Raw data to clean data conversion using python EDA

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

In [2]:
```
pd.__version__
```

Out[2]:
```
'1.4.2'
```

In [3]:
```
#pip install --upgrade openpyxl
```

In [4]:
```
emp = pd.read_excel(r"C:\Users\sidra\Downloads\Rawdata.xlsx")
```

In [5]:
```
emp
```

Out[5]:

	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 [6]:
```
id(emp)
```

Out[6]:
```
2627178245184
```

In [7]:
```
emp.columns
```

Out[7]:
```
Index(['Name', 'Domain', 'Age', 'Location', 'Salary', 'Exp'], dtype='objec
t')
```

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

Out[8]:
```
(6, 6)
```

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

Out[9]:
```
<bound method NDFrame.head of      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 [10]:
```
emp.tail
```

Out[10]:
```
<bound method NDFrame.tail of      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 [11]:
```
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: 416.0+ bytes
```

In [12]:
```
emp
```

Out[12]:

	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 [14]:
```
emp.isnull
```

Out[14]:
```
<bound method DataFrame.isnull of      Name          Domain       Age   Loca
tion   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 [15]:

```
emp.isnull()
```

Out[15]:

	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 [16]:

```
emp.isna()
```

Out[16]:

	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 [17]:

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

Out[17]:

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

In [18]:

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

Out[18]:

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

In [19]:

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

In [20]:
```
emp['Name']
```
Out[20]:
```
0     Mike
1    Teddy
2     Umar
3     Jane
4    Uttam
5      Kim
Name: Name, dtype: object
```

In [21]:
```
emp
```
Out[21]:

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

In [26]:
```
emp['Age']
```
Out[26]:
```
0    34years
1       45yr
2        NaN
3        NaN
4       67yr
5       55yr
Name: Age, dtype: object
```

In [28]:
```
emp['Age'] = emp['Age'].str.extract('(\\d+)')
```

In [29]:
```
emp['Age']
```

Out[29]:
```
0     34
1     45
2    NaN
3    NaN
4     67
5     55
Name: Age, dtype: object
```

In [30]:
```
emp
```

Out[30]:

	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 [31]:
```
emp['Location'] = emp['Location'].str.replace(r'\W','',regex=True)
```

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

Out[32]:
```
0       Mumbai
1    Bangalore
2          NaN
3     Hyderbad
4          NaN
5        Delhi
Name: Location, dtype: object
```

In [33]:
```
emp['Salary']
```

Out[33]:
```
0     5^00#0
1    10%%000
2    1$5%000
3     2000^0
4     30000-
5    6000^$0
Name: Salary, dtype: object
```

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.head()
```

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

In [37]:
```
emp['Exp']
```

Out[37]:
```
0         2+
1         <3
2     4> yrs
3        NaN
4    5+ year
5        10+
Name: Exp, dtype: object
```

In [40]:
```
emp['Exp'] = emp['Exp'].str.extract('(\\d+)')
```

In [42]:
```
emp['Exp']
```

Out[42]:
```
0      2
1      3
2      4
3    NaN
4      5
5     10
Name: Exp, dtype: object
```

In [43]:
```
clean_data = emp.copy()
```

In [44]:
```
clean_data
```

Out[44]:

	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 [45]:
```
# EDA TECHNIQUE
```

In [46]:
```
clean_data
```

Out[46]:

	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 [47]:
```
clean_data.isnull().sum()
```

Out[47]:
```
Name        0
Domain      0
Age         2
Location    2
Salary      0
Exp         1
dtype: int64
```

In [48]:
```
clean_data['Age']
```

Out[48]:
```
0     34
1     45
2    NaN
3    NaN
4     67
5     55
Name: Age, dtype: object
```

In [49]:
```
import numpy as np
```

In [50]:
```
clean_data['Age'] = clean_data['Age'].fillna(np.mean(pd.to_numeric(clean_data[
In [51]: clean_data['Age']
```

```
Out[51]: 0       34
         1       45
         2    50.25
         3    50.25
         4       67
         5       55
         Name: Age, dtype: object
```

```
In [52]: clean_data['Exp']
```

```
Out[52]: 0      2
         1      3
         2      4
         3    NaN
         4      5
         5     10
         Name: Exp, dtype: object
```

```
In [53]: clean_data['Exp'] = clean_data['Exp'].fillna(np.mean(pd.to_numeric(clean_data[
```

```
In [54]: clean_data['Exp']
```

```
Out[54]: 0      2
         1      3
         2      4
         3    4.8
         4      5
         5     10
         Name: Exp, dtype: object
```

```
In [55]: clean_data
```

Out[55]:

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

```
Out[56]: 2
```

```
In [57]: clean_data['Location']
```

```
Out[57]: 0       Mumbai
         1    Bangalore
         2          NaN
         3     Hyderbad
         4          NaN
         5        Delhi
         Name: Location, dtype: object
```

```
In [58]: clean_data['Location'] = clean_data['Location'].fillna(clean_data['Location'].
```

```
In [59]: clean_data['Location']
```

```
Out[59]: 0       Mumbai
         1    Bangalore
         2    Bangalore
         3     Hyderbad
         4    Bangalore
         5        Delhi
         Name: Location, dtype: object
```

```
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

```
In [61]: 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: 416.0+ bytes
```

In [62]:

```
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: 416.0+ bytes
```

In [63]:

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

In [64]:

```
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      object
 5   Exp       6 non-null      object
dtypes: int32(1), object(5)
memory usage: 392.0+ bytes
```

In [65]:

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

In [66]:

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

In [67]:

```
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: 344.0+ bytes
```

In [68]:

```
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 [69]:

```
clean_data
```

Out[69]:

	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

In [70]:

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

In [71]:

```
import os
os.getcwd()    # from the os give the saved current working directly
```

Out[71]:

```
'C:\\Users\\sidra'
```

In [72]:

```
clean_data
```

Out[72]:

	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