

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

```
In [3]: emp = pd.read_excel(r'C:\Users\Acer\Downloads\Rawdata.xlsx')
```

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

```
Out[5]:
```

Name	0
Domain	0
Age	2
Location	2
Salary	0
Exp	1
dtype: int64	

```
In [6]: emp.isna().sum()
```

```
Out[6]:
```

Name	0
Domain	0
Age	2
Location	2
Salary	0
Exp	1
dtype: int64	

```
In [8]: emp.columns
```

```
Out[8]: Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='object')
```

```
In [9]: emp.info
```

```
Out[9]: <bound method DataFrame.info of
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 [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['Name']`

```
Out[11]: 0      Mike
          1      Teddy^
          2      Uma#r
          3      Jane
          4      Uttam*
          5      Kim
Name: Name, dtype: object
```

In [12]: `emp['Name'] = emp['Name'].str.replace(r'\W', '', regex=True)`

In [13]: `emp['Name']`

```
Out[13]: 0      Mike
          1      Teddy
          2      Umar
          3      Jane
          4      Uttam
          5      Kim
Name: Name, dtype: object
```

In [14]: `emp`

	Name	Domain	Age	Location	Salary	Exp
<b>0</b>	Mike	Datascience#\$	34 years	Mumbai	5^00#0	2+
<b>1</b>	Teddy	Testing	45' yr	Bangalore	10%000	<3
<b>2</b>	Umar	Dataanalyst^^#	NaN	NaN	1\$5%000	4> yrs
<b>3</b>	Jane	Ana^^lytics	NaN	Hyderbad	2000^0	NaN
<b>4</b>	Uttam	Statistics	67-yr	NaN	30000-	5+ year
<b>5</b>	Kim	NLP	55yr	Delhi	6000^\$0	10+

In [15]: `emp['Domain'] = emp['Domain'].str.replace(r'\W', '', regex=True)`

In [16]: `emp`

Out[16]:

	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 [17]:

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

In [18]:

```
emp
```

Out[18]:

	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
5	Kim	NLP	55yr	Delhi	60000	10+

In [19]:

```
emp['Age'] = emp['Age'].str.extract('(\d+)') #To remove the characters after di
```

In [20]:

```
emp
```

Out[20]:

	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 [21]:

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

In [22]:

```
emp
```

Out[22]:

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascienc	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

## EDA Techniques

In [23]:

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

Out[23]:

	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascienc	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

## 1.Missing Value Treatment

In [24]:

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

Out[24]:

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

In [25]:

```
import numpy as np
```

In [26]:

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

In [27]:

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

```
Out[27]: 0      34
         1      45
         2    50.25
         3    50.25
         4      67
         5      55
Name: Age, dtype: object
```

```
In [28]: clean_data['Exp'] = clean_data['Exp'].fillna(np.mean(pd.to_numeric(clean_data['E
```

```
In [29]: clean_data['Exp']
```

```
Out[29]: 0      2
         1      3
         2      4
         3    4.8
         4      5
         5     10
Name: Exp, dtype: object
```

```
In [30]: 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	50.25	NaN	15000	4
3	Jane	Analytics	50.25	Hyderabad	20000	4.8
4	Uttam	Statistics	67	NaN	30000	5
5	Kim	NLP	55	Delhi	60000	10

```
In [31]: clean_data['Location'] = clean_data['Location'].fillna(np.mode(pd.to_numeric(cle
```

```
-----
AttributeError                                                 Traceback (most recent call last)
Cell In[31], line 1
----> 1 clean_data['Location'] = clean_data['Location'].fillna(np.mode(pd.to_numeric(clean_data['Location'])))

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:808, in __getattr__(attr)
  805     import numpy.char as char
  806     return char.chararray
--> 808 raise AttributeError(f"module {__name__!r} has no attribute {attr!r}")

AttributeError: module 'numpy' has no attribute 'mode'
```

```
In [33]: clean_data['Location'] = clean_data['Location'].fillna(clean_data['Location'].mo
```

**we need to use mode()[0], index to pass the value**

```
In [34]: clean_data['Location']
```

```
Out[34]: 0      Mumbai
         1    Bangalore
         2    Bangalore
         3   Hyderbad
         4    Bangalore
         5      Delhi
Name: Location, dtype: object
```

```
In [35]: clean_data
```

```
Out[35]:    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 [ ]:
```