### 20th May

```
import pandas as pd
In [1]:
         pd.__version__
In [2]:
Out[2]:
         '2.2.2'
In [3]: # !pip install---upgrade openpyxl
In [4]: emp=pd.read_excel(r"C:\Users\J. Rajesh\Downloads\Rawdata.xlsx")
In [5]:
         emp
Out[5]:
             Name
                           Domain
                                       Age
                                             Location
                                                          Salary
                                                                    Exp
         0
              Mike
                     Datascience#$ 34 years
                                                                     2+
                                              Mumbai
                                                         5^00#0
           Teddy^
                            Testing
                                      45' yr
                                            Bangalore
                                                       10%%000
                                                                     <3
                    Dataanalyst^^#
            Uma#r
                                      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^$0
                                                                    10+
In [6]:
         id(emp)
Out[6]: 2388361076400
In [7]:
         emp.columns
Out[7]: Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
In [8]:
        emp.shape
Out[8]: (6, 6)
In [9]:
        emp.head()
Out[9]:
                           Domain
             Name
                                       Age
                                             Location
                                                          Salary
                                                                     Exp
              Mike
                     Datascience#$
                                                                     2+
                                   34 years
                                              Mumbai
                                                         5^00#0
            Teddy^
                                            Bangalore
                           Testing
                                      45' yr
                                                       10%%000
                                                                     <3
                    Dataanalyst^^#
                                                       1$5%000
            Uma#r
                                      NaN
                                                 NaN
                                                                  4> yrs
                       Ana^^lytics
              Jane
                                     NaN Hyderbad
                                                         2000^0
                                                                    NaN
            Uttam*
                          Statistics
                                      67-yr
                                                 NaN
                                                         30000-
                                                                 5+ year
```

```
In [10]: emp.tail()
```

Out[10]:		Name	Domain	Age	Location	Salary	Ехр
	1	Teddy^	Testing	45' yr	Bangalore	10%%000	<3
	<b>2</b> Uma		Dataanalyst^^#	NaN	NaN	1\$5%000	4> yrs
	3	Jane	Ana^^lytics	NaN	Hyderbad	2000^0	NaN
	4	Uttam*	Statistics	67-yr	NaN	30000-	5+ year
	5	Kim	NLP	55vr	Delhi	6000^\$0	10+

In [11]: 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	object
2	Age	4 non-null	object
3	Location	4 non-null	object
4	Salary	6 non-null	object
5	Exp	5 non-null	object
d+vn	oc. object	(6)	

dtypes: object(6)

memory usage: 420.0+ bytes

In [12]: emp

Out[12]:

	Name	Domain	Age Location		Salary	Ехр
0	Mike	Datascience#\$	34 years	Mumbai	5^00#0	2+
1	Teddy^	Testing	45' yr	Bangalore	10%%000	<3
2	Uma#r	Dataanalyst^^#	NaN	NaN	1\$5%000	4> yrs
3	Jane	Ana^^lytics	NaN	Hyderbad	2000^0	NaN
4	Uttam*	Statistics	67-yr	NaN	30000-	5+ year
5	Kim	NLP	55yr	Delhi	6000^\$0	10+

In [13]: emp.isnull() # it will give the true in the missing values

Out[13]:		Name	Domain	Age	Location	Salary	Ехр	_	
	0	False	False	False	False	False	False		
	1	False	False	False	False	False	False		
	2	False	False	True	True	False	False		
	3	False	False	True	False	False	True		
	4	False	False	False	True	False	False		
	5	False	False	False	False	False	False		
[14].		n :	1/\ =/\						
[14]:	em	p.isnui	1().sum()	)					
ut[14]:	Do Ag Lo Sa Ex	cation lary	0 0 2 2 0 1						
L5]:	em	p.colum	ns						
[15]:	In	dex(['N	lame', 'Do	omain'	, 'Age',	'Locatio	on',	'Salary',	'Exp'],
[16]:	em	р							
t[16]:		Name	ı	Domain	Age	Locat	ion	Salary	Ехр
	0	Mike	Datasc	ience#\$	34 years	Mum	bai	5^00#0	2+
	1	Teddy^		Testing	45' yr	Bangal	ore 1	0%%000	<3
	2	Uma#r	Dataana	lyst^^#	NaN	N	laN	1\$5%000	4> yrs
	3	Jane	Ana	^^lytics	. NaN	Hyderk	oad	2000^0	NaN
	4	Uttam*	S	Statistics	67-yr	N	laN	30000-	5+ year
	5	Kim		NLP	55yr	D	elhi	6000^\$0	10+

# **Data Cleaning or Data Cleansing**

In [17]: emp

```
Out[17]:
              Name
                            Domain
                                         Age
                                               Location
                                                            Salary
                                                                       Exp
          0
               Mike
                       Datascience#$ 34 years
                                                Mumbai
                                                           5^00#0
                                                                        2+
          1
             Teddy^
                             Testing
                                        45' yr Bangalore
                                                         10%%000
                                                                        <3
              Uma#r Dataanalyst^^#
          2
                                         NaN
                                                   NaN
                                                          1$5%000
                                                                     4> yrs
          3
                Jane
                         Ana^^lytics
                                         NaN Hyderbad
                                                           2000^0
                                                                      NaN
          4
              Uttam*
                            Statistics
                                        67-yr
                                                   NaN
                                                            30000-
                                                                    5+ year
          5
                 Kim
                                NLP
                                         55yr
                                                   Delhi
                                                          6000^$0
                                                                       10+
In [18]:
          emp['Name']
Out[18]:
          0
                  Mike
          1
                Teddy^
          2
                 Uma#r
          3
                  Jane
                Uttam*
          4
                   Kim
          Name: Name, dtype: object
In [19]:
          emp['Name']=emp['Name'].str.replace(r'\W','',regex=True) # to remove all the sp
In [20]:
          emp['Name']
Out[20]:
          0
                 Mike
          1
                Teddy
          2
                 Umar
          3
                 Jane
          4
                Uttam
                  Kim
          Name: Name, dtype: object
In [21]:
          emp
Out[21]:
             Name
                           Domain
                                        Age
                                               Location
                                                           Salary
                                                                      Ехр
          0
              Mike
                      Datascience#$
                                    34 years
                                                          5^00#0
                                                                       2+
                                               Mumbai
                                              Bangalore
              Teddy
                            Testing
                                       45' yr
                                                         10%%000
                                                                       <3
          2
              Umar
                     Dataanalyst^^#
                                                   NaN
                                        NaN
                                                         1$5%000
                                                                    4> yrs
                        Ana^^lytics
          3
               Jane
                                        NaN
                                              Hyderbad
                                                          2000^0
                                                                      NaN
                           Statistics
                                                           30000-
                                                                   5+ year
          4
             Uttam
                                       67-yr
                                                   NaN
                               NLP
                                                         6000^$0
          5
                Kim
                                        55yr
                                                  Delhi
                                                                      10+
          emp['Domain']
In [22]:
```

```
Out[22]: 0
              Datascience#$
          1
                     Testing
          2
              Dataanalyst^^#
                 Ana^^lytics
          3
                   Statistics
          4
                          NLP
          Name: Domain, dtype: object
In [23]: emp['Domain']=emp['Domain'].str.replace(r'\W','',regex=True)
         emp['Domain']
Out[23]: 0
               Datascience
                  Testing
          2
             Dataanalyst
          3
                Analytics
                Statistics
          4
                       NLP
          Name: Domain, dtype: object
         emp['Age']=emp['Age'].str.replace(r'\W','',regex=True)
In [24]:
         emp['Age']
Out[24]: 0
               34years
          1
                  45yr
          2
                   NaN
          3
                   NaN
          4
                  67yr
          5
                  55yr
          Name: Age, dtype: object
In [25]: emp['Age']=emp['Age'].str.extract('(\d+)') #to remove the irregular format
         emp['Age']
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        C:\Users\J. Rajesh\AppData\Local\Temp\ipykernel 13564\1973738011.py:1: SyntaxWarn
        ing: invalid escape sequence '\d'
          emp['Age']=emp['Age'].str.extract('(\d+)') #to remove the irregular format
Out[25]: 0
                34
                45
          1
          2
               NaN
          3
              NaN
          4
                67
                55
          5
          Name: Age, dtype: object
In [26]:
         emp
```

```
Out[26]:
             Name
                        Domain Age
                                       Location
                                                    Salary
                                                               Exp
          0
              Mike Datascience
                                  34
                                        Mumbai
                                                   5^00#0
                                                                2+
             Teddy
                         Testing
                                  45 Bangalore
                                                 10%%000
                                                                <3
                     Dataanalyst NaN
          2
              Umar
                                           NaN
                                                  1$5%000
                                                             4> yrs
          3
              Jane
                       Analytics
                                 NaN
                                       Hyderbad
                                                   2000^0
                                                              NaN
             Uttam
                       Statistics
                                           NaN
                                                    30000- 5+ year
          4
                                  67
          5
               Kim
                           NLP
                                   55
                                           Delhi
                                                  6000^$0
                                                               10+
          emp['Salary']=emp['Salary'].str.replace(r'\W','',regex=True)
In [30]:
          emp['Salary']
In [31]:
Out[31]:
          0
                 5000
                10000
          1
          2
                15000
          3
                20000
          4
                30000
          5
               60000
          Name: Salary, dtype: object
In [32]:
Out[32]:
             Name
                                       Location Salary
                        Domain
                                 Age
                                                            Exp
              Mike
          0
                    Datascience
                                  34
                                        Mumbai
                                                   5000
                                                             2+
             Teddy
                         Testing
                                  45
                                       Bangalore
                                                  10000
                                                             <3
          1
          2
                     Dataanalyst
                                                  15000
                                                          4> yrs
              Umar
                                 NaN
                                           NaN
          3
               Jane
                       Analytics
                                 NaN
                                       Hyderbad
                                                  20000
                                                            NaN
          4
             Uttam
                       Statistics
                                   67
                                           NaN
                                                  30000
                                                         5+ year
          5
                           NLP
               Kim
                                   55
                                           Delhi
                                                  60000
                                                            10+
In [33]:
          emp['Exp']
Out[33]:
          0
                     2+
          1
                     <3
          2
                 4> yrs
          3
                    NaN
          4
                5+ year
          5
                    10+
          Name: Exp, dtype: object
In [34]:
          emp['Exp']=emp['Exp'].str.extract('(\d+)')
          emp['Exp']
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        C:\Users\J. Rajesh\AppData\Local\Temp\ipykernel_13564\2863867557.py:1: SyntaxWarn
        ing: invalid escape sequence '\d'
          emp['Exp']=emp['Exp'].str.extract('(\d+)')
```

```
2
Out[34]:
                  3
           1
                  4
           3
                NaN
                  5
           4
           5
                 10
           Name: Exp, dtype: object
In [35]:
Out[35]:
              Name
                        Domain
                                  Age
                                         Location
                                                  Salary
                                                           Exp
           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 [36]:
          clean_data=emp.copy()
In [37]:
          clean_data
Out[37]:
              Name
                        Domain
                                  Age
                                         Location
                                                  Salary
                                                           Exp
               Mike Datascience
                                         Mumbai
                                                    5000
                                                              2
           0
                                    34
                                        Bangalore
                                                   10000
                                                              3
              Teddy
                          Testing
                                    45
                                                              4
           2
              Umar
                      Dataanalyst
                                  NaN
                                             NaN
                                                   15000
           3
               Jane
                        Analytics
                                  NaN
                                        Hyderbad
                                                   20000
             Uttam
                                                   30000
                                                              5
                        Statistics
                                    67
                                             NaN
```

### %% md

Kim

NLP

55

5

till now we have raw data we use regex to clean the data and removed all noise characted from the dataset

60000

10

you can also work in same things in sql query as well

```
In [38]: clean_data['Age']
```

Delhi

```
Out[38]: 0
                                                          34
                                                         45
                                    1
                                    2
                                                      NaN
                                    3
                                                      NaN
                                    4
                                                         67
                                    5
                                                          55
                                    Name: Age, dtype: object
In [40]: import numpy as np
                                clean_data['Age']=clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.mean(pd.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age']).fillna(np.to_numeric(clean_data['Age'])).fillna(np.to_numeric
In [41]:
In [42]: clean_data['Age']
Out[42]: 0
                                                                 34
                                    1
                                                                 45
                                    2
                                                      50.25
                                    3
                                                      50.25
                                    4
                                                                 67
                                                                 55
                                    Name: Age, dtype: object
In [43]:
                                  clean_data['Exp']
                                   0
                                                             2
Out[43]:
                                    1
                                                             3
                                    2
                                                             4
                                    3
                                                      NaN
                                    4
                                                             5
                                                          10
                                    Name: Exp, dtype: object
                                  clean_data['Exp']=clean_data['Exp'].fillna(np.mean(pd.to_numeric(clean_data['Exp
                                   clean_data['Exp']
Out[44]: 0
                                                             2
                                    1
                                                             3
                                    2
                                                             4
                                    3
                                                      4.8
                                    4
                                                             5
                                                          10
                                    Name: Exp, dtype: object
In [45]:
                                  clean data
Out[45]:
                                              Name
                                                                                  Domain
                                                                                                                    Age
                                                                                                                                         Location Salary Exp
                                    0
                                                 Mike
                                                                       Datascience
                                                                                                                         34
                                                                                                                                            Mumbai
                                                                                                                                                                                 5000
                                                                                                                                                                                                             2
                                               Teddy
                                                                                                                                                                             10000
                                                                                                                                                                                                             3
                                                                                      Testing
                                                                                                                         45
                                                                                                                                       Bangalore
                                    2
                                                                        Dataanalyst
                                                                                                                                                                             15000
                                                Umar
                                                                                                                 50.25
                                                                                                                                                        NaN
                                                                                                                                                                                                             4
                                    3
                                                                                Analytics 50.25
                                                   Jane
                                                                                                                                        Hyderbad
                                                                                                                                                                             20000
                                                                                                                                                                                                        4.8
                                                                                                                                                                                                             5
                                                                                                                                                                             30000
                                    4
                                              Uttam
                                                                                 Statistics
                                                                                                                         67
                                                                                                                                                        NaN
                                                                                              NLP
                                    5
                                                                                                                          55
                                                                                                                                                                             60000
                                                                                                                                                                                                          10
                                                     Kim
                                                                                                                                                      Delhi
```

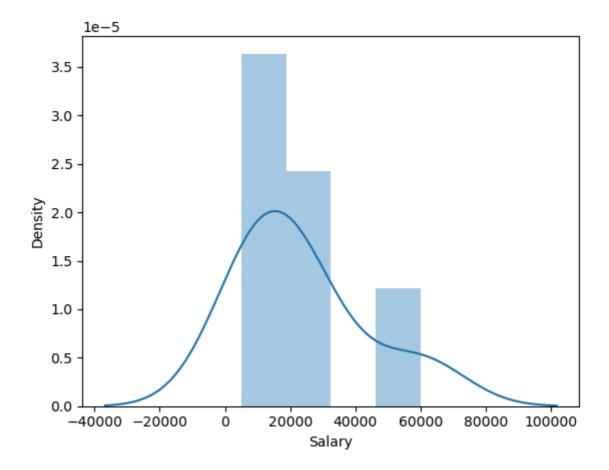
```
clean_data['Location'].isnull().sum()
Out[46]: 2
In [47]:
          clean_data['Location']
Out[47]:
                  Mumbai
               Bangalore
          1
          2
                      NaN
          3
                Hyderbad
          4
                      NaN
          5
                   Delhi
          Name: Location, dtype: object
In [48]:
          clean_data['Location'] = clean_data['Location'].fillna(clean_data['Location'].mo
          clean_data['Location']
Out[48]:
          0
                  Mumbai
          1
               Bangalore
          2
               Bangalore
          3
                Hyderbad
          4
               Bangalore
          5
                   Delhi
          Name: Location, dtype: object
In [49]:
          clean_data
Out[49]:
             Name
                       Domain
                                 Age
                                        Location Salary Exp
              Mike Datascience
                                                   5000
          0
                                   34
                                        Mumbai
                                                           2
          1
             Teddy
                        Testing
                                   45
                                       Bangalore
                                                  10000
                                                           3
          2
              Umar
                    Dataanalyst
                                50.25
                                       Bangalore
                                                  15000
                                                           4
          3
                                50.25
              Jane
                       Analytics
                                       Hyderbad
                                                  20000
                                                          4.8
          4
                                                           5
             Uttam
                       Statistics
                                       Bangalore
                                                  30000
                                   67
          5
               Kim
                           NLP
                                   55
                                                  60000
                                           Delhi
                                                          10
In [50]:
          clean_data.info()
         <class 'pandas.core.frame.DataFrame'>
        RangeIndex: 6 entries, 0 to 5
        Data columns (total 6 columns):
                        Non-Null Count Dtype
         #
              Column
         _ _ _
             _____
         0
             Name
                        6 non-null
                                         object
         1
             Domain
                        6 non-null
                                         object
                        6 non-null
                                         object
         2
              Age
              Location 6 non-null
                                         object
                        6 non-null
                                         object
         4
              Salary
         5
                        6 non-null
                                         object
              Exp
        dtypes: object(6)
        memory usage: 420.0+ bytes
In [51]:
          clean_data['Age'] = clean_data['Age'].astype(int)
```

```
In [52]: clean_data.info()
       <class 'pandas.core.frame.DataFrame'>
       RangeIndex: 6 entries, 0 to 5
       Data columns (total 6 columns):
           Column Non-Null Count Dtype
           _____
                    -----
          Name
                   6 non-null object
        0
        1 Domain 6 non-null
                                 object
        2 Age 6 non-null int32
3 Location 6 non-null object
        4 Salary 6 non-null
                                 object
                   6 non-null
                                   object
        5 Exp
       dtypes: int32(1), object(5)
       memory usage: 396.0+ bytes
In [53]: clean data['Salary'] = clean data['Salary'].astype(int)
        clean_data['Exp'] = clean_data['Exp'].astype(int)
In [54]: clean_data.info()
       <class 'pandas.core.frame.DataFrame'>
       RangeIndex: 6 entries, 0 to 5
       Data columns (total 6 columns):
        # Column Non-Null Count Dtype
       --- -----
                    -----
           Name 6 non-null object
Domain 6 non-null object
        0
        1 Domain 6 non-null
                   6 non-null
        2 Age
                                  int32
        3 Location 6 non-null
                                  object
        4
          Salary 6 non-null
                                   int32
        5
                   6 non-null
                                   int32
       dtypes: int32(3), object(3)
       memory usage: 348.0+ bytes
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
                  6 non-null
        2 Age
                                   int32
           Location 6 non-null
        3
                                 category
        4 Salary 6 non-null
                                   int32
        5
                     6 non-null
                                   int32
            Exp
       dtypes: category(3), int32(3)
       memory usage: 866.0 bytes
In [58]: clean_data
```

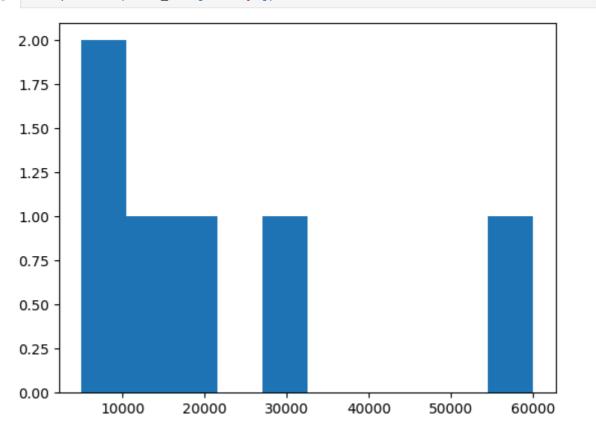
Out[58]:	]: Name		Domain	Age	Location	Salary	Ехр
	0	Mike	Datascience	34	Mumbai	5000	2
	1	Teddy	Testing	45	Bangalore	10000	3
	2	Umar	Dataanalyst	50	Bangalore	15000	4
	3	Jane	Analytics	50	Hyderbad	20000	4
	4	Uttam	Statistics	67	Bangalore	30000	5
	5	Kim	NLP	55	Delhi	60000	10
In [59]:	cl	ean_dat	a.to_csv('c	lean_c	data.csv')		
In [60]:		<b>port</b> os .getcwd					
Out[60]:	'C	:\\User	`s∖\J. Rajes	h'			
In [61]:	cl	ean_dat	a				
Out[61]:		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
	3	Jane Uttam	Analytics Statistics	50 67	Hyderbad Bangalore	20000 30000	5

## LETS APPLY EDA TECHNIQUE

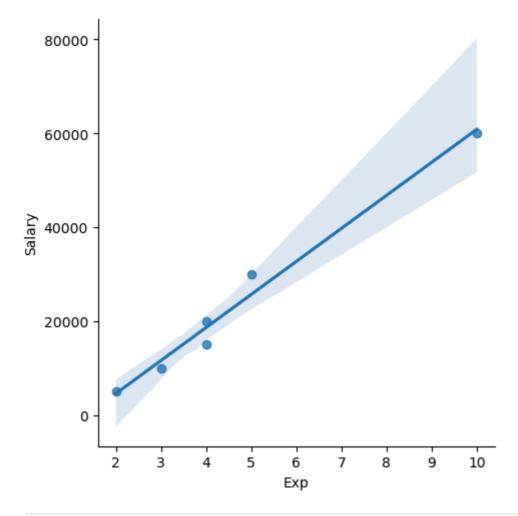
```
In [62]:
         import matplotlib.pyplot as plt # for visualization
         import seaborn as sns
In [63]: import warnings
         warnings.filterwarnings('ignore')
In [64]: clean_data['Salary']
Out[64]: 0
               5000
          1
              10000
          2
              15000
              20000
              30000
              60000
          Name: Salary, dtype: int32
         vis1=sns.distplot(clean_data['Salary'])
In [69]:
```



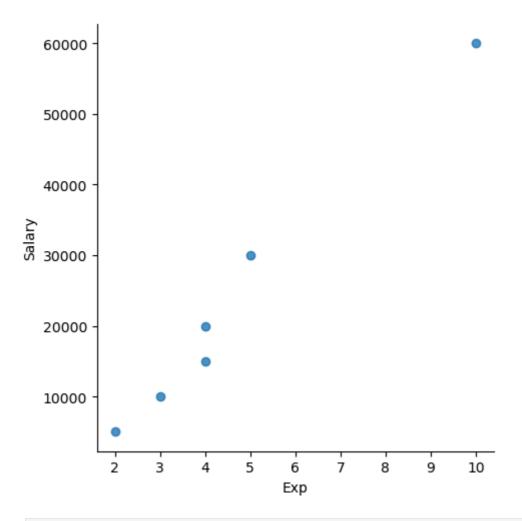
In [71]: vis2=plt.hist(clean\_data['Salary'])



In [72]: vis4=sns.lmplot(data=clean\_data,x='Exp',y='Salary')



In [74]: vis5=sns.lmplot(data=clean\_data,x='Exp',y='Salary',fit\_reg=False)



In [75]: clean\_data[:]

Out[75]:		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 [76]: clean\_data[0:6:2]

Out[76]:		Name	Domain	Age	Location	Salary	Exp
	0	Mike	Datascience	34	Mumbai	5000	2
	2	Umar	Dataanalyst	50	Bangalore	15000	4
	4	Uttam	Statistics	67	Bangalore	30000	5

```
In [78]: clean_data[::-1]
```

```
Out[78]:
             Name
                       Domain Age
                                      Location Salary Exp
                           NLP
          5
               Kim
                                  55
                                          Delhi
                                                 60000
                                                         10
          4 Uttam
                       Statistics
                                  67
                                      Bangalore
                                                 30000
                                                          5
          3
              Jane
                       Analytics
                                  50
                                      Hyderbad
                                                 20000
                                                          4
          2
              Umar
                     Dataanalyst
                                  50
                                      Bangalore
                                                 15000
          1
             Teddy
                        Testing
                                  45
                                      Bangalore
                                                 10000
                                                          3
              Mike
                    Datascience
                                  34
                                        Mumbai
                                                  5000
                                                          2
In [79]:
         clean_data.columns
Out[79]: Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
In [80]: X_iv=clean_data[['Name','Domain','Age','Location','Exp']]
In [81]: X_iv
Out[81]:
                                     Location Exp
             Name
                       Domain Age
              Mike
                    Datascience
                                  34
                                        Mumbai
                                                   2
             Teddy
                        Testing
                                  45
                                      Bangalore
          2
              Umar
                     Dataanalyst
                                  50
                                      Bangalore
          3
              Jane
                       Analytics
                                  50
                                      Hyderbad
          4
             Uttam
                       Statistics
                                  67
                                      Bangalore
                                                   5
          5
               Kim
                           NLP
                                  55
                                          Delhi
                                                  10
         y_dv=clean_data[['Salary']]
In [82]:
In [83]:
         y_dv
Out[83]:
             Salary
              5000
          0
             10000
             15000
          2
          3
             20000
             30000
             60000
In [84]:
          emp
```

Out[84]:		Name	Domain	Age	Location	Salary	Ехр
	0	Mike	Datascience	34	Mumbai	5000 10000 15000 20000 Na 30000	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 [85]: clean\_data

#### Out[85]:

	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 [86]: X\_iv

#### Out[86]:

	Name	Domain	Age	Location	Ехр
0	Mike	Datascience	34	Mumbai	2
1	Teddy	Testing	45	Bangalore	3
2	Umar	Dataanalyst	50	Bangalore	4
3	Jane	Analytics	50	Hyderbad	4
4	Uttam	Statistics	67	Bangalore	5
5	Kim	NLP	55	Delhi	10

In [87]: y\_dv

#### In [88]: clean\_data

#### Out[88]:

	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 [89]: imputation=pd.get\_dummies(clean\_data)

In [90]: imputation

#### Out[90]:

:		Age	Salary	Ехр	Name_Jane	Name_Kim	Name_Mike	Name_Teddy	Name_Umar
	0	34	5000	2	False	False	True	False	False
	1	45	10000	3	False	False	False	True	False
	2	50	15000	4	False	False	False	False	True
	3	50	20000	4	True	False	False	False	False
	4	67	30000	5	False	False	False	False	False
	5	55	60000	10	False	True	False	False	False
	•								•

In [91]: clean\_data

Out[91]:		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 [92]: imputation

A.	 г	$\cap$	$\sim$	п.	
1 11		ч	- /	- 1	

	Age	Salary	Ехр	Name_Jane	Name_Kim	Name_Mike	Name_Teddy	Name_Umar
0	34	5000	2	False	False	True	False	False
1	45	10000	3	False	False	False	True	False
2	50	15000	4	False	False	False	False	True
3	50	20000	4	True	False	False	False	False
4	67	30000	5	False	False	False	False	False
5	55	60000	10	False	True	False	False	False
4								•

### %% md

raw data with lot of regex, missing, uncleandata regex, clean

fill missing numerical & cateigroica clean\_dataset ( data cleaning) 3 month - 5mont outlier treatement, univati, bivariate, corelation split the data into x\_i.v & y\_dv impute cateogrica data to numerical eda part complete

## %% md

## Next step

- we splitn x\_iv -- x\_train, x\_test
- we split y\_dv -- y\_train, y\_test
- build the ml model with x\_train & y\_train

In [ ]: