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

In [2]:
```
emp = pd.read_excel(r'C:\Users\soham\OneDrive\Desktop\Rawdata.xlsx')
```

In [3]:
```
emp
```

Out[3]:

	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 [4]:
```
emp.shape  #total lenth of dimension
```

Out[4]:
```
(6, 6)
```

In [6]:
```
len(emp)
```

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

In [8]:
```
emp.columns
```

Out[8]:
```
Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
```

In [13]:
```
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 [15]:
```
emp
```

Out[15]:

	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 [19]:

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

Out[19]:

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

In [21]:

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

Out[21]:

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

In [23]:

```
emp['Location']
```

Out[23]:

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

In [25]:

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

Out[25]:
```
0      5^00#0
1     10%%000
2     1$5%000
3      2000^0
4      30000-
5     6000^$0
Name: Salary, dtype: object
```

In [27]:
```
emp['Exp']
```

Out[27]:
```
0          2+
1          <3
2      4> yrs
3         NaN
4     5+ year
5         10+
Name: Exp, dtype: object
```

In [29]:
```
emp[['Name','Age']]
```

Out[29]:

	Name	Age
0	Mike	34 years
1	Teddy^	45' yr
2	Uma#r	NaN
3	Jane	NaN
4	Uttam*	67-yr
5	Kim	55yr

In [31]:
```
emp[['Name','Domain','Age','Location','Salary','Exp']]
```

Out[31]:

	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 of row formate#

In [34]:
```
emp['Name']
```

Out[34]:

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

In [36]:

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

In [38]:

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

Out[38]:

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

In [40]:

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

In [42]:

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

Out[42]:

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

In [44]:

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

In [46]:

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

Out[46]:

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

In [48]:

```
emp['Age'] = emp['Age'].str.extract('(\d+)')    #using age cancel year
```

```
<>:1: SyntaxWarning: invalid escape sequence '\d'
<>:1: SyntaxWarning: invalid escape sequence '\d'
C:\Users\soham\AppData\Local\Temp\ipykernel_22004\4102034463.py:1: SyntaxWarning:
invalid escape sequence '\d'
  emp['Age'] = emp['Age'].str.extract('(\d+)')    #using age cancel year
```

In [50]:

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

```
Out[50]: 0     34
         1     45
         2    NaN
         3    NaN
         4     67
         5     55
         Name: Age, dtype: object
```

```
In [52]: emp['Location'] = emp['Location'].str.replace(r'\W','')
```

```
In [54]: emp['Location']
```

```
Out[54]: 0       Mumbai
         1    Bangalore
         2          NaN
         3     Hyderbad
         4          NaN
         5        Delhi
         Name: Location, dtype: object
```

```
In [56]: emp['Salary'] = emp['Salary'].str.replace(r'\W','',regex=True)
```

```
In [58]: emp['Salary']
```

```
Out[58]: 0     5000
         1    10000
         2    15000
         3    20000
         4    30000
         5    60000
         Name: Salary, dtype: object
```

```
In [60]: emp['Exp'] = emp['Exp'].str.replace(r'\W','',regex=True)
```

```
In [62]: emp['Exp'] = emp['Exp'].str.extract('(\d+)')
```

```
<>:1: SyntaxWarning: invalid escape sequence '\d'
<>:1: SyntaxWarning: invalid escape sequence '\d'
C:\Users\soham\AppData\Local\Temp\ipykernel_22004\3836251810.py:1: SyntaxWarning:
invalid escape sequence '\d'
  emp['Exp'] = emp['Exp'].str.extract('(\d+)')
```

```
In [64]: emp['Exp']
```

```
Out[64]: 0      2
         1      3
         2      4
         3    NaN
         4      5
         5     10
         Name: Exp, dtype: object
```

```
In [66]: emp   #cleaning of 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	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 [68]:

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

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	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 [72]:

```
# Missing values treatment #
```

In [74]:

```
clean_data
```

Out[74]:

	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 [76]:

```
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 [78]:
```
clean_data.head(2)
```

Out[78]:

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

In [80]:
```
import numpy as np
import pandas as pd
```

In [82]:
```
clean_data['Age']
```

Out[82]:
```
0     34
1     45
2    NaN
3    NaN
4     67
5     55
Name: Age, dtype: object
```

In [84]:
```
clean_data['Age']  = clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['
```

In [86]:
```
import numpy as np
import pandas as pd
```

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	50.25	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

In [90]:
```
clean_data['Age']
```

Out[90]:
```
0       34
1       45
2    50.25
3    50.25
4       67
5       55
Name: Age, dtype: object
```

In [92]:
```
clean_data['Exp']  = clean_data['Exp'].fillna(np.mean(pd.to_numeric(clean_data['
```

In [94]:
```
clean_data['Exp']
```

Out[94]:
```
0      2
1      3
2      4
3    4.8
4      5
5     10
Name: Exp, dtype: object
```

In [96]:
```
clean_data
```

Out[96]:

	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 [98]:
```
clean_data['Location']  = clean_data['Location'].fillna(clean_data['Location'].m
```

In [100…
```
clean_data['Location']
```

Out[100…
```
0       Mumbai
1    Bangalore
2    Bangalore
3     Hyderbad
4    Bangalore
5        Delhi
Name: Location, dtype: object
```

In [102…
```
clean_data
```

Out[102…

	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 [104…

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

In [106…

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

In [108…

```
clean_data['Name'] = clean_data['Name'].astype('category')
```

In [110…

```
clean_data['Location'] = clean_data['Location'].astype('category')
```

In [112…

```
clean_data['Domain'] = clean_data['Domain'].astype('category')
```

In [114…

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

In [116…

```
clean_data
```

Out[116…

	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 [118…

```
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 [120…

```
clean_data
```

Out[120…

	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 [122…

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

In [124…

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

Out[124…

```
'C:\\Users\\soham'
```

In [126…

```
clean_data.columns
```

Out[126…

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

In [128…

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

In [129…

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

In [130…

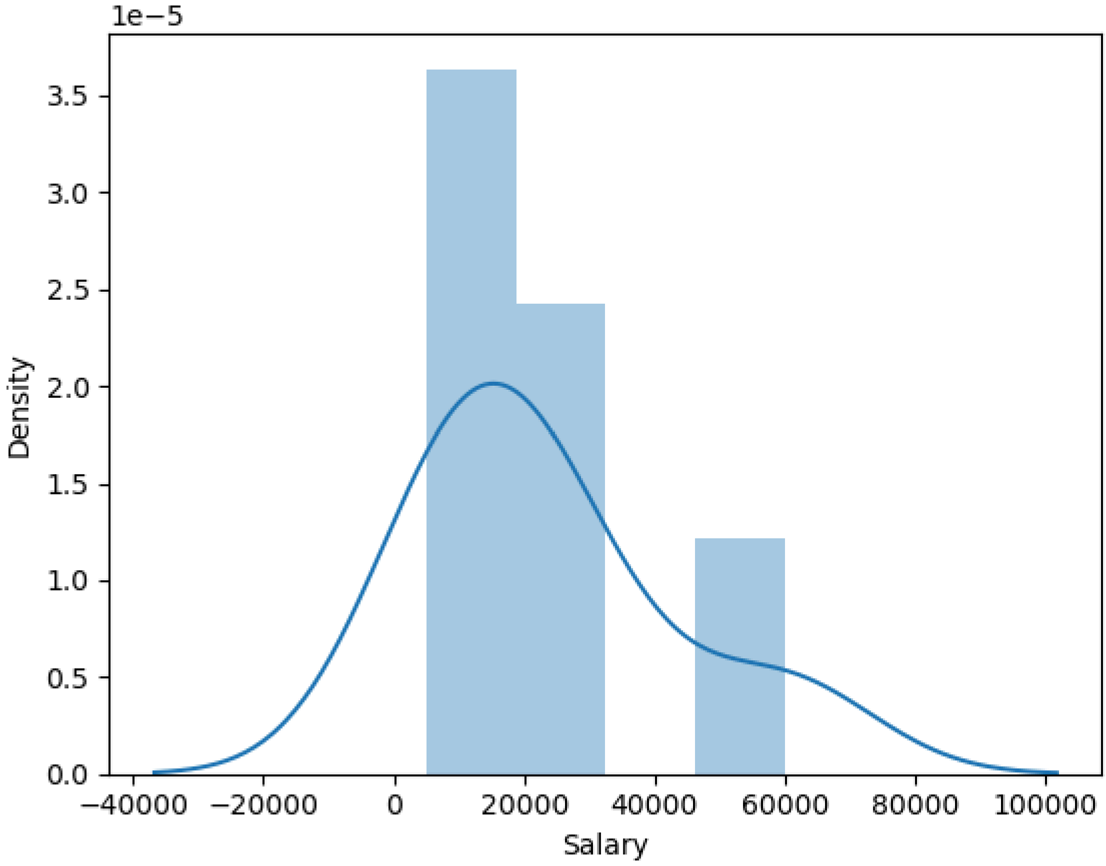
```
clean_data['Salary']
```

Out[130…

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

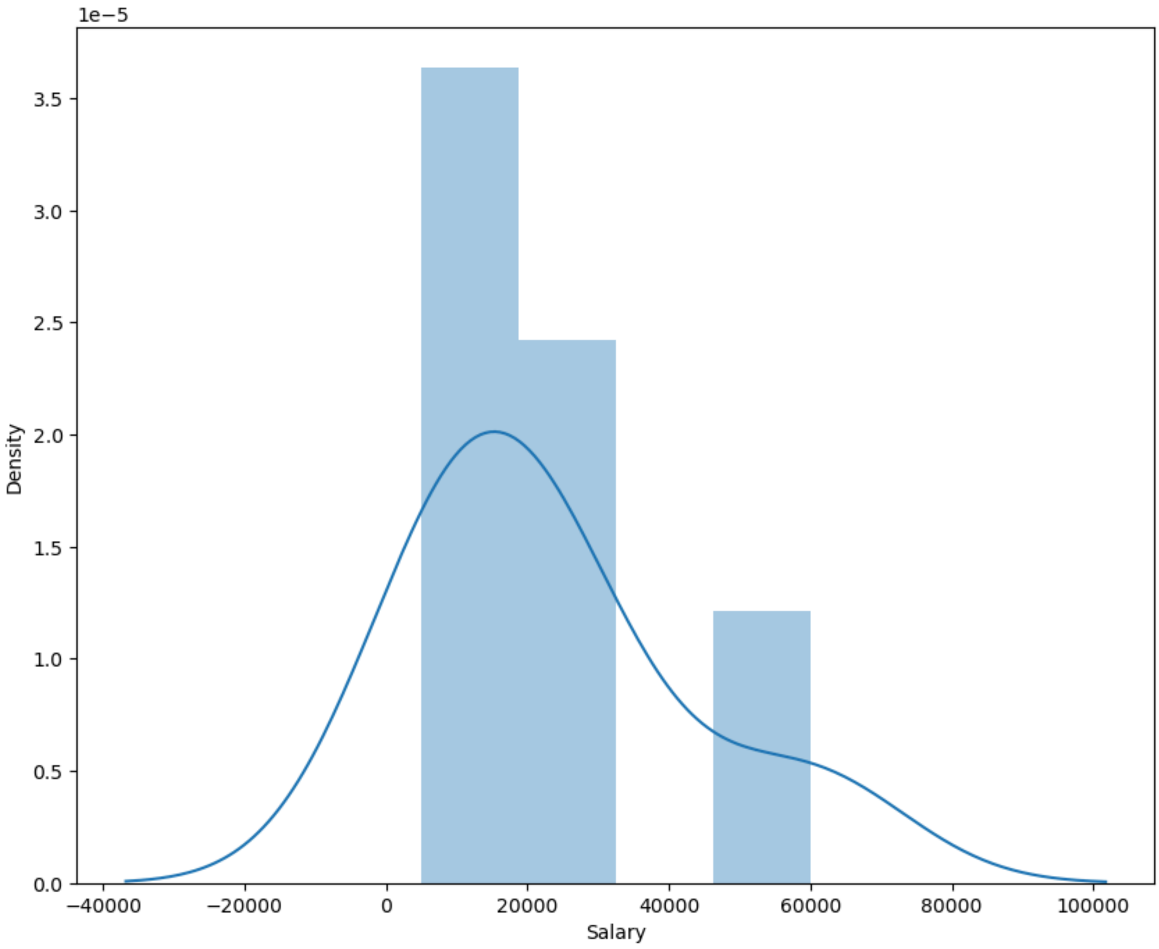
In [134…

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



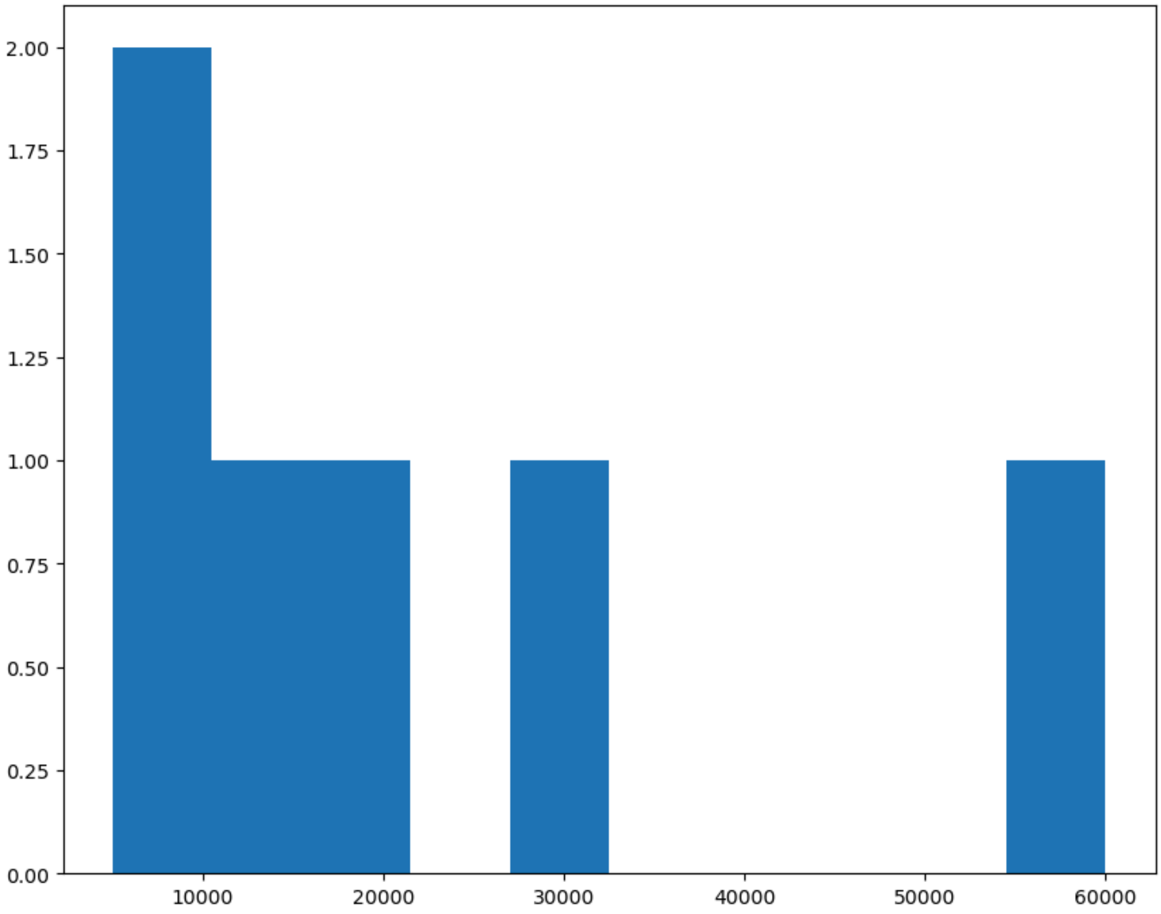
In [138…
```
plt.rcParams['figure.figsize'] = 10,8
```

In [140…
```
vis1 = sns.distplot(clean_data['Salary'])
```



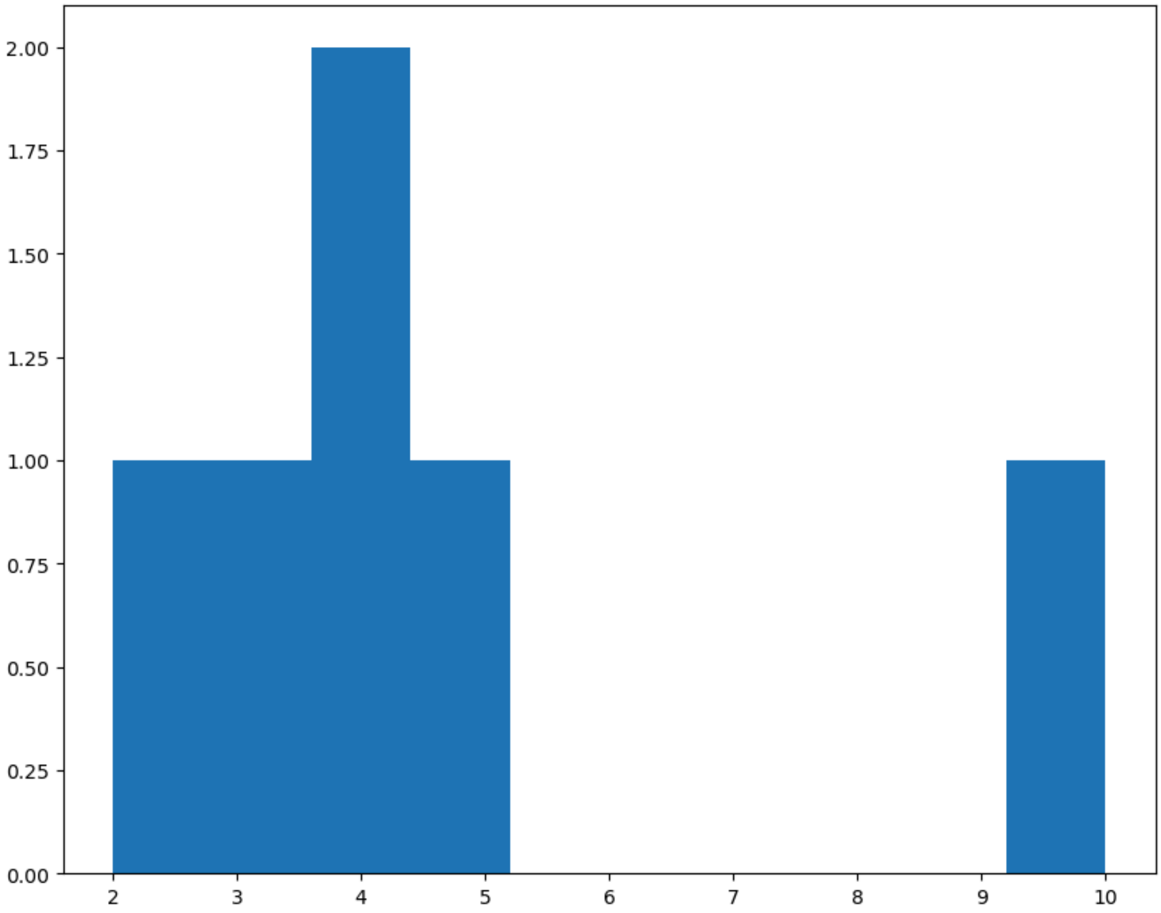
In [142…

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



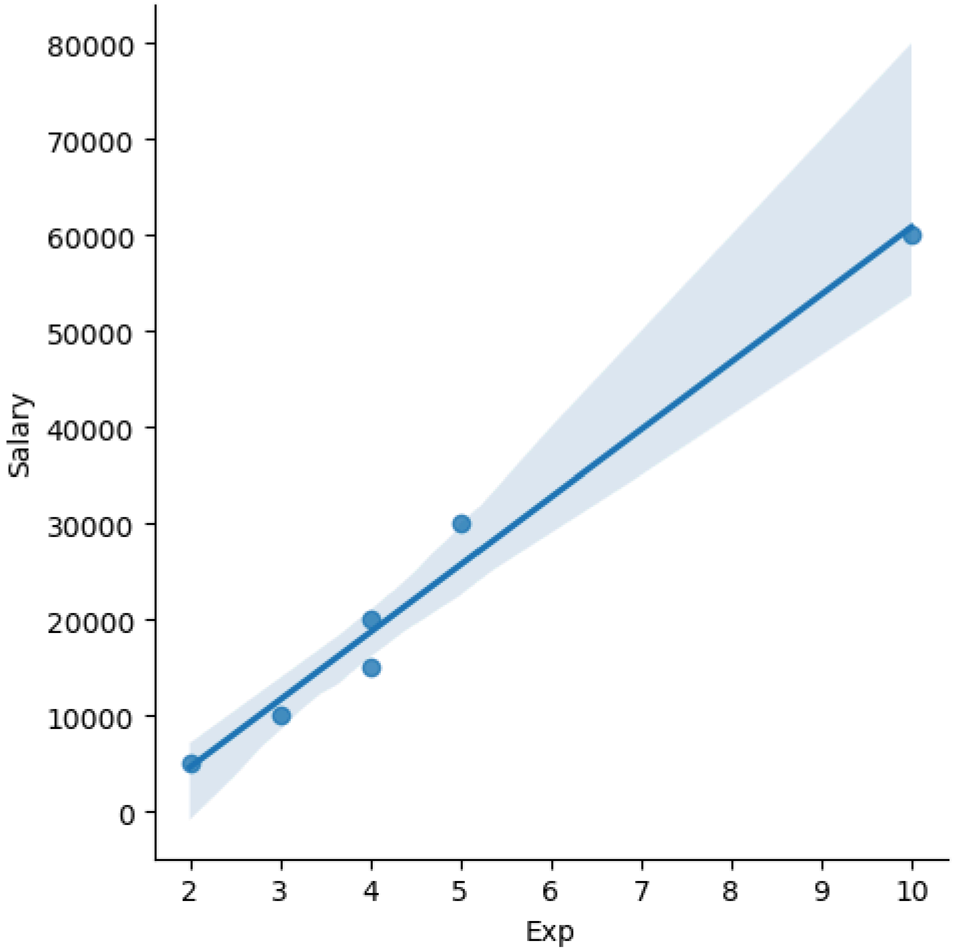
In [144…

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



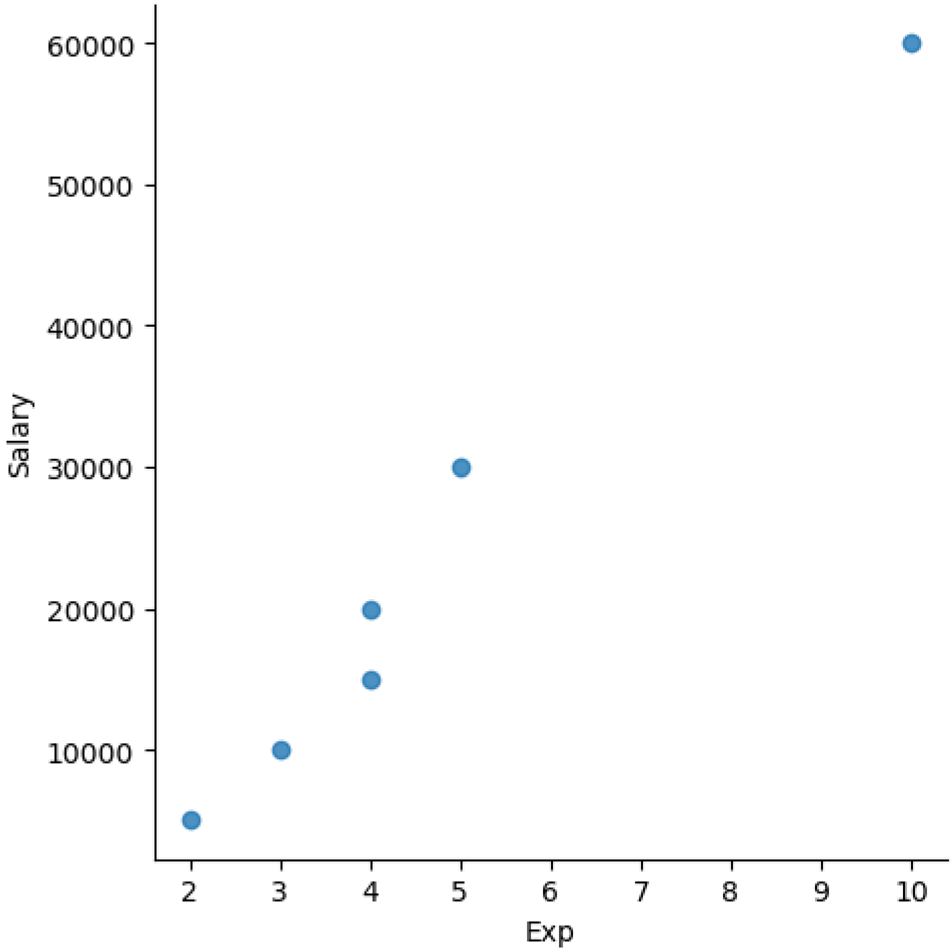
In [148…

```
vis4 = sns.lmplot(data=clean_data,x='Exp',y='Salary')
```



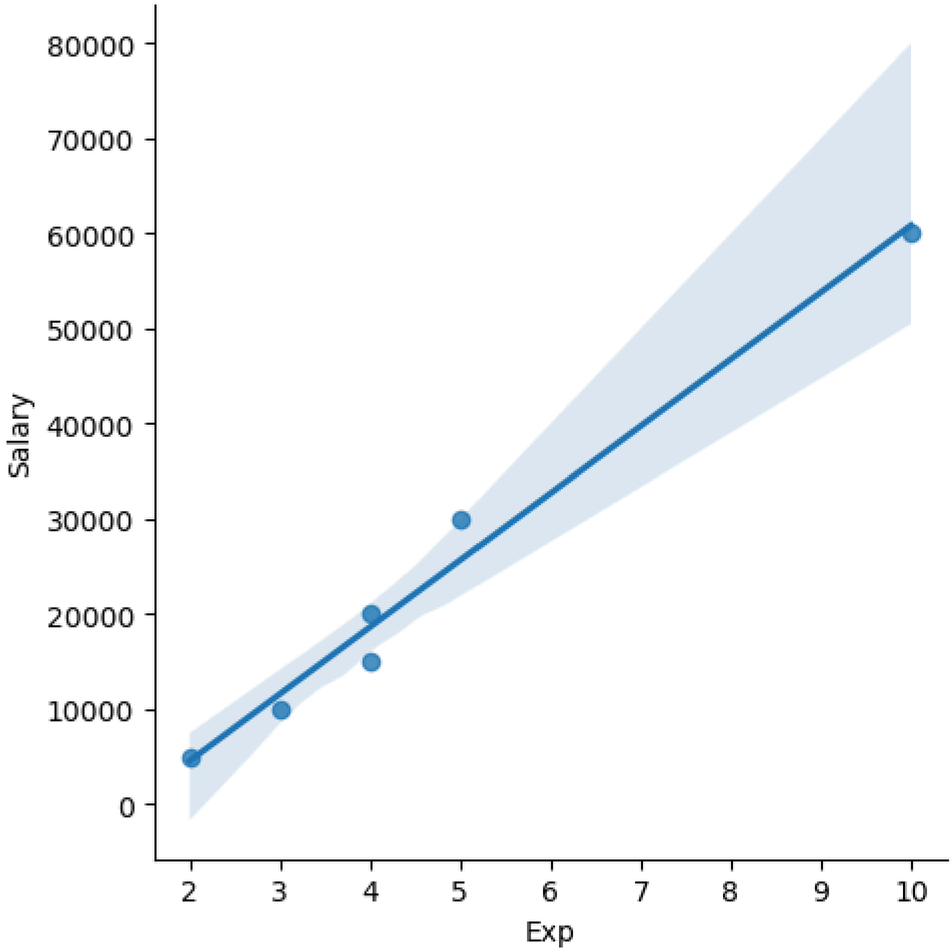
In [150…

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



In [152…

```
vis6 = sns.lmplot(clean_data,x='Exp',y='Salary',fit_reg = True)
```



In [154…

```
clean_data
```

Out[154…

	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 [156…

```
clean_data[:]
```

Out[156…

	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 [158…

```
clean_data[:2]
```

Out[158…

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

In [160…

```
clean_data[2:]
```

Out[160…

	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 [162…

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

Out[162…

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

In [164…

```
clean_data[0:3]
```

Out[164…

	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

In [172…

```
clean_data
```

Out[172…

	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 [174…

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

In [176…

```
clean_data
```

Out[176…

	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 [178…

```
x
```

Out[178…

	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 [180…

```
x.columns
```

Out[180…

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

In [182…

```
clean_data.columns
```

Out[182…

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

In [186…

```
clean_data
```

Out[186…

	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 [188…

```
y = clean_data.drop(['Name','Age','Location'],axis=1)
```

In [190…

```
y
```

Out[190…

	Domain	Salary	Exp
0	Datascience	5000	2
1	Testing	10000	3
2	Dataanalyst	15000	4
3	Analytics	20000	4
4	Statistics	30000	5
5	NLP	60000	10

In [199…

```
clean_data
```

Out[199…

	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 [201…

```
x
```

Out[201…

	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 [203…

```
y
```

Out[203…

	Domain	Salary	Exp
0	Datascience	5000	2
1	Testing	10000	3
2	Dataanalyst	15000	4
3	Analytics	20000	4
4	Statistics	30000	5
5	NLP	60000	10

In [207…

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

In [209…

```
imputation
```

Out[209…

	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 [ ]: