          5
          4
             Uttam
                       Statistics
                                 67
                                          NaN
                                                30000
          5
                          NLP
                                  55
               Kim
                                          Delhi
                                                60000
                                                         10
```

missing value data treatment

```
In [33]: clean_data.info()
                       <class 'pandas.core.frame.DataFrame'>
                       RangeIndex: 6 entries, 0 to 5
                       Data columns (total 6 columns):
                                    Column
                                                               Non-Null Count Dtype
                                                                 -----
                          0
                                 Name
                                                             6 non-null
                                                                                                              object
                                   Domain 6 non-null
                                                                                                              object
                          2
                                   Age
                                                            4 non-null
                                                                                                              object
                          3
                                    Location 4 non-null
                                                                                                              object
                          4
                                 Salary
                                                                 6 non-null
                                                                                                              object
                          5
                                                                 5 non-null
                                                                                                              object
                                     Exp
                       dtypes: object(6)
                       memory usage: 420.0+ bytes
In [34]: import numpy as np #to take care of multi dim array
In [35]: #numpy and pndas is used for data cleaning
In [36]: clean_data.head(1)
Out[36]:
                                   Name
                                                               Domain Age Location Salary Exp
                                                                                                                                   5000
                                                                                                                                                          2
                                      Mike Datascience
                                                                                           34
                                                                                                        Mumbai
                        clean_data
In [37]:
Out[37]:
                                                                                                        Location Salary
                                   Name
                                                               Domain
                                                                                       Age
                                                                                                                                                        Exp
                            0
                                      Mike Datascience
                                                                                            34
                                                                                                           Mumbai
                                                                                                                                      5000
                                                                                                                                                              2
                                    Teddy
                                                                  Testing
                                                                                            45
                                                                                                       Bangalore
                                                                                                                                    10000
                                                                                                                                                              3
                                     Umar
                                                      Dataanalyst
                                                                                                                                   15000
                            2
                                                                                       NaN
                                                                                                                   NaN
                                                                                                                                                              4
                            3
                                       Jane
                                                              Analytics NaN
                                                                                                       Hyderbad
                                                                                                                                   20000
                                                                                                                                                      NaN
                            4
                                   Uttam
                                                              Statistics
                                                                                            67
                                                                                                                   NaN
                                                                                                                                   30000
                                                                                                                                                              5
                                                                         NLP
                            5
                                         Kim
                                                                                            55
                                                                                                                  Delhi
                                                                                                                                   60000
                                                                                                                                                            10
In [38]: clean data['Age'] = clean data['Age'].fillna(np.mean(pd.to numeric(clean data['Age'].fillna(np.to numeric(clean data['Age'].f
In [39]: clean_data['Age']
Out[39]: 0
                                                  34
                            1
                                                  45
                            2
                                         50.25
                            3
                                          50.25
                            4
                                                  67
                                                  55
                            Name: Age, dtype: object
In [40]:
                          emp
```

| Out[40]: | | Name | Domain | Age | Location | Salary | Ехр |
|---------------------------------------|----------------------------------------|----------------------------------------------------------------|-----------------------------------------------------------------|----------------------|---------------------------------|----------------------|------------------|
| | 0 | Mike | Datascience | 34 | Mumbai | 5000 | 2 |
| | 1 | Teddy | Testing | 45 | Bangalore | 10000 | 3 |
| | 2 | Umar | Dataanalyst | NaN | NaN | 15000 | 4 |
| | 3 | Jane | Analytics | NaN | Hyderbad | 20000 | NaN |
| | 4 | Uttam | Statistics | 67 | NaN | 30000 | 5 |
| | 5 | Kim | NLP | 55 | Delhi | 60000 | 10 |
| | | | | | | | |
| In [41]: | cl | ean_dat | a | | | | |
| Out[41]: | | Name | Domain | Age | Location | Salary | Ехр |
| | 0 | Mike | Datascience | 34 | Mumbai | 5000 | 2 |
| | 1 | Teddy | Testing | 45 | Bangalore | 10000 | 3 |
| | 2 | Umar | Dataanalyst | 50.25 | NaN | 15000 | 4 |
| | 3 | Jane | Analytics | 50.25 | Hyderbad | 20000 | NaN |
| | 4 | Uttam | Statistics | 67 | NaN | 30000 | 5 |
| | 5 | Kim | NLP | 55 | Delhi | 60000 | 10 |
| | | | | | | | |
| Tn [42]· | c1 | ean dat | a['Exn'] = (| -lean (| data['Exn' | 1.filln | a(nn.m |
| In [42]: | | | a['Exp'] = 0 | | | | |
| In [42]: In [45]: | | | a['Exp'] = 0 | | | | |
| In [45]: | c1 0 | ean_dat 2 | | | | | |
| In [45]: | cl 0 1 2 | ean_dat 2 3 4 | | | | | |
| | cl 0 1 2 3 | ean_dat 2 3 4 4.8 | | | | | |
| In [45]: | cl 0 1 2 | ean_dat 2 3 4 | | | | | |
| In [45]: | cl 0 1 2 3 4 5 | ean_dat 2 3 4 4.8 5 10 | | Leanin | | | |
| In [45]: Out[45]: | cl 0 1 2 3 4 5 Na | ean_dat 2 3 4 4.8 5 10 | a['Exp'] #c | Leanin | | | |
| In [45]: Out[45]: In [44]: | cl 0 1 2 3 4 5 Na | ean_dat 2 3 4 4.8 5 10 | a['Exp'] #c | Leanin | | e in ex _l | D |
| In [45]: Out[45]: In [44]: | cl 0 1 2 3 4 5 Na | ean_dat 2 3 4 4.8 5 10 nme: Exp | a['Exp'] #co | Jeaning | g Nan vaLu | e in ex _l | D |
| In [45]: | cl 0 1 2 3 4 5 Na | ean_dat 2 3 4 4.8 5 10 mme: Exp ean_dat Name | a['Exp'] #c | ject Age | g Nan value | e in ex | Ехр |
| In [45]: Out[45]: In [44]: | cl 0 1 2 3 4 5 Na cl | ean_dat 2 3 4 4.8 5 10 nme: Exp ean_dat Name | a['Exp'] #c | ject Age 34 45 | Location Mumbai | Salary 5000 | Exp 2 |
| In [45]: Out[45]: In [44]: | cl 0 1 2 3 4 5 Na cl | ean_dat 2 3 4 4.8 5 10 nme: Exp ean_dat Name Mike Teddy | a['Exp'] #c | ject Age 34 45 50.25 | Location Mumbai Bangalore | Salary 5000 10000 | Exp 2 3 |
| <pre>In [45]: Out[45]: In [44]:</pre> | cl 0 1 2 3 4 5 Na cl 0 1 2 | ean_dat 2 3 4 4.8 5 10 ame: Exp ean_dat Name Mike Teddy Umar | a ['Exp'] #contact a Domain Datascience Testing Dataanalyst | ject Age 34 45 50.25 | Location Mumbai Bangalore NaN | Salary 5000 10000 | Exp 2 3 4 |

```
In [46]: clean_data['Location']=clean_data['Location'].fillna(clean_data['Location'].mode
  In [47]: clean_data['Location']
  Out[47]: 0
                    Mumbai
            1
                 Bangalore
            2
                 Bangalore
            3
                 Hyderbad
            4
                 Bangalore
            5
                      Delhi
            Name: Location, dtype: object
----to chnage dtype of attributes-----
  In [59]: clean_data.info()
           <class 'pandas.core.frame.DataFrame'>
           RangeIndex: 6 entries, 0 to 5
           Data columns (total 6 columns):
               Column Non-Null Count Dtype
           --- -----
           0 Name 6 non-null category
1 Domain 6 non-null category
2 Age 6 non-null int32
3 Location 6 non-null category
            4
              Salary 6 non-null int32
            5
                Exp
                        6 non-null
                                          int32
           dtypes: category(3), int32(3)
           memory usage: 866.0 bytes
#dtypes - how to convert Name to category, domain also...age-number,location,cateogry,salary-no.....by defualt python gives
objewct as defualt dtype
  In [52]:
            clean_data['Age'] = clean_data['Age'].astype(int)
            clean_data['Exp'] = clean_data['Exp'].astype(int)
            clean_data['Salary'] = clean_data['Salary'].astype(int)
  In [53]: clean_data.info()
           <class 'pandas.core.frame.DataFrame'>
           RangeIndex: 6 entries, 0 to 5
           Data columns (total 6 columns):
            # Column Non-Null Count Dtype
                          -----
           ---
               _____
            0 Name
                         6 non-null
                                          object
           1 Domain 6 non-null
                                          object
           2 Age 6 non-null int32
3 Location 6 non-null object
                                           object
            4 Salary
                         6 non-null
                                           int32
                          6 non-null
                                           int32
           dtypes: int32(3), object(3)
           memory usage: 348.0+ bytes
----datatype is changed now----
  In [56]:
            clean data['Name'] = clean data['Name'].astype('category')
            clean_data['Domain'] = clean_data['Domain'].astype('category')
            clean_data['Location'] = clean_data['Location'].astype('category')
  In [57]: clean_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6 entries, 0 to 5
Data columns (total 6 columns):
   Column Non-Null Count Dtype
--- ----- -----
0 Name 6 non-null category
1 Domain 6 non-null category
2 Age 6 non-null
                          int32
3 Location 6 non-null
                          category
   Salary 6 non-null
                           int32
                           int32
5
    Exp
            6 non-null
dtypes: category(3), int32(3)
memory usage: 866.0 bytes
```

WHAT HAVE WE DONE TILL NOW????? ---we have raw data (unclean, filled with noise char(*^\%#)--regex, replace, extrac)-----numeric(mean) || category(mode) ----change inbuild dtype to respective dtype

| In [60]: | cl | ean_dat | a | | | | |
|----------|-----|----------------------------|-------------|--------|------------|--------|-----|
| Out[60]: | | Name | Domain | Age | Location | Salary | Ехр |
| | 0 | Mike | Datascience | 34 | Mumbai | 5000 | 2 |
| | 1 | Teddy | Testing | 45 | Bangalore | 10000 | 3 |
| | 2 | Umar | Dataanalyst | 50 | Bangalore | 15000 | 4 |
| | 3 | Jane | Analytics | 50 | Hyderbad | 20000 | 4 |
| | 4 | Uttam | Statistics | 67 | Bangalore | 30000 | 5 |
| | 5 | Kim | NLP | 55 | Delhi | 60000 | 10 |
| | _ | | | | | | |
| In [61]: | cl | ean_dat | a.to_csv('c | lean_c | lata.csv') | | |
| In [62]: | | <pre>port os .getcwd</pre> | | | | | |
| Out[62]: | ' (| :\\User | s\\Affan' | | | | |

once data set is cleaned we visualize thenmon-25 deg, tues,35 deg(predictive model)

Var identification

```
In [64]: clean_data.columns
Out[64]: Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
In [65]: #for visulization we need matplotlib and seaborn
In [66]: import matplotlib.pyplot as plt import seaborn as sns
In [67]: import warnings warnings.filterwarnings('ignore')
In [68]: clean_data
```

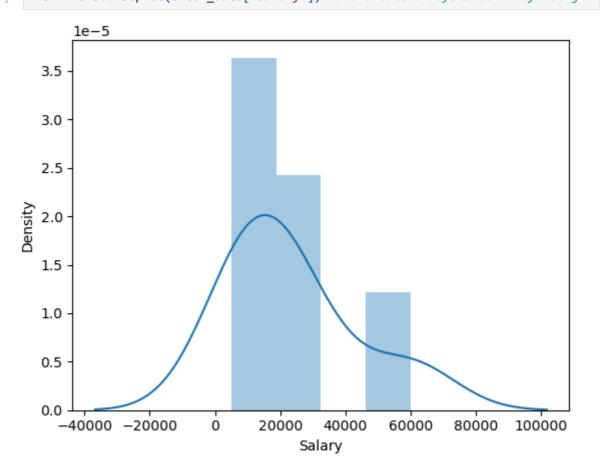
| Out[68]: | | Name | Domain | Age | Location | Salary | Ехр |
|----------|---|-------|-------------|-----|-----------|--------|-----|
| | 0 | Mike | Datascience | 34 | Mumbai | 5000 | 2 |
| | 1 | Teddy | Testing | 45 | Bangalore | 10000 | 3 |
| | 2 | Umar | Dataanalyst | 50 | Bangalore | 15000 | 4 |
| | 3 | Jane | Analytics | 50 | Hyderbad | 20000 | 4 |
| | 4 | Uttam | Statistics | 67 | Bangalore | 30000 | 5 |
| | 5 | Kim | NLP | 55 | Delhi | 60000 | 10 |

```
In [69]: clean_data['Salary']
```

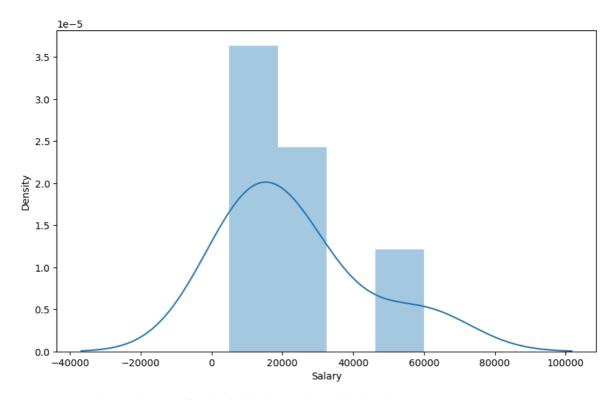
Out[69]: 0 5000 1 10000 2 15000 3 20000 4 30000

> 5 60000 Name: Salary, dtype: int32

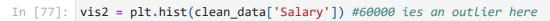
In [71]: vis1 = sns.distplot(clean_data['Salary']) #univariate analysis coz only using 1

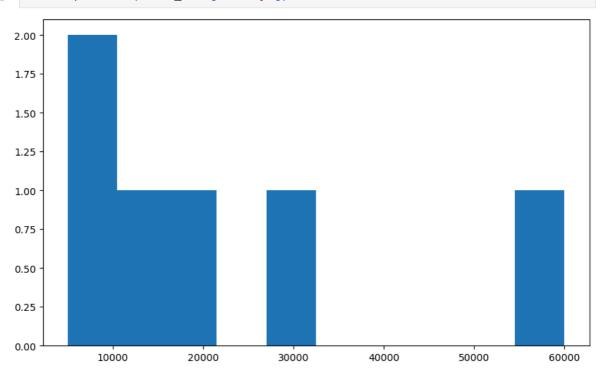


```
In [74]: plt.rcParams['figure.figsize']=10,6
  vis1 = sns.distplot(clean_data['Salary'])
```

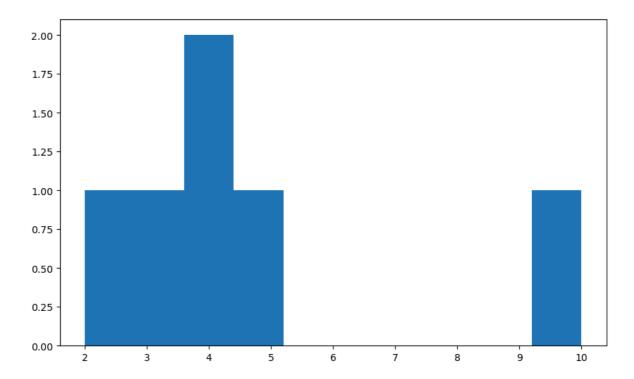


OUTLIER DETECTION --- how to detect outlier by visulization tecnique (dist lm plot)

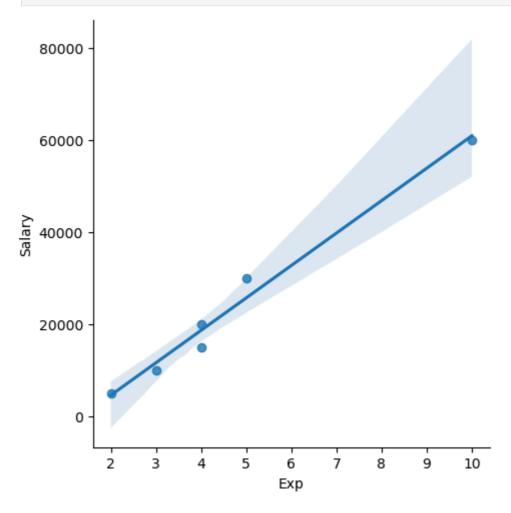




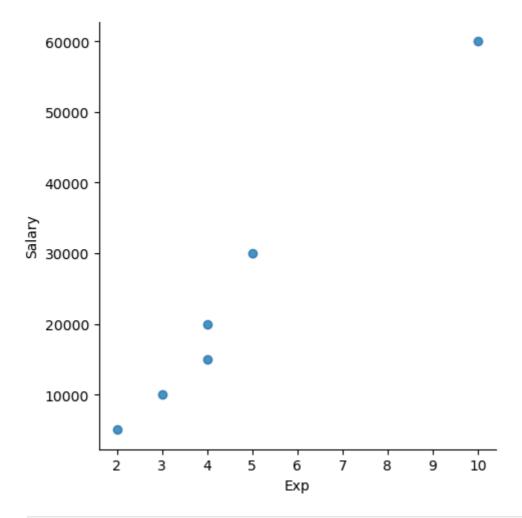
In [78]: vis3 = plt.hist(clean_data['Exp'])



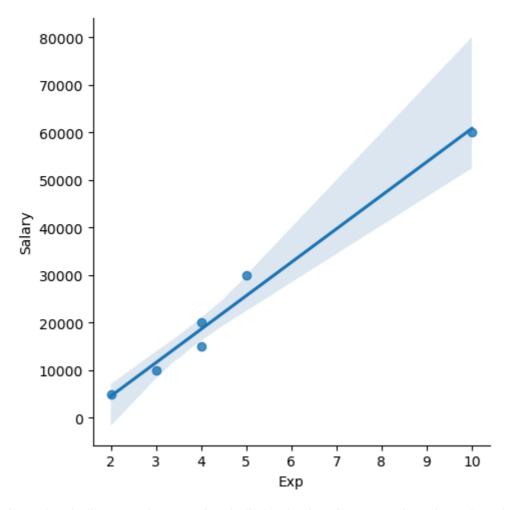
In [79]: vis4 = sns.lmplot(data=clean_data,x='Exp',y='Salary')



In [80]: vis5 = sns.lmplot(data=clean_data,x='Exp',y='Salary',fit_reg=False)



In [81]: vis4 = sns.lmplot(data=clean_data,x='Exp',y='Salary',fit_reg=True)



heatmap can be used to visuliaze more than 2 varafter visuliazing the clean data, Var tranformation and creation---relevant, irrelevant data

| In [82]: | cl | ean_dat | a | | | | |
|----------|----|---------|-------------|-----|-----------|--------|-----|
| Out[82]: | | Name | Domain | Age | Location | Salary | Ехр |
| | 0 | Mike | Datascience | 34 | Mumbai | 5000 | 2 |
| | 1 | Teddy | Testing | 45 | Bangalore | 10000 | 3 |
| | 2 | Umar | Dataanalyst | 50 | Bangalore | 15000 | 4 |
| | 3 | Jane | Analytics | 50 | Hyderbad | 20000 | 4 |
| | 4 | Uttam | Statistics | 67 | Bangalore | 30000 | 5 |
| | 5 | Kim | NLP | 55 | Delhi | 60000 | 10 |
| | | | | | | | |
| In [83]: | cl | ean_dat | a[:] | | | | |

| Out[83]: | | Name | Domain | Age | Location | Salary | Ехр |
|-------------------|-------------|------------------------------|--------------------------------------|----------------------|------------------------------------|----------------------------------|--------------|
| | 0 | Mike | Datascience | 34 | Mumbai | 5000 | 2 |
| | 1 | Teddy | Testing | 45 | Bangalore | 10000 | 3 |
| | 2 | Umar | Dataanalyst | 50 | Bangalore | 15000 | 4 |
| | 3 | Jane | Analytics | 50 | Hyderbad | 20000 | 4 |
| | 4 | Uttam | Statistics | 67 | Bangalore | 30000 | 5 |
| | 5 | Kim | NLP | 55 | Delhi | 60000 | 10 |
| | | | | | | | |
| In [84]: | cl | ean_dat | a[2:] | | | | |
| | | | | | | | |
| Out[84]: | | Name | Domain | Age | Location | Salary | Ехр |
| Out[84]: | 2 | Name Umar | | Age 50 | Location Bangalore | Salary 15000 | Exp 4 |
| Out[84]: | 2 | | | | | | |
| Out[84]: | | Umar | Dataanalyst | 50 | Bangalore | 15000 | 4 |
| Out[84]: | 3 | Umar | Dataanalyst Analytics | 50 50 | Bangalore Hyderbad | 15000 | 4 |
| Out[84]: | 3 | Umar Jane Uttam | Dataanalyst Analytics Statistics | 50 50 67 | Bangalore Hyderbad Bangalore | 15000 20000 30000 | 4 4 5 |
| Out[84]: In [86]: | 3 4 5 | Umar Jane Uttam | Dataanalyst Analytics Statistics NLP | 50 50 67 | Bangalore Hyderbad Bangalore | 15000 20000 30000 | 4 4 5 |
| | 3 4 5 | Umar Jane Uttam Kim | Dataanalyst Analytics Statistics NLP | 50 50 67 55 | Bangalore Hyderbad Bangalore | 15000 20000 30000 60000 | 4 4 5 10 |

split the data into idependent var adn dependent var

```
In [91]: x_iv = clean_data.drop(['Salary'],axis=1) #salary is dependent var so if we drop
In [88]: x_iv
Out[88]:
                                     Location Exp
             Name
                       Domain Age
              Mike Datascience
                                  34
                                        Mumbai
                                                   2
             Teddy
                                  45
                                      Bangalore
                                                   3
                        Testing
          2
                     Dataanalyst
                                      Bangalore
              Umar
                                  50
                                                   4
          3
                       Analytics
                                      Hyderbad
              Jane
                                  50
                       Statistics
                                      Bangalore
                                                   5
             Uttam
                                  67
                           NLP
                                  55
                                          Delhi
                                                  10
               Kim
In [89]: x_iv.columns
```

```
Out[89]: Index(['Name', 'Domain', 'Age', 'Location', 'Exp'], dtype='object')
         clean_data.columns
In [90]:
Out[90]: Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
In [92]: y_dv= clean_data.drop(['Name', 'Domain', 'Age', 'Location', 'Exp'],axis=1)
Out[92]:
             Salary
              5000
             10000
             15000
          2
             20000
             30000
             60000
In [93]: x_iv
                                     Location Exp
Out[93]:
             Name
                       Domain Age
              Mike
                    Datascience
                                       Mumbai
                                                  2
                                  34
                                                  3
             Teddy
                                      Bangalore
                        Testing
                                  45
                                      Bangalore
          2
             Umar
                    Dataanalyst
                                  50
                                                  4
                                      Hyderbad
          3
              Jane
                       Analytics
                                  50
                                                  4
             Uttam
                       Statistics
                                  67
                                      Bangalore
                                                  5
          5
               Kim
                           NLP
                                  55
                                          Delhi
                                                 10
In [94]: y_dv
Out[94]:
             Salary
          0
              5000
             10000
             15000
          2
             20000
          3
             30000
             60000
In [95]:
         clean_data
```

| ut[95]: | | Name | Do | omain | Age | Location | Salary | Ехр | | | |
|--------------------|-------------|-----------------|--------------------------------------|------------------|--------|------------------------|----------------------|--------|--------------------------|-------------|----------|
| | 0 | Mike | Dataso | cience | 34 | Mumbai | 5000 | 2 | | | |
| | 1 | Teddy | Te | esting | 45 | Bangalore | 10000 | 3 | | | |
| | 2 | Umar | Dataa | nalyst | 50 | Bangalore | 15000 | 4 | | | |
| | 3 | Jane | Ana | alytics | 50 | Hyderbad | 20000 | 4 | | | |
| | 4 | Uttam | Sta | atistics | 67 | Bangalore | 30000 | 5 | | | |
| | 5 | Kim | | NLP | 55 | Delhi | 60000 | 10 | | | |
| n [97]: | | | | l.get_d | dummie | es(clean_da | ata,dty _l | pe=int | :) | | |
| | | putatio | on | | | | | | | Name Teddy | Name Uma |
| | | putatio | on | | | | | | | Name_Teddy | |
| n [97]: ut[97]: | im | putation Age | Salary | Ехр | | Jane Nam | ne_Kim | | _Mike | | |
| | im 0 | Age 34 | Salary 5000 | Exp 2 | | _ Jane Nam 0 | ne_Kim 0 | | _ Mike | 0 | |
| | 0 1 | Age 34 45 50 | Salary 5000 10000 | Exp 2 3 | | Jane Nam 0 0 | 0 0 | | _ Mike 1 0 | 0 | |
| | 0 1 2 | Age 34 45 50 | Salary 5000 10000 | 2 3 4 | | Jane Nam 0 0 0 | 0 0 0 | | _ Mike 1 0 0 | 0 1 0 | Name_Uma |
| | 0 1 2 | Age 34 45 50 50 | Salary 5000 10000 15000 20000 | 2 3 4 4 | | Jane Nam 0 0 0 1 | 0 0 0 | | _ Mike 1 0 0 0 | 0 1 0 | |

EDA Workshop Completed

In []: