In [128…
```
import pandas as pd
```

In [129…
```
pd.__version__
```

Out[129… '2.2.2'

In [130…
```
dataFile = pd.read_excel(r'C:\Python_Learning\Week5_EDA\20thFeb\RawData.xlsx')
```

In [131…
```
dataFile
```

Out[131…

	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 [132…
```
id(dataFile)
```

Out[132… 1455084775776

In [133…
```
dataFile.shape
```

Out[133… (6, 6)

In [134…
```
len(dataFile)
```

Out[134… 6

In [135…
```
dataFile.columns
```

Out[135… Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')

In [136…
```
dataFile.describe()
```

Out[136…

	Name	Domain	Age	Location	Salary	Exp
count	6	6	4	4	6	5
unique	6	6	4	4	6	5
top	Mike	Datascience#$	34 years	Mumbai	5^00#0	2+
freq	1	1	1	1	1	1

In [137…
```
dataFile.head()
```

Out[137…

	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 [138…
```
dataFile.tail()
```

Out[138…

	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 [139…
```
dataFile.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 [140…
```
dataFile.isnull()
```

Out[140…

	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 [141…
```
dataFile.isnull().sum()
```

Out[141…
```
Name        0
Domain      0
Age         2
Location    2
Salary      0
Exp         1
dtype: int64
```

In [142…
```
dataFile.isna()
```

Out[142…

	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 [143…
```
dataFile['Name']
```

Out[143…
```
0      Mike
1    Teddy^
2     Uma#r
3      Jane
4    Uttam*
5       Kim
Name: Name, dtype: object
```

In [144…
```
dataFile['Name'] = dataFile['Name'].str.replace(r'[^a-zA-Z]', '', regex=True)
```

In [145…
```
dataFile['Name']
```

Out[145…
```
0     Mike
1    Teddy
2     Umar
3     Jane
4    Uttam
5      Kim
Name: Name, dtype: object
```

In [146…
```
dataFile['Domain'] = dataFile['Domain'].str.replace(r'\W', '', regex=True)
```

In [147…
```
dataFile
```

Out[147…

	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 [148…
```
dataFile['Age'] = dataFile['Age'].str.replace(r'[^0-9]', '', regex=True)
```

In [149…
```
dataFile
```

Out[149…

	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 [150…
```
dataFile['Age'] = dataFile['Age'].str.extract(r'(\d+)')
```

In [151…
```
dataFile['Location'] = dataFile['Location'].str.replace(r'\W', '', regex=True)
dataFile['Salary'] = dataFile['Salary'].str.replace(r'\W', '', regex=True)
```

In [152…
```
dataFile['Exp'] = dataFile['Exp'].str.extract(r'(\d+)')
```

In [153…
```
dataFile
```

Out[153…

	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 [154…
```
clean_data = dataFile.copy()
```

In [155…
```
clean_data
```

Out[155…

	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 [156…
```
# EDA TECHNIQUES
clean_data.isnull().sum()
```

Out[156…
```
Name        0
Domain      0
Age         2
Location    2
Salary      0
Exp         1
dtype: int64
```

In [125…
```
clean_data['Age'].mean()
```

Out[125…
```
50.166666666666664
```

In [32]:
```
import numpy as np
# FILL MISSING VALS
```

In [159…
```
clean_data['Age'] = pd.to_numeric(clean_data['Age'])
clean_data['Age'] = clean_data['Age'].fillna(clean_data['Age'].mean())
```

In [160…
```
clean_data['Age']
```

Out[160…
```
0    34.00
1    45.00
2    50.25
3    50.25
4    67.00
5    55.00
Name: Age, dtype: float64
```

In [35]:
```
clean_data['Exp'] = clean_data['Exp'].fillna(np.mean(pd.to_numeric(clean_data['Exp'
```

In [36]:
```
clean_data
```

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	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 [37]:
```
clean_data['Location'].isnull().sum()
```

Out[37]:
```
2
```

In [38]:
```
clean_data['Location'] = clean_data['Location'].fillna(clean_data['Location'].mode(
```

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	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 [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       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 [41]:
```
clean_data['Age'] = clean_data['Age'].astype(int)
clean_data['Salary'] = clean_data['Salary'].astype(int)
clean_data['Exp'] = clean_data['Exp'].astype(int)
```

In [42]:
```
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 [43]:
```
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 [44]:
```
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 [45]:
```
clean_data.to_csv(r'C:\Python_Learning\Week5_EDA\20thFeb\CleanData.xlsx')
```

In [46]:
```
import os
os.getcwd()
```

Out[46]:
```
'C:\\Users\\bmittipa'
```

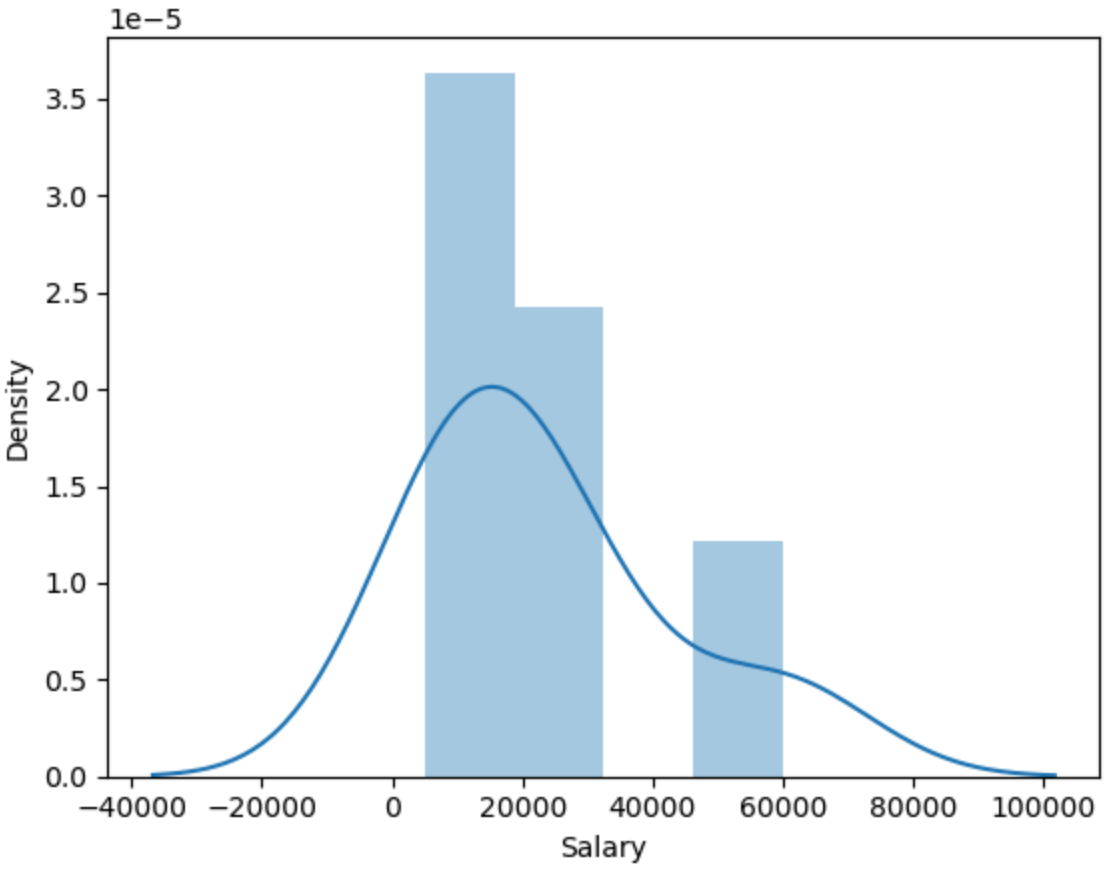
In [47]:
```
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

In [48]:
```
# Visualization
################### UNIVARIATE ANALYSIS#######################
clean_data['Salary']
```

Out[48]:
```
0     5000
1    10000
2    15000
3    20000
4    30000
5    60000
Name: Salary, dtype: int32
```

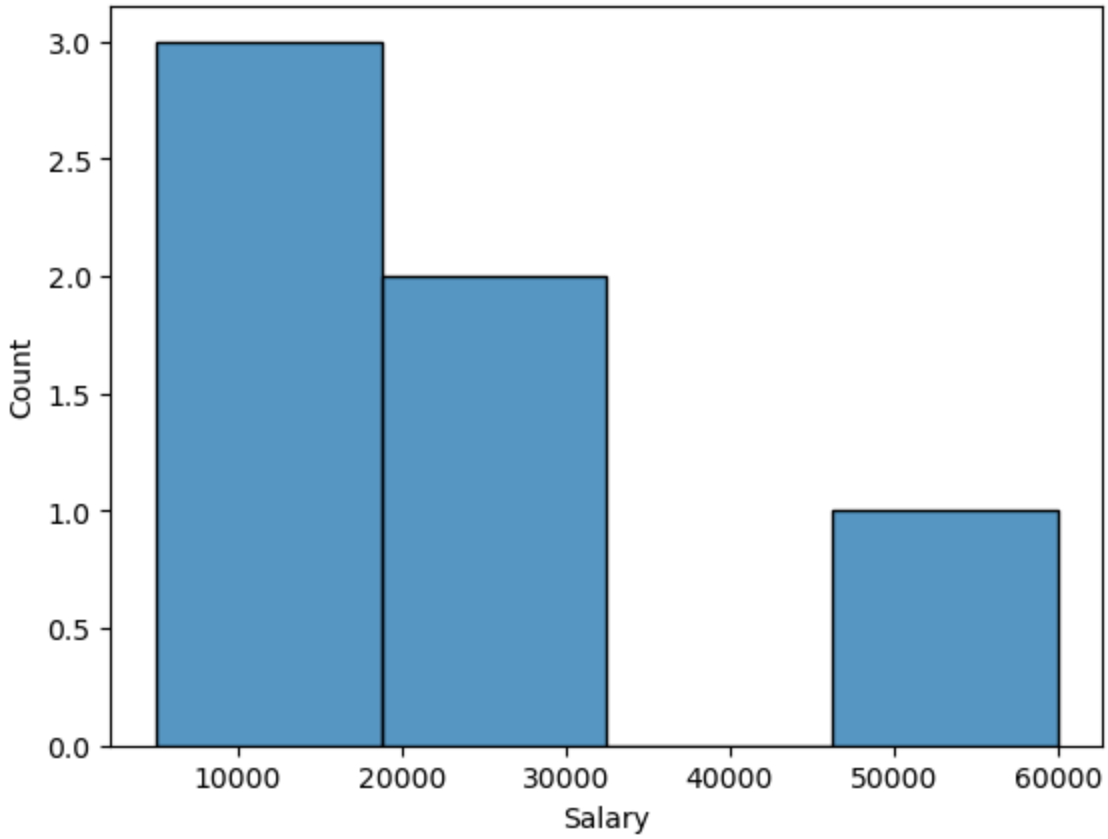
In [49]:
```
sns.distplot(clean_data['Salary'])
```

Out[49]:
```
<Axes: xlabel='Salary', ylabel='Density'>
```



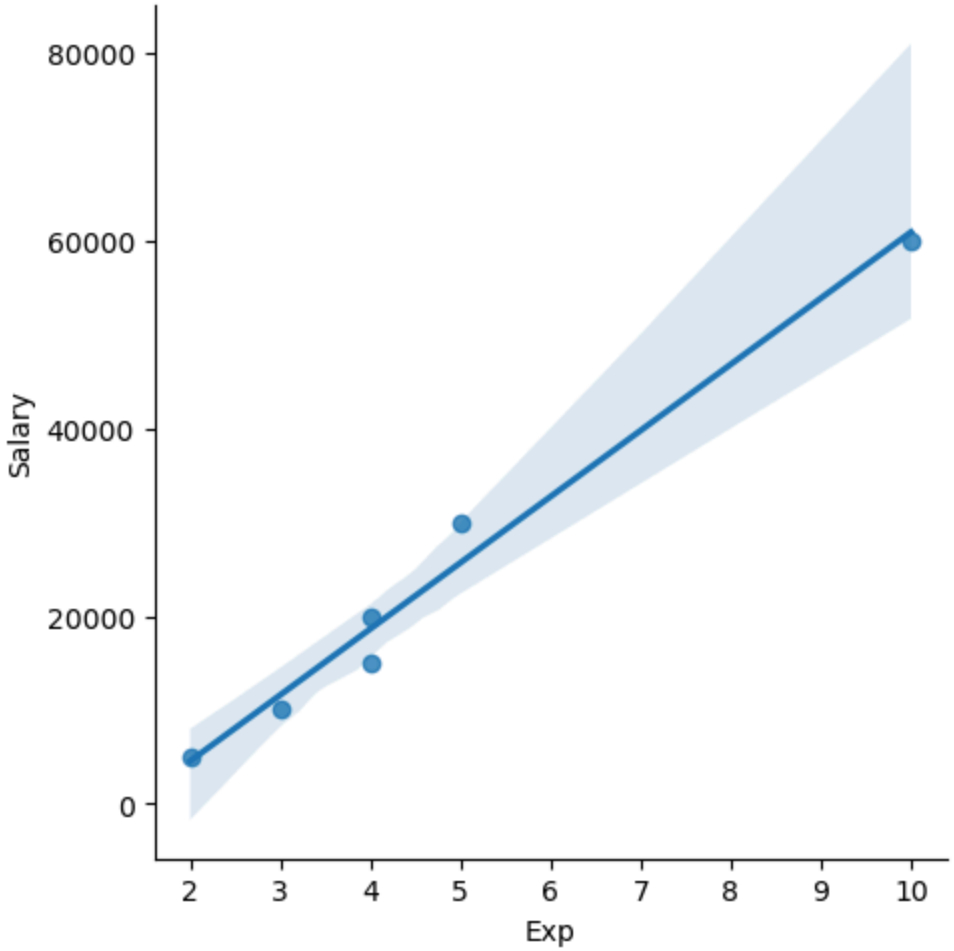
In [50]:
```
sns.histplot(clean_data['Salary'])
```

Out[50]: <Axes: xlabel='Salary', ylabel='Count'>



In [51]:
```
################### BIVARIATE ANALYSIS#########################
sns.lmplot(data=clean_data, x='Exp',y='Salary')
```

Out[51]: <seaborn.axisgrid.FacetGrid at 0x152c665d3d0>



In [62]:
```
clean_data.columns
```

Out[62]:
```
Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
```

In [55]:
```
X_vars = clean_data[['Name', 'Domain', 'Age', 'Location', 'Exp']]
```

In [61]:
```
X_vars
```

Out[61]:

	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 [59]:
```
Y_vars = clean_data[['Salary']]
```

In [60]:
```
Y_vars
```

Out[60]:

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

In [63]:
```
# IMPUTATION
imputation = pd.get_dummies(clean_data)
```

In [64]:
```
imputation
```

Out[64]:

	Age	Salary	Exp	Name_Jane	Name_Kim	Name_Mike	Name_Teddy	Name_Umar	Nan
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 [ ]: