Page No.
Date

Devanshu Surana
PC-23, Panel C, Batch C1
1032210755

DEC Lab Assignment 2

Problem Statement : Use housing Dataset for Data Pre-processing. Apply Various data cleaning functions for following:

Handle missing values: Ignore, Defaults, Impute.
Handle duplicate: Identify, Remove smoothing noisy data, Resolve inconsistencies.

Objective

To clean data and make it noise free.
Prepare data for Analysis.

Theory:

What is Data Preprocessing?

→ Data preprocessing is a crucial step in data analysis and machine learning pipeline. It involves a series of operations and techniques applied to raw data to make it suitable for analysis or training machine learning models.

Here are some key steps and techniques involved in data preprocessing.

1. Data Cleaning
2. Data Transformation

Teacher's Sign.: _____

3. Data Reduction
4. Data Integration
5. Data formatting
6. Data Exploration
7. Handling Imbalanced data
8. Normalization
9. Data Splitting
10. Data Imputation.

Need of Data Pre-Processing

→ 1. Handling Missing data: Real world dataset often contain missing values due to various reasons such as sensor failures.

2. Removing duplicate data: Duplicate records can skew analysis result and mislead the model or user.

3. Data Transformation: Raw data often requires transf--ormation to make it suitable for analysis or modelling.

4. Data formatting: Ensure appropriate data for analysis / modelling.

5. Data Imputation: When missing values are present, data preprocessing techniques can impute or fill in those values.

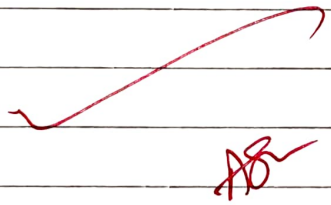
Page No.
Date

3. List of steps in Data Cleaning with example:

→ Data cleaning is process of correcting errors or inconsistency in data.

Steps:

- Removal of irrelevant or duplicate data
- fixing structural errors
- Dealing with missing data
- filtering out data outliers
- Validating data
- standardizing capitalization
- Converting data type
- Language translation



Teacher's Sign.: _____

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from google.colab import files
uploaded=files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.
Saving Iris (1).csv to Iris (1) (1).csv

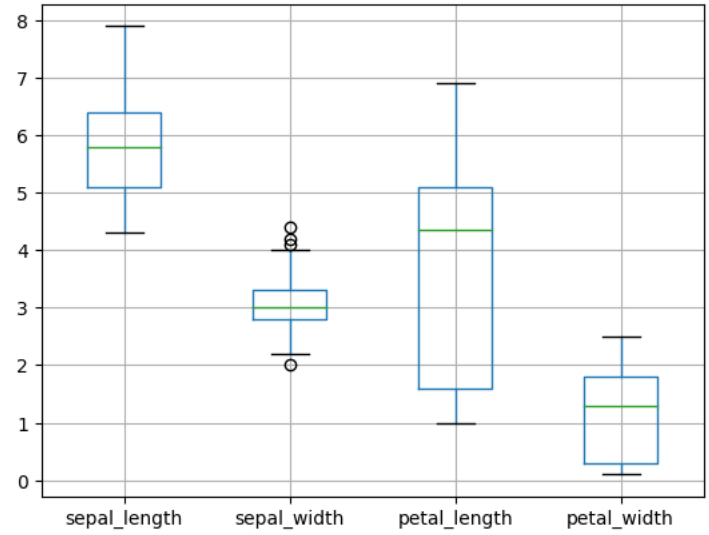
```
dataset=pd.read_csv('Iris (1).csv')
print(dataset)
```

```
     sepal_length  sepal_width  petal_length  petal_width         species
0             5.1          3.5           1.4          0.2     Iris-setosa
1             4.9          3.0           1.4          0.2     Iris-setosa
2             4.7          3.2           1.3          0.2     Iris-setosa
3             4.6          3.1           1.5          0.2     Iris-setosa
4             5.0          3.6           1.4          0.2     Iris-setosa
..            ...          ...           ...          ...             ...
145           6.7          3.0           5.2          2.3  Iris-virginica
146           6.3          2.5           5.0          1.9  Iris-virginica
147           6.5          3.0           5.2          2.0  Iris-virginica
148           6.2          3.4           5.4          2.3  Iris-virginica
149           5.9          3.0           5.1          1.8  Iris-virginica

[150 rows x 5 columns]
```

```
dataset.boxplot()
```

```
<Axes: >
```




```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from google.colab import files
uploaded=files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.
Saving sample_data.csv to sample_data.csv

```
ds=pd.read_csv('sample_data.csv')
```

```
ds.head()
```

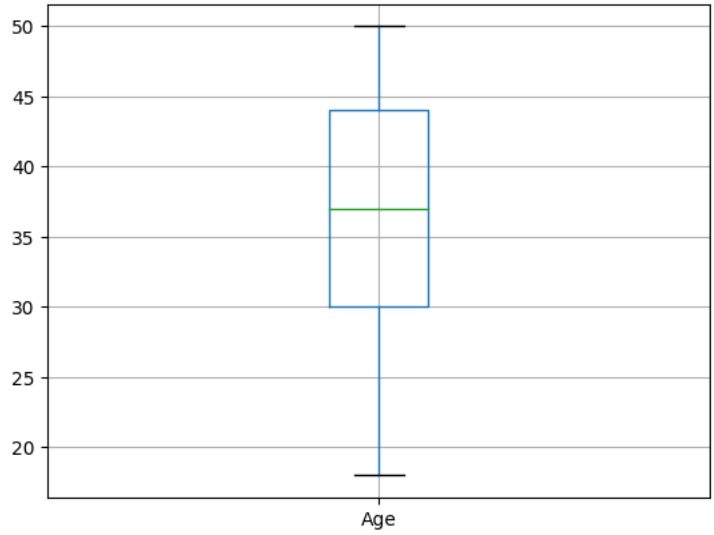
	Country	Age	Salary	Purchased
0	France	44.0	72000.0	No
1	Spain	27.0	48000.0	Yes
2	Germany	30.0	54000.0	No
3	Spain	38.0	61000.0	No
4	Nigeria	18.0	15000.0	No

```
ds.shape
```

```
(29, 4)
```

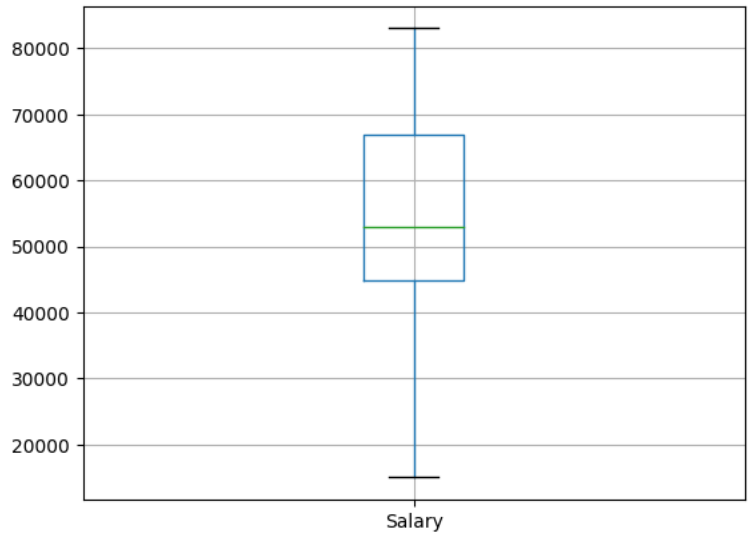
```
ds.boxplot("Age")
```

```
<Axes: >
```




```
ds.boxplot("Salary")
```

```
<Axes: >
```




```
ds.describe()
```

	Age	Salary
count	27.000000	28.000000
mean	36.925926	53642.857143
std	8.757089	19216.532785
min	18.000000	15000.000000
25%	30.000000	44750.000000
50%	37.000000	53000.000000
75%	44.000000	67000.000000
max	50.000000	83000.000000

```
dn=ds.isnull().sum()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-1-8982da57b083> in <cell line: 1>()
```

```
print(dn)
```

```
Country      1
Age          2
Salary       1
Purchased    1
dtype: int64
```

```
ds['Country'].fillna(99,inplace=True)
```

```
ds.head(15)
```

	Country	Age	Salary	Purchased
0	France	44.0	72000.0	No
1	Spain	27.0	48000.0	Yes
2	Germany	30.0	54000.0	No
3	Spain	38.0	61000.0	No
4	Nigeria	18.0	15000.0	No
5	Germany	40.0	NaN	Yes
6	France	35.0	58000.0	Yes
7	Spain	NaN	52000.0	No
8	France	48.0	79000.0	Yes
9	Germany	50.0	83000.0	No
10	France	37.0	67000.0	Yes
11	Nigeria	50.0	60000.0	Yes
12	France	22.0	30000.0	No
13	99	44.0	45000.0	Yes
14	France	47.0	78000.0	NaN

```
print("fill the missing values")
median=ds["Age"].median()
print(median)
```

```
fill the missing values
37.0
```

```
ds["Age"].fillna(99, inplace=True)
```

```
print(ds)
```

```
    Country   Age   Salary Purchased
0    France  44.0  72000.0        No
1     Spain  27.0  48000.0       Yes
2   Germany  30.0  54000.0        No
3     Spain  38.0  61000.0        No
4   Nigeria  18.0  15000.0        No
5   Germany  40.0      NaN       Yes
6    France  35.0  58000.0       Yes
7     Spain  99.0  52000.0        No
8    France  48.0  79000.0       Yes
9   Germany  50.0  83000.0        No
10   France  37.0  67000.0       Yes
11  Nigeria  50.0  60000.0       Yes
12   France  22.0  30000.0        No
13       99  44.0  45000.0       Yes
14   France  47.0  78000.0       NaN
15  Nigeria  35.0  43000.0       Yes
16    Spain  34.0  44000.0       Yes
17    Spain  27.0  48000.0       Yes
18    Spain  33.0  48000.0       Yes
19  Nigeria  29.0  77000.0       Yes
20    Spain  99.0  57000.0       Yes
21   France  44.0  48000.0       Yes
22  Germany  50.0  83000.0        No
23   France  37.0  67000.0       Yes
24   France  37.0  23000.0       Yes
25  Germany  45.0  50000.0        No
26   France  37.0  67000.0       Yes
```

```
27  Nigeria  30.0  30000.0       Yes
28  Nigeria  29.0  15000.0        No
```