**Assessment 1:**

1. **Write a Python program to calculate the area of a rectangle given its length and width.**
2. #area of rectangle
3. # Python program to find the area of a rectangle using a lambda function
4. area\_simple\_multiplication = lambda length, width: length \* width
5. length = 5
6. width = 8
7. result = area\_simple\_multiplication(length, width)
8. print(f"Area of the Rectangle: {result:.2f}")

**O/P: Area of the Rectangle: 40.00**

**2. Write a program to convert miles to kilometers.**

#converting miles to kilometers

miles = float(input("Enter the value in miles: "))

conversion\_factor = 1.60934

kilometers = miles \* conversion\_factor

print('%.4f miles = %0.4f kilometers' %(miles, kilometers))

**O/P: Enter the value in miles: 9**

**9.0000 miles = 14.4841 kilometer**

**3. Write a function to check if a given string is a palindrome.**

#checking palindrome

def is\_palindrome(str):

    for i in range(0, int(len(str) / 2)):

        if str[i] != str[len(str) - i - 1]:

            return False

    return True

s = "malayalam"

ans = is\_palindrome(s)

if ans:

    print("Yes")

else:

    print("No")

**O/P: Yes**

**4. Write a Python program to find the second largest element in a list.**

#second largest number

def find\_second\_largest(arr):

    mx = max(arr[0], arr[1])

    second\_max = min(arr[0], arr[1])

    n = len(arr)

    for i in range(2, n):

        if arr[i] > mx:

            second\_max = mx

            mx = arr[i]

        elif arr[i] > second\_max and mx != arr[i]:

            second\_max = arr[i]

    return second\_max

my\_list = [10, 20, 4, 45, 99]

print("Second largest number is:", find\_second\_largest(my\_list))

**O/P: Second largest number is: 45**

**5. Explain what indentation means in Python.**

#python indentation

#Indentation refers to the spaces at the beginning of a code line.Where in other programming languages the indentation in code is for readability only, the indentation in Python is very important.Python uses indentation to indicate a block of code

**6. Write a program to perform set difference operation.**

#second largest number

def find\_second\_largest(arr):

    mx = max(arr[0], arr[1])

    second\_max = min(arr[0], arr[1])

    n = len(arr)

    for i in range(2, n):

        if arr[i] > mx:

            second\_max = mx

            mx = arr[i]

        elif arr[i] > second\_max and mx != arr[i]:

            second\_max = arr[i]

    return second\_max

my\_list = [10, 20, 4, 45, 99]

print("Second largest number is:", find\_second\_largest(my\_list))

O/P: largest number is: 45

**7. Write a Python program to print numbers from 1 to 10 using a while loop.**

#using while loop

# Python program to print numbers from 1 to 10 without a loop

def print\_num(n):

    if n > 0:

        print\_num(n - 1)

        print(n, end=' ')

print('Numbers from 1 to 10:')

print\_num(10)

O/P:Numbers from 1 to 10:

1 2 3 4 5 6 7 8 9 10

**8. Write a program to calculate the factorial of a number using a while loop.**

#factorial of a number using while loop

fact = 1

i = 1

# To take input from the user

num = int(input("Enter a number: "))

# check if the number is negative or positive

if num < 0:

    print("Sorry, factorial does not exist for negative numbers")

else:

    while i <= num:

        fact = fact\*i

        i = i + 1

    print("The factorial of", num, "is", fact)

O/P:Enter a number: 12

The factorial of 12 is 479001600

**9. Write a Python program to check if a number is positive, negative, or zero using if-elif-else**

**statements.** #if elif else

num = float(input("Enter a number: "))

if num > 0:

    print("Positive number")

elif num == 0:

    print("Zero")

else:

    print("Negative number")

O/P:Enter a number: -3

Negative number

**10. Write a program to determine the largest among three numbers using conditional**

**statements.**

#using conditional satement

# Program for finding the largest of three numbers in Python

a = int(input("Enter A: "))

b = int(input("Enter B: "))

c = int(input("Enter C: "))

# Conditions to find the largest

if a > b:

    if a > c:

        largest = a

    else:

        largest = c

else:

    if b > c:

        largest = b

    else:

        largest = c

print("Greater =", largest)

O/P:Enter A: 68

Enter B: 89

Enter C: 56

Greater = 89

**11.Write a Python program to create a numpy array filled with ones of given shape.**

#creare a numpy array

import numpy as np

# Create a 2D array with shape (3, 4) filled with ones

array\_shape = (3, 4)

ones\_array = np.ones(array\_shape)

print("Array of ones with shape", array\_shape)

print(ones\_array)

O/P:Array of ones with shape (3, 4)

[[1. 1. 1. 1.]

[1. 1. 1. 1.]

[1. 1. 1. 1.]]

**12.Write a program to create a 2D numpy array initialized with random integers.**

#2d array

import numpy as np

N = 5  # Change this to the desired size of your array

random\_2d\_array = np.random.random((N, N))

**13. Write a Python program to generate an array of evenly spaced numbers over a specified**

**range using linspace.**

#line space

import numpy as np

# Define the range and the desired number of points

lower\_bound = 0

upper\_bound = 5

num\_points = 10

# Generate an array of evenly spaced numbers (including the right endpoint)

evenly\_spaced\_array = np.linspace(lower\_bound, upper\_bound, num\_points)

print("Evenly spaced array (including the right endpoint):")

print(evenly\_spaced\_array)

# If you prefer to exclude the right endpoint:

evenly\_spaced\_array\_excluding\_endpoint = np.linspace(lower\_bound, upper\_bound, num\_points, endpoint=False)

print("\nEvenly spaced array (excluding the right endpoint):")

print(evenly\_spaced\_array\_excluding\_endpoint)

O/P:Evenly spaced array (including the right endpoint):

[0. 0.55555556 1.11111111 1.66666667 2.22222222 2.77777778

3.33333333 3.88888889 4.44444444 5. ]

Evenly spaced array (excluding the right endpoint):

[0. 0.5 1. 1.5 2. 2.5 3. 3.5 4. 4.5]

**14. Write a program to generate an array of 10 equally spaced values between 1 and 100 using**

**linspace.**

#to generate an array of 10 equally spaced values using line space

import numpy as np

# Generate an array of 10 equally spaced values between 1 and 100

my\_array = np.linspace(1, 100, 10)

# Print the resulting array

print(f"Equally spaced values between 1 and 100: {my\_array}")

O/P: Equally spaced values between 1 and 100: [ 1. 12. 23. 34. 45. 56. 67. 78. 89. 100.]

**15. Write a Python program to create an array containing even numbers from 2 to 20 using**

**arange.**

#python program to create  an array containing even numbers from 2 to 20 using arange

# Initialize an empty list

even\_numbers\_list = []

# Append even numbers from 2 to 20

for i in range(2, 21, 2):

    even\_numbers\_list.append(i)

print("Array of even numbers (using loop and append):", even\_numbers\_list)

O/P: Array of even numbers (using loop and append): [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]

**16. Write a program to create an array containing numbers from 1 to 10 with a step size of 0.5**

**using arange.** #python program to create  an array containing  numbers from 1 to 10  with a step size of 0.5 using range

import numpy as np

# Create an array with values from 1 to 10 (inclusive) and a step size of 0.5

number\_array = np.arange(1, 10.5, 0.5).tolist()

print(number\_array)

O/P: [1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0, 8.5, 9.0, 9.5, 10.0]