



```
In [35]: #PRACTICAL N02:Write a program in Python for Data Collection, Cleaning, and Ha
```

```
In [37]: #Name: Ritika R. Junekar
#Sub: PD
#Roll_No:29
```

```
In [39]: import pandas as pd
import numpy as np
from sklearn.impute import SimpleImputer
```

```
In [53]: # Create a sample DataFrame for demonstration
data = {
    'Age':      [25, 30, np.nan, 45, 50, 22],
    'Salary':   [50000, 60000, 75000, np.nan, 90000, 45000],
    'City':     ['New York', 'London', 'Paris', 'New York', np.nan, 'London'],
    'Experience': [2, 5, 10, 15, 20, np.nan]
}
```

```
In [55]: df = pd.DataFrame(data)
```

```
In [57]: print("--- Original Data ---")
print(df)
```

```
--- Original Data ---
   Age  Salary   City  Experience
0  25.0  50000.0 New York         2.0
1  30.0  60000.0  London         5.0
2   NaN  75000.0   Paris        10.0
3  45.0     NaN New York        15.0
4  50.0  90000.0    NaN         20.0
5  22.0  45000.0  London         NaN
```

```
In [59]: # Check missing values
print("\n--- Missing Values Count ---")
print(df.isnull().sum())
```

```
--- Missing Values Count ---
Age          1
Salary       1
City         1
Experience   1
dtype: int64
```

```
In [61]: # a) Numerical columns: mean imputation
numerical_cols = ['Age', 'Salary', 'Experience']
imputer_mean = SimpleImputer(missing_values=np.nan, strategy='mean')
df[numerical_cols] = imputer_mean.fit_transform(df[numerical_cols])
```

```
In [65]: # b) Categorical column: mode imputation
df['City'].fillna(df['City'].mode()[0], inplace=True)
print("\n--- Data After Imputation ---")
print(df)
```

```
print("\n--- Final Missing Values Check ---")
print(df.isnull().sum())
```

```
--- Data After Imputation ---
```

	Age	Salary	City	Experience
0	25.0	50000.0	New York	2.0
1	30.0	60000.0	London	5.0
2	34.4	75000.0	Paris	10.0
3	45.0	64000.0	New York	15.0
4	50.0	90000.0	London	20.0
5	22.0	45000.0	London	10.4

```
--- Final Missing Values Check ---
```

```
Age          0
Salary       0
City         0
Experience   0
dtype: int64
```

In []: