Exploratory Data Analysis

In [1]:

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

In [2]:

```
import pandas as pd
```

In [3]:

```
emp = pd.read_excel("Rawdata.xlsx")
```

In [4]:

```
emp
```

Out[4]:

	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 [5]:

```
emp.shape
```

Out[5]:

```
(6, 6)
```

In [6]:

```
len(emp)
```

Out[6]:

```
6
```

In [7]:

```
emp.columns
```

Out[7]:

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

In [8]:

```
len(emp.columns)
```

Out[8]:

```
6
```

In [9]:

```
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 [10]:
```
emp
```

Out[10]:

	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 [11]:
```
emp['Name']
```

Out[11]:
```
0      Mike
1    Teddy^
2     Uma#r
3      Jane
4    Uttam*
5       Kim
Name: Name, dtype: object
```

In [12]:
```
emp['Domain']
```

Out[12]:
```
0     Datascience#$
1           Testing
2    Dataanalyst^^#
3       Ana^^lytics
4        Statistics
5               NLP
Name: Domain, dtype: object
```

In [13]:
```
emp['Age']
```

Out[13]:
```
0    34 years
1      45' yr
2         NaN
3         NaN
4       67-yr
5        55yr
Name: Age, dtype: object
```

In [14]:
```
emp['Location']
```

Out[14]:
```
0       Mumbai
1    Bangalore
2          NaN
3     Hyderbad
4          NaN
5        Delhi
Name: Location, dtype: object
```

In [15]:
```
emp['Salary']
```

Out[15]:
```
0     5^00#0
1    10%%000
2    1$5%000
3     2000^0
4     30000-
5    6000^$0
Name: Salary, dtype: object
```

In [16]:
```
emp['Exp']
```

Out[16]:
```
0          2+
1          <3
2      4> yrs
3         NaN
4     5+ year
5         10+
Name: Exp, dtype: object
```

In [17]:
```
emp[['Name','Domain']]
```

Out[17]:

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

In [18]:
```
emp[['Name','Domain','Age']]
```

Out[18]:

	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 [19]:
```
emp[['Name','Domain','Age','Location','Salary','Exp']]
```

Out[19]:

	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+

Data Cleansing

In [20]:
```
emp['Name']
```

Out[20]:

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

In [21]:

```
emp['Name'] = emp['Name'].str.replace(r'\W','')
```

In [22]:

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

Out[22]:

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

In [23]:

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

In [24]:

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

Out[24]:

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

In [25]:

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

In [26]:

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

Out[26]:

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

In [27]:

```
emp['Age'] = emp['Age'].str.extract('(\d+)')
```

In [28]:

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

Out[28]:

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

In [29]:

```
emp
```

Out[29]:

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5^00#0	2+
1	Teddy	Testing	45	Bangalore	10%%000	<3
2	Umar	Dataanalyst	NaN	NaN	1$5%000	4> yrs
3	Jane	Analytics	NaN	Hyderbad	2000^0	NaN
4	Uttam	Statistics	67	NaN	30000-	5+ year
5	Kim	NLP	55	Delhi	6000^$0	10+

In [30]:

```python
emp['Location'] = emp['Location'].str.replace(r'\W','')
```

In [31]:

```python
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','')
```

In [33]:

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

Out[33]:

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

In [34]:

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

In [35]:

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

Out[35]:

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

In [36]:

```python
emp
```

Out[36]:

	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 [37]:

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

In [38]:

```
clean_data
```

Out[38]:

	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

Missing Value Treatment

In [39]:

```
clean_data
```

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
5	Kim	NLP	55	Delhi	60000	10

In [40]:

```
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 [41]:

```
import numpy as np
```

In [42]:

```
clean_data
```

Out[42]:

	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 [43]:

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

Out[43]:

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

In [44]:

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

In [45]:

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

Out[45]:

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

In [46]:

```
emp
```

Out[46]:

	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 [48]:
```
clean_data['Exp'] = clean_data['Exp'].fillna(np.mean(pd.to_numeric(clean_data['Exp'
```

In [49]:
```
clean_data['Exp']
```

Out[49]:
```
0       2
1       3
2       4
3     4.8
4       5
5      10
Name: Exp, dtype: object
```

In [50]:
```
clean_data['Location'] = clean_data['Location'].fillna(clean_data['Location'].mode(
```

In [51]:
```
clean_data['Location']
```

Out[51]:
```
0       Mumbai
1    Bangalore
2    Bangalore
3     Hyderbad
4    Bangalore
5        Delhi
Name: Location, dtype: object
```

In [52]:
```
clean_data
```

Out[52]:

	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 [53]:
```
clean_data['Age'] = clean_data['Age'].astype(int)
```

In [54]:
```
clean_data['Salary'] = clean_data['Salary'].astype(int)
```

In [55]:
```
clean_data['Exp'] = clean_data['Exp'].astype(int)
```

In [56]:
```
clean_data
```

Out[56]:

	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 [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      object
 1   Domain    6 non-null      object
 2   Age       6 non-null      int32 
 3   Location  6 non-null      object
 4   Salary    6 non-null      int32 
 5   Exp       6 non-null      int32 
dtypes: int32(3), object(3)
memory usage: 348.0+ bytes
```

In [58]:

```
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 [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
```

In [60]:

```
clean_data
```

Out[60]:

	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 [61]:

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

In [62]:

```
import os
os.getcwd()
```

Out[62]:

```
'C:\\Users\\JANHAVI\\NIT'
```

In [68]:

```
import matplotlib.pyplot as plt # visualization
import seaborn as sns # Advanced visualization
```

In [69]:

```
clean_data.columns
```

Out[69]:

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

In [70]:

```
clean_data
```

Out[70]:

	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 [71]:

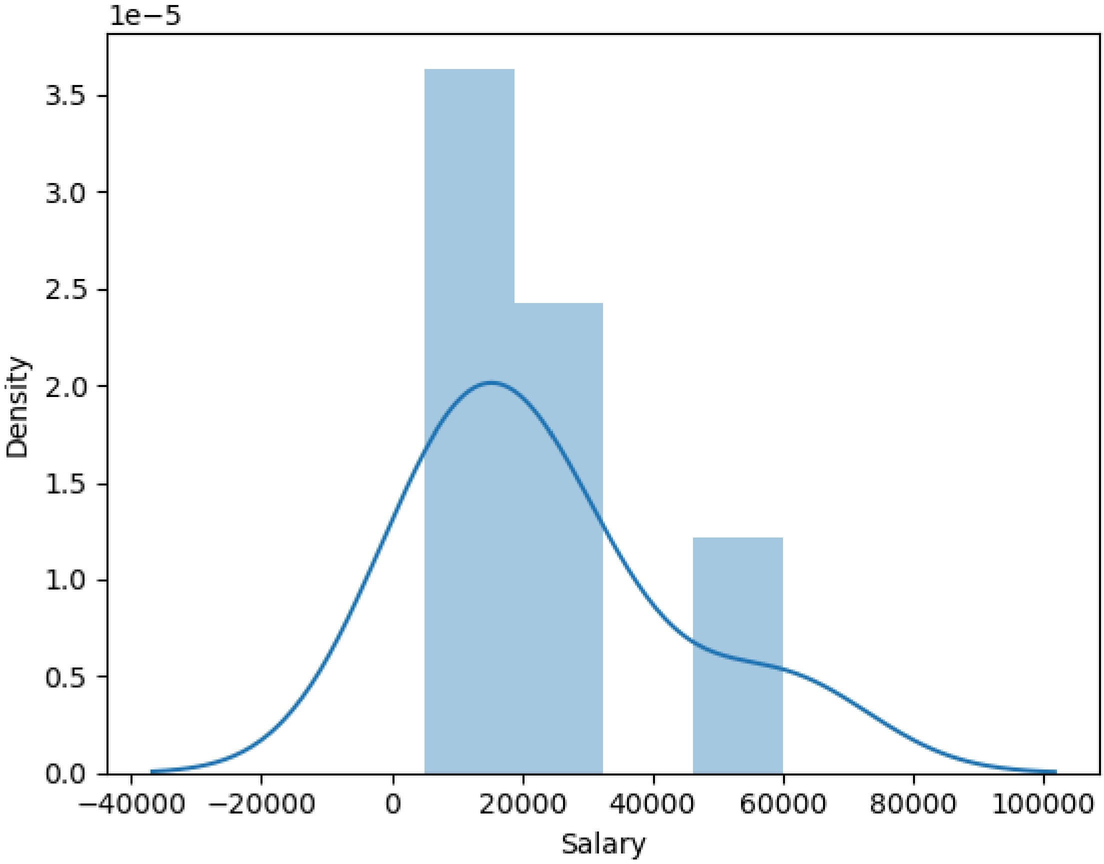
```
clean_data['Salary']
```

Out[71]:

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

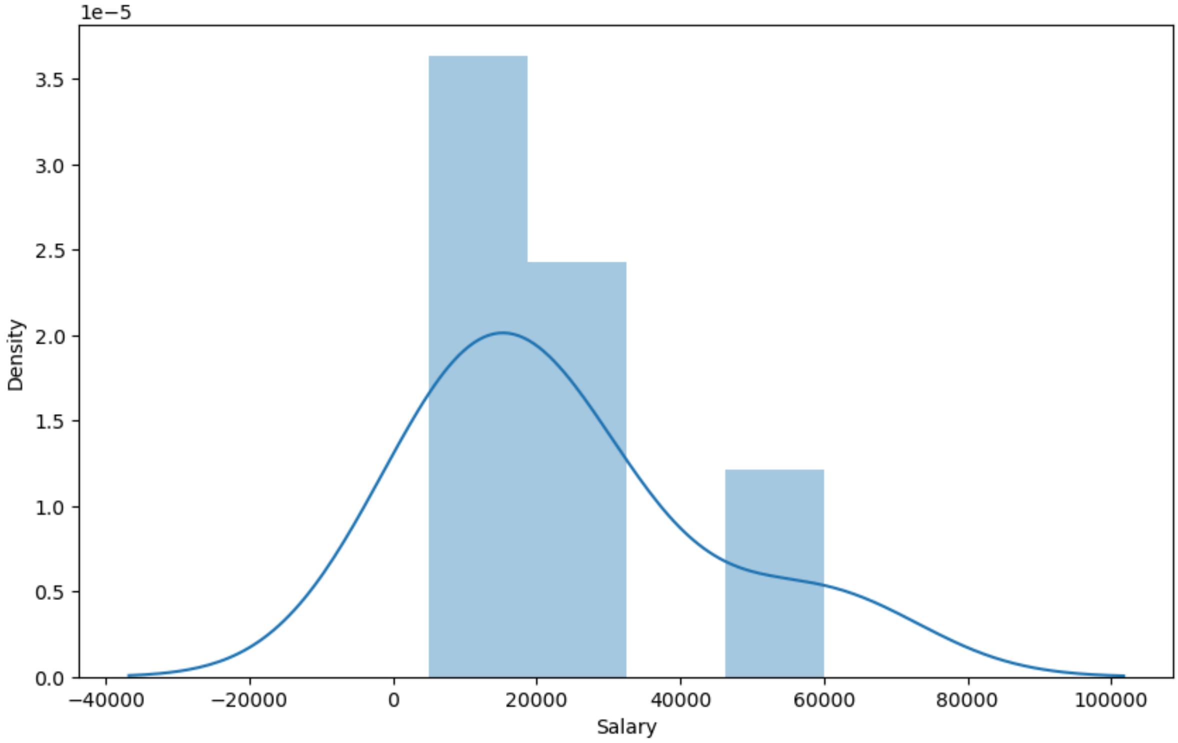
In [72]:

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

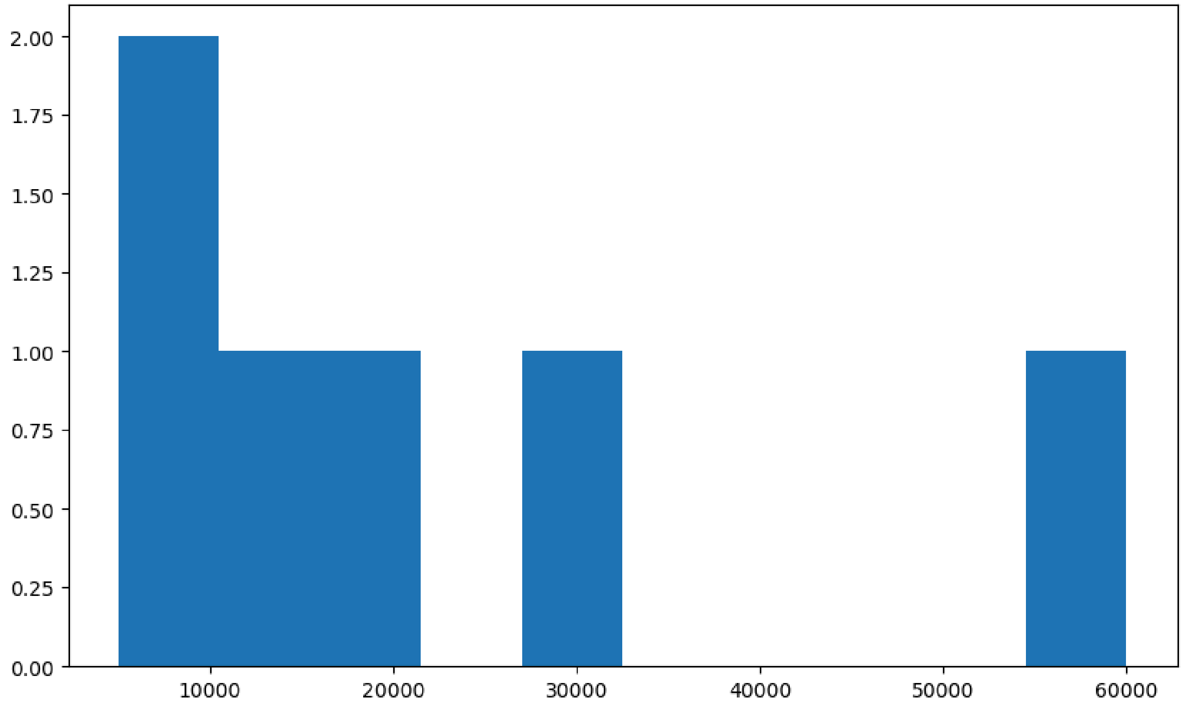


```
In [73]: plt.rcParams['figure.figsize'] = 10,6
```

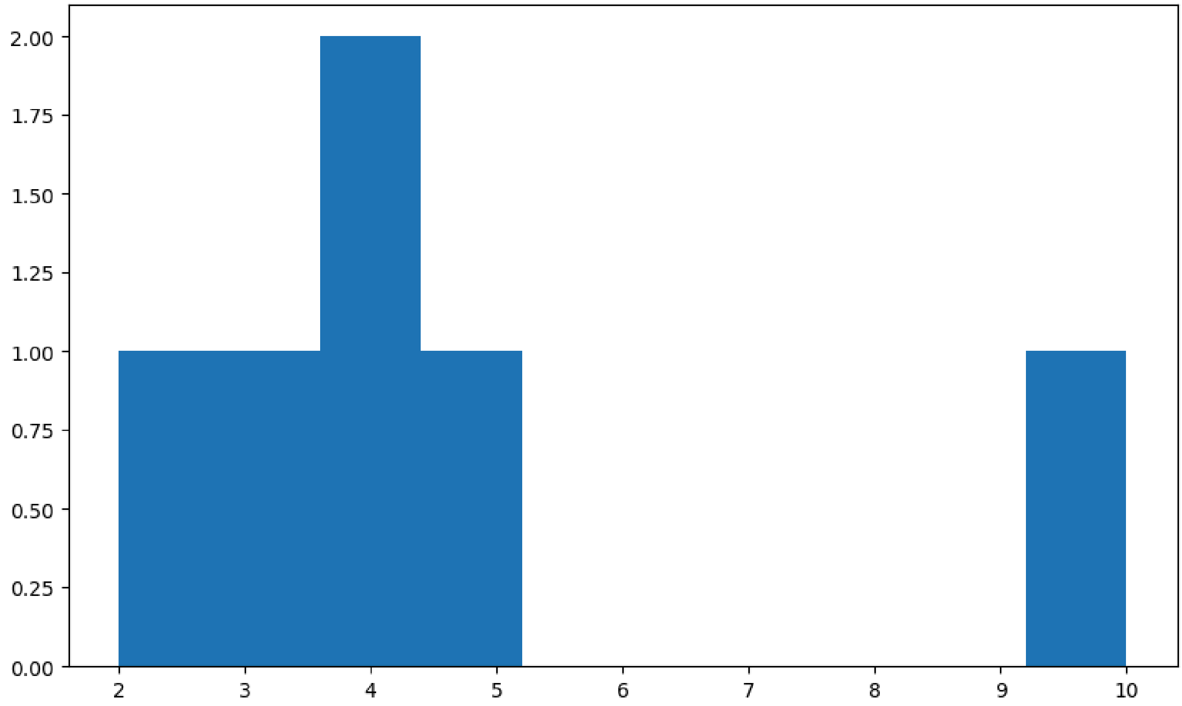
```
In [74]: vis1 = sns.distplot(clean_data['Salary'])
```



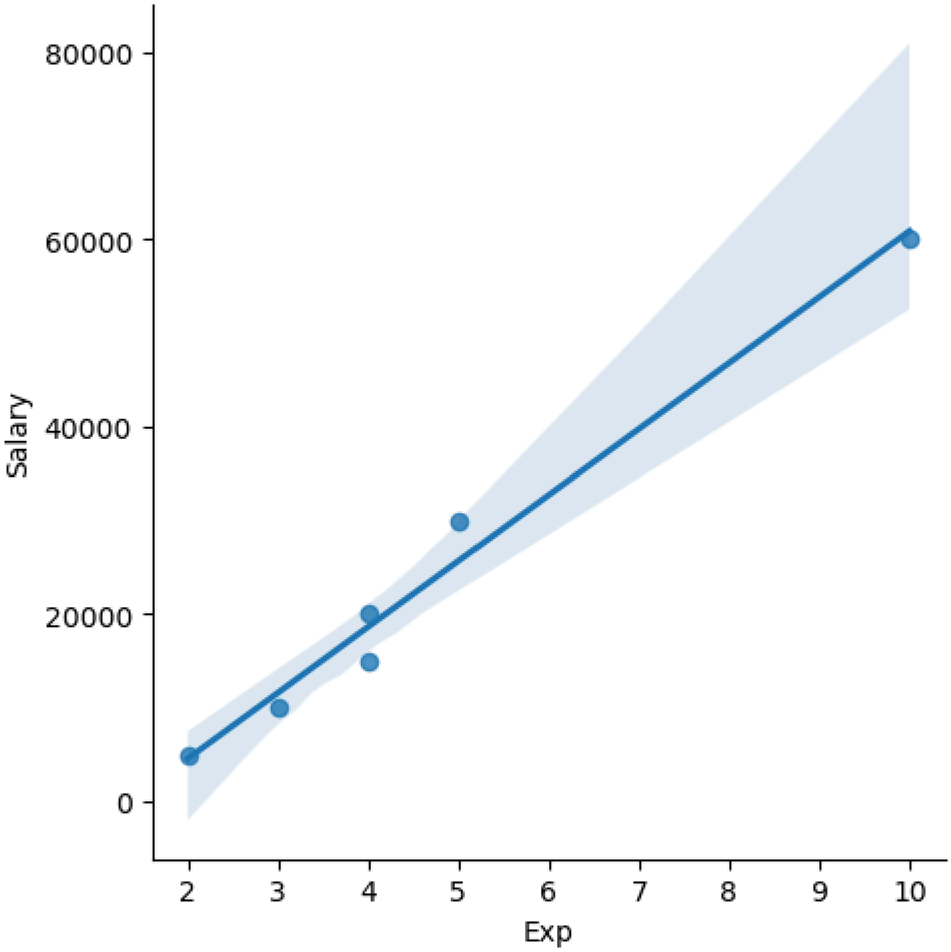
```
In [75]: vis2 = plt.hist(clean_data['Salary'])
```



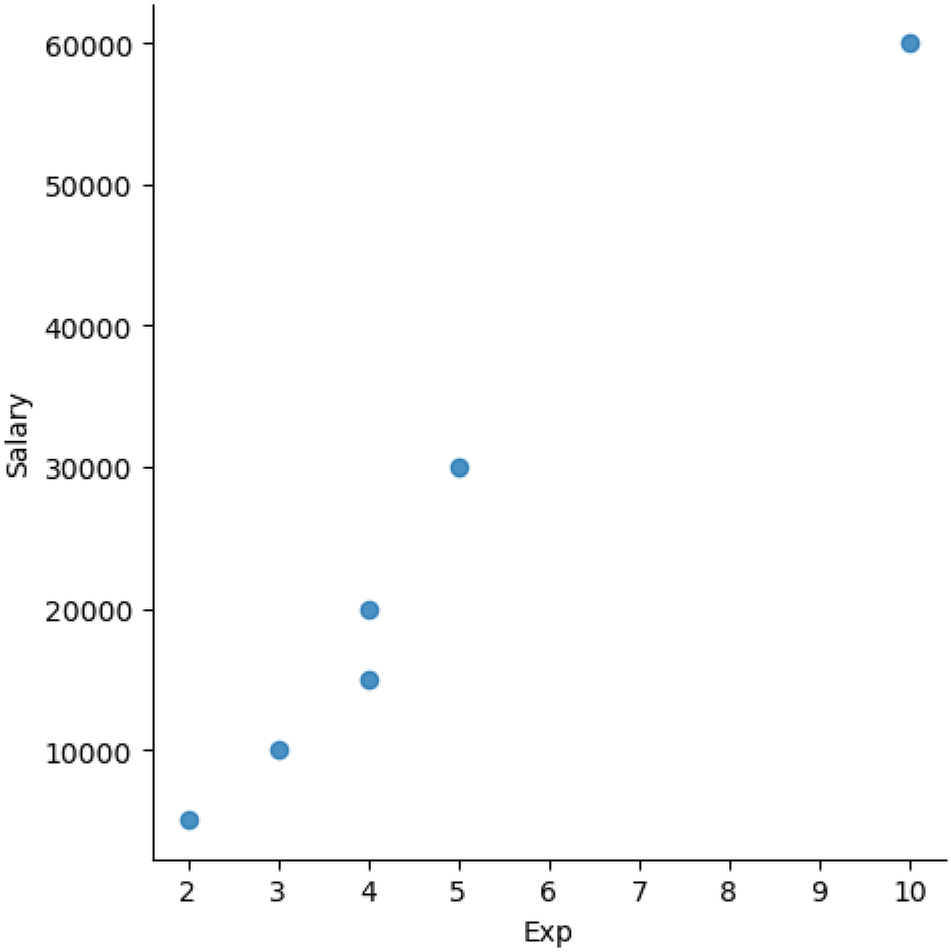
```
In [76]: vis3 = plt.hist(clean_data['Exp'])
```



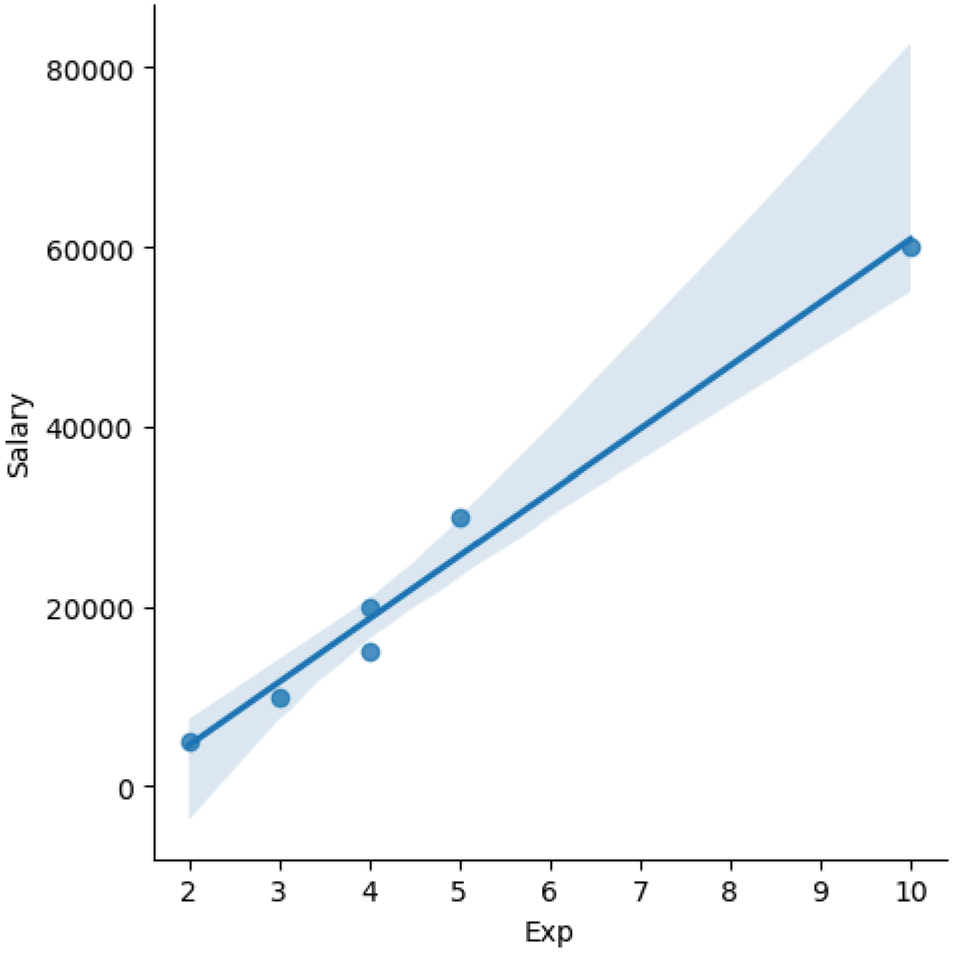
```
In [77]: vis4 = sns.lmplot(data=clean_data,x = 'Exp', y='Salary')
```



```
In [78]: vis5 = sns.lmplot(data=clean_data,x = 'Exp', y='Salary', fit_reg = False)
```



In [79]:
```
vis6 = sns.lmplot(data=clean_data,x = 'Exp', y='Salary', fit_reg = True)
```



In [80]:
```
clean_data
```

Out[80]:

	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 [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]:

```
clean_data[:2]
```

Out[82]:

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000	2
1	Teddy	Testing	45	Bangalore	10000	3

In [83]:

```
clean_data[2:]
```

Out[83]:

	Name	Domain	Age	Location	Salary	Exp
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]:

```
clean_data[:]
```

Out[84]:

	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 [85]:

```
clean_data[0:1]
```

Out[85]:

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000	2

In [88]:

```
clean_data
```

Out[88]:

	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 [89]:

```
x_iv = clean_data.drop(['Salary'],axis=1)
```

In [90]:

```
x_iv
```

Out[90]:

	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 [91]:
```
y_dv = clean_data.drop(['Name', 'Domain', 'Age', 'Location','Exp'],axis=1)
```

In [92]:
```
y_dv
```

Out[92]:

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

In [93]:
```
clean_data
```

Out[93]:

	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 [96]:
```
imputation = pd.get_dummies(clean_data)
```

In [97]:
```
imputation
```

Out[97]:

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

In [ ]: