

SMART INTERNZ - APSCHE

AI / ML Training

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

```
In [4]: length = float(input("Enter length : "))
width = float(input("Enter width : "))
area = length * width

print(f"Area = {area}")
```

```
Enter length : 12
Enter width : 7
Area = 84.0
```

2. Write a program to convert miles to kilometers

```
In [7]: miles = float(input("Enter the distance in miles: "))
kilometers = miles * 1.60934

print(f"{miles} miles = {kilometers}km")
```

```
Enter the distance in miles: 12
12.0 miles = 19.31208km
```

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

```
In [9]: def is_palindrome(s):
        s = s.replace(" ", "").lower()
        return s == s[::-1]

str = input("Enter a string: ")

if is_palindrome(str):
    print(f"{str} is a palindrome.")
else:
    print(f"{str} is not a palindrome.")
```

```
Enter a string: tenet
tenet is a palindrome.
```

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

```
In [12]: l = [1, 4, 6, 2, 5, 7, 0, 9, 8]
print(sorted(l)[-2])
```

8

5. Explain what indentation means in Python.

- Indentation is used to define the structure and scope of the code.
- Unlike many other programming languages that use braces or keywords to indicate the beginning and end of blocks of code (such as if statements, loops, and functions), Python uses indentation.
- Indentation refers to the spaces or tabs at the beginning of a line of code.
- It is used to group statements into blocks.
 - Blocks of code with the same level of indentation are considered part of the same block or scope.

6. Write a program to perform set difference operation.

```
In [13]: set1 = {1, 2, 3, 4, 5}
set2 = {3, 4, 5, 6, 7}
set_difference = set1 - set2

print(f"Set Difference : {set_difference}")
```

Set Difference : {1, 2}

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

```
In [15]: n = 1
while n <= 10:
    print(n, end= " ")
    n += 1
```

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.

```
In [18]: def factorial(n):
        if n < 0:
            return "Enter positive number."
        elif n == 0 or n == 1:
            return 1
        else:
            fact = 1

            while n >= 2:
                fact *= n
                n -= 1

            return fact

n = int(input("Enter a number : "))
result = factorial(n)
print(f"The factorial of {n} is: {result}")
```

```
Enter a number : 6
The factorial of 6 is: 720
```

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

```
In [19]: number = float(input("Enter a number: "))
        if number > 0:
            print(f"{number} is a positive number.")
        elif number < 0:
            print(f"{number} is a negative number.")
        else:
            print("The entered number is zero.")
```

```
Enter a number: -12
-12.0 is a negative number.
```

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

```
In [21]: num1 = int(input("Enter first number: "))
num2 = int(input("Enter second number: "))
num3 = int(input("Enter third number: "))

if num1 >= num2 and num1 >= num3:
    largest = num1
elif num2 >= num1 and num2 >= num3:
    largest = num2
else:
    largest = num3

print(f"The largest number is : {largest}")
```

```
Enter first number: 12
Enter second number: 22
Enter third number: 7
The largest number is : 22
```

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

```
In [25]: import numpy as np

shape = (4, 2)
ones_array = np.ones(shape)
print(f"Array filled with ones : \n{ones_array}")
```

```
Array filled with ones :
[[1. 1.]
 [1. 1.]
 [1. 1.]
 [1. 1.]]
```

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

```
In [29]: import numpy as np

shape = (3, 4)
random_array = np.random.randint(1, 25, shape)

# result = random_integers_array(shape, low, high)
print(f"2D array : \n{random_array}")
```

```
2D array :
[[ 3 14 18 18]
 [22  9 16 16]
 [12 24  6 14]]
```

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

```
In [34]: import numpy as np

arr = np.linspace(0, 12, 6)

print(f"Array {arr}")
```

```
Array [ 0.   2.4  4.8  7.2  9.6 12. ]
```

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

```
In [32]: import numpy as np

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

print(f"Array :\n{arr}")
```

```
Array :
[  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.

```
In [35]: import numpy as np

arr = np.arange(2, 21, 2)

print(f"Array {arr}")
```

```
Array [ 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

```
In [36]: import numpy as np

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

print(f"Array {arr}")
```

```
Array [ 1.   1.5  2.   2.5  3.   3.5  4.   4.5  5.   5.5  6.   6.5  7.
 7.5
 8.   8.5  9.   9.5 10. ]
```