```
!pip install --upgrade openpyxl
```

```
Requirement already satisfied: openpyxl in /usr/local/lib/python3.11/dist-packages (3.1.5)
Requirement already satisfied: et-xmlfile in /usr/local/lib/python3.11/dist-packages (from openpyxl) (2.0.0)
```

```
import pandas as pd
```

```
emp=pd.read_excel(r"/content/Rawdata.xlsx")
emp
```

	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+

Next steps: Generate code with emp | View recommended plots | New interactive sheet

```
id(emp)
```

```
136837415323472
```

```
emp.columns
```

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

```
emp.shape
```

```
(6, 6)
```

```
emp.head()
```

	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

Next steps: Generate code with emp | View recommended plots | New interactive sheet

```
emp.tail()
```

	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+

```
emp.head()
```

	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

Next steps: Generate code with emp | View recommended plots | New interactive sheet

```
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
```

```
emp
```

	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+

Next steps: Generate code with emp | View recommended plots | New interactive sheet

```
emp.isnull()  #check missing value(false means missing value)
```

	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

```
emp.isna()
```

	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

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

	0
Name	0
Domain	0
Age	2
Location	2
Salary	0
Exp	1

dtype: int64

```
emp["Name"]
```

	Name
0	Mike
1	Teddy^
2	Uma#r
3	Jane
4	Uttam*
5	Kim

dtype: object

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

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

	Name
0	Mike
1	Teddy
2	Umar
3	Jane
4	Uttam
5	Kim

dtype: object

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

```
emp["Age"]
```

	Age
0	34
1	45
2	NaN
3	NaN
4	67
5	55

dtype: object

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

```
emp["Location"]
```

	Location
0	Mumbai
1	Bangalore
2	NaN
3	Hyderbad
4	NaN
5	Delhi

dtype: object

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

```
emp["Domain"]
```

	Domain
0	Datascience
1	Testing
2	Dataanalyst
3	Analytics
4	Statistics
5	NLP

dtype: object

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

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

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

dtype: object

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

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

	Exp
0	2
1	3
2	4
3	NaN
4	5
5	10

dtype: object

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

```
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	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

Next steps: Generate code with clean_data | View recommended plots | New interactive sheet

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

	0
Name	0
Domain	0
Age	2
Location	2
Salary	0
Exp	1

dtype: int64

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

	Age
0	34
1	45
2	NaN
3	NaN
4	67
5	55

dtype: object

```
emp
```

	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

Next steps: Generate code with emp | View recommended plots | New interactive sheet

```
import numpy as np

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

	Age
0	34
1	45
2	50.25
3	50.25
4	67
5	55

dtype: object

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

	0
Name	0
Domain	0
Age	0
Location	2
Salary	0
Exp	1

dtype: int64

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

	Exp
0	2
1	3
2	4
3	4.8
4	5
5	10

dtype: object

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

	Location
0	Mumbai
1	Bangalore
2	Bangalore
3	Hyderbad
4	Bangalore
5	Delhi

dtype: object

```
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
```

```
clean_data['Age']=clean_data['Age'].astype(int)
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      int64
 3   Location  6 non-null      object
 4   Salary    6 non-null      object
 5   Exp       6 non-null      object
dtypes: int64(1), object(5)
memory usage: 420.0+ bytes
```

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

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

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

dtype: int64

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

```
clean_data['Exp']
```

	Exp
0	2
1	3
2	4
3	4
4	5
5	10

dtype: int64

```
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

Next steps: Generate code with clean_data | View recommended plots | New interactive sheet

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

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

```
'/content'
```

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

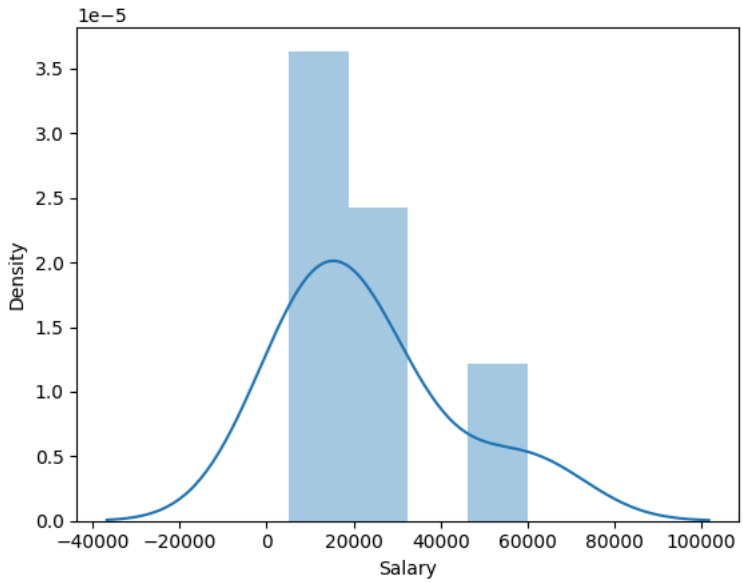
```
import warnings
warnings.filterwarnings('ignore')
```

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

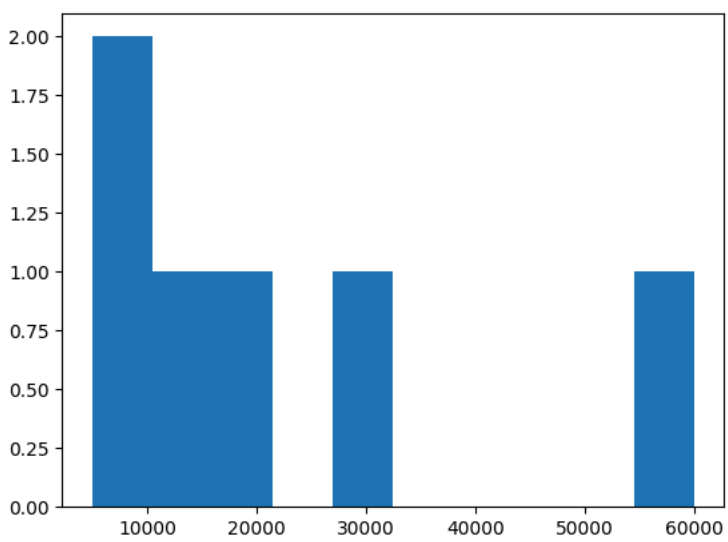
	Salary
0	5000
1	10000
2	15000
3	20000
4	30000
5	60000

dtype: int64

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




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

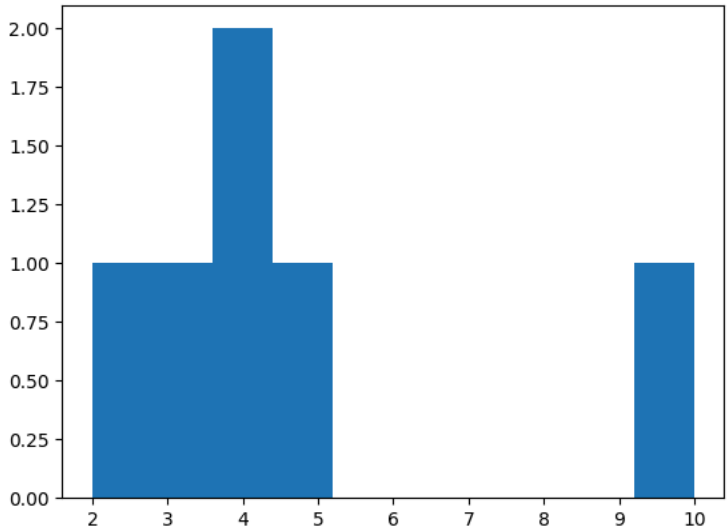



```
clean_data
```

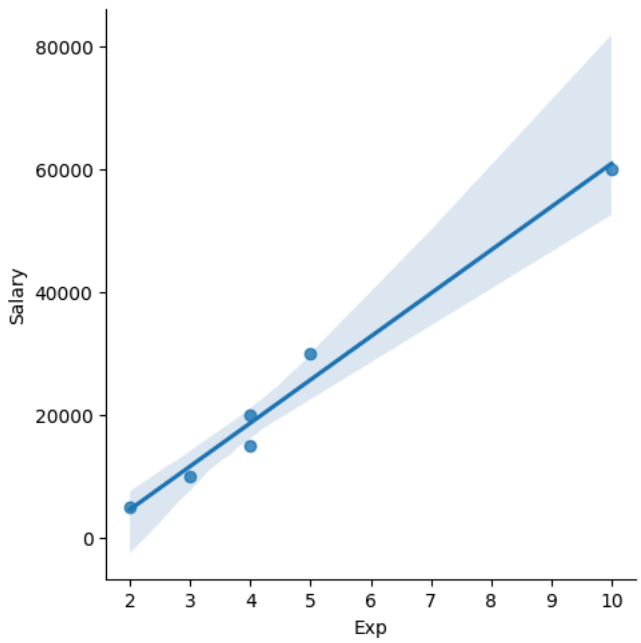
Show hidden output

Next steps: Generate code with clean_data | View recommended plots | New interactive sheet

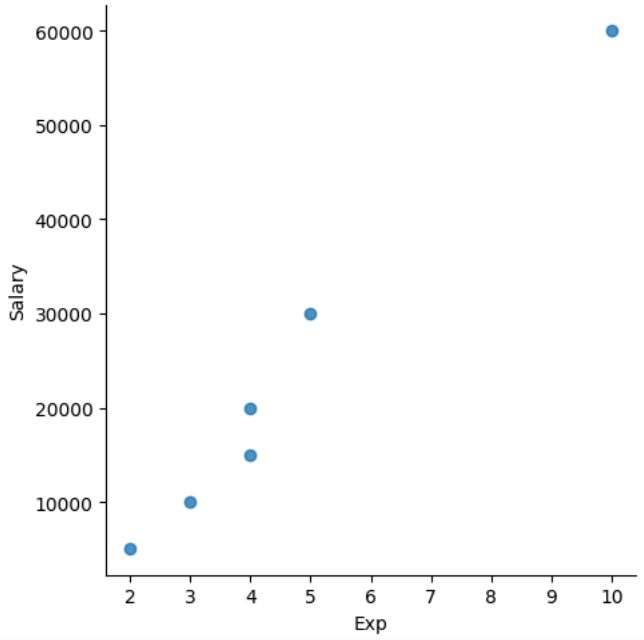
```
vis3=plt.hist(clean_data['Exp'])
```




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




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




```
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

```
clean_data[0:6:2]
```

	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

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

```
x_iv
```

	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

Next steps: Generate code with x_iv | View recommended plots | New interactive sheet

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

```
y_dv
```

	Salary
0	5000
1	10000
2	15000
3	20000
4	30000

Next steps: Generate code with y_dv | View recommended plots | New interactive sheet

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

Next steps: Generate code with clean_data | View recommended plots | New interactive sheet

x_iv

	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

Next steps: Generate code with x_iv | View recommended plots | New interactive sheet

y_dv

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

Next steps: Generate code with y_dv | View recommended plots | New interactive sheet

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