Dylan Torres

7/10/2025

**1. Description of the Program**

This program simulates a basic **bank account system** using Python. It allows a user to create a bank account and perform standard operations, such as depositing, withdrawing, adjusting the interest rate, checking the balance, and calculating interest over time. The system is interactive and uses user input to perform actions, making it easy to test and demonstrate real-world banking logic. It includes a BankAcct class that encapsulates all relevant account details and behaviors, and a test function that drives the user menu and input system.

**2. Functions Created**

**\_\_init\_\_(self, name, account\_number, amount, interest\_rate)**

* **Purpose**: Initializes a new bank account object with provided values.
* **Parameters**:
  + name (str): The name of the account holder.
  + account\_number (str): The account number.
  + amount (float): Initial deposit amount.
  + interest\_rate (float): Annual interest rate in percentage.
* **Return**: None (constructor)

**deposit(self, amount)**

* **Purpose**: Adds the specified amount to the account balance.
* **Parameters**:
  + amount (float): The amount to deposit.
* **Return**: None

**withdraw(self, amount)**

* **Purpose**: Deducts the specified amount from the account balance if sufficient funds exist.
* **Parameters**:
  + amount (float): The amount to withdraw.
* **Return**: None

**adjust\_interest\_rate(self, new\_rate)**

* **Purpose**: Changes the account’s interest rate to the new value.
* **Parameters**:
  + new\_rate (float): New annual interest rate as a percentage.
* **Return**: None

**get\_balance(self)**

* **Purpose**: Returns the current balance of the account.
* **Parameters**: None
* **Return**:
  + float: The current balance

**calculate\_interest(self, days)**

* **Purpose**: Calculates how much interest the account would earn over a given number of days.
* **Parameters**:
  + days (int): The number of days to calculate interest for.
* **Return**:
  + float: Calculated interest amount

**\_\_str\_\_(self)**

* **Purpose**: Returns a string representation of the account, including holder info, balance, and interest rate.
* **Parameters**: None
* **Return**:
  + str: Formatted summary string

**test\_bank\_acct()**

* **Purpose**: Drives the main program. Collects user input, creates an account, and provides a menu for interaction.
* **Parameters**: None
* **Return**: None

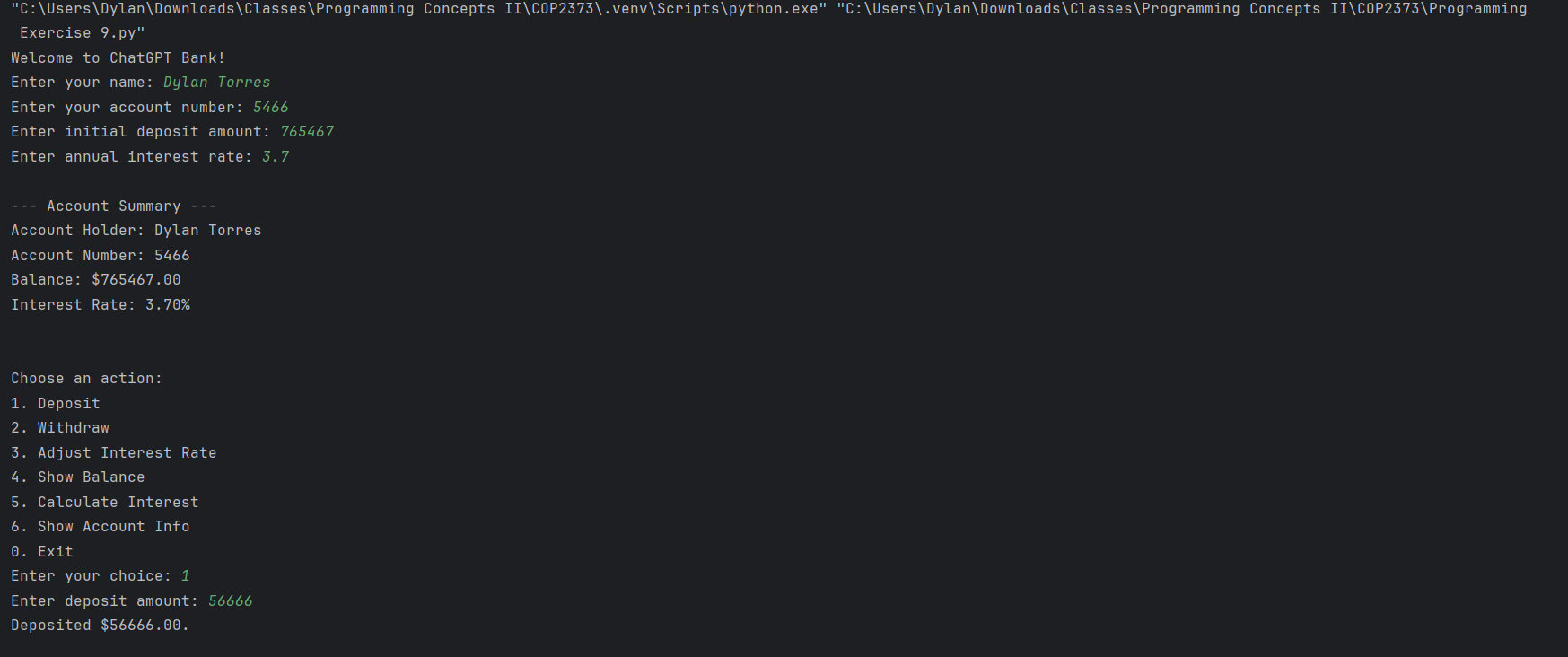
**3. Logical Steps of the Program**

1. **Start program execution** by running the test\_bank\_acct() function under the \_\_main\_\_ block.
2. **Prompt the user** for:
   * Name
   * Account number
   * Initial deposit
   * Interest rate
3. **Create a BankAcct object** using the provided input.
4. **Display a menu of options**:
   * Deposit
   * Withdraw
   * Adjust interest rate
   * Show balance
   * Calculate interest
   * Show full account info
   * Exit
5. **Handle the user’s menu choice**:
   * Validate and parse input values.
   * Call the appropriate method on the BankAcct object.
   * Loop back to the menu until the user chooses to exit.
6. **End the program** with a goodbye message.

**4. Link to your COP2373 repository:**

<https://github.com/Shinymon/COP2373>

**5. Screenshots:**



A screenshot of a computer

AI-generated content may be incorrect.