## Activity 6: Inventory Management

**Objective:**

The goal is to create a program that manages an inventory system using multi-dimensional arrays. The program will store product details (including names and stock quantities), display them to the user, allow updates to the inventory stock, and provide an option to reset the inventory to its initial values.

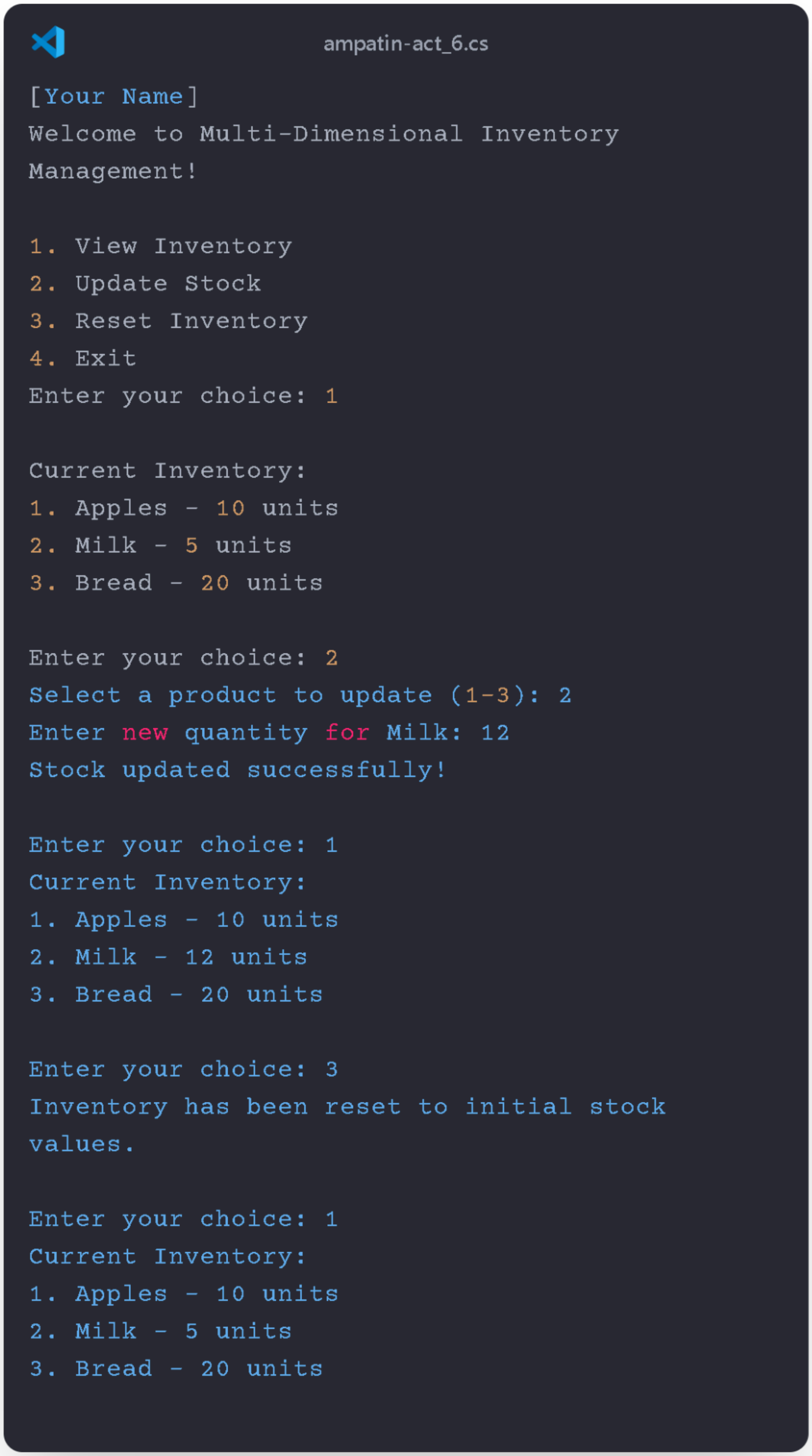
---

**Implementation Steps:**

1. **Create a new C# Console Application.**
   - Set up a new C# console application in Visual Studio or any C# IDE of your choice.
2. **Define a Two-Dimensional Array to Store Product Details:**
   - The two-dimensional array will store product information across two rows:
     - **Row 1** will store product names (e.g., Apples, Milk, Bread).
     - **Row 2** will store the corresponding stock quantities for each product (e.g., 10, 5, 20).
   - **Array Structure:**
     - `products[0, 0] = "Apples"` — The first row stores the product names.
     - `products[1, 0] = "10"` — The second row stores the stock quantities for the corresponding product in the first row.
3. **Display Inventory Details:**
   - The program will display the inventory in a structured format, showing product names along with their stock quantities.
   - This will be done by iterating over the rows of the two-dimensional array and printing each product with its stock.
4. **Create a Menu with Options:**
   - **Option 1: View Inventory** – This option will display the current inventory, listing all products along with their respective stock quantities.
   - **Option 2: Update Stock** – This option will allow the user to update the stock for a specific product by selecting the product by number (1, 2, or 3). The user can enter a new stock quantity for the selected product.
   - **Option 3: Reset Inventory** – This new option will allow the user to reset all products' stock quantities to their original values (as defined in the `initialStock` array). This is useful if the inventory changes were accidental or if the user wants to start over with the initial stock levels.
   - **Option 4: Exit** – This option will allow the user to exit the program.

5. **Keep the Program Running:**
   - Use a loop to keep the program running until the user selects the "Exit" option. The loop will continuously display the menu options and perform the chosen actions.

---



```
[Your Name]
Welcome to Multi-Dimensional Inventory
Management!

1. View Inventory
2. Update Stock
3. Reset Inventory
4. Exit
Enter your choice: 1

Current Inventory:
1. Apples - 10 units
2. Milk - 5 units
3. Bread - 20 units

Enter your choice: 2
Select a product to update (1-3): 2
Enter new quantity for Milk: 12
Stock updated successfully!

Enter your choice: 1
Current Inventory:
1. Apples - 10 units
2. Milk - 12 units
3. Bread - 20 units

Enter your choice: 3
Inventory has been reset to initial stock
values.

Enter your choice: 1
Current Inventory:
1. Apples - 10 units
2. Milk - 5 units
3. Bread - 20 units
```