```
import pandas as pd
In [1]:
In [2]: emp = pd.read_excel(r'C:\Users\kripal\EDA\Rawdata.xlsx')
In [3]:
        emp
Out[3]:
                          Domain
            Name
                                      Age
                                            Location
                                                        Salary
                                                                   Exp
         0
              Mike
                     Datascience#$ 34 years
                                             Mumbai
                                                       5^00#0
                                                                    2+
         1 Teddy^
                                     45' yr Bangalore
                                                                    <3
                           Testing
                                                      10%%000
         2
            Uma#r
                   Dataanalyst^^#
                                      NaN
                                                NaN
                                                      1$5%000
                                                                 4> yrs
         3
                       Ana^^lytics
                                      NaN Hyderbad
              Jane
                                                       2000^0
                                                                  NaN
         4
            Uttam*
                          Statistics
                                     67-yr
                                                                5+ year
                                                NaN
                                                        30000-
         5
                              NLP
                                                                   10+
               Kim
                                      55yr
                                                Delhi
                                                      6000^$0
In [4]:
        emp.shape
Out[4]: (6, 6)
In [5]:
        len(emp)
Out[5]: 6
In [6]: emp.columns
Out[6]: Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
In [7]: 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
            Domain
                      6 non-null
                                       object
        1
        2
            Age
                      4 non-null
                                       object
        3
            Location 4 non-null
                                       object
        4
                      6 non-null
                                       object
            Salary
            Exp
                      5 non-null
                                       object
       dtypes: object(6)
       memory usage: 420.0+ bytes
In [8]: emp.isnull().sum()
Out[8]:
        Name
         Domain
                     0
         Age
                     2
                     2
         Location
                     0
         Salary
         Exp
         dtype: int64
```

```
In [9]:
          emp
 Out[9]:
              Name
                            Domain
                                               Location
                                         Age
                                                            Salary
                                                                       Ехр
          0
               Mike
                       Datascience#$
                                     34 years
                                                Mumbai
                                                           5^00#0
                                                                        2+
             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^$0
                                                                       10+
          emp['Name']
In [10]:
Out[10]:
                  Mike
               Teddy^
          1
          2
                 Uma#r
          3
                  Jane
          4
                Uttam*
          5
                   Kim
          Name: Name, dtype: object
In [11]:
          emp['Name'] = emp['Name'].str.replace(r'\W','',regex=True)
Out[11]:
             Name
                           Domain
                                        Age
                                              Location
                                                           Salary
                                                                      Exp
          0
              Mike
                      Datascience#$
                                    34 years
                                               Mumbai
                                                          5^00#0
                                                                       2+
          1
              Teddy
                            Testing
                                       45' yr
                                              Bangalore
                                                        10%%000
                                                                       <3
                     Dataanalyst^^#
          2
              Umar
                                        NaN
                                                  NaN
                                                         1$5%000
                                                                    4> yrs
          3
               Jane
                        Ana^^lytics
                                              Hyderbad
                                                          2000^0
                                                                     NaN
                                        NaN
          4
             Uttam
                           Statistics
                                       67-yr
                                                  NaN
                                                           30000-
                                                                   5+ year
                               NLP
          5
               Kim
                                        55yr
                                                  Delhi
                                                         6000^$0
                                                                      10+
In [12]:
          emp['Domain'] = emp['Domain'].str.replace(r'\W','',regex=True)
          emp['Age'] = emp['Age'].str.replace(r'\W','',regex=True)
In [13]:
          emp['Location'] = emp['Location'].str.replace(r'\W','',regex=True)
          emp['Salary'] = emp['Salary'].str.replace(r'\W','',regex=True)
In [14]:
          emp
```

```
Out[14]:
             Name
                       Domain
                                   Age
                                         Location Salary
                                                             Exp
          0
              Mike Datascience 34years
                                          Mumbai
                                                    5000
                                                              2+
             Teddy
                        Testing
                                   45yr Bangalore
                                                   10000
                                                              <3
          2
             Umar
                    Dataanalyst
                                   NaN
                                             NaN
                                                   15000
                                                           4> yrs
          3
              Jane
                       Analytics
                                        Hyderbad
                                                   20000
                                                             NaN
                                   NaN
          4
            Uttam
                       Statistics
                                   67yr
                                             NaN
                                                   30000 5+ year
          5
               Kim
                           NLP
                                   55yr
                                            Delhi
                                                   60000
                                                             10+
          emp['Age'] = emp['Age'].str.extract('(\d+)') # give the output only numbers rest
In [15]:
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        <>:1: SyntaxWarning: invalid escape sequence '\d'
        C:\Users\kripal\AppData\Local\Temp\ipykernel_1896\3836655718.py:1: SyntaxWarning:
        invalid escape sequence '\d'
          emp['Age'] = emp['Age'].str.extract('(\d+)') # give the output only numbers res
        t of item gone #extract the digits
Out[15]:
             Name
                       Domain
                                Age
                                      Location Salary
                                                           Exp
          0
              Mike Datascience
                                  34
                                       Mumbai
                                                 5000
                                                           2+
                                                10000
             Teddy
                        Testing
                                  45 Bangalore
                                                            <3
          1
          2
             Umar
                    Dataanalyst NaN
                                          NaN
                                                15000
                                                        4> yrs
                                      Hyderbad
                                                          NaN
          3
              Jane
                       Analytics NaN
                                                20000
          4
            Uttam
                       Statistics
                                  67
                                          NaN
                                                30000
                                                       5+ year
          5
               Kim
                           NLP
                                  55
                                          Delhi
                                                60000
                                                          10+
         emp['Exp'] = emp['Exp'].str.replace(r'\W','',regex=True)
In [16]:
          emp['Exp'] = emp['Exp'].str.extract('(\d+)')
        <>:2: SyntaxWarning: invalid escape sequence '\d'
        <>:2: SyntaxWarning: invalid escape sequence '\d'
        C:\Users\kripal\AppData\Local\Temp\ipykernel_1896\3744867905.py:2: SyntaxWarning:
        invalid escape sequence '\d'
          emp['Exp'] = emp['Exp'].str.extract('(\d+)')
In [17]:
         emp
```

Out[17]:		Name	Domain	Age	Location	on Sala	ry	Ехр
	0	Mike	Datascience	34	Mumb	bai 50	00	2
	1	Teddy	Testing	45	Bangalo	ore 100	00	3
	2	Umar	Dataanalyst	NaN	Na	aN 150	00	4
	3	Jane	Analytics	NaN	Hyderb	ad 200	00	NaN
	4	Uttam	Statistics	67	Na	aN 300	00	5
	5	Kim	NLP	55	De	elhi 600	00	10
Tn [19].	ρm	n[['	Name', 'Doma:	in'	1	Ago!	L	ווית
TII [TO].	CIII	PLL	ivalile , Dollia.	LII ,		Age ,		h 11
Out[18]:	Cili	Name	Domain			Age ,		h 11
	0	Name				Age ,		P]]
		Name Mike	Domain	Age 34	Ехр	Age ,		h 11
	0	Name Mike	Domain Datascience	Age 34 45	Exp 2	Age ,		Ь 11
	0	Name Mike Teddy Umar	Domain Datascience Testing	Age 34 45 NaN	2 3 4	Age ,		h]]
	0 1 2	Name Mike Teddy Umar	Domain Datascience Testing Dataanalyst	Age 34 45 NaN	2 3 4	Age		h]]

Missing Value Treatment

```
In [20]:
         clean_data = emp.copy()
          clean_data
In [21]:
Out[21]:
             Name
                        Domain
                                 Age
                                       Location Salary
                                                          Ехр
          0
              Mike
                                                            2
                    Datascience
                                   34
                                        Mumbai
                                                   5000
                                                            3
             Teddy
                         Testing
                                   45
                                       Bangalore
                                                  10000
                                                            4
          2
              Umar
                     Dataanalyst
                                 NaN
                                           NaN
                                                  15000
               Jane
                       Analytics
                                 NaN
                                       Hyderbad
                                                  20000 NaN
                                                            5
             Uttam
                       Statistics
                                   67
                                           NaN
                                                  30000
                           NLP
               Kim
                                   55
                                           Delhi
                                                  60000
                                                           10
In [22]:
          clean_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
                             RangeIndex: 6 entries, 0 to 5
                            Data columns (total 6 columns):
                                             Column
                                                                          Non-Null Count Dtype
                                                                                 6 non-null
                                0 Name
                                                                                                                                         object
                                1 Domain 6 non-null
                                                                                                                                         object
                                2 Age
                                                                            4 non-null
                                                                                                                                         object
                                3 Location 4 non-null
                                                                                                                                         object
                                           Salary
                                                                                6 non-null
                                                                                                                                         object
                                5
                                                                                 5 non-null
                                                                                                                                         object
                                              Exp
                             dtypes: object(6)
                             memory usage: 420.0+ bytes
                                 import numpy as np
In [23]:
                               clean_data.head(1)
In [24]:
Out[24]:
                                             Name
                                                                               Domain Age Location Salary Exp
                                  0
                                                Mike Datascience
                                                                                                                  34
                                                                                                                                 Mumbai
                                                                                                                                                                    5000
                                                                                                                                                                                                2
In [25]:
                               clean_data['Age']
Out[25]: 0
                                                       34
                                  1
                                                       45
                                  2
                                                   NaN
                                  3
                                                   NaN
                                                       67
                                  4
                                  5
                                                       55
                                  Name: Age, dtype: object
In [26]: clean_data['Age'] = clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(np.to_numeric(clean_data['Age'].fillna(
In [27]:
                               clean_data['Age']
Out[27]: 0
                                                               34
                                  1
                                                              45
                                  2
                                                   50.25
                                  3
                                                    50.25
                                  4
                                                               67
                                                              55
                                  Name: Age, dtype: object
In [28]:
                               emp
```

Out[28]:		Name	Domain	Age	Location	Salary	Ехр
	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 [29]:	6 1	ean_dat	2				
	CI						
Out[29]:		Name	Domain	Age	Location	Salary	Ехр
	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
				_			
n [30]:	cl	ean_dat	a['Exp'] = 0	clean_o	data['Exp'].filln	a(np.m
n [31]:	cl	ean_dat	a				
ut[31]:		Name	Domain	Age	Location	Salary	Ехр
	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
n [32]:	cl	ean_dat	a['Location] = c	lean_data['Locati	on'].f
In [33]:	cl	ean_dat	a				

```
Out[33]:
            Name
                     Domain
                              Age
                                    Location Salary Exp
         0
             Mike Datascience
                                34
                                     Mumbai
                                              5000
                                                      2
            Teddy
                      Testing
                                45 Bangalore
                                             10000
                                                      3
         2
            Umar
                  Dataanalyst 50.25
                                   Bangalore
                                                      4
                                             15000
         3
             Jane
                     Analytics 50.25
                                    Hyderbad
                                             20000
                                                     4.8
           Uttam
                     Statistics
                                   Bangalore
                                             30000
                                                      5
                                67
              Kim
                         NLP
                                55
                                             60000
                                                     10
                                       Delhi
In [34]: 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
            Domain
                     6 non-null
                                     object
        1
                    6 non-null
                                     object
        2 Age
        3 Location 6 non-null
                                     object
        4 Salary
                      6 non-null
                                     object
                      6 non-null
        5
            Exp
                                     object
        dtypes: object(6)
        memory usage: 420.0+ bytes
In [35]:
         clean_data['Age'] = clean_data['Age'].astype(int)
         clean_data['Salary'] = clean_data['Salary'].astype(int)
         clean_data['Exp'] = clean_data['Exp'].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
                                     int32
        3
            Location 6 non-null
                                     object
                      6 non-null
                                     int32
        4
            Salary
        5
                      6 non-null
                                     int32
            Exp
        dtypes: int32(3), object(3)
        memory usage: 348.0+ bytes
In [36]:
        clean_data[['Name','Domain']] = clean_data[['Name','Domain']].astype('category')
In [37]: clean_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
       RangeIndex: 6 entries, 0 to 5
       Data columns (total 6 columns):
           Column Non-Null Count Dtype
       --- -----
                   -----
                   6 non-null
        0 Name
                                  category
        1 Domain 6 non-null
                                 category
        2 Age
                   6 non-null
                                  int32
        3 Location 6 non-null
                                  object
          Salary
                    6 non-null
                                   int32
        5 Exp
                   6 non-null
                                   int32
       dtypes: category(2), int32(3), object(1)
       memory usage: 704.0+ bytes
In [38]: clean_data['Location'] = clean_data['Location'].astype('category') #capital mist
In [39]: clean_data.info()
       <class 'pandas.core.frame.DataFrame'>
       RangeIndex: 6 entries, 0 to 5
       Data columns (total 6 columns):
        # Column
                   Non-Null Count Dtype
                    -----
       --- -----
                   6 non-null
        0 Name
                                  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
           Exp
                    6 non-null
                                   int32
       dtypes: category(3), int32(3)
       memory usage: 866.0 bytes
In [40]: clean_data['Location'] = clean_data['Location'].astype('category')
In [41]: clean data = clean data.drop('location',axis=1)
```

```
KeyError
                                          Traceback (most recent call last)
Cell In[41], line 1
----> 1 clean_data = clean_data.drop('location',axis=1)
File ~\anaconda3\Lib\site-packages\pandas\core\frame.py:5581, in DataFrame.drop(s
elf, labels, axis, index, columns, level, inplace, errors)
   5433 def drop(
   5434
            self,
   5435
            labels: IndexLabel | None = None,
   (\ldots)
   5442
            errors: IgnoreRaise = "raise",
   5443 ) -> DataFrame | None:
            0.00
   5444
   5445
            Drop specified labels from rows or columns.
   5446
   (\ldots)
   5579
                    weight 1.0
                                    0.8
            0.00
   5580
-> 5581
            return super().drop(
   5582
                labels=labels,
   5583
                axis=axis,
                index=index,
   5584
   5585
                columns=columns,
                level=level,
   5586
   5587
                inplace=inplace,
   5588
                errors=errors,
   5589
            )
File ~\anaconda3\Lib\site-packages\pandas\core\generic.py:4788, in NDFrame.drop(s
elf, labels, axis, index, columns, level, inplace, errors)
   4786 for axis, labels in axes.items():
   4787
            if labels is not None:
                obj = obj._drop_axis(labels, axis, level=level, errors=errors)
-> 4788
   4790 if inplace:
            self._update_inplace(obj)
   4791
File ~\anaconda3\Lib\site-packages\pandas\core\generic.py:4830, in NDFrame._drop_
axis(self, labels, axis, level, errors, only_slice)
   4828
                new axis = axis.drop(labels, level=level, errors=errors)
   4829
            else:
-> 4830
                new axis = axis.drop(labels, errors=errors)
   4831
            indexer = axis.get_indexer(new_axis)
   4833 # Case for non-unique axis
   4834 else:
File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:7070, in Index.dro
p(self, labels, errors)
   7068 if mask.any():
  7069
            if errors != "ignore":
-> 7070
                raise KeyError(f"{labels[mask].tolist()} not found in axis")
   7071
            indexer = indexer[~mask]
   7072 return self.delete(indexer)
KeyError: "['location'] not found in axis"
```

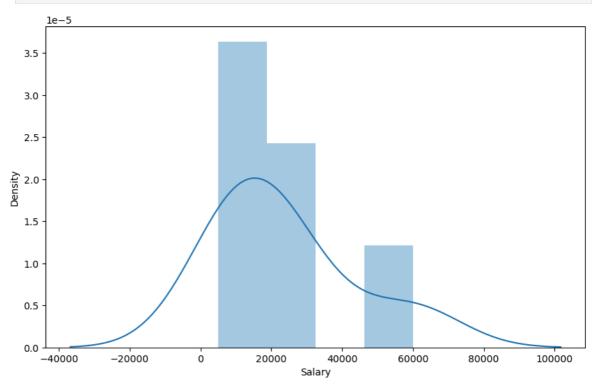
```
In [101... clean_data.info()
```

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 6 entries, 0 to 5

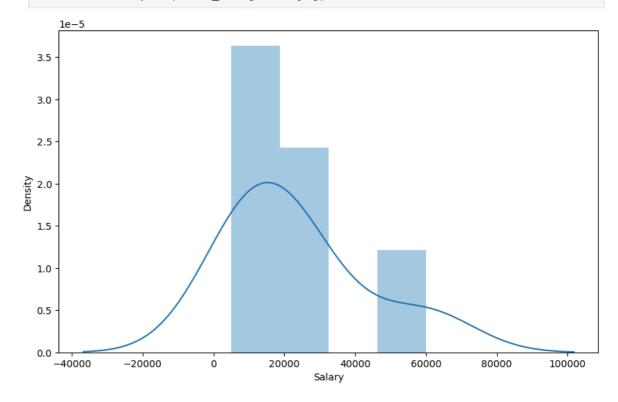
```
Data columns (total 6 columns):
             Column
                     Non-Null Count Dtype
         --- -----
                       -----
          0
             Name
                       6 non-null
                                       category
                                     category
          1 Domain 6 non-null
          2 Age
                      6 non-null
                                       int32
             Location 6 non-null
                                       category
          3
          4
             Salary
                       6 non-null
                                       int32
          5
                                       int32
                       6 non-null
              Exp
         dtypes: category(3), int32(3)
         memory usage: 866.0 bytes
In [103...
          clean_data
Out[103...
             Name
                       Domain Age Location Salary Exp
          0
              Mike Datascience
                                 34
                                      Mumbai
                                                5000
                                                        2
             Teddy
                        Testing
                                 45 Bangalore 10000
                                                        3
          1
          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 [105...
          clean_data.to_csv('clean_data.csv')
In [107...
          import os
          os.getcwd()
Out[107...
          'C:\\Users\\kripal'
In [108...
          clean_data.columns
Out[108...
          Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
In [109...
          import matplotlib.pyplot as plt
          %matplotlib inline
          import seaborn as sns
In [111...
          import warnings
          warnings.filterwarnings('ignore')
In [113...
          clean_data['Salary']
          0
Out[113...
                5000
          1
               10000
          2
               15000
          3
               20000
          4
               30000
               60000
          Name: Salary, dtype: int32
```

In [115... vis1 = sns.distplot(clean_data['Salary'])

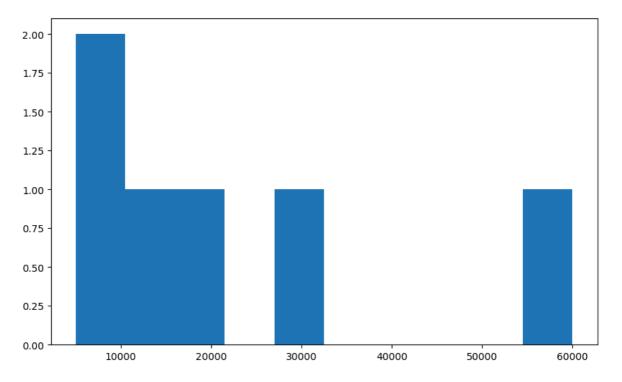


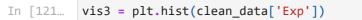
In [116... plt.rcParams['figure.figsize']=10,6

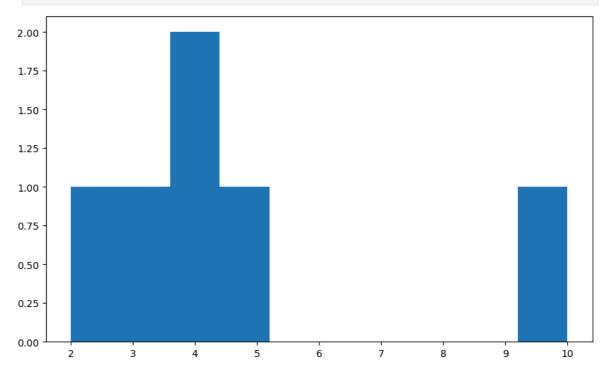
In [118... vis1 = sns.distplot(clean_data['Salary'])



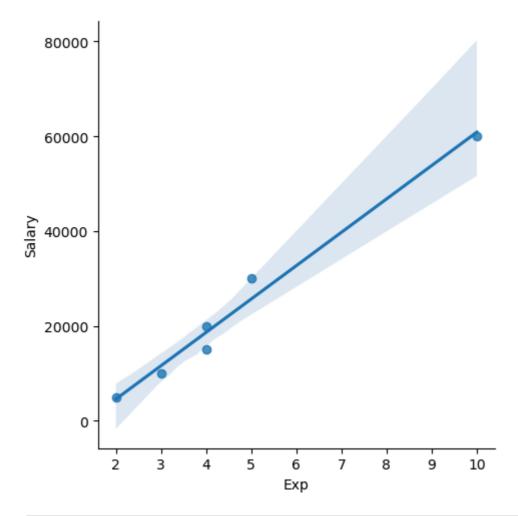
In [119... vis2 = plt.hist(clean_data['Salary'])

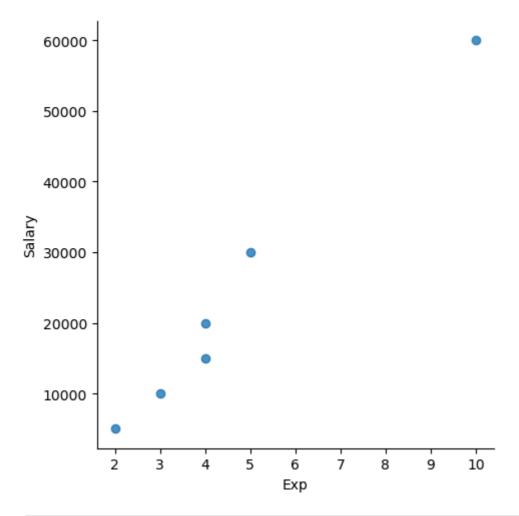


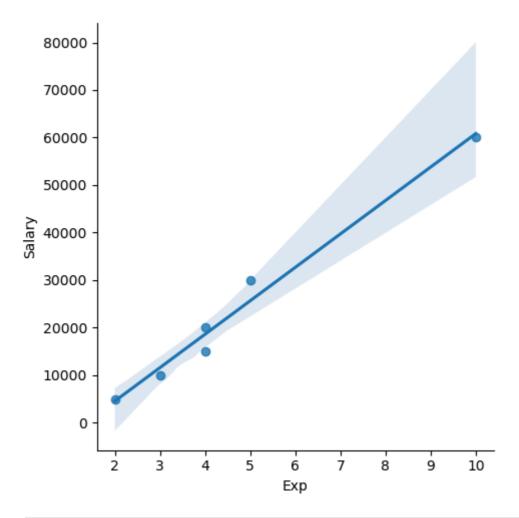




In [122... vis4 = sns.lmplot(data=clean_data,x='Exp',y = 'Salary')







In [125... clean_data[:]

Out[125...

	Name	Domain	Age	Location	Salary	Ехр
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 [126... clean_data[:2]

Out[126...

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

In [127... clean_data

```
Out[127...
               Name
                         Domain Age
                                         Location Salary Exp
           0
                Mike
                      Datascience
                                     34
                                          Mumbai
                                                     5000
                                                              2
               Teddy
                           Testing
                                     45
                                         Bangalore
                                                    10000
                                                              3
            2
               Umar
                       Dataanalyst
                                     50
                                         Bangalore
                                                    15000
                                                              4
            3
                Jane
                         Analytics
                                     50
                                         Hyderbad
                                                    20000
                                                              4
               Uttam
                         Statistics
                                     67
                                         Bangalore
                                                    30000
                                                              5
            5
                 Kim
                             NLP
                                     55
                                             Delhi
                                                    60000
                                                             10
           x_iv = clean_data.drop(['Salary'],axis=1)
In [128...
In [129...
           x_iv
Out[129...
               Name
                         Domain Age
                                         Location Exp
                Mike
                                                      2
            0
                      Datascience
                                     34
                                          Mumbai
               Teddy
                                         Bangalore
                                                      3
            1
                          Testing
                                    45
            2
               Umar
                       Dataanalyst
                                     50
                                         Bangalore
                                                      4
            3
                         Analytics
                                     50
                                         Hyderbad
                Jane
                                                      4
            4
               Uttam
                         Statistics
                                         Bangalore
                                                      5
                                     67
            5
                             NLP
                                     55
                                                     10
                 Kim
                                             Delhi
In [130...
           x_iv.columns
            Index(['Name', 'Domain', 'Age', 'Location', 'Exp'], dtype='object')
Out[130...
In [131...
           clean_data.columns
Out[131...
            Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
In [136...
           x_dv = clean_data.drop(['Name', 'Domain', 'Age', 'Location', 'Exp'],axis=1)
In [138...
           x_dv
Out[138...
               Salary
            0
                5000
               10000
            1
               15000
            2
               20000
            3
               30000
            4
               60000
In [139...
           clean_data
```

Out[139		Name	. De	omain	Age	Locati	on Salary	Ехр			
	0	Mike	Datas	cience	34	Muml	oai 5000	2			
	1	Teddy	, Т	esting	45	Bangalo	ore 10000	3			
	2	Umar	Dataa	analyst	50	Bangalo	ore 15000	4			
	3	Jane	e An	alytics	50	Hyderb	ad 20000	4			
	4	Uttam	Sta	atistics	67	Bangalo	ore 30000	5			
	5	Kim	ı	NLP	55	De	elhi 60000	10			
In [140	im	putati	on = po	d.get_	dummie	es(clear	n_data,dtyp	oe=int	:)		
In [141	im	putati	on								
Out[141		Age	Salary	Ехр	Name _.	_Jane N	Name_Kim	Name	_Mike	Name_Teddy	Name_Umar
Out[141	0	Age 34	Salary 5000	Exp 2	Name _.	_ Jane N	Name_Kim	Name	_Mike	Name_Teddy	Name_Umar
Out[141	0				Name _.			Name			
Out[141		34	5000	2	Name _.	0	0	Name	1	0	0
Out[141	1	34 45	5000	2	Name _.	0	0	Name	1 0	0	0
Out[141	1	34 45 50	5000 10000 15000	2 3 4	Name _.	0 0 0	0 0	Name	1 0 0	0 1 0	0 0 1
Out[141	1 2 3	34 45 50 50	5000 10000 15000 20000	2 3 4 4	Name_	0 0 0	0 0 0	Name	1 0 0	0 1 0	0 0 1 0
Out[141	1 2 3 4	34 45 50 50 67	5000 10000 15000 20000 30000	2 3 4 4 5	Name _.	0 0 0 1	0 0 0 0	Name	1 0 0 0	0 1 0 0	0 0 1 0