Data Structures and Algorithms

SuperSet ID:6412063

Exercise 1: Inventory Management System

Code:

import java.util.HashMap;

import java.util.Map;

import java.util.Scanner;

class Product {

int productId;

String productName;

int quantity;

double price;

public Product(int productId, String productName, int quantity, double price) {

this.productId = productId;

this.productName = productName;

this.quantity = quantity;

this.price = price;

}

public String toString() {

return "Product ID: " + productId + ", Name: " + productName +

", Quantity: " + quantity + ", Price: ₹" + price;

}

}

public class InventorySystem {

private static Map<Integer, Product> inventory = new HashMap<>();

public static void addProduct(Product product) {

inventory.put(product.productId, product);

System.out.println("Product added successfully.");

}

public static void updateProduct(int productId, int quantity, double price) {

if (inventory.containsKey(productId)) {

Product product = inventory.get(productId);

product.quantity = quantity;

product.price = price;

System.out.println("Product updated successfully.");

} else {

System.out.println("Product not found.");

}

}

public static void deleteProduct(int productId) {

if (inventory.remove(productId) != null) {

System.out.println("Product deleted successfully.");

} else {

System.out.println("Product not found.");

}

}

public static void displayInventory() {

if (inventory.isEmpty()) {

System.out.println("Inventory is empty.");

} else {

for (Product product : inventory.values()) {

System.out.println(product);

}

}

}

public static void main(String[] args) {

addProduct(new Product(101, "Laptop", 10, 55000));

addProduct(new Product(102, "Mouse", 50, 500));

addProduct(new Product(103, "Keyboard", 30, 1500));

System.out.println("\nInventory:");

displayInventory();

System.out.println("\nUpdating product 102...");

updateProduct(102, 40, 450);

System.out.println("\nDeleting product 103...");

deleteProduct(103);

System.out.println("\nInventory after updates:");

displayInventory();

}

}

Output:

