**1. Build a Login System with Maximum Retry Limit**

**Objective:**  
Create a secure login system that gives users a limited number of attempts to enter the correct username and password.

**Instructions:**

* Store a correct username and password in the program (hardcoded).
* Allow the user **up to 3 attempts** to log in.
* After each failed attempt, show how many tries are left.
* If the login is successful, continue to the main program.
* If the user fails 3 times, display an error and exit the program.

**Expected Output:**

* On success: "Login successful!"
* On failure: "Too many failed attempts. Exiting program."

**2. Implement a Menu Loop Using while True**

**Objective:**  
Design an interactive, menu-driven system that allows continuous user interaction until they choose to exit.

**Instructions:**

* Use a while True loop to display a menu with options.
* Ask the user to choose from options like:
  1. Check Balance
  2. Withdraw
  3. Deposit
  4. View All Accounts
  5. Exit
* Use if, elif, and else to handle each menu option.
* The menu should repeat after each operation unless the user chooses "Exit".

**Expected Behavior:**

* The menu appears after each action.
* The loop exits only when the user selects option 5 (Exit).

**3. Loop Through Account Balances**

**Objective:**  
Simulate a system where a user may have multiple accounts, and display all balances in a list format.

**Instructions:**

* Create a list of balances for multiple accounts, e.g., [1000, 1500, 800].
* Provide a menu option to **view all account balances**.
* When the user selects that option, loop through the list using a for loop and display each account and its balance.

**Expected Output:**

All Account Balances:

Account 1: $1000

Account 2: $1500

Account 3: $800

ANSWER

# --- Task 1: Login system with max retry ---

correct\_username = "user123"

correct\_password = "pass123"

max\_attempts = 3

attempts = 0

while attempts < max\_attempts:

username = input("Enter username: ")

password = input("Enter password: ")

if username == correct\_username and password == correct\_password:

print("\n Login successful!\n")

break

else:

attempts += 1

remaining = max\_attempts - attempts

print(f" Invalid credentials. {remaining} attempt(s) left.\n")

if attempts == max\_attempts:

print(" Too many failed attempts. Exiting program.")

exit()

# --- Task 2 & 3: Menu loop + Loop through account balances ---

account\_balances = [1000, 1500, 800] # Example: 3 account balances

selected\_account = 0 # We’re working with Account 1 by default

while True:

print("========== ATM MENU ==========")

print("1. Check Balance")

print("2. Withdraw Money")

print("3. Deposit Money")

print("4. View All Account Balances")

print("5. Exit")

choice = input("Enter your choice (1-5): ")

if choice == "1":

print(f"Account {selected\_account + 1} Balance: ${account\_balances[selected\_account]}")

elif choice == "2":

amount = float(input("Enter amount to withdraw: $"))

if amount <= account\_balances[selected\_account]:

account\_balances[selected\_account] -= amount

print(f"Withdrawal successful. New balance: ${account\_balances[selected\_account]}")

else:

print("Insufficient funds!")

elif choice == "3":

deposit = float(input("Enter amount to deposit: $"))

account\_balances[selected\_account] += deposit

print(f"Deposit successful. New balance: ${account\_balances[selected\_account]}")

elif choice == "4":

print("\n📋 All Account Balances:")

for i, balance in enumerate(account\_balances):

print(f"Account {i + 1}: ${balance}")

print()

elif choice == "5":

print("Thank you for using the ATM. Goodbye!")

break

else:

print("Invalid choice. Please select a number between 1 and 5.\n")