# SMART INTERNZ-APSCHE

## AI/ML TRAINING

ASSESSMENT-1

1.Write a Python program to calculate the area of a rectangle given its length and width.

def calculate\_rectangle\_area(length, width):

area = length \* width

return area

length = float(input("Enter the length of the rectangle: "))

width = float(input("Enter the width of the rectangle: "))

area = calculate\_rectangle\_area(length, width)

print("The area of the rectangle is:", area)

Input:

the length and width of the rectangle.

Output:

the calculated area.

2. Write a program to convert miles to kilometers.

def miles\_to\_kilometers(miles):

kilometers = miles \* 1.60934

return kilometers

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

kilometers = miles\_to\_kilometers(miles)

print(f"{miles} miles is equal to {kilometers} kilometers")

Input:

The distance in miles,

Output:

The equivalent distance in kilometers.

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

def is\_palindrome(s):

s = s.lower() # Convert to lowercase for case-insensitive comparison

# Remove non-alphanumeric characters

s = ''.join(char for char in s if char.isalnum())

return s == s[::-1]

# Example usage:

test\_string = "A man, a plan, a canal, Panama!"

print(is\_palindrome(test\_string)) # Output: True

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

def second\_largest(numbers):

# Make sure the list has at least two elements

if len(numbers) < 2:

return "List must have at least two elements"

largest = second\_largest = float('-inf')

for num in numbers:

if num > largest:

second\_largest = largest

largest = num

elif num > second\_largest and num != largest:

second\_largest = num

if second\_largest == float('-inf'):

return "There is no second largest element"

else:

return second\_largest

# Example usage:

numbers = [10, 5, 20, 8, 25]

print("Second largest element:", second\_largest(numbers))

5. Explain what indentation means in Python.

In Python, indentation is used to define blocks of code. It is not just for readability; it's a fundamental part of the language's syntax. Indentation is used to group statements together to form a block of code, such as within loops, conditional statements, function definitions, and class definitions.

Python uses indentation to determine the structure and hierarchy of the code. Blocks of code at the same level of indentation are considered part of the same block or scope. Incorrect indentation can lead to syntax errors or change the logical structure of the code, leading to unexpected behavior.

6. Write a program to perform set difference operation.

def set\_difference(set1, set2):

return set1 - set2

# Example usage:

set1 = {1, 2, 3, 4, 5}

set2 = {4, 5, 6, 7, 8}

result = set\_difference(set1, set2)

print("Set difference:", result)

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

num = 1

while num <= 10:

print(num)

num += 1

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

def factorial(n):

if n < 0:

return "Factorial is not defined for negative numbers."

elif n == 0:

return 1

else:

result = 1

while n > 0:

result \*= n

n -= 1

return result

# Example usage:

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

print("Factorial of", number, "is", factorial(number))

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

def check\_number(num):

if num > 0:

return "Positive"

elif num < 0:

return "Negative"

else:

return "Zero"

# Example usage:

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

print("The number is", check\_number(number))

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

def find\_largest(num1, num2, num3):

if num1 >= num2 and num1 >= num3:

return num1

elif num2 >= num1 and num2 >= num3:

return num2

else:

return num3

# Example usage:

number1 = float(input("Enter the first number: "))

number2 = float(input("Enter the second number: "))

number3 = float(input("Enter the third number: "))

largest = find\_largest(number1, number2, number3)

print("The largest number is:", largest)

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

import numpy as np

def create\_ones\_array(shape):

return np.ones(shape)

# Example usage:

shape = (3, 4) # Change the shape as per your requirement

ones\_array = create\_ones\_array(shape)

print("Array filled with ones:")

print(ones\_array)

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

import numpy as np

def create\_random\_int\_array(shape, low, high):

return np.random.randint(low, high, size=shape)

# Example usage:

shape = (3, 4) # Change the shape as per your requirement

low = 1 # Lowest possible integer

high = 10 # One more than the highest possible integer

random\_int\_array = create\_random\_int\_array(shape, low, high)

print("2D numpy array initialized with random integers:")

print(random\_int\_array)

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

range using linspace.

import numpy as np

def generate\_linspace(start, stop, num):

return np.linspace(start, stop, num)

# Example usage:

start = 0 # Start of the range

stop = 10 # End of the range

num = 5 # Number of samples to generate

linspace\_array = generate\_linspace(start, stop, num)

print("Array of evenly spaced numbers using linspace:")

print(linspace\_array)

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

linspace.

import numpy as np

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

values = np.linspace(1, 100, 10)

print("Array of 10 equally spaced values between 1 and 100:")

print(values)

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

arange.

import numpy as np

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

values = np.linspace(1, 100, 10)

print("Array of 10 equally spaced values between 1 and 100:")

print(values)

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

using arange.

import numpy as np

# Create an array containing numbers from 1 to 10 with a step size of 0.5

numbers = np.arange(1, 10.5, 0.5)

print("Array containing numbers from 1 to 10 with a step size of 0.5:")

print(numbers).

Submitted By

Nagisetti Yamini

20HU1A4226