

```
!pip install --upgrade openpyxl
```

Requirement already satisfied: openpyxl in /usr/local/lib/python3.11/dist-packages (3.1.5)
Requirement already satisfied: et-xmlfile in /usr/local/lib/python3.11/dist-packages (1.0.1)

```
pd.__version__
```

'2.2.2'

```
import pandas as pd
```

```
emp=pd.read_excel(r"/content/Rawdata.xlsx")
```

Code cell output actions

	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+

Next steps:

[Generate code with emp](#)

[View recommended plots](#)

[New interactive sheet](#)

```
id(emp)
```

139941439312720


```
emp.columns
```

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



```
emp.shape
```

(6, 6)

```
emp.head()
```




	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



Next steps: [Generate code with emp](#) [View recommended plots](#) [New interactive sheet](#)


emp.tail()



	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+


Code cell output actions

emp.head()



	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

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

emp



	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^alytics	NaN	Hyderbad	2000^0	NaN	
4	Uttam*	Statistics	67-yr	NaN	30000-	5+ year	
5	Kim	NLP	55yr	Delhi	60000^\$0	10+	

Code cell output actions

Next steps:

[Generate code with emp](#)[View recommended plots](#)[New interactive sheet](#)

emp.isnull() #check missing value(false means missing value)



	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

emp.isna()



	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

emp.isnull().sum()



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

```
emp["Name"]
```

Code cell output actions



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

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

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



	Name
0	Mike
1	Teddy
2	Umar
3	Jane
4	Uttam
5	Kim
dtype:	object

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

```
emp["Age"]
```



	Age
0	34
1	45
2	NaN
3	NaN
4	67
5	55

dtype: object

Code cell output actions

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

```
emp["Location"]
```



	Location
0	Mumbai
1	Bangalore
2	NaN
3	Hyderbad
4	NaN
5	Delhi

dtype: object

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

```
emp["Domain"]
```



	Domain
0	Datascience
1	Testing
2	Dataanalyst
3	Analytics
4	Statistics
5	NLP

dtype: object

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

Code cell output actions

```
emp['Salary']
```



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

dtype: object

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


```
emp['Exp']
```



	Exp
0	2
1	3
2	4
3	NaN
4	5
5	10

dtype: object

emp




	Name	Domain	Age	Location	Salary	Exp
0	Mike	Datascience#	34		5^00#0	2
1	Teddy	Testing	45		10%%000	3
2	Umar	Dataanalyst^^	NaN	NaN	1\$5%000	4
3	Jane	Ana^alytics	NaN		2000^0	NaN
4	Uttam	Statistics	67	NaN	30000-	5
5	Kim	NLP	55		6000^\$0	10




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

Code cell output actions

clean_data



	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
3	Jane	Analytics	NaN	Hyderbad	20000	NaN
4	Uttam	Statistics	67-yr	NaN	30000	5
5	Kim	NLP	55yr	Delhi	60000	10



Next steps:

Generate code with clean_data

 View recommended plots

New interactive sheet

Start coding or [generate](#) with AI.

Start coding or [generate](#) with AI.