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

In [2]:
```
emp = pd.read_excel(r"C:\Users\indus\Downloads\python sir notes\Eda\Rawdata.xlsx")
```

In [3]:
```
emp  # employment chain
```

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]:
```
# we are removing the reg ex it is spl characters to clean the raw data
```

In [5]:
```
len(emp)
```

Out[5]:
```
6
```

In [6]:
```
emp.columns
```

Out[6]:
```
Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
```

In [7]:
```
id(emp)
```

Out[7]:
```
1962302322064
```

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.isnull()
```

Out[12]:

	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 [13]:
```
emp.isnull().sum()
```

Out[13]:
```
Name        0
Domain      0
Age         2
Location    2
Salary      0
Exp         1
dtype: int64
```

Out[14]:

In [14]:

```
                ke
emp['Na  1    Teddy^
me']     2     Uma#r
         3      Jane
 0       M  4    Uttam*
            i
         5Kim
         Name: Name, dtype: object
```

In [15]:
```
emp['Name'] =emp['Name'].str.replace(r'\W','',regex=True) # W is non word it  represents the characters eg like /,*
```

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

Out[16]:
```
0     Mike
1    Teddy
2     Umar
3     Jane
4    Uttam
5Kim
Name: Name, dtype: object
```

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

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

Out[20]:
```
0    Datascience
1        Testing
2    Dataanalyst
3      Analytics
4     Statistics
5            NLP
Name: Domain, dtype: object
```

In [21]:
```
emp['Salary']
```

Out[21]:
```
0     5^00#0
1    10%%000
2    1$5%000
3     2000^0
4     30000-
56000^$0
Name: Salary, dtype: object
```

In [22]:
```
emp['Salary'] =emp['Salary'].str.replace(r'\W','',regex=True)
```

In [23]:
```
emp['Salary']
```

Out[23]:
```
0     5000
1    10000
2    15000
3    20000
4    30000
560000
Name: Salary, dtype: object
```

In [24]:
```
emp.head()
```

Out[24]:

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34 years	Mumbai	5000	2+
1	Teddy	Testing	45' yr	Bangalore	10000	<3
2	Umar	Dataanalyst	NaN	NaN	15000	4> yrs
3	Jane	Analytics	NaN	Hyderbad	20000	NaN
4	Uttam	Statistics	67-yr	NaN	30000	5+ year

In [25]:
```
emp['Age']
```

Out[25]:
```
0    34 years
1      45' yr
2         NaN
3         NaN
4       67-yr
555yr
Name: Age, dtype: object
```

In [26]:
```
emp['Age'] =emp['Age'].str.replace(r'\W','',regex=True)
```

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

Out[27]:
```
0    34years
1       45yr
2        NaN
3        NaN
4       67yr
555yr
```

```
Name:    Age, dtype: object
```

In [28]:
```
emp['Age'] =emp['Age'].str.extract('(\d+)') # d is the only digit extracting
```

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

Out[29]:
```
0     34
1     45
2    NaN
3    NaN
4     67
555
Name: Age, dtype: object
```

In [30]:
```
emp['Location']
```

Out[30]:
```
0       Mumbai
1    Bangalore
2          NaN
3     Hyderbad
4          NaN
5Delhi
Name: Location, dtype: object
```

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

In [32]:
```
emp['Location']
```

Out[32]:
```
0       Mumbai
1    Bangalore
2          NaN
3     Hyderbad
4          NaN
5Delhi
Name: Location, dtype: object
```

In [33]:
```
emp['Exp']
```

Out[33]:
```
0         2+
1         <3
24> yrs
3        NaN
45+ year
510+
Name: Exp, dtype: object
```

In [34]:
```
emp['Exp'] =emp['Exp'].str.replace(r'\W','',regex=True)
```

In [35]:
```
emp['Exp']
```

Out[35]:
```
0        2
1        3
2     4yrs
3      NaN
4    5year
510
Name: Exp, dtype: object
```

In [36]:
```
emp['Exp'] =emp['Exp'].str.extract('(\d+)')
```

In [37]:
```
emp['Exp']
```

Out[37]:
```
0      2
1      3
2      4
3    NaN
4      5
510
Name: Exp, dtype: object
```

In [38]:
```
emp.head()
```

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

In [39]:
```
emp['Salary'] =emp['Salary'].str.extract('(\d+)')
```

In [40]:
```
emp['Salary']
```

Out[40]:
```
0     5000
1    10000
2    15000
3    20000
4    30000
560000
Name: Salary, dtype: object
```

In [41]:
```
emp.head()
```

Out[41]:

	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 [42]:
```
clean_data =emp.copy()
```

Eda technique

In [43]:
```
clean_data.isnull()
```

Out[43]:

	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 [44]:
```
clean_data.isnull().sum()
```

Out[44]:
```
Name        0
Domain      0
Age         2
Location    2
Salary      0
Exp         1
dtype: int64
```

In [45]:
```
clean_data['Age']
```

Out[45]:
```
0     34
1     45
2    NaN
3    NaN
4     67
555
Name: Age, dtype: object
```

In [46]:
```
import numpy as np
```

In [47]:
```
clean_data['Age'] = clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'])))
```

In [48]:
```
clean_data['Age']
```

Out[48]:
```
0       34
1       45
2    50.25
3    50.25
4       67
555
Name: Age, dtype: object
```

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

Out[49]:
```
0      2
1      3
2      4
3    NaN
4      5
510
Name: Exp, dtype: object
```

In [50]:
```
clean_data['Exp']=clean_data['Exp'].fillna(np.mean(pd.to_numeric(clean_data['Exp'])))
```

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

Out[51]:
```
0      2
1      3
2      4
3    4.8
4      5
510
Name: Exp, 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	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 [53]:
```
clean_data['Location'].isnull().sum()
```

Out[53]:
```
2
```

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

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

In [55]:
```
clean_data['Location']=clean_data['Location'].fillna(clean_data['Location'].mode()[0])
```

In [56]:
```
clean_data['Location']
```

Out[56]:
```
0       Mumbai
1    Bangalore
2    Bangalore
3     Hyderbad
```

```
4    Ba       ore
     ng  5Delhi
     al  Name: Location, dtype: object
```

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	50.25	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 [58]:
```
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 [59]:
```
clean_data['Age'] = clean_data['Age'].astype(int)
```

In [60]:
```
clean_data['Age']
```

Out[60]:
```
0    34
1    45
2    50
3    50
4    67
555
Name: Age, dtype: int32
```

In [61]:
```
clean_data['Exp'] = clean_data['Exp'].astype(int)
```

In [62]:
```
clean_data['Exp']
```

Out[62]:
```
0     2
1     3
2     4
3     4
4     5
510
Name: Exp, dtype: int32
```

In [63]:
```
clean_data
```

Out[63]:

	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 [64]:
```
clean_data['Salary'] = clean_data['Salary'].astype(int)
```

In [65]:
```
clean_data['Salary']
```

Out[65]:
```
0     5000
1    10000
2    15000
3    20000
4    30000
560000
Name: Salary, dtype: int32
```

In [66]:
```
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 [67]:
```
clean_data
```

Out[67]:

	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 [68]:
```
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 [69]:
```
clean_data.to_csv('clean_data_csv') # saving it in to desktop
```

In [70]:
```
import os
```

In [71]:
```
import os
os.getcwd()
```

Out[71]: 'C:\\Users\\indus'

In [72]:
```
clean_data
```

Out[72]:

	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 [73]:
```
import matplotlib.pyplot as plt
import seaborn as sns
```

In [74]:
```
import warnings
warnings.filterwarnings("ignore")
```

In [75]:
```
clean_data
```

Out[75]:

	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

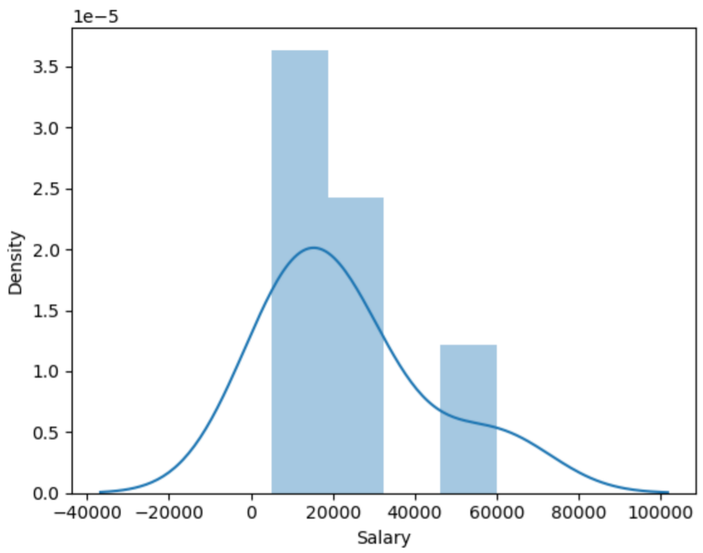
In [77]:

In [82]:

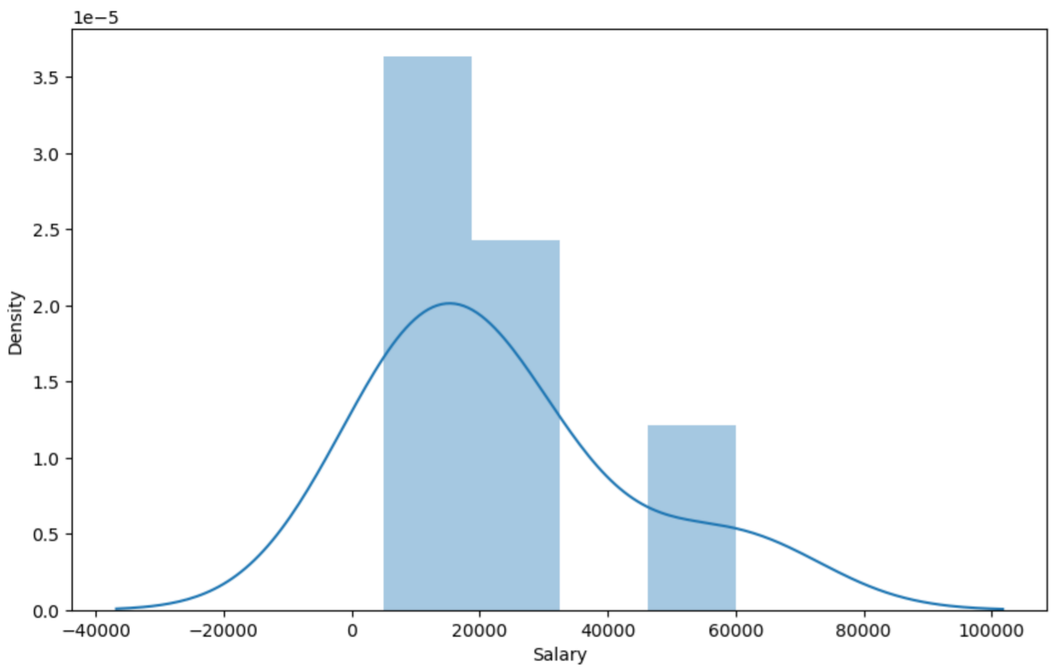
In [80]:

In [83]:

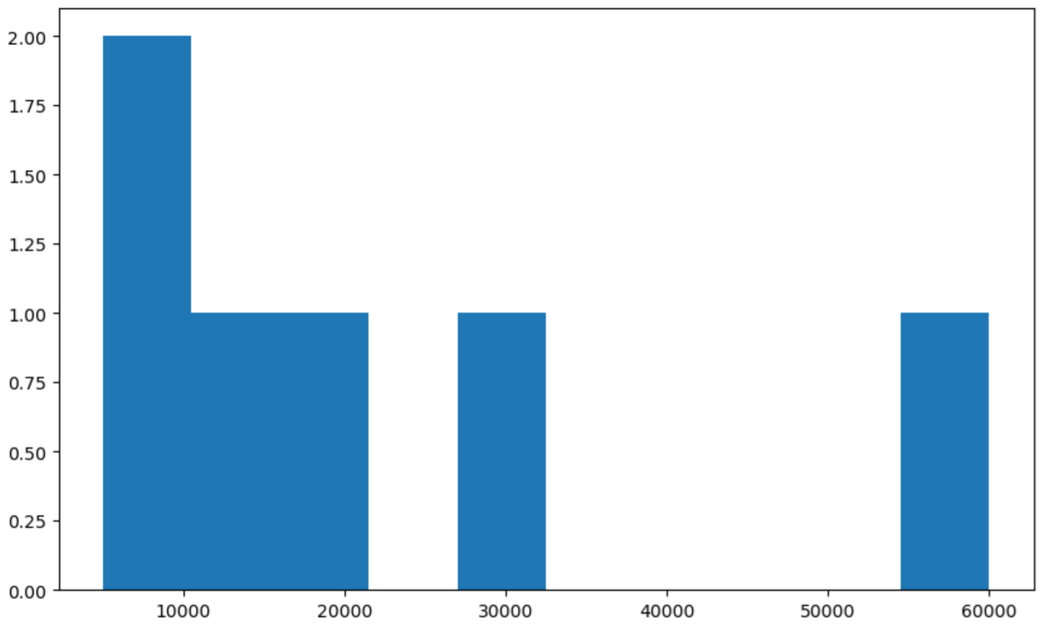
In [84]:
```
v = sns.distplot(clean_data['Salary'])
```



In [85]:
```
plt.rcParams['figure.figsize']=10,6
v1 = sns.distplot(clean_data['Salary'])
```



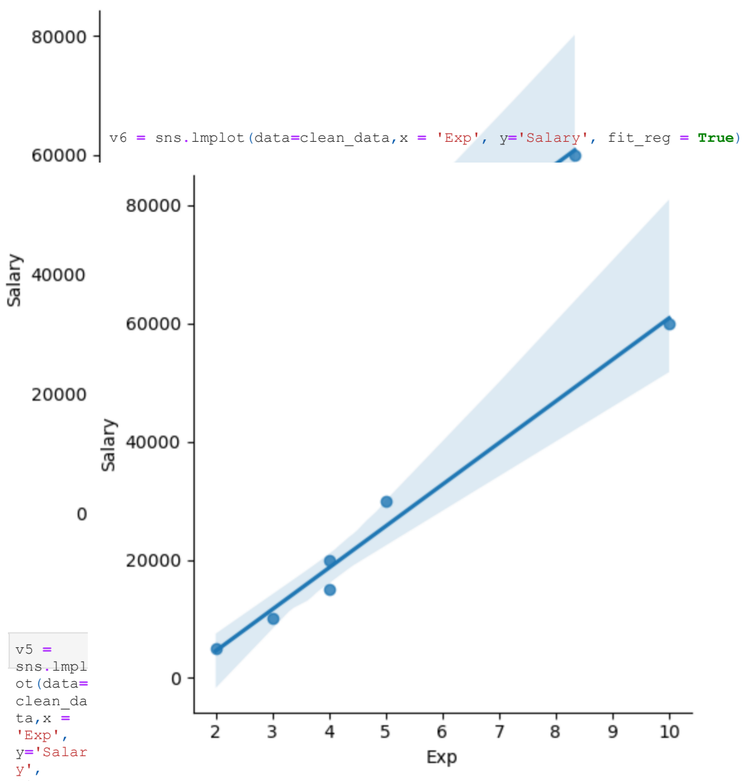
In [86]:
```
v2 = plt.hist(clean_data['Salary'])
```



In [87]:
```
v3 = plt.hist(clean_data['Exp'])
```

Out[87]:

```
v4 =
sns.lmp
lot(dat
a=clean
_data,x
=
'Exp',
y='Sala
ry')
```




```
v5 =
sns.lmpl
ot(data=
clean_da
ta,x =
'Exp',
y='Salar
y',
fit_reg
= False)
```

```
v6 = sns.lmplot(data=clean_data,x = 'Exp', y='Salary', fit_reg = True)
```



```
clean_data
```

	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 [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]:
```
clean_data[:-2]
```

Out[89]:

	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

In [92]:
```
clean_data[0:1]
```

Out[92]:

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

In [94]:

In [93]:
```
Out[94]:
```

```
p(['Salary'],axis=1)
```

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

```
clean_data
```

	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 [95]:
```
x_iv
```

Out[95]:

	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 [97]:
```
x_iv.columns
```

Out[97]:
```
Index(['Name', 'Domain', 'Age', 'Location', 'Exp'], dtype='object')
```

In [98]:
```
clean_data.columns
```

Out[98]:
```
Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
```

In [99]:
```
y_dv = clean_data.drop(['Name','Domain','Age','Location','Exp'],axis=1)
```

In [100…
```
y_dv
```

Out[100…

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

In [104…
```
y_dv
```

Out[104…

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

```
x_iv
```

In [103…

Out[103…

	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 [105…
```
clean_data
```

Out[105…

	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 [109…

In [110…
```
imputation = pd.get_dummies(clean_data , dtype=int)
```

Out[110…
```
imputation
```

	Age	Salary	Exp	Name_Jane	Name_Kim	Name_Mike	Name_Teddy	Name_Umar	Name_Uttam	Domain_Analytics	Domain_Dataanalyst	Domain_Datascience	Domain_NLP	Domain_Statistics	Domain_Testing	Location_Bangalore	Location_Delhi	Location_Hyderbad	Location_Mumbai
0	34	5000	2	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1
1	45	10000	3	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
2	50	15000	4	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0
3	50	20000	4	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
4	67	30000	5	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0
5	55	60000	10	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0

In [111…
```
imputation = pd.get_dummies(clean_data)
imputation
```

Out[111…

	Age	Salary	Exp	Name_Jane	Name_Kim	Name_Mike	Name_Teddy	Name_Umar	Name_Uttam	Domain_Analytics	Domain_Dataanalyst	Domain_Datascience	Domain_NLP	Domain_Statistics	Domain_Testing	Location_Bangalore	Location_Delhi	Location_Hyderbad	Location_Mumbai
0	34	5000	2	False	False	True	False	False	False	False	False	True	False	False	False	False	False	False	True
1	45	10000	3	False	False	False	True	False	False	False	False	False	False	False	True	True	False	False	False
2	50	15000	4	False	False	False	False	True	False	False	True	False	False	False	False	True	False	False	False
3	50	20000	4	True	False	False	False	False	False	True	False	False	False	False	False	False	False	True	False
4	67	30000	5	False	False	False	False	False	True	False	False	False	False	True	False	True	False	False	False
5	55	60000	10	False	True	False	False	False	False	False	False	False	True	False	False	False	True	False	False

In [ ]: