**PRACTICAL: 01**

**AIM:** Create one-dimensional data using series and perform various operations on it.

**THEORY**

A one-dimensional array (or array) is a data structure that stores a sequence of (references to) objects. We refer to the objects within an array as its elements. The method that we use to refer to elements in an array is numbering and then indexing them.

Creating a series from Scalar value: In order to create a series from scalar value, an index must be provided. The scalar value will be repeated to match the length of the index.

**SOURCE CODE & OUTPUT**

**# Series**

import numpy as np

import pandas as pd

data = pd.Series([15,32,45,62,25,3,12,21])

data

0 15

1 32

2 45

3 62

4 25

5 3

6 12

7 21

dtype: int64

**# Accessing elements**

print("First element:-", data[0])

print("second Last element:-", data[len(data) - 2])

First element:- 15

second Last element:- 12

**# Sliced**

print("Sliced data (index 3 to 7):-")

print(data[3:7])

Sliced data (index 3 to 7):-

3 62

4 25

5 3

6 12

dtype: int64

**# Arithmetic operations:**

print("Add 2 to each element:-")

print(data+2)

print("Multiply each element by 3:-")

print(data \*3)

Add 2 to each element:-

0 17

1 34

2 47

3 64

4 27

5 5

6 14

7 23

dtype: int64

Multiply each element by 3:-

0 45

1 96

2 135

3 186

4 75

5 9

6 36

7 63

dtype: int64

**# Statistical operations**

print("Mean:-", data.mean())

print("Sum:-", data.sum())

print("Maximum :-", data.max())

print("Minimum :-", data.min())

Mean:- 26.875

Sum:- 215

Maximum :- 62

Minimum :- 3

**# Conditions**

print("Elements greater than 21:-")

print(data[data>21])

Elements greater than 21:-

1 32

2 45

3 62

4 25

dtype: int64