AI Assisted Coding

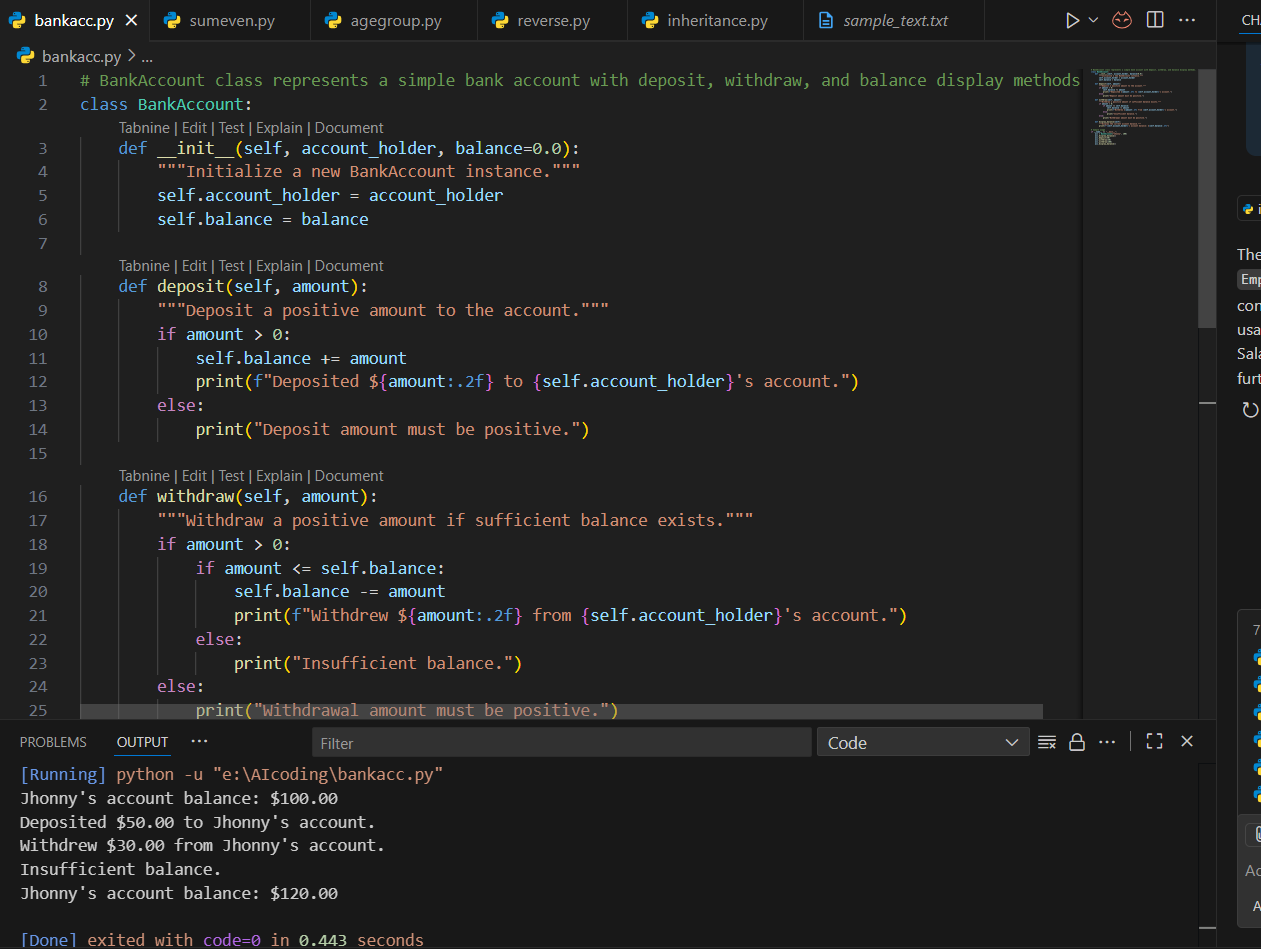
# K.Sindhu meenan(2403A51250)-Batch11

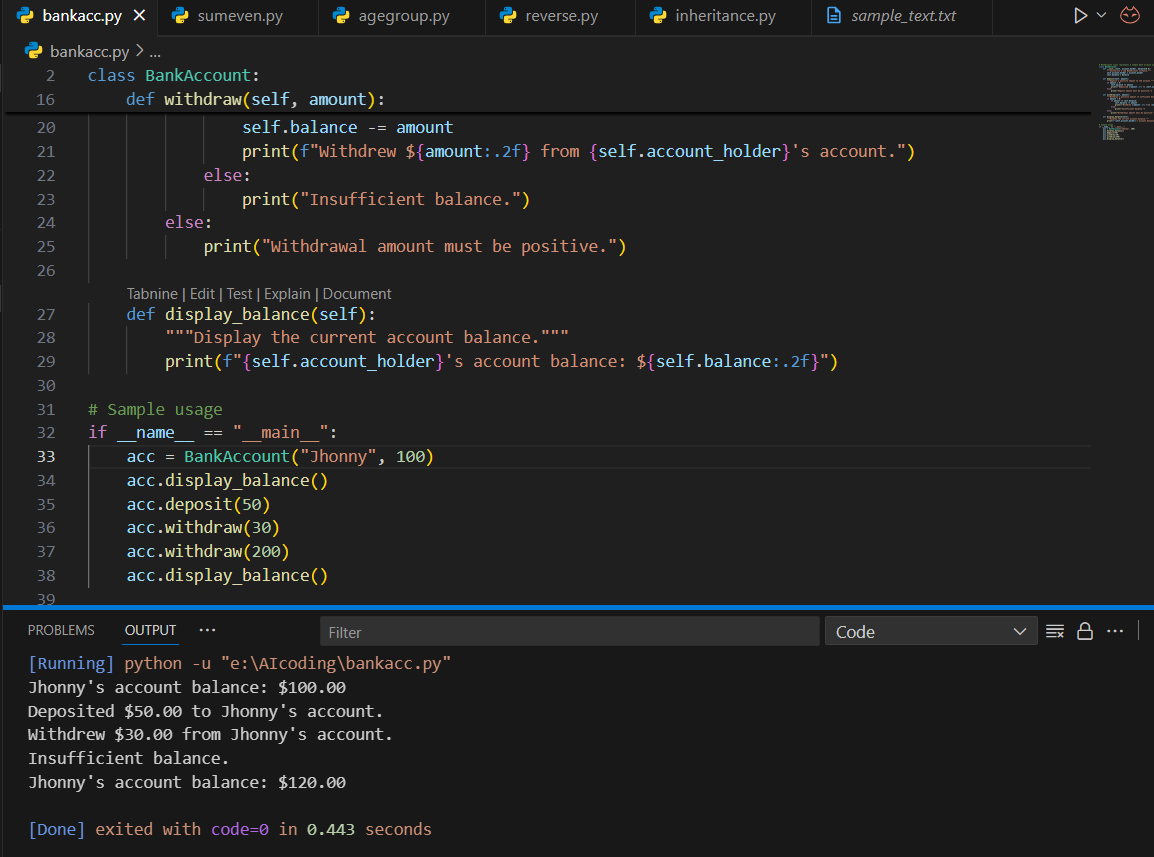
Task Description #1

Auto-Complete a Python Class for Bank Account  
Write a class definition comment and start the constructor for a class called BankAccount with  
account\_holder and balance attributes. Use GitHub Copilot to auto-complete the rest of the  
class, including methods to deposit, withdraw, and display balance

Expected output:

Complete class with methods like:  
deposit(self, amount)  
withdraw(self, amount)  
display\_balance(self)





Code Explanation:

The BankAccount class represents a simple bank account with basic operations:

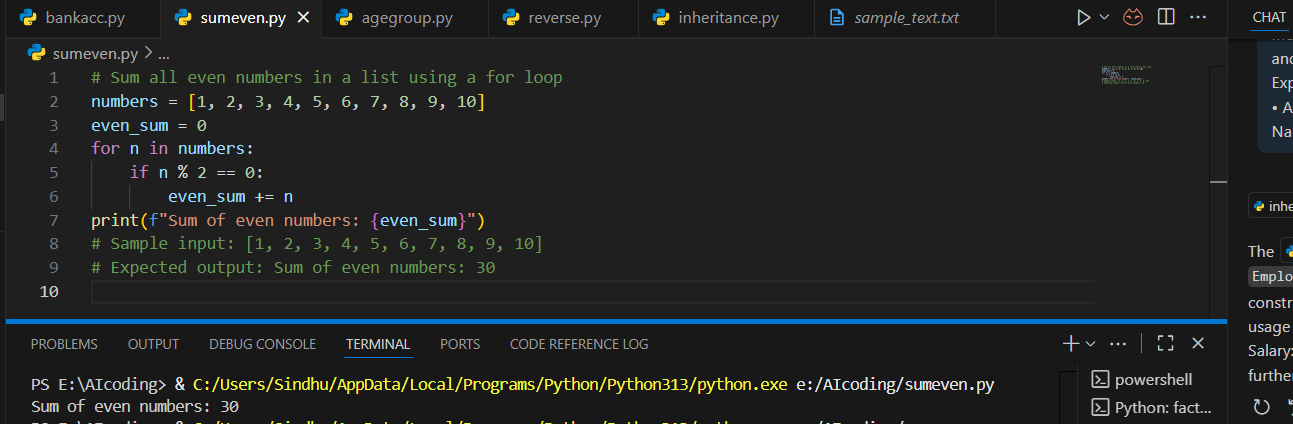
* **Constructor (\_\_init\_\_)**  
  Initializes a new bank account with an account holder’s name and an optional starting balance (default is 0.0).
* **deposit(amount)**  
  Adds a positive amount to the balance. Prints a confirmation message.
* **withdraw(amount)**  
  Subtracts a positive amount from the balance if enough funds are available. Prints success or an error message.
* **display\_balance()**  
  Prints the current balance of the account.

Task Description #2

Auto-Complete a For Loop to Sum Even Numbers in a List  
• Write a comment and the initial line of a loop to iterate over a list. Allow GitHub Copilot to  
complete the logic to sum all even numbers in the list

Expected output:

Code that:  
• Iterates over a list  
• Checks if the number is even using % 2 == 0  
• Accumulates the sum



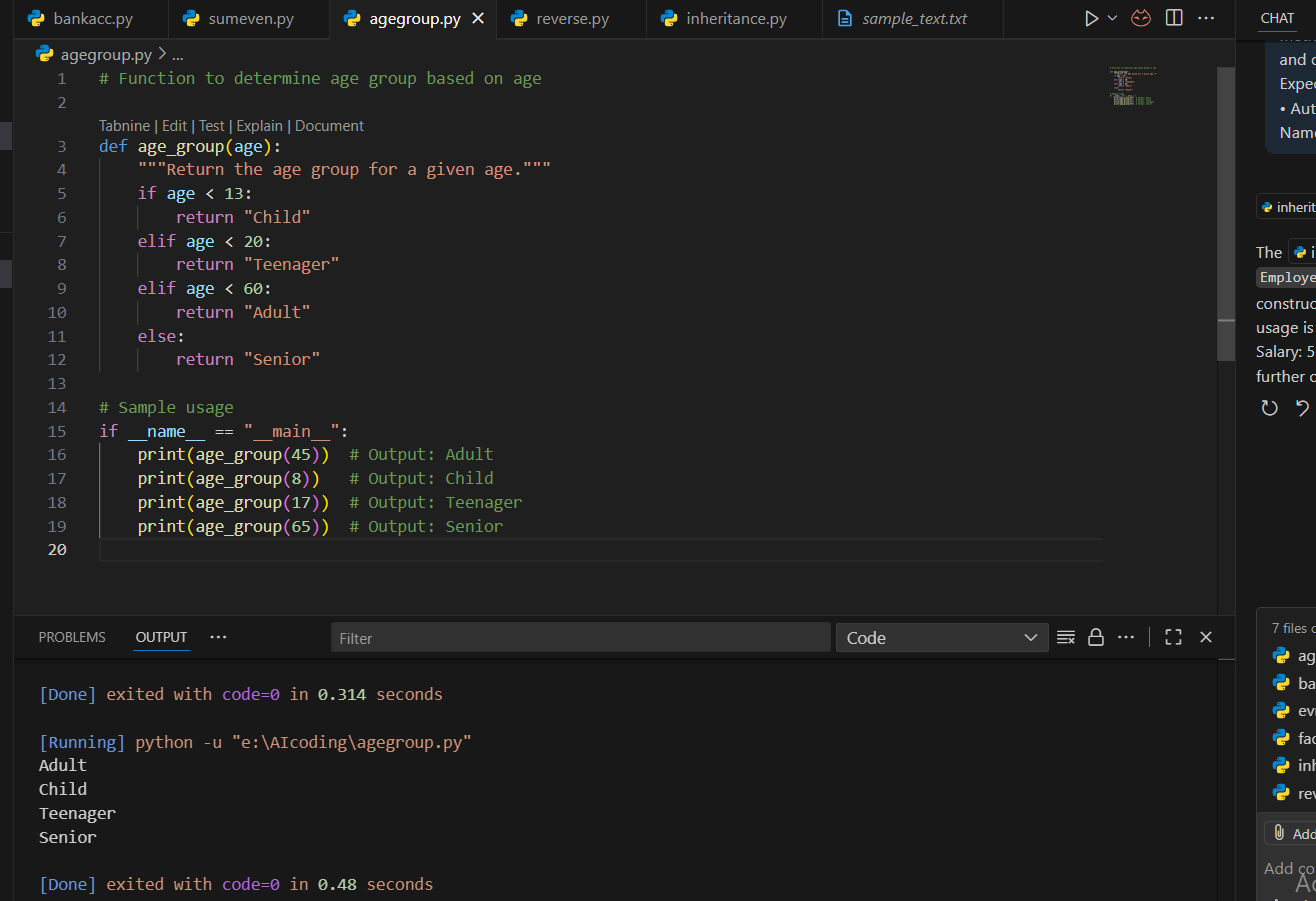
**Code explanation:**

* A list of numbers is defined.
* A variable even\_sum is initialized to 0 to hold the total.
* A for loop goes through each number in the list.
* If the number is even (n % 2 == 0), it is added to even\_sum.
* Finally, the program prints the total sum of even numbers.

Task Description #3

Auto-Complete Conditional Logic to Check Age Group  
• Start a function that takes age as input and returns whether the person is a child, teenager,  
adult, or senior using if-elif-else. Use Copilot to complete the conditionals

Expected output:  
• Function like: Output for age\_group (45) ➝ "Adult"



Code explanation:

 The function **age\_group(age)** categorizes a person into an age group based on their age.

 Conditions used:

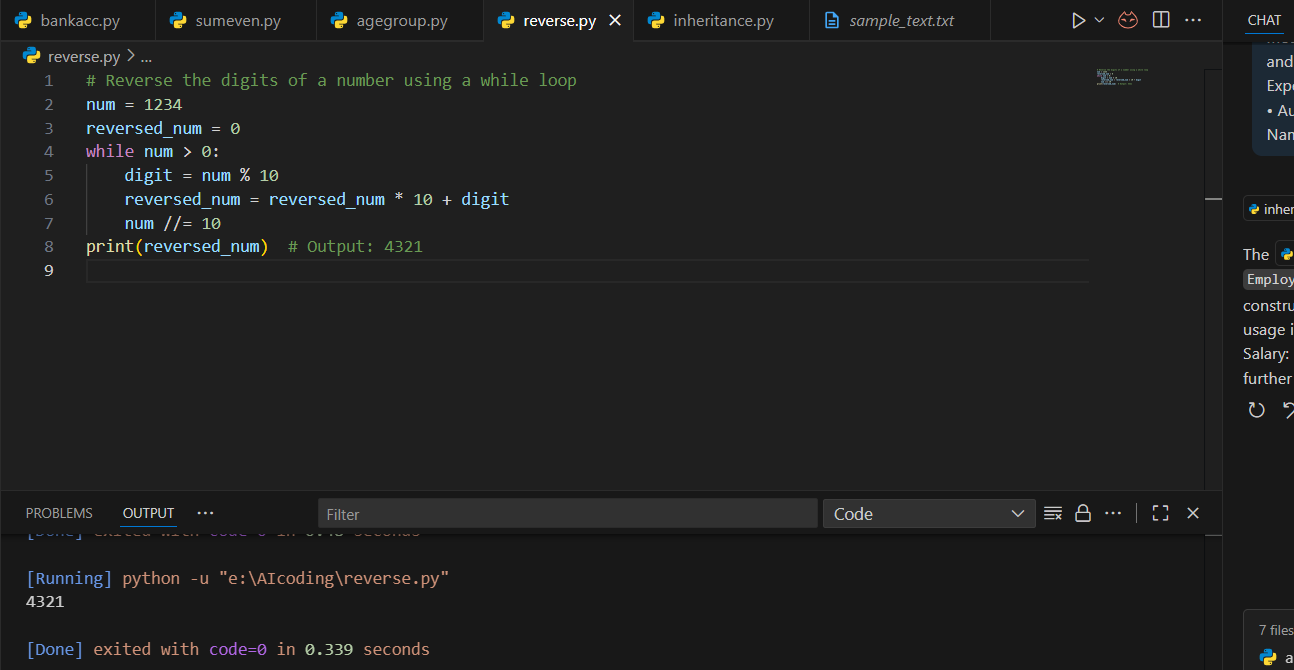
* **< 13 → "Child"**
* **< 20 → "Teenager"**
* **< 60 → "Adult"**
* **Otherwise → "Senior"**

 Returns the corresponding group as a string.

Task 4: Auto-Complete a While Loop to Reverse Digits of a Number  
Task Description #4  
• Write a comment and start a while loop to reverse the digits of a number. Let Copilot  
complete the loop logic

Expected account:

Functional loop: Output: 4321



Code explanation:

 A number is given (num = 1234).

 reversed\_num is initialized to 0.

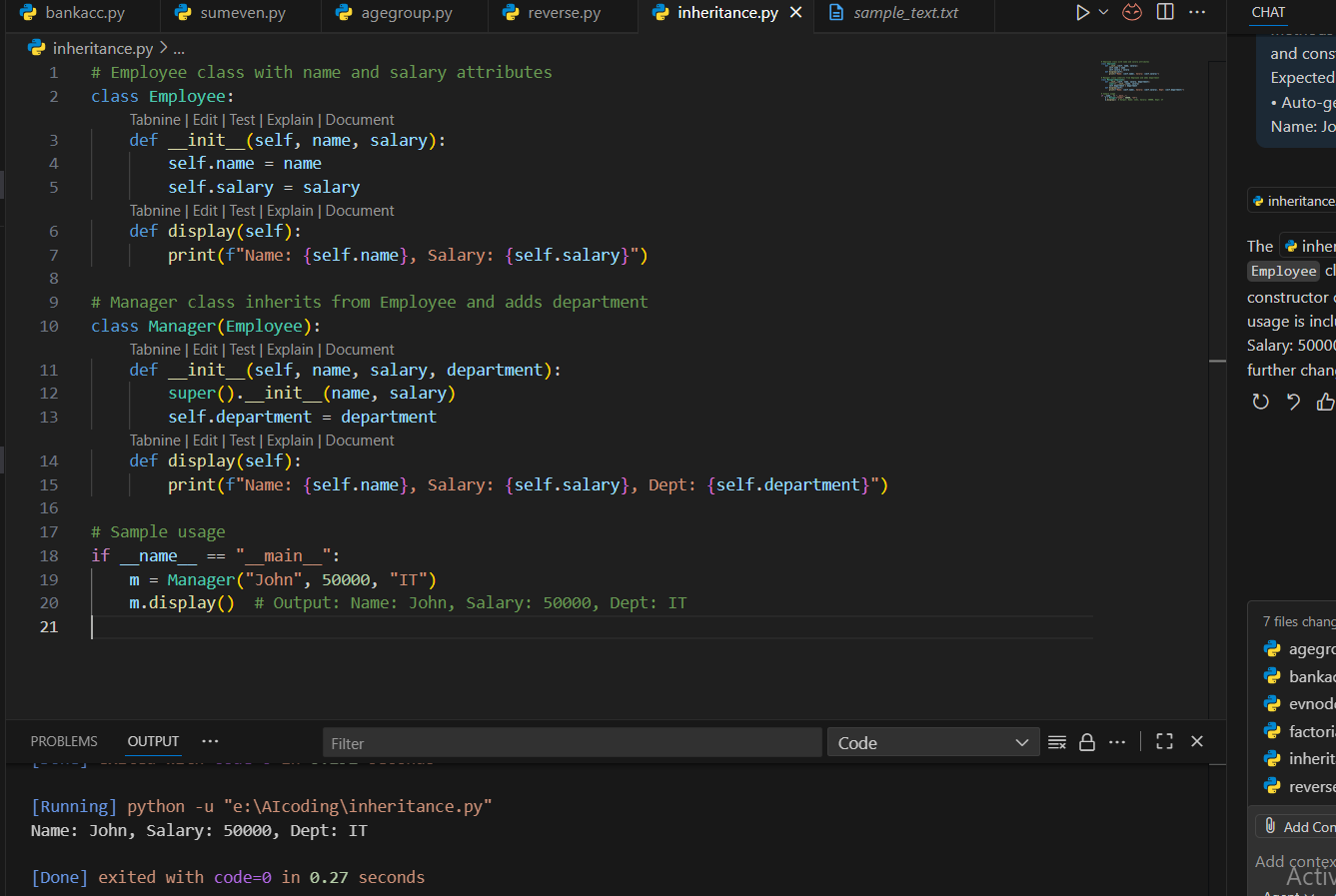
 A **while loop** runs until num becomes 0:

* Extract the last digit using num % 10.
* Append it to reversed\_num (reversed\_num = reversed\_num \* 10 + digit).
* Remove the last digit from num using integer division (num //= 10).

 Finally, the reversed number is printed.

Task Description #5

Auto-Complete Class with Inheritance (Employee → Manager)  
• Begin a class Employee with attributes name and salary. Then, start a derived class Manager  
that inherits from Employee and adds department. Let GitHub Copilot complete the methods  
and constructor chaining.



Code explanation:

 **Employee Class**: Defines an employee with name and salary. The display() method prints these details.

 **Manager Class**: Inherits from Employee and adds a department attribute. It overrides display() to show name, salary, and department.

 **Usage**: Creating Manager("John", 50000, "IT") and calling display() prints