

**AI ASSISTED CODING**

**LAB-4:** ***AI-Based Code Auto-Completion – Classes, Loops, and Conditionals in Python using GitHub Copilot***

**Roll No:** 2503A51L34

**Name:** Dhangar Nagamrutha

**Batch:** 24BTCAICSB20

**Task #1:**

**Prompt:** Write a class definition comment and start the constructor for a class called BankAccount with account\_holder and balance attributes. Auto-complete the rest of the class, including methods to deposit, withdraw, and display balance.And also ask for the inputs from the users.

**A screen shot of a computer program

AI-generated content may be incorrect.Code Generated: Output:**

**A computer screen shot of a program

AI-generated content may be incorrect.**

**Observation:**

In this task, I learned how to create a Python class with a constructor and methods for deposit, withdraw, and display balance. I observed that small corrections like using \_\_init\_\_ and \_\_name\_\_ == "\_\_main\_\_" are important for the code to work. This showed me how object-oriented programming makes code more organized and interactive.

**Task #2:**

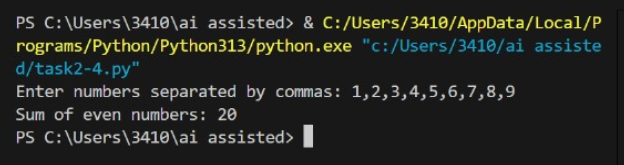
**Prompt:** Write a comment and the initial line of a loop to iterate over a list. Complete the logic to sum all even numbers in the list. Ask for the list from the user.

**Code Generated:**

**A computer screen with colorful text

AI-generated content may be incorrect.**

**Output:**

****

**Observation:**

In this task, I learned how to take user input as a list of numbers, convert them into integers, and use a loop with a condition to check even numbers. I observed that by using an if statement inside the loop, only even numbers are added to the total, giving the sum of all even numbers in the list.

**Task #3:**

**Prompt:** Start a function that takes age as input and returns whether the person is a child, teenager, adult, or senior using if-elif-else. Complete the conditionals and user gives the input.

**Code Generated:**

**A computer screen shot of a program code

AI-generated content may be incorrect.**

**Output:**

**A screen shot of a computer

AI-generated content may be incorrect.**

**Observation:**

In this task, I learned how to use conditional statements to categorize input values into different groups. The program checks the entered age and returns whether the person is a Child, Teenager, Adult, or Senior. I observed that proper use of if-elif-else helps in making clear decisions based on ranges of values.

**Task #4:** Write a comment and start a while loop to reverse the digits of a number. Complete the loop logic and ask for the number from the user.

**Code Generated:**

**A computer screen with colorful text

AI-generated content may be incorrect.**

**Output:**

**A computer screen shot of a program code

AI-generated content may be incorrect.**

**Observation:**

In this task, Copilot provided the logic to reverse digits using a while loop with modulus and integer division. I only began with a comment and loop, and it handled the rest. Running the code gave the expected reversed number. I observed how Copilot can easily generate solutions for number-based problems that usually take time to think through.

**Task #5:**

**Prompt:** Begin a class Employee with attributes name and salary. Then, start a derived class Manager that inherits from Employee and adds a department. Complete the methods and constructor chaining.

**A screen shot of a computer program

AI-generated content may be incorrect.Code Generated:**

**A screen shot of a computer

AI-generated content may be incorrect.Output:**

**Observation:**

In this task, I learned how inheritance works in Python by creating a base class Employee and a derived class Manager. The derived class reused attributes of the base class through super() and added its own property department. I observed that inheritance reduces code duplication and allows extending functionality while maintaining organized and reusable code.