```
In [1]: import pandas as pd
```

```
In [2]: pd.__version__
```

```
Out[2]: '2.2.2'
```

```
In [3]: pip install --upgrade openpyxl
```

```
Requirement already satisfied: openpyxl in c:\users\hanshu\anaconda3\lib\site-pac
kages (3.1.5)
Requirement already satisfied: et-xmlfile in c:\users\hanshu\anaconda3\lib\site-p
ackages (from openpyxl) (1.1.0)
Note: you may need to restart the kernel to use updated packages.
```

```
In [4]: pd.__version__
```

```
Out[4]: '2.2.2'
```

```
In [5]: emp = pd.read_excel(r"C:\Users\Hanshu\Desktop\excel data\Rawdata.xlsx")
        emp
```

Out[5]:

	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
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 [6]: id(emp)
```

```
Out[6]: 2456864696192
```

```
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]:

	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
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

In [10]:

```
emp.tail()
```

Out[10]:

	Name	Domain	Age	Location	Salary	Exp
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 [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
dtypes: object(6)
memory usage: 420.0+ bytes
```

In [12]:

```
emp
```

Out[12]:

	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
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()
```

Out[13]:

	Name	Domain	Age	Location	Salary	Exp
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

In [14]:

```
emp.isna()
```

Out[14]:

	Name	Domain	Age	Location	Salary	Exp
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

In [15]:

```
emp.isnull().sum()
```

Out[15]:

```
Name        0
Domain      0
Age         2
Location    2
Salary      0
Exp         1
dtype: int64
```

In [16]:

```
emp
```

Out[16]:

	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
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 [17]:

```
emp['Name']
```

Out[17]:

```
0      Mike
1    Teddy^
2     Uma#r
3      Jane
4    Uttam*
5       Kim
Name: Name, dtype: object
```

In [18]:

```
emp['Domain']
```

Out[18]:

```
0     Datascience#$
1           Testing
2    Dataanalyst^^#
3       Ana^^lytics
4        Statistics
5               NLP
Name: Domain, dtype: object
```

In [19]:

```
emp['Age']
```

Out[19]:

```
0    34 years
1      45' yr
2         NaN
3         NaN
4       67-yr
5        55yr
Name: Age, dtype: object
```

In [20]:

```
emp['Salary']
```

Out[20]:

```
0     5^00#0
1    10%%000
2    1$5%000
3     2000^0
4     30000-
5    6000^$0
Name: Salary, dtype: object
```

In [21]:

```
emp['Exp']
```

Out[21]:

```
0         2+
1         <3
2     4> yrs
3        NaN
4    5+ year
5        10+
Name: Exp, dtype: object
```

In [22]:

```
emp
```

Out[22]:

	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
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 [23]:

```
emp[['Name','Domain']]
```

Out[23]:

	Name	Domain
0	Mike	Datascience#$
1	Teddy^	Testing
2	Uma#r	Dataanalyst^^#
3	Jane	Ana^^lytics
4	Uttam*	Statistics
5	Kim	NLP

In [24]:

```
emp[['Name','Domain','Age']]
```

Out[24]:

	Name	Domain	Age
0	Mike	Datascience#$	34 years
1	Teddy^	Testing	45' yr
2	Uma#r	Dataanalyst^^#	NaN
3	Jane	Ana^^lytics	NaN
4	Uttam*	Statistics	67-yr
5	Kim	NLP	55yr

In [25]:

```
emp[['Name','Domain','Age','Location']]
```

Out[25]:

	Name	Domain	Age	Location
0	Mike	Datascience#$	34 years	Mumbai
1	Teddy^	Testing	45' yr	Bangalore
2	Uma#r	Dataanalyst^^#	NaN	NaN
3	Jane	Ana^^lytics	NaN	Hyderbad
4	Uttam*	Statistics	67-yr	NaN
5	Kim	NLP	55yr	Delhi

DATA CLEANING & DATA CLEANSING

In [26]:

```
emp['Name']
```

Out[26]:

```
0      Mike
1    Teddy^
2     Uma#r
3      Jane
4    Uttam*
5       Kim
Name: Name, dtype: object
```

In [27]:

```
emp['Name'] = emp['Name'].str.replace(r'\W','',regex=True) # remove caps and...
emp['Name']
```

Out[27]:

```
0     Mike
1    Teddy
2     Umar
3     Jane
4    Uttam
5      Kim
Name: Name, dtype: object
```

In [28]:

```
emp['Domain'] = emp['Domain'].str.replace(r'\W','',regex=True)
emp['Domain']
```

Out[28]:

```
0    Datascience
1        Testing
2    Dataanalyst
3      Analytics
4     Statistics
5            NLP
Name: Domain, dtype: object
```

In [29]:

```
emp['Age'] = emp['Age'].str.replace(r'\W','',regex=True)
emp['Age']
```

Out[29]:

```
0    34years
1      45yr
2       NaN
3       NaN
4      67yr
5      55yr
Name: Age, dtype: object
```

In [30]:
```python
emp['Age'] = emp['Age'].str.extract('(\\d+)') # extract used for remove categori
emp['Age']
```

Out[30]:
```
0     34
1     45
2    NaN
3    NaN
4     67
5     55
Name: Age, dtype: object
```

In [31]:
```python
emp['Location'] = emp['Location'].str.replace(r'\W','',regex=True)
emp['Location']
```

Out[31]:
```
0       Mumbai
1    Bangalore
2          NaN
3     Hyderbad
4          NaN
5        Delhi
Name: Location, dtype: object
```

In [32]:
```python
emp['Salary']=emp['Salary'].str.replace(r'\W','',regex=True)
emp['Salary']
```

Out[32]:
```
0     5000
1    10000
2    15000
3    20000
4    30000
5    60000
Name: Salary, dtype: object
```

In [33]:
```python
emp['Exp'] = emp['Exp'].str.extract('(\\d+)')
emp['Exp']
```

Out[33]:
```
0      2
1      3
2      4
3    NaN
4      5
5     10
Name: Exp, dtype: object
```

In [34]:
```python
emp
```

Out[34]:

	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 [35]:

```python
clean_data = emp.copy()
clean_data
```

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

EDA TECHNIQUES

Missing Value Treatement

In [36]:

```python
clean_data.isnull().sum()
```

Out[36]:

```
Name        0
Domain      0
Age         2
Location    2
Salary      0
Exp         1
dtype: int64
```

In [37]:

```python
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       4 non-null      object
 3   Location  4 non-null      object
 4   Salary    6 non-null      object
 5   Exp       5 non-null      object
dtypes: object(6)
memory usage: 420.0+ bytes
```

In [38]:

```python
import numpy as np
```

In [39]:

```python
clean_data.head()
```

Out[39]:

	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

In [40]:

```
clean_data['Age']=clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age
```

In [41]:

```
clean_data['Age']
```

Out[41]:

```
0       34
1       45
2    50.25
3    50.25
4       67
5       55
Name: Age, dtype: object
```

In [42]:

```
clean_data['Exp']=clean_data['Exp'].fillna(np.mean(pd.to_numeric(clean_data['Exp
clean_data['Exp']
```

Out[42]:

```
0      2
1      3
2      4
3    4.8
4      5
5     10
Name: Exp, dtype: object
```

In [43]:

```
clean_data
```

Out[43]:

	Name	Domain	Age	Location	Salary	Exp
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	4.8
4	Uttam	Statistics	67	NaN	30000	5
5	Kim	NLP	55	Delhi	60000	10

In [44]:

```
clean_data['Location'].isnull().sum
```

```
Out[44]: <bound method Series.sum of 0    False
1    False
2     True
3    False
4     True
5    False
Name: Location, dtype: bool>
```

```
In [45]: clean_data['Location'].isnull().sum()
```

```
Out[45]: 2
```

```
In [46]: clean_data['Location']
```

```
Out[46]: 0       Mumbai
1    Bangalore
2          NaN
3     Hyderbad
4          NaN
5        Delhi
Name: Location, dtype: object
```

```
In [47]: clean_data['Location'] = clean_data['Location'].fillna(clean_data['Location'].mc
clean_data['Location']
```

```
Out[47]: 0       Mumbai
1    Bangalore
2    Bangalore
3     Hyderbad
4    Bangalore
5        Delhi
Name: Location, dtype: object
```

```
In [48]: clean_data
```

Out[48]:

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000	2
1	Teddy	Testing	45	Bangalore	10000	3
2	Umar	Dataanalyst	50.25	Bangalore	15000	4
3	Jane	Analytics	50.25	Hyderbad	20000	4.8
4	Uttam	Statistics	67	Bangalore	30000	5
5	Kim	NLP	55	Delhi	60000	10

```
In [49]: 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
dtypes: object(6)
memory usage: 420.0+ bytes
```

In [50]:

```
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      object
 3   Location  6 non-null      object
 4   Salary    6 non-null      object
 5   Exp       6 non-null      object
dtypes: object(6)
memory usage: 420.0+ bytes
```

In [51]:

```
clean_data['Age']=clean_data['Age'].astype(int)
clean_data['Age']
```

Out[51]:

```
0    34
1    45
2    50
3    50
4    67
5    55
Name: Age, dtype: int32
```

In [52]:

```
clean_data['Salary']=clean_data['Salary'].astype(int)
clean_data['Salary']
```

Out[52]:

```
0     5000
1    10000
2    15000
3    20000
4    30000
5    60000
Name: Salary, dtype: int32
```

In [53]:

```
clean_data['Exp']=clean_data['Exp'].astype(int)
clean_data['Exp']
```

Out[53]:

```
0     2
1     3
2     4
3     4
4     5
5    10
Name: Exp, dtype: int32
```

In [54]:

```
clean_data
```

Out[54]:

	Name	Domain	Age	Location	Salary	Exp
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 [55]:

```
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 [56]:

```
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
```

In [57]:

```
clean_data
```

Out[57]:

	Name	Domain	Age	Location	Salary	Exp
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 [58]:

```
clean_data.to_csv('clean_data.csv')
```

In [59]:

```
import os
os.getcwd()  # from os to get saved current working directory
```

Out[59]:

```
'C:\\Users\\Hanshu\\basics'
```

EDA TECNIQUES APPLYING

In [60]:

```
import matplotlib.pyplot as plt    # visualiztion
import seaborn as sns
```

In [61]:

```
import warnings
warnings.filterwarnings('ignore')
```

In [62]:

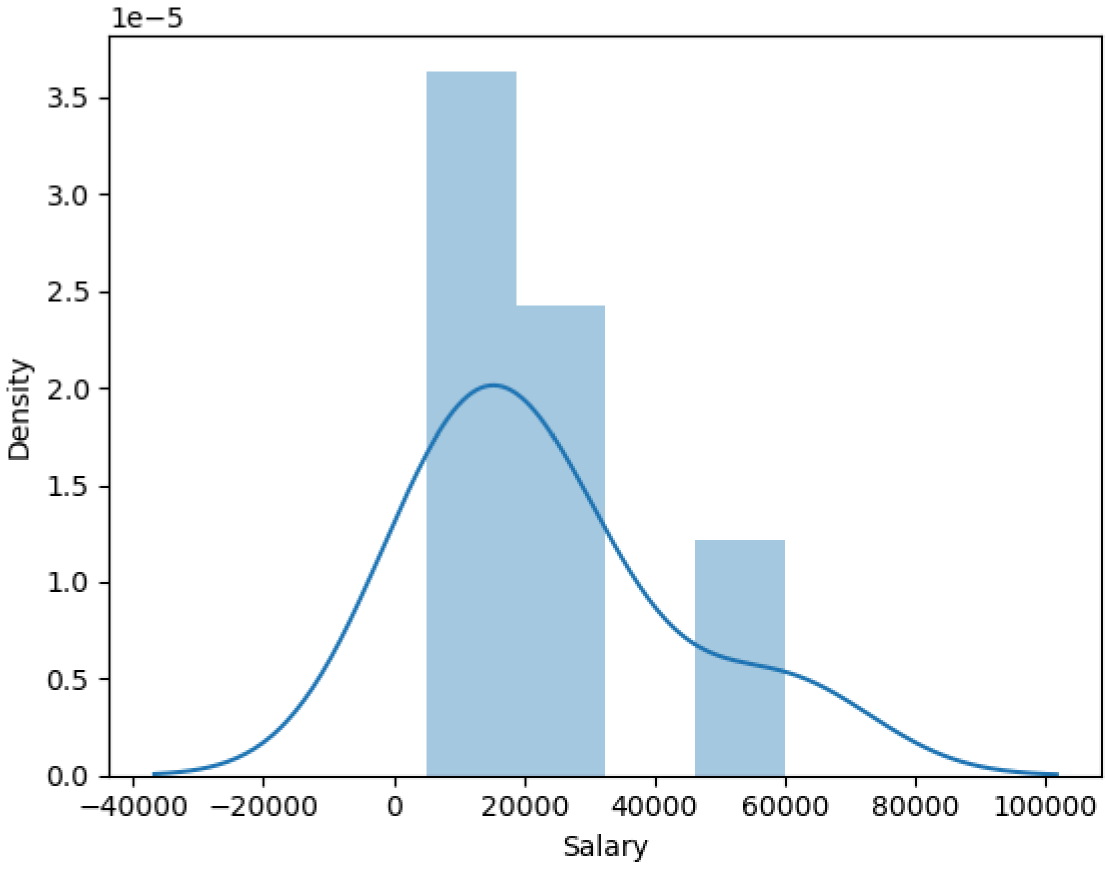
```
clean_data['Salary']
```

Out[62]:

```
0     5000
1    10000
2    15000
3    20000
4    30000
5    60000
Name: Salary, dtype: int32
```

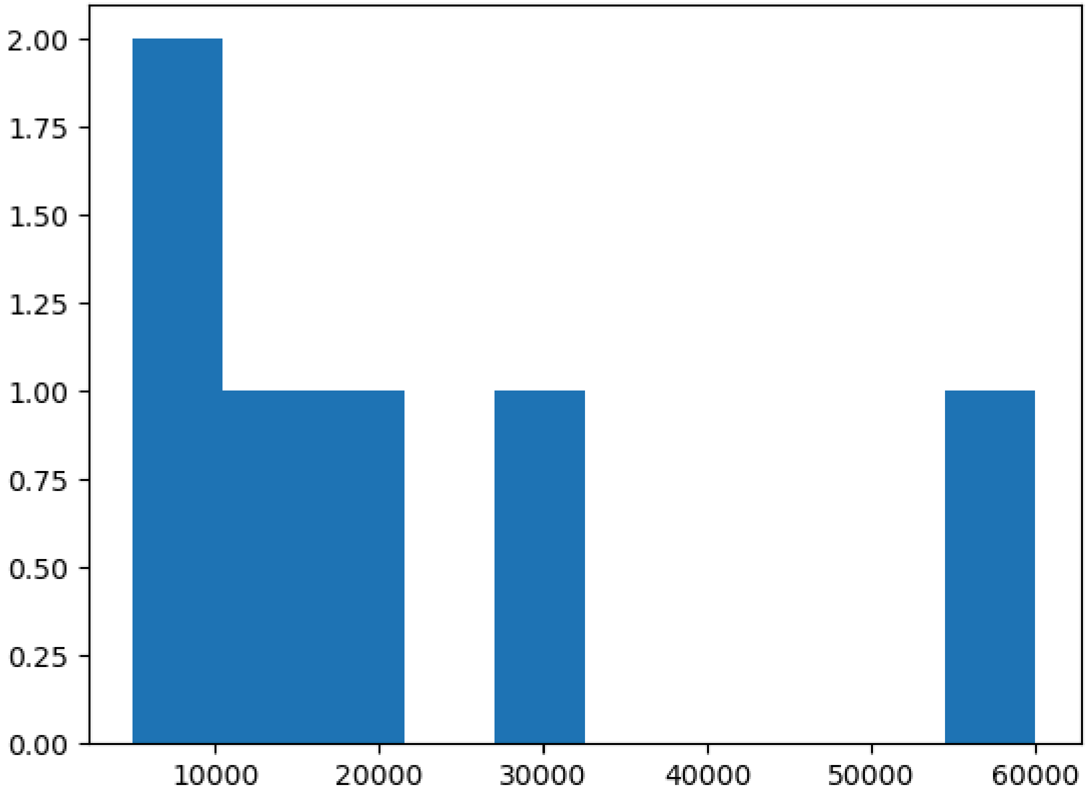
In [63]:

```
vis1 = sns.distplot(clean_data['Salary'])
```

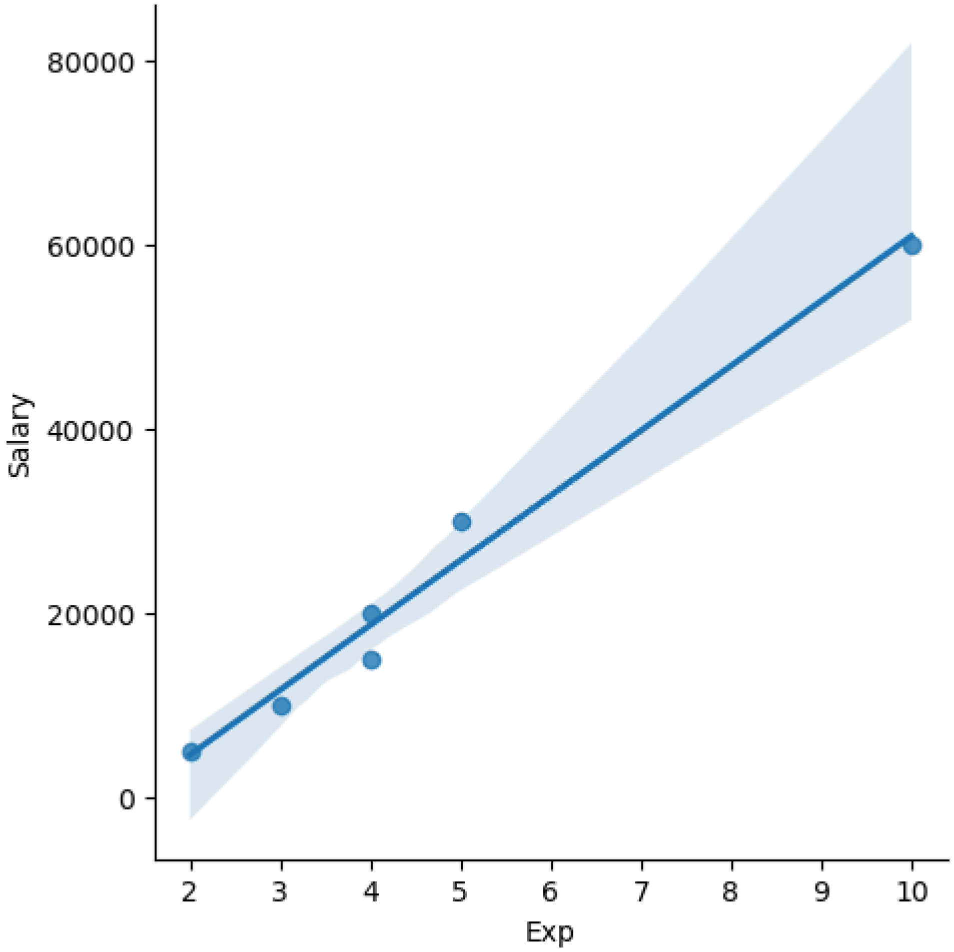


In [64]:

```
vis2 = plt.hist(clean_data['Salary'])
```



```
In [65]: vis4 = sns.lmplot(data=clean_data,x='Exp',y='Salary')
```



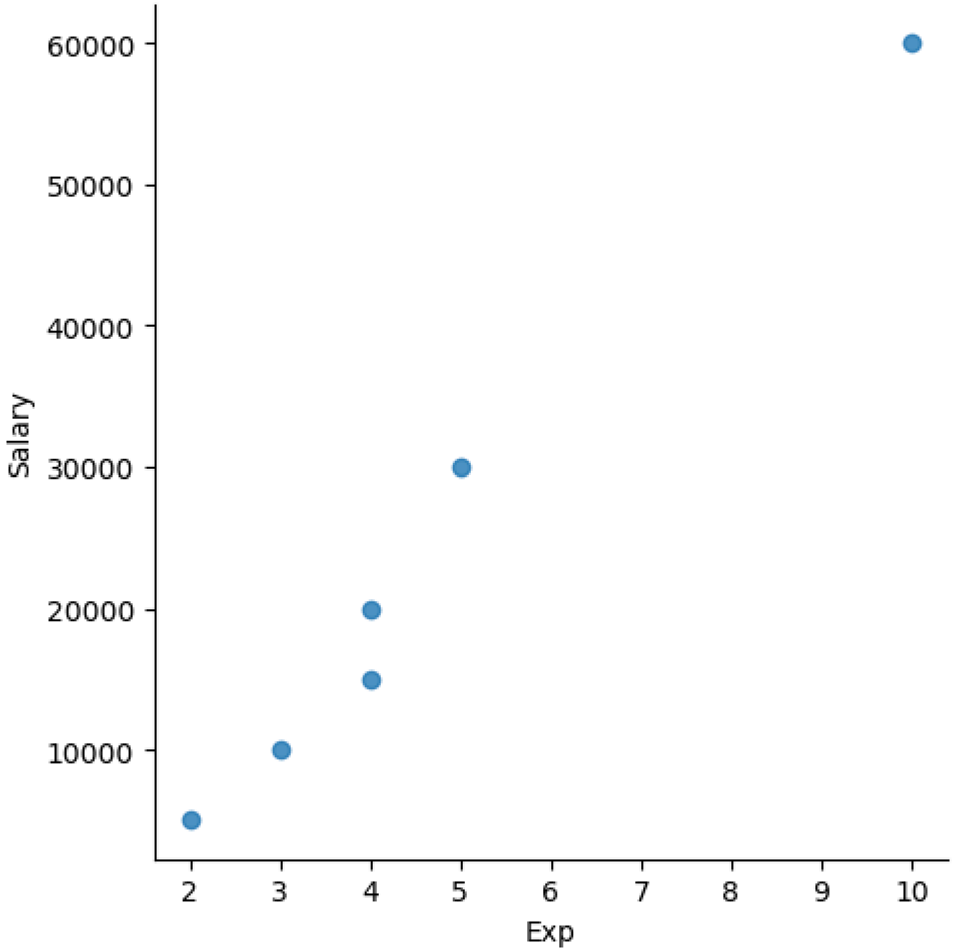
```
In [66]: clean_data
```

Out[66]:

	Name	Domain	Age	Location	Salary	Exp
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 [67]:

```
vis5=sns.lmplot(data=clean_data,x='Exp',y='Salary',fit_reg=False)
```



In [68]:

```
clean_data[:]
```

Out[68]:

	Name	Domain	Age	Location	Salary	Exp
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[0:6:2]
```

Out[69]:

	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 [70]:

```
clean_data[::-1]
```

Out[70]:

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

In [71]:

```
clean_data.columns
```

Out[71]:

```
Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
```

In [73]:

```
x_iv = clean_data[['Name','Domain','Age','Location','Exp']]
```

In [75]:

```
x_iv          # variable identification
```

Out[75]:

	Name	Domain	Age	Location	Exp
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 [76]:

```
y_dv = clean_data[['Salary']]
y_dv
```

Out[76]:

	Salary
0	5000
1	10000
2	15000
3	20000
4	30000
5	60000

In [77]:

```
emp
```

Out[77]:

	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 [78]:

```
clean_data
```

Out[78]:

	Name	Domain	Age	Location	Salary	Exp
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 [79]:

```
x_iv
```

Out[79]:

	Name	Domain	Age	Location	Exp
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 [80]:

```
y_dv
```

Out[80]:

	Salary
0	5000
1	10000
2	15000
3	20000
4	30000
5	60000

In [81]:

```
clean_data
```

Out[81]:

	Name	Domain	Age	Location	Salary	Exp
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 [82]:

```
imputation = pd.get_dummies(clean_data)
```

In [83]:

```
imputation
```

Out[83]:

	Age	Salary	Exp	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 [ ]:

```
clean_
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```