INVENTORY MANAGEMENT SYSTEM IN PYTHON

Presented by
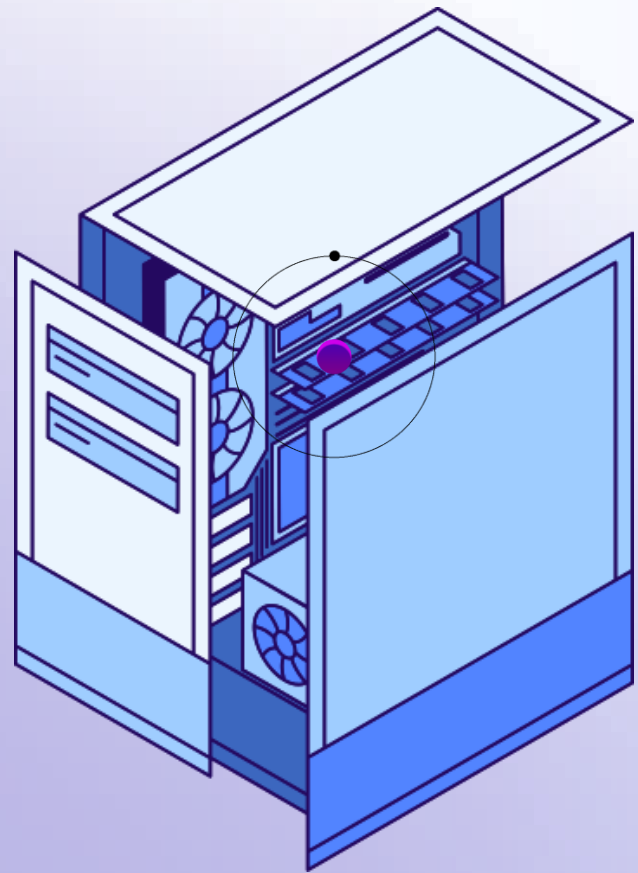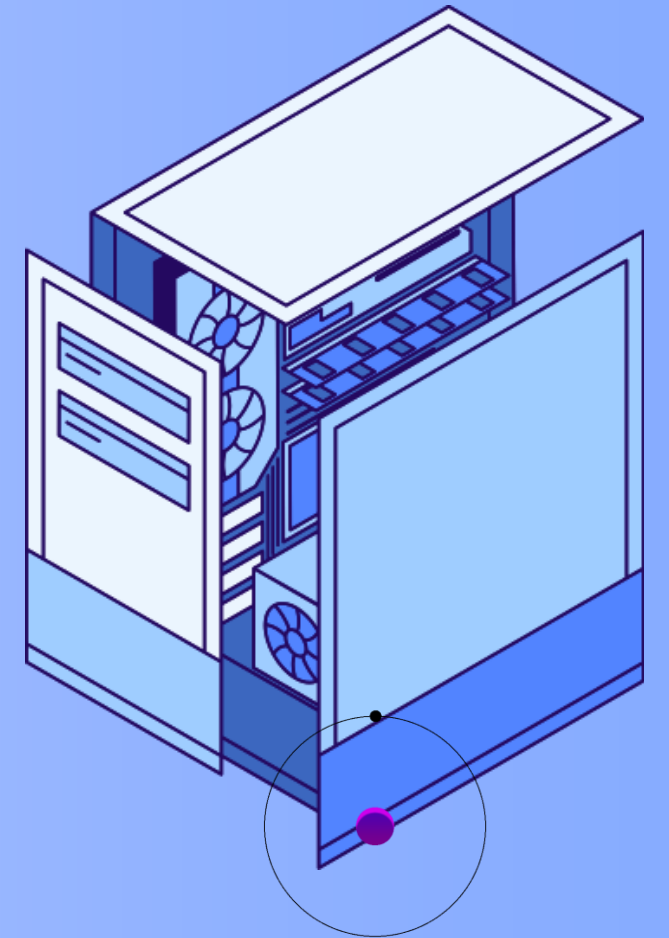
S.M. TAUKIR RAHMAN
241-56-005

INTRODUCTION

What is an Inventory Management System?
- A system to track and manage items in an inventory.
- Helps businesses keep track of the stock of products, update quantities, and remove items as necessary.
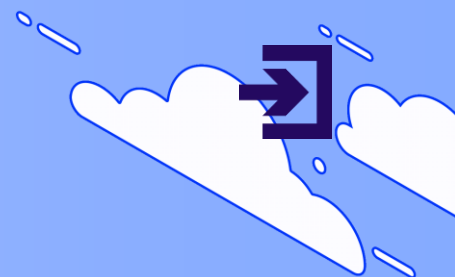
Objective:
- To build a simple Inventory Management System using Python's object-oriented programming (OOP) principles.

CONCEPTS COVERED

- **Classes & Objects**
  - Using Python classes to structure the system.
- **Dictionary for Storage**
  - Storing inventory data in a dictionary.
- **Basic Operations**
  - Add Item
  - Update Item
  - Remove Item
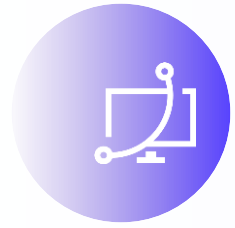 - View Inventory

THE CODE STRUCTURE

1. **Inventory Class**
   - Defines the inventory system and its operations:
     - add_item()
     - update_item()
     - remove_item()
     - view_inventory()
2. **Menu System**
   - A simple interactive menu that allows the user to:
   - Add items, update, remove, or view the inventory.

CLASS DEFINITION - INVENTORY

```python
class Inventory:
    def __init__(self):
        self.items = {}  # Store items in format: item_name: quantity

    def add_item(self, item_name, quantity):
        if item_name in self.items:
            print(f"{item_name} already exists. Use 'Update Item' to change quantity."
        else:
            self.items[item_name] = quantity
            print(f"{item_name} added with quantity {quantity}.")

    def update_item(self, item_name, quantity):
        if item_name in self.items:
            self.items[item_name] = quantity
            print(f"{item_name} updated to quantity {quantity}.")
        else:
            print(f"{item_name} not found in inventory.")

    def remove_item(self, item_name):
        if item_name in self.items:
            del self.items[item_name]
            print(f"{item_name} removed from inventory.")
        else:
            print(f"{item_name} not found in inventory.")

    def view_inventory(self):
        if not self.items:
            print("Inventory is empty.")
        else:
            print("\n--- Inventory List ---")
            for name, qty in self.items.items():
                print(f"{name}: {qty}")
                print("----------------------")
```

MENU SYSTEM
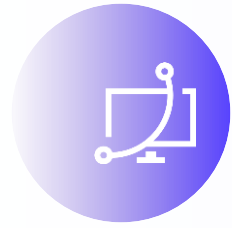
```python
35    # Main menu loop
36    def main():
37        inventory = Inventory()
38
39        while True:
40            print("\n=== Inventory Menu ===")
41            print("1. Add Item")
42            print("2. Update Item")
43            print("3. Remove Item")
44            print("4. View Inventory")
45            print("5. Exit")
46            choice = input("Enter your choice (1-5): ")
47
48            if choice == '1':
49                name = input("Enter item name: ")
50                qty = int(input("Enter quantity: "))
51                inventory.add_item(name, qty)
52
53            elif choice == '2':
54                name = input("Enter item name to update: ")
55                qty = int(input("Enter new quantity: "))
56                inventory.update_item(name, qty)
57
58            elif choice == '3':
59                name = input("Enter item name to remove: ")
60                inventory.remove_item(name)
61
62            elif choice == '4':
63                inventory.view_inventory()
64
65            elif choice == '5':
66                print("Exiting Inventory System. Goodbye!")
67                break
68
69            else:
70                print("Invalid choice. Please select from 1 to 5.")
71
72    if __name__ == "__main__":
73        main()
```

SAMPLE OUTPUT

```
=== Inventory Menu ===
1. Add Item
2. Update Item
3. Remove Item
4. View Inventory
5. Exit
Enter your choice (1-5): 1
Enter item name: apple
Enter quantity: 10
apple added with quantity 10.

=== Inventory Menu ===
1. Add Item
2. Update Item
3. Remove Item
4. View Inventory
5. Exit
Enter your choice (1-5): 4

--- Inventory List ---
apple: 10
-----------------------

=== Inventory Menu ===
1. Add Item
2. Update Item
3. Remove Item
4. View Inventory
5. Exit
Enter your choice (1-5):
```

THANK YOU!