PORIKA CHARAN-2303A51570-BATCH.09

Lab Assignment 6.3

Task Description #1: Classes (Student Class)

Scenario

You are developing a simple student information management module.

Task

• Use an AI tool (GitHub Copilot / Cursor AI / Gemini) to complete a Student class.

• The class should include attributes such as name, roll number, and branch.

• Add a method display_details() to print student information.

• Execute the code and verify the output.

• Analyze the code generated by the AI tool for correctness and clarity.

Prompt: Create a Python class named `Student`. This class should have the following attributes: `name` (string), `roll_number` (integer), and `branch` (string). Additionally, include a method named `display_details()` that, when called, prints out the student's name, roll number, and branch in a readable format. After defining the class, provide example code to create an instance of the `Student` class and call its `display_details()` method to demonstrate its functionality

Task Description #2: Loops (Multiples of a Number)

Scenario

You are writing a utility function to display multiples of a given number.

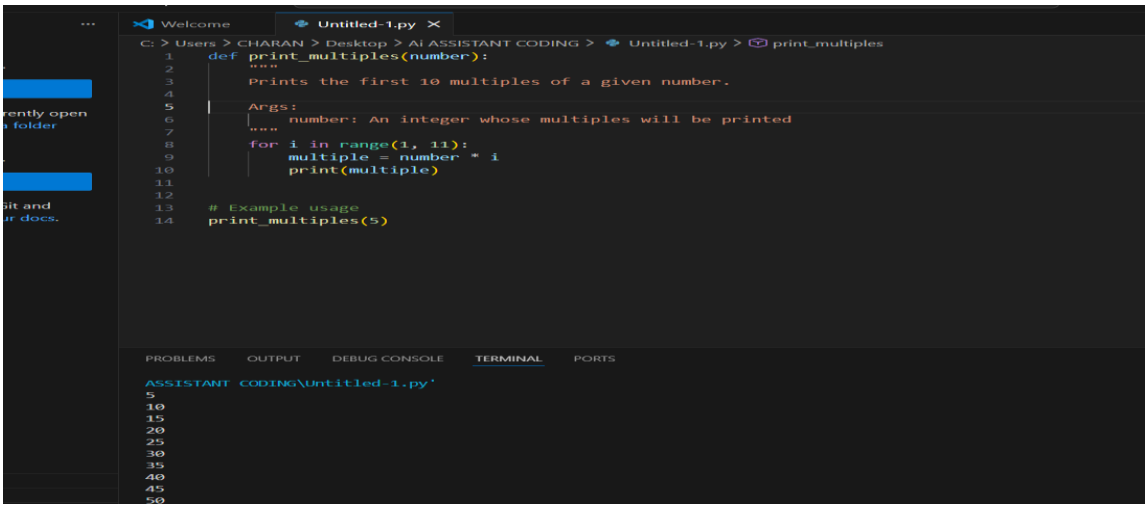
Task

• Prompt the AI tool to generate a function that prints the first 10 multiples of a given number using a loop.

• Analyze the generated loop logic.

• Ask the AI to generate the same functionality using another controlled looping structure

Prompt: wite a Python programmme that prints the first 10 multiples of a given number using a for loop.it should Accept one integer as input and calculate multiples print each in a new line



Task Description #3: Conditional Statements (Age Classification)

Scenario

You are building a basic classification system based on age.

Task

• Ask the AI tool to generate nested if-elif-else conditional statements to classify age groups (e.g., child, teenager, adult, senior).

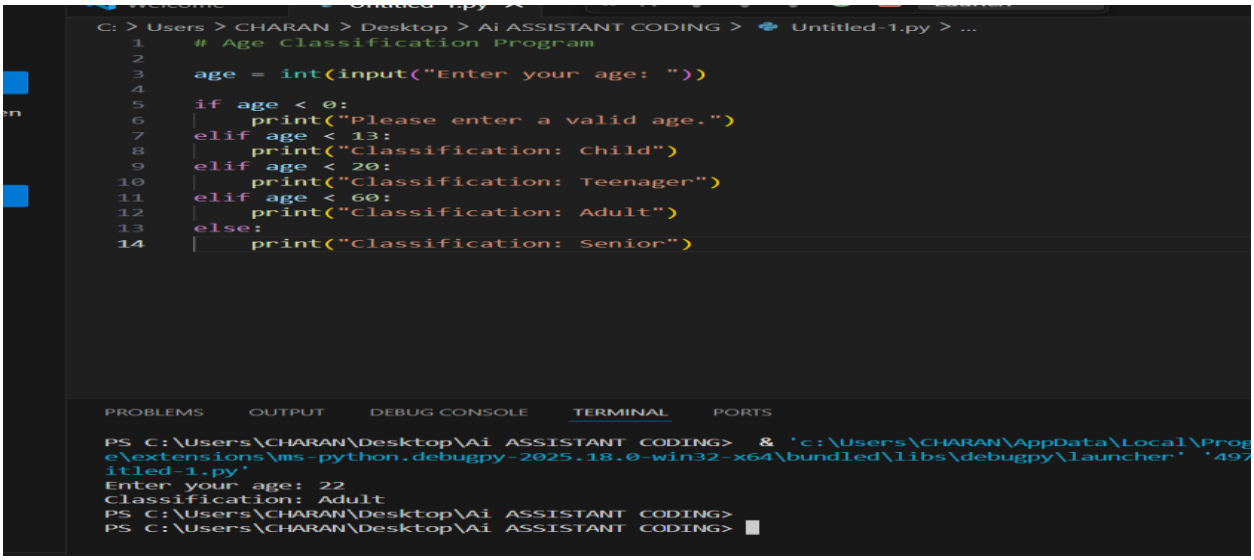
• Analyze the generated conditions and logic.

• Ask the AI to generate the same classification using alternative conditional structures (e.g.,

simplified conditions or dictionary-based logic)

prompt: Generate a Python program that classifies a person based on their age using nested

conditional statements take age as integer and classify the age as child,teenager,adult and senior Print the age group as output



Task Description #4: For and While Loops (Sum of First n Numbers)

Scenario

You need to calculate the sum of the first n natural numbers.

Task

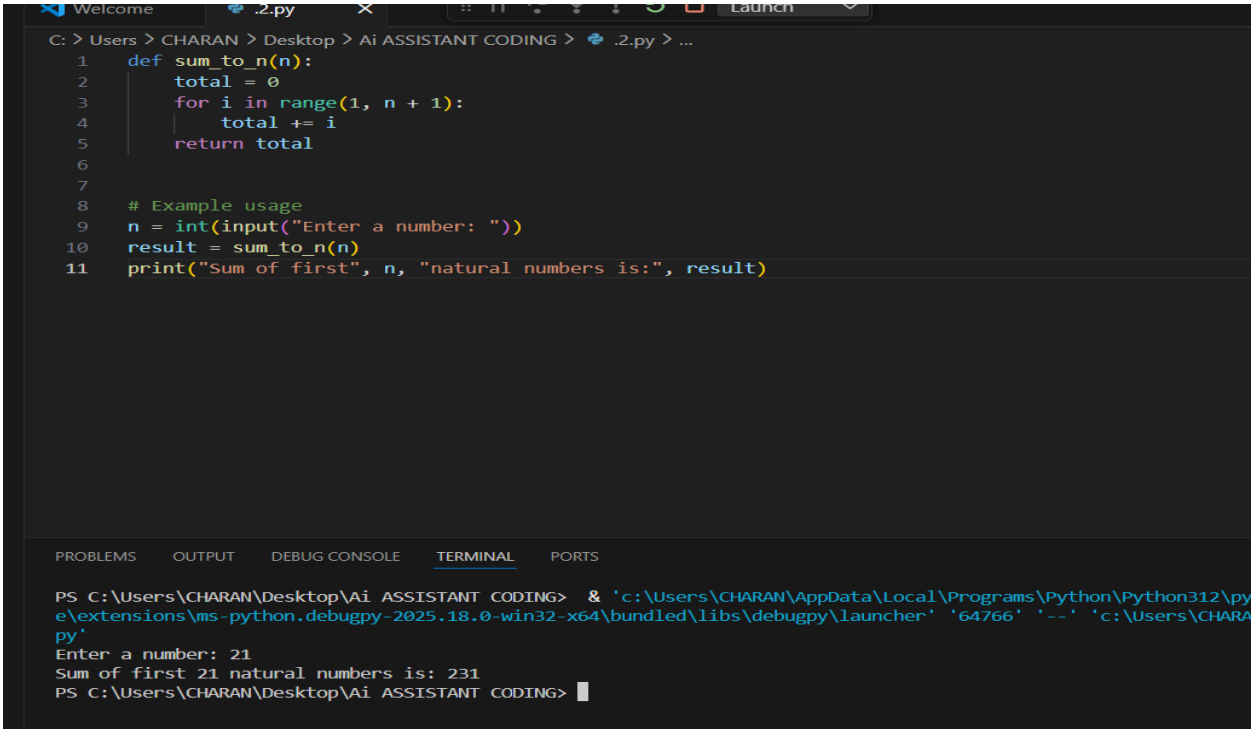
• Use AI assistance to generate a sum_to_n() function using a for loop.

• Analyze the generated code.

• Ask the AI to suggest an alternative implementation using a while loop or a mathematical

formula.

Prompt: write a python functiom named as sum to n() it have to calculate sum of first n natural numbers,the function should take n as integer and use for loop to calculate the sum and return final sum



Task Description #5: Classes (Bank Account Class)

Scenario

You are designing a basic banking application.

Task

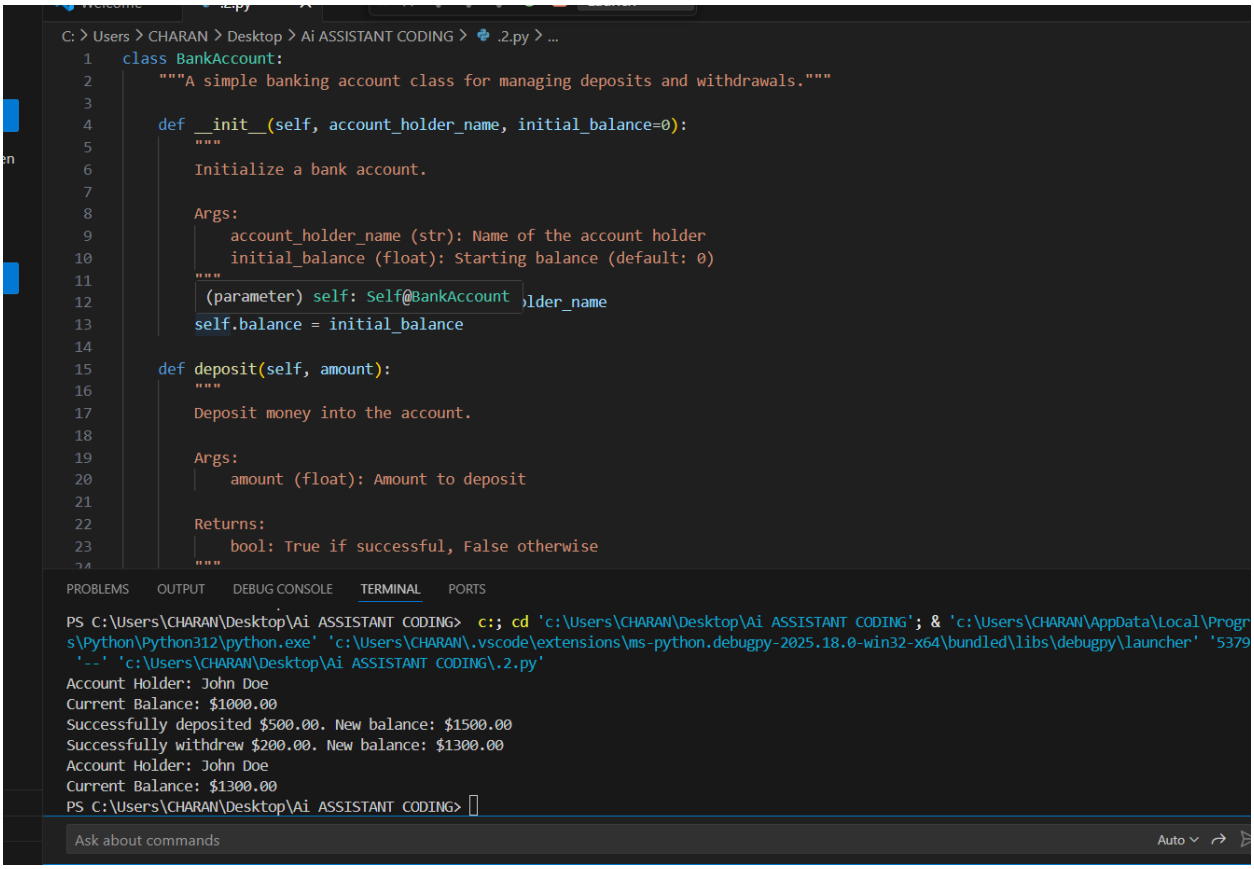
• Use AI tools to generate a Bank Account class with methods such as deposit(), withdraw(),

and check_balance().

• Analyze the AI-generated class structure and logic.

• Add meaningful comments and explain the working of the code.

PROMPT: I need a bank application which performs three main tasks withdrawal ,deposit and check balance.take input from the user as name and security pin then show the three performance user need to select any one of them .after deposit/withdrawal you have to show the remain balance


C: > Users > CHARAN > Desktop > Ai ASSISTANT CODING > .2.py > ...
class BankAccount:
"""A simple banking account class for managing deposits and withdrawals."""
def __init__(self, account_holder_name, initial_balance=0):
"""
Initialize a bank account.
Args:
account_holder_name (str): Name of the account holder
initial_balance (float): Starting balance (default: 0)
"""
(parameter) self: Self@BankAccount
self.balance = initial_balance
def deposit(self, amount):
"""
Deposit money into the account.
Args:
amount (float): Amount to deposit
Returns:
bool: True if successful, False otherwise
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\CHARAN\Desktop\Ai ASSISTANT CODING> c:; cd 'c:\Users\CHARAN\Desktop\Ai ASSISTANT CODING'; & 'c:\Users\CHARAN\AppData\Local\Progr
s\Python\Python312\python.exe' 'c:\Users\CHARAN\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '5379
'--' 'c:\Users\CHARAN\Desktop\Ai ASSISTANT CODING\.2.py'
Account Holder: John Doe
Current Balance: $1000.00
Successfully deposited $500.00. New balance: $1500.00
Successfully withdrew $200.00. New balance: $1300.00
Account Holder: John Doe
Current Balance: $1300.00
PS C:\Users\CHARAN\Desktop\Ai ASSISTANT CODING>
Ask about commands
Auto