Assignment 6.2

Name:P.Pooja

RollNumber: 2303A510F7

Batch - 03

Task Description-1 (Classes – Data Validation)

• Prompt AI to generate a Student class with attributes: name,

roll_no, and marks. Add a method is_pass() that returns whether

the student has passed (marks ≥ 40).

Prompt: Write a Python program using a class Student with attributes name, roll_no, and marks. Include a method is_pass() that returns True if marks are 40 or above, otherwise False. Create an object and print the result.

Code:

```python
#Write a Python program using a class Student with attributes name, roll_no, and marks. Include a method is_pass() that returns
class Student:
    def __init__(self, name, roll_no, marks):
        self.name = name
        self.roll_no = roll_no
        self.marks = marks

    def is_pass(self):
        return self.marks >= 40

student1 = Student("Pooja",2042, 85)
print(f"{student1.name}(Roll No:{student1.roll_no})-Marks:{student1.marks}-Pass:{student1.is_pass()}")
```

Output:

```
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting> & C:/Users/Apple/AppData/Local/Microsoft/WindowsApps/python3.12.exe "c:/Users/Apple/OneDrive/Desktop/Ai Assisting/student.py"
Pooja(Roll No:2042)-Marks:85-Pass:True
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting>
```

Code Explanation:

This program defines a Student class to store student details like name, roll number, and marks. The __init__ method initializes these values when a student object is created. The is_pass method checks whether the student has passed by verifying if the marks are 40 or more and returns True or False. An object is created using the class, and the result of the pass check is printed.

Task Description-2 (Loops – Pattern Generation)

• Ask AI to generate a function that prints a right-angled triangle

star pattern using a for loop. Then regenerate the same pattern

using a while loop.

Prompt: Write a Python function using a while loop to print a right-angled triangle pattern of stars for a given number.

Code:

```python
#Write a Python function using a while loop to print a right-angled triangle pattern of stars for a given number.
def print_triangle(n):
    i = 1
    while i <= n:
        print('*' * i)
        i += 1
n = 5
print_triangle(n)
```

Output:

```
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting> & C:/Users/Apple/AppData/Local/Microsoft/WindowsApps/python3.12.exe "c:/Users/Apple/OneDrive/Desktop/Ai Assisting/student.py"
*
**
***
****
*****
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting>
```

Code Explanation:

This program defines a function triangle_while that takes a number n as input. Inside the function, a variable i starts from 1 and a while loop runs until i is less than or equal to n. In each loop iteration, it prints i number of * characters to form a triangle pattern. After printing each line, i is increased by 1. Finally, the function is called with the value 5, which prints a triangle of 5 rows.

Task Description-3 (Conditional Statements – Number Analysis)

• Ask AI to write a function that checks whether a given number

is positive, negative, or zero using if-elif-else. Test the function

with multiple inputs.

Prompt: Write a Python function that checks whether a number is positive, negative, or zero and prints the result.

Code:

```python
#Write a Python function that checks whether a number is positive, negative, or zero and prints the result.
def check_number(num):
    if num > 0:
        print(f"{num} is a positive number.")
    elif num < 0:
        print(f"{num} is a negative number.")
    else:
        print("The number is zero.")
number = -10
check_number(number)
```

Output:

```
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting> & C:/Users/Apple/AppData/Local/Microsoft/WindowsApps/python3.12.exe "c:/Users/Apple/OneDrive/Desktop/Ai Assisting/student.py"
-10 is a negative number.
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting>
```

Code Explanation:

This program defines a function check_number that takes a number as input. It uses if, elif, and else conditions to check whether the number is greater than zero, less than zero, or equal to zero. If the number is greater than zero, it returns "Positive", if it is less than zero, it returns "Negative", and if it is zero, it returns "Zero". The function is then called with different values, and the results are printed.

Task Description-4 (Nested Conditionals)

• Generate a function check_discount(age, is_member) that

determines discount eligibility:

• Age ≥ 60 → Senior discount

• Member → Additional discount

Use nested if statements.

Prompt: Write a Python function that determines the type of discount based on a person's age and membership status using conditional statements.

Code:

```python
#Write a Python function that determines the type of discount based on a person's age and membership status using conditional statements.
def calculate_discount(age, is_member):
    if age < 12:
        return "Child discount: 50%"
    elif age >= 60:
        return "Senior discount: 30%"
    elif is_member:
        return "Member discount: 20%"
    else:
        return "No discount"
age = 65
membership_status = True
discount = calculate_discount(age, membership_status)
print(f"Age: {age}, Membership Status: {membership_status} - Discount: {discount}")
```

Output:

```
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting> & C:/Users/Apple/AppData/Local/Microsoft/WindowsApps/python3.12.exe "c:/Users/Apple/OneDrive/Desktop/Ai Assisting/student.py"
Age: 65, Membership Status: True - Discount: Senior discount: 30%
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting>
```

Code Explanation:

This program defines a function check_discount that takes two inputs: age and is_member. It first checks if the age is 60 or above to identify a senior citizen. If the person is a senior and also a member, it returns **"Senior + Member Discount"**; if they are only a senior, it returns **"Senior Discount"**. If the age is below 60, the function then checks membership status: members get **"Member Discount"**, and non-members get **"No Discount"**. The function is called with different values, and the appropriate discount message is printed.

Task Description-5 (Class – Mathematical Opera)

• Ask AI to create a Circle class with methods to calculate area ()

and circumference () given the radius.

Prompt: Create a Python class Circle that takes radius as input and includes methods to calculate and return the area and circumference of the circle.

Code:

```python
#Create a Python class Circle that takes radius as input and includes methods to calculate and return the area and circumference of the circle.
import math
class Circle:
    def __init__(self, radius):
        self.radius = radius

    def area(self):
        return math.pi * (self.radius ** 2)

    def circumference(self):
        return 2 * math.pi * self.radius
radius = 5
circle = Circle(radius)
print(f"Circle with radius {radius}: Area = {circle.area():.2f}, Circumference = {circle.circumference():.2f}")
```

Output:

```
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting> & C:/Users/Apple/AppData/Local/Microsoft/WindowsApps/python3.12.exe "c:/Users/Apple/OneDrive/Desktop/Ai Assisting/student.py"
Circle with radius 5: Area = 78.54, Circumference = 31.42
PS C:\Users\Apple\OneDrive\Desktop\Ai Assisting>
```

Code Explanation:

This program imports the math module to use the value of π (pi). It defines a class Circle with an __init__ method that stores the radius of the circle. The area method calculates and returns the area using the formula π × radius × radius.