

21-11-2025 raw data to clean data conversion using python EDA

In [1]: `import pandas as pd`

In [2]: `pd.__version__`

Out[2]: '2.2.3'

In [3]: `emp = pd.read_excel(r"F:\Full Stack Data Science 9AM-Prakash senapathi\2.Novembe`

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]: `id(emp)`

Out[5]: 1915138883072

In [6]: `emp.columns`

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

In [7]: `emp.shape`

Out[7]: (6, 6)

In [8]: `emp.head()`

Out[8]:

	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 [9]: `emp.tail()`

Out[9]:

	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 [10]: 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 [11]: emp

Out[11]:

	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 [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.isna()
```

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

```
Out[14]: Name      0
Domain    0
Age       2
Location  2
Salary    0
Exp       1
dtype: int64
```

Data Cleaning

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

```
Out[16]: 0      Mike
1      Teddy^
2      Uma#r
3      Jane
4      Uttam*
5      Kim
Name: Name, dtype: object
```

```
In [17]: emp['Name'] = emp['Name'].str.replace(r'\W', '', regex=True)
```

```
In [18]: emp['Name']
```

```
Out[18]: 0    Mike
         1    Teddy
         2    Umar
         3    Jane
         4    Uttam
         5    Kim
         Name: Name, dtype: object
```

```
In [19]: emp
```

```
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	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 [20]: emp.columns
```

```
Out[20]: Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
```

```
In [21]: emp.head(1)
```

```
Out[21]:
```

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience#\$	34 years	Mumbai	5^00#0	2+

```
In [22]: emp['Domain']
```

```
Out[22]: 0    Datascience#$
         1    Testing
         2    Dataanalyst^^#
         3    Ana^^lytics
         4    Statistics
         5    NLP
         Name: Domain, dtype: object
```

```
In [23]: emp['Domain'] = emp['Domain'].str.replace(r'\W', '', regex=True)
```

```
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
```

Out[25]:

	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	Analytics	NaN	Hyderbad	2000^0	NaN
4	Uttam	Statistics	67-yr	NaN	30000-	5+ year
5	Kim	NLP	55yr	Delhi	6000^\$0	10+

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

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

Out[27]:

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

In [28]: `emp`

Out[28]:

	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	Analytics	NaN	Hyderbad	2000^0	NaN
4	Uttam	Statistics	67-yr	NaN	30000-	5+ year
5	Kim	NLP	55yr	Delhi	6000^\$0	10+

In [29]: `emp['Age'] = emp['Age'].str.replace(r'\W', '', regex=True)`

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

Out[30]:

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

In [31]: `emp['Age'] = emp['Age'].str.extract('(\d+)') # r(r'(\d+)')`

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

```
Out[32]: 0      34
         1      45
         2     NaN
         3     NaN
         4      67
         5      55
         Name: Age, dtype: object
```

```
In [33]: emp
```

```
Out[33]:
```

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

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

```
Out[35]: 0      5000
         1     10000
         2     15000
         3     20000
         4     30000
         5     60000
         Name: Salary, dtype: object
```

```
In [36]: 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> yrs
3	Jane	Analytics	NaN	Hyderbad	20000	NaN
4	Uttam	Statistics	67	NaN	30000	5+ year
5	Kim	NLP	55	Delhi	60000	10+

```
In [37]: emp['Exp'] = emp['Exp'].str.extract('(\d+)')
```

```
In [38]: emp['Exp']
```

```
Out[38]: 0      2
         1      3
         2      4
         3    NaN
         4      5
         5     10
         Name: Exp, dtype: object
```

```
In [39]: emp
```

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

```
In [41]: clean_data
```

```
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
5	Kim	NLP	55	Delhi	60000	10

24-11-2025

Missing value treatment

```
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.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 [44]: `import numpy as np`

In [45]: `clean_data`

Out[45]:

	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 [46]: `clean_data.head(1)`

Out[46]:

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

In [47]: `clean_data['Age']`

```
Out[47]: 0      34
         1      45
         2      NaN
         3      NaN
         4      67
         5      55
         Name: Age, dtype: object
```

```
In [48]: clean_data['Age'] = clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['A
```

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

```
Out[49]: 0      34
         1      45
         2     50.25
         3     50.25
         4      67
         5      55
         Name: Age, dtype: object
```

```
In [50]: emp
```

```
Out[50]:
```

	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 [51]: clean_data
```

```
Out[51]:
```

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

```
In [53]: clean_data['Exp']
```

```
Out[53]: 0      2
         1      3
         2      4
         3      4.8
         4      5
         5     10
         Name: Exp, dtype: object
```

```
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	50.25	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 [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    Bangalore
         5      Delhi
         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.to_csv('clean_data.csv')
```

```
In [59]: import os
         os.getcwd()
```

```
Out[59]: 'C:\\Users\\karthik reddy'
```

```
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	50.25	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

EDA TECHNIQUE LETS APPLY

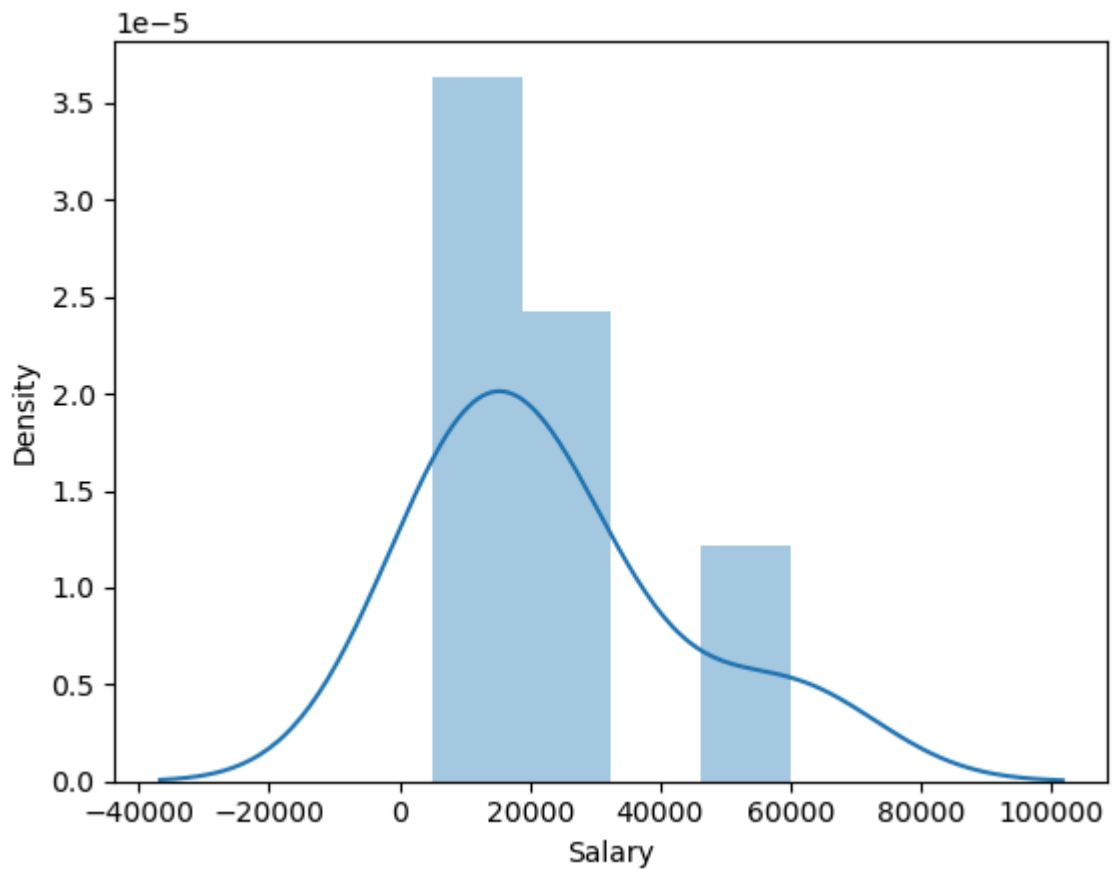
```
In [61]: import matplotlib.pyplot as plt # Visualization
import seaborn as sns
```

```
In [62]: import warnings
warnings.filterwarnings('ignore')
```

```
In [63]: clean_data['Salary']
```

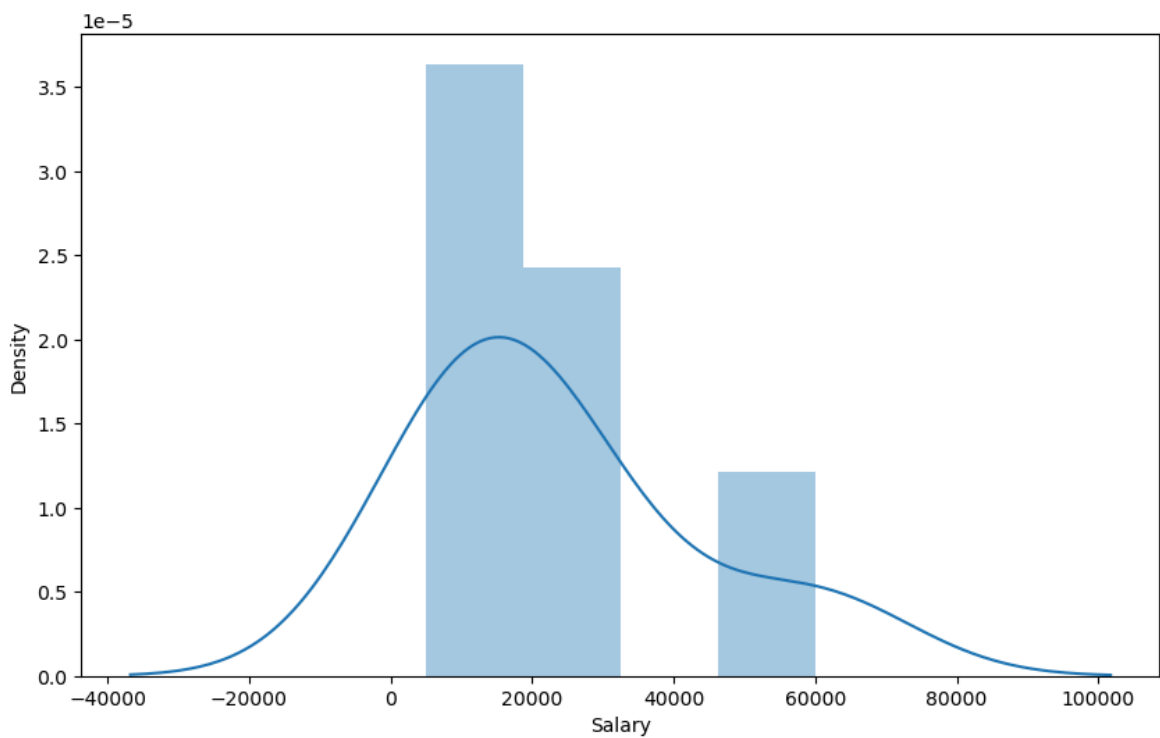
```
Out[63]: 0      5000
1     10000
2     15000
3     20000
4     30000
5     60000
Name: Salary, dtype: object
```

```
In [64]: vis1=sns.distplot(clean_data['Salary'])
```

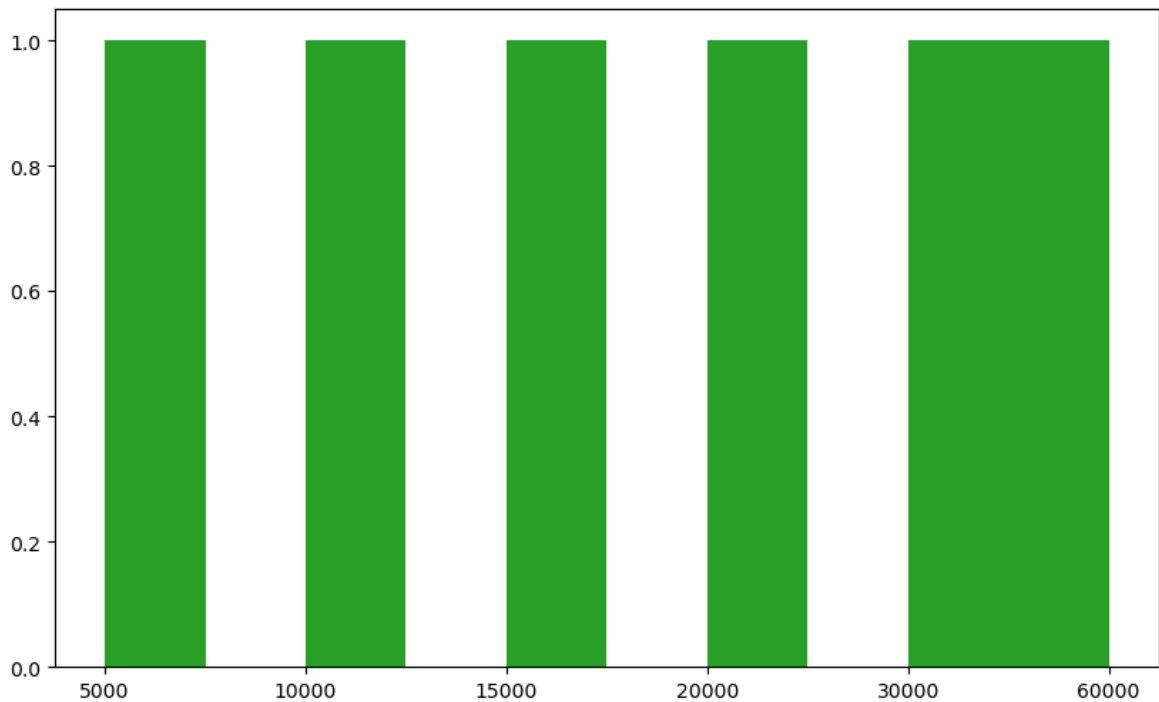


```
In [65]: plt.rcParams['figure.figsize'] = 10,6
```

```
In [66]: vis1 = sns.distplot(clean_data['Salary'])
```

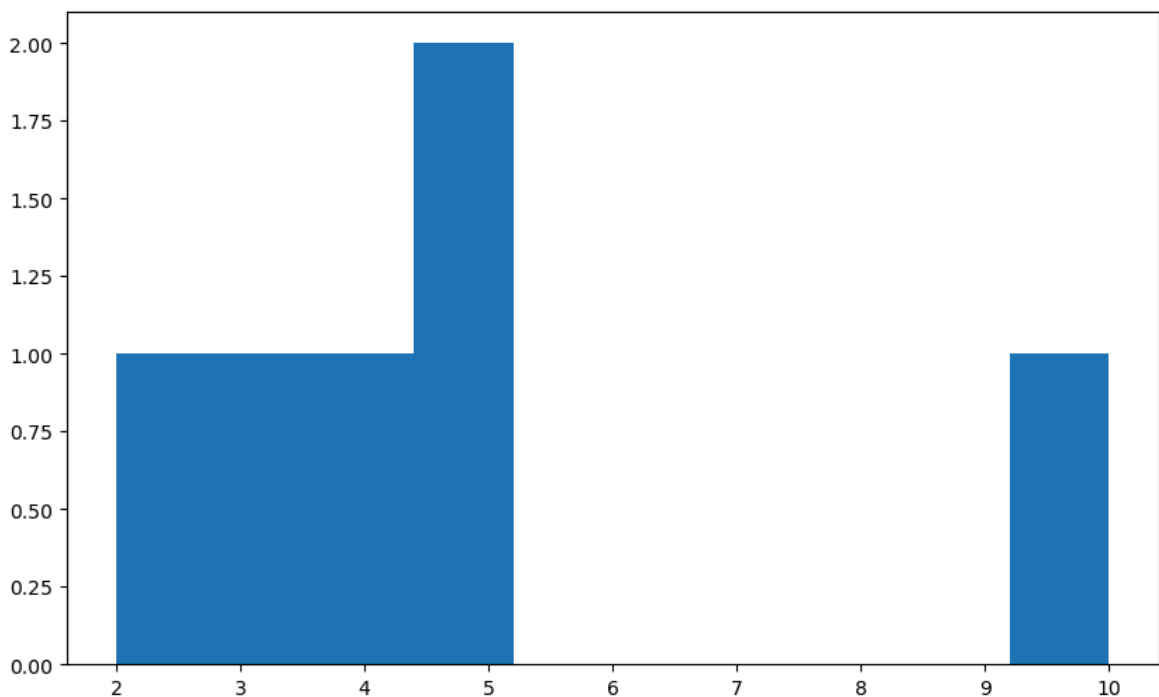


```
In [84]: vis2 = plt.hist(clean_data['Salary'])  
vis2  
plt.show()
```



```
In [85]: vis3 = plt.hist(clean_data['Exp'])
```

```
In [86]: vis3
plt.show()
```



```
In [88]: import seaborn as sns
import matplotlib.pyplot as plt
```

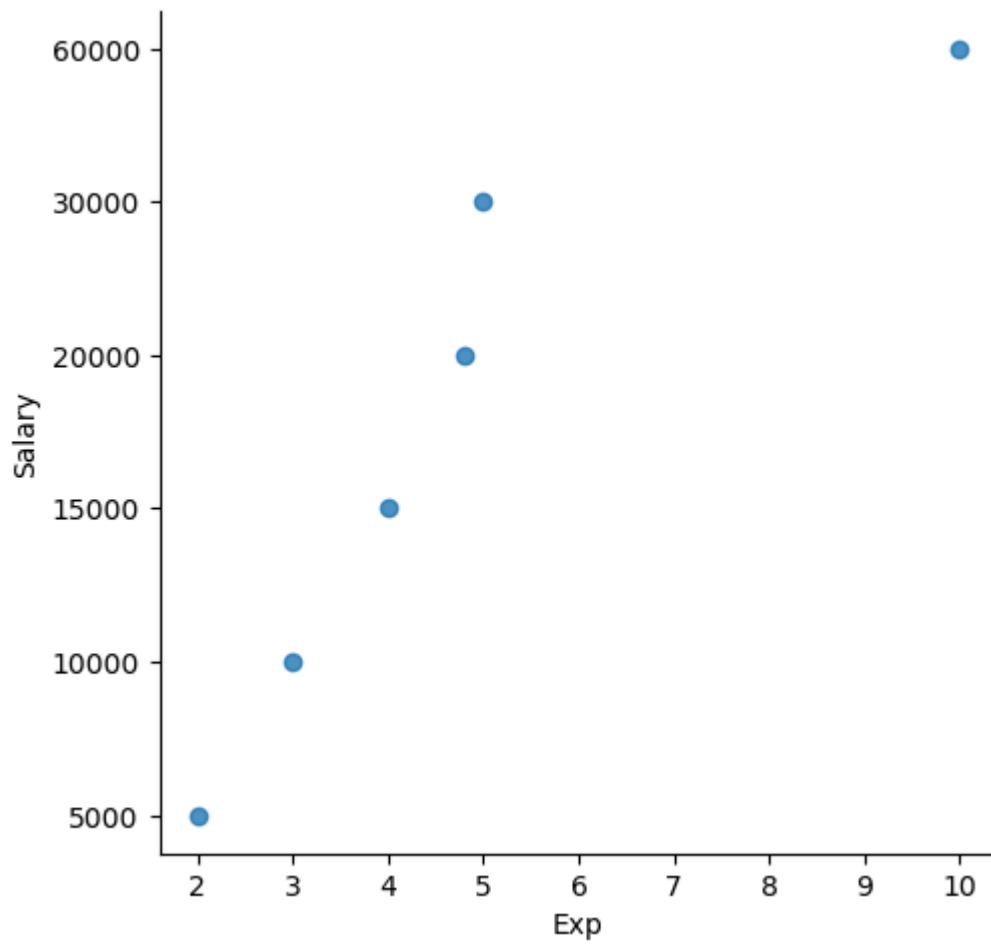
```
In [89]: clean_data.columns
```

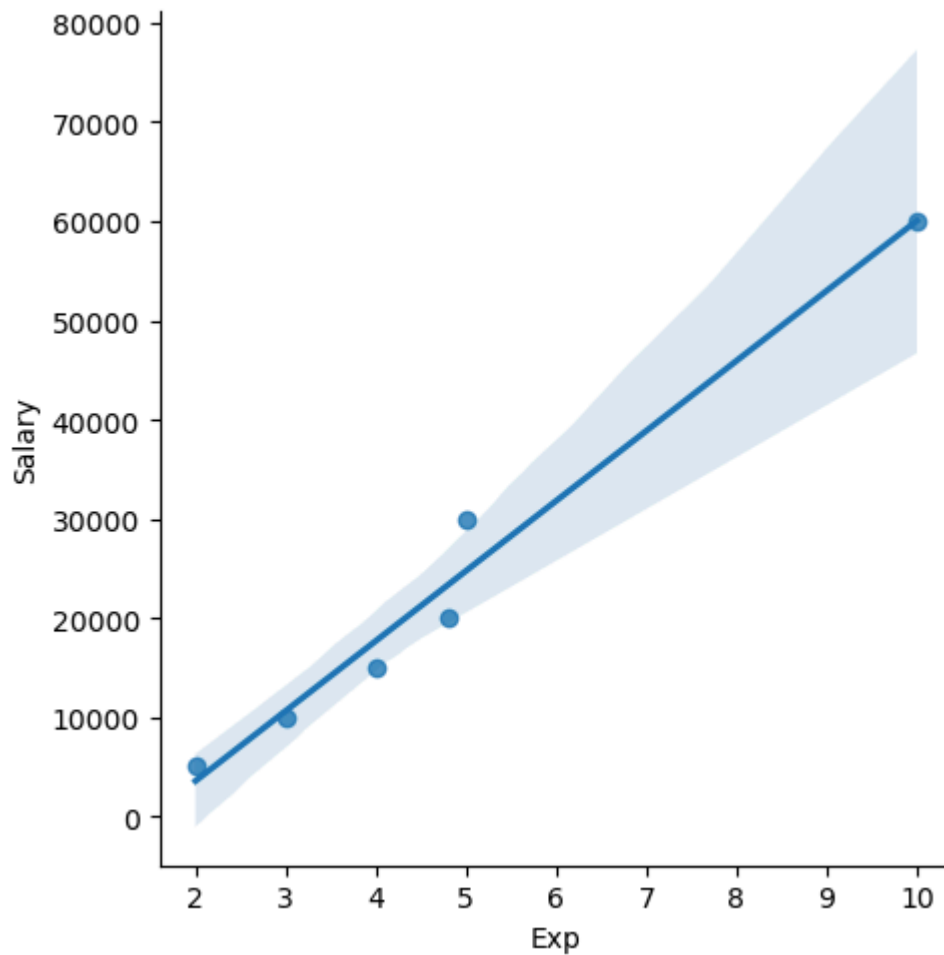
```
Out[89]: Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
```

```
In [90]: clean_data = clean_data.rename(columns={' Experience ': 'Exp', ' Salary ': 'Sala
```

```
In [91]: clean_data['Exp'] = clean_data['Exp'].astype(float)
clean_data['Salary'] = clean_data['Salary'].astype(float)
```

```
In [92]: sns.lmplot(x='Exp', y='Salary', data=clean_data)
plt.show()
```



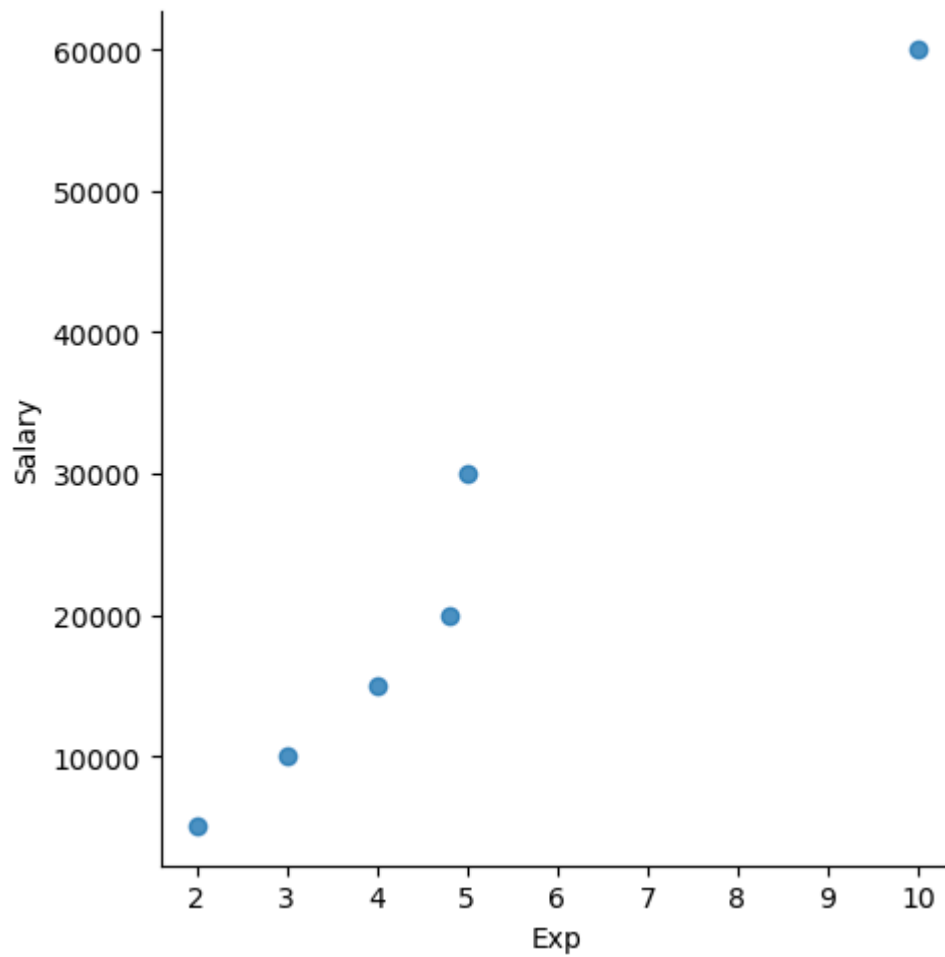


```
In [93]: vis5 = sns.lmplot(data=clean_data, x = 'Exp', y='Salary', fit_reg = False)
```

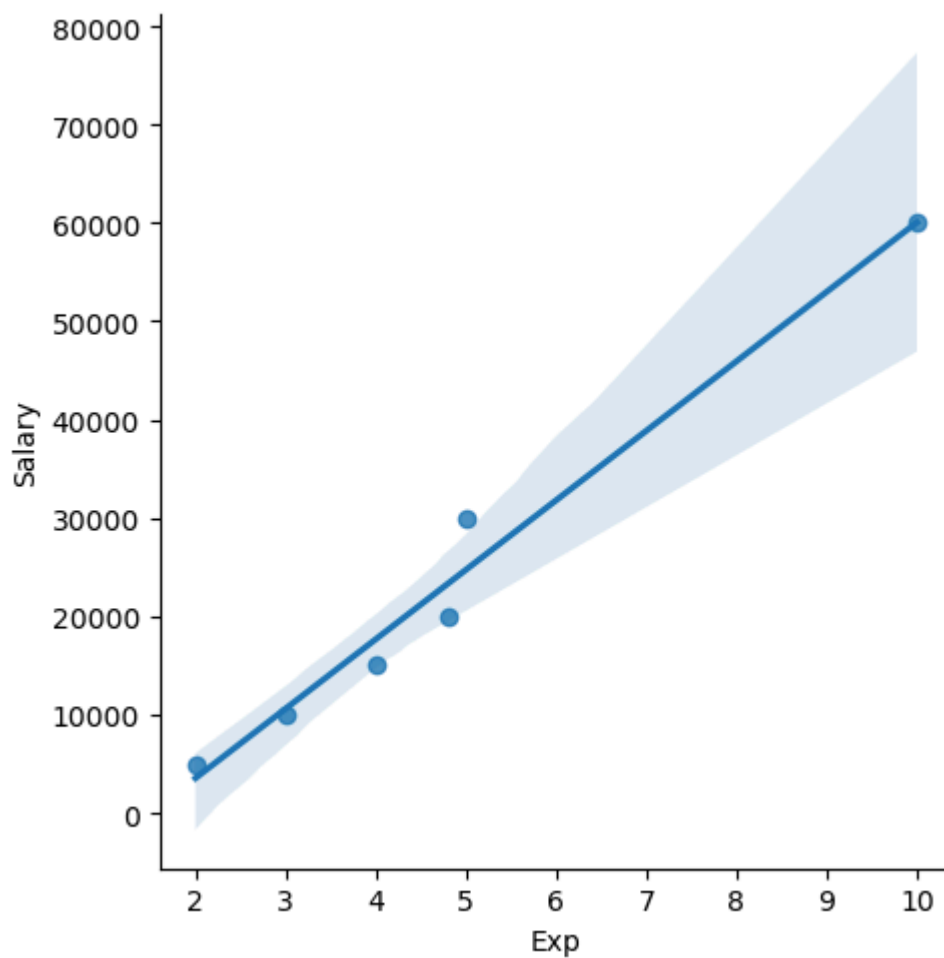
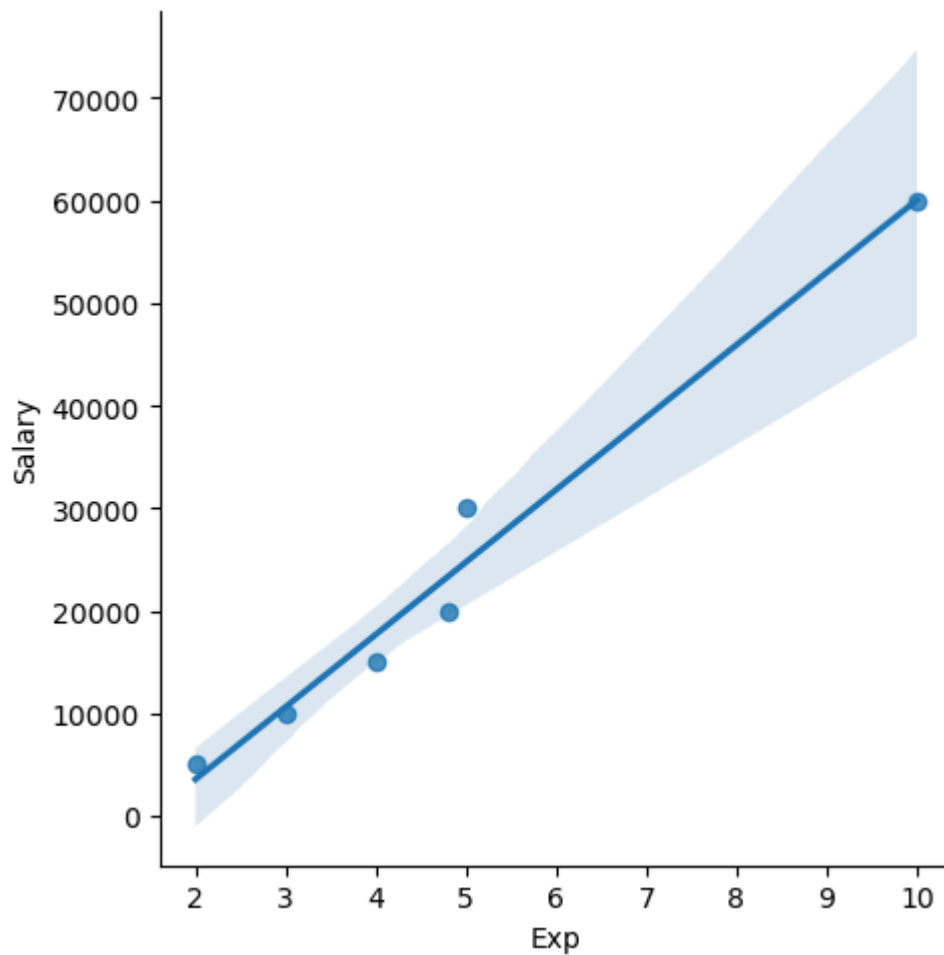
```
In [94]: vis5
```

```
Out[94]: <seaborn.axisgrid.FacetGrid at 0x1bdf7d8c050>
```

```
In [95]: plt.show()
```



```
In [97]: vis6 = sns.lmplot(data=clean_data, x = 'Exp', y='Salary', fit_reg = True)
vis6
plt.show()
```



```
In [98]: clean_data
```

Out[98]:

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000.0	2.0
1	Teddy	Testing	45	Bangalore	10000.0	3.0
2	Umar	Dataanalyst	50.25	Bangalore	15000.0	4.0
3	Jane	Analytics	50.25	Hyderbad	20000.0	4.8
4	Uttam	Statistics	67	Bangalore	30000.0	5.0
5	Kim	NLP	55	Delhi	60000.0	10.0

In [99]:

clean_data[:,]

Out[99]:

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000.0	2.0
1	Teddy	Testing	45	Bangalore	10000.0	3.0
2	Umar	Dataanalyst	50.25	Bangalore	15000.0	4.0
3	Jane	Analytics	50.25	Hyderbad	20000.0	4.8
4	Uttam	Statistics	67	Bangalore	30000.0	5.0
5	Kim	NLP	55	Delhi	60000.0	10.0

In [100...]

clean_data[:,2]

Out[100...]

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000.0	2.0
1	Teddy	Testing	45	Bangalore	10000.0	3.0

In [101...]

clean_data[2:,]

Out[101...]

	Name	Domain	Age	Location	Salary	Exp
2	Umar	Dataanalyst	50.25	Bangalore	15000.0	4.0
3	Jane	Analytics	50.25	Hyderbad	20000.0	4.8
4	Uttam	Statistics	67	Bangalore	30000.0	5.0
5	Kim	NLP	55	Delhi	60000.0	10.0

In [102...]

clean_data[:,]

Out[102...

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000.0	2.0
1	Teddy	Testing	45	Bangalore	10000.0	3.0
2	Umar	Dataanalyst	50.25	Bangalore	15000.0	4.0
3	Jane	Analytics	50.25	Hyderbad	20000.0	4.8
4	Uttam	Statistics	67	Bangalore	30000.0	5.0
5	Kim	NLP	55	Delhi	60000.0	10.0

In [103...

clean_data[0:1]

Out[103...

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000.0	2.0

In [106...

clean_data[0:6:2]

Out[106...

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000.0	2.0
2	Umar	Dataanalyst	50.25	Bangalore	15000.0	4.0
4	Uttam	Statistics	67	Bangalore	30000.0	5.0

In [107...

clean_data[::-1]

Out[107...

	Name	Domain	Age	Location	Salary	Exp
5	Kim	NLP	55	Delhi	60000.0	10.0
4	Uttam	Statistics	67	Bangalore	30000.0	5.0
3	Jane	Analytics	50.25	Hyderbad	20000.0	4.8
2	Umar	Dataanalyst	50.25	Bangalore	15000.0	4.0
1	Teddy	Testing	45	Bangalore	10000.0	3.0
0	Mike	Datascience	34	Mumbai	5000.0	2.0

In [108...

clean_data.columns

Out[108...

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

In [109...

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

In [110...

x_iv

Out[110...

	Name	Domain	Age	Location	Exp
0	Mike	Datascience	34	Mumbai	2.0
1	Teddy	Testing	45	Bangalore	3.0
2	Umar	Dataanalyst	50.25	Bangalore	4.0
3	Jane	Analytics	50.25	Hyderbad	4.8
4	Uttam	Statistics	67	Bangalore	5.0
5	Kim	NLP	55	Delhi	10.0

In [111...

y_dv=clean_data['Salary']

In [112...

y_dv

Out[112...

```
0    5000.0
1   10000.0
2   15000.0
3   20000.0
4   30000.0
5   60000.0
Name: Salary, dtype: float64
```

In [113...

emp

Out[113...

	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 [114...

clean_data

Out[114...

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000.0	2.0
1	Teddy	Testing	45	Bangalore	10000.0	3.0
2	Umar	Dataanalyst	50.25	Bangalore	15000.0	4.0
3	Jane	Analytics	50.25	Hyderbad	20000.0	4.8
4	Uttam	Statistics	67	Bangalore	30000.0	5.0
5	Kim	NLP	55	Delhi	60000.0	10.0

In [115...

x_iv

Out[115...

	Name	Domain	Age	Location	Exp
0	Mike	Datascience	34	Mumbai	2.0
1	Teddy	Testing	45	Bangalore	3.0
2	Umar	Dataanalyst	50.25	Bangalore	4.0
3	Jane	Analytics	50.25	Hyderbad	4.8
4	Uttam	Statistics	67	Bangalore	5.0
5	Kim	NLP	55	Delhi	10.0

In [116...

y_dv

Out[116...

```
0    5000.0
1   10000.0
2   15000.0
3   20000.0
4   30000.0
5   60000.0
Name: Salary, dtype: float64
```

In [117...

clean_data

Out[117...

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000.0	2.0
1	Teddy	Testing	45	Bangalore	10000.0	3.0
2	Umar	Dataanalyst	50.25	Bangalore	15000.0	4.0
3	Jane	Analytics	50.25	Hyderbad	20000.0	4.8
4	Uttam	Statistics	67	Bangalore	30000.0	5.0
5	Kim	NLP	55	Delhi	60000.0	10.0

In [118...

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

Out[118...

	Salary	Exp	Name_Jane	Name_Kim	Name_Mike	Name_Teddy	Name_Umar	Nam
0	5000.0	2.0	False	False	True	False	False	
1	10000.0	3.0	False	False	False	True	False	
2	15000.0	4.0	False	False	False	False	True	
3	20000.0	4.8	True	False	False	False	False	
4	30000.0	5.0	False	False	False	False	False	
5	60000.0	10.0	False	True	False	False	False	

6 rows × 23 columns



```
In [119... imputation = pd.get_dummies(clean_data).astype('int')
```

```
In [120... imputation
```

```
Out[120...      Salary  Exp  Name_Jane  Name_Kim  Name_Mike  Name_Teddy  Name_Umar  Name_
```

	Salary	Exp	Name_Jane	Name_Kim	Name_Mike	Name_Teddy	Name_Umar	Name_
0	5000	2	0	0	1	0	0	
1	10000	3	0	0	0	1	0	
2	15000	4	0	0	0	0	1	
3	20000	4	1	0	0	0	0	
4	30000	5	0	0	0	0	0	
5	60000	10	0	1	0	0	0	

6 rows × 23 columns



```
In [121... x_iv=clean_data.drop(['Salary'],axis=1)
```

```
In [122... clean_data
```

```
Out[122...      Name  Domain  Age  Location  Salary  Exp
```

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience	34	Mumbai	5000.0	2.0
1	Teddy	Testing	45	Bangalore	10000.0	3.0
2	Umar	Dataanalyst	50.25	Bangalore	15000.0	4.0
3	Jane	Analytics	50.25	Hyderbad	20000.0	4.8
4	Uttam	Statistics	67	Bangalore	30000.0	5.0
5	Kim	NLP	55	Delhi	60000.0	10.0

```
In [ ]:
```