**Exercise 1: Inventory Management System**

**Product.java**

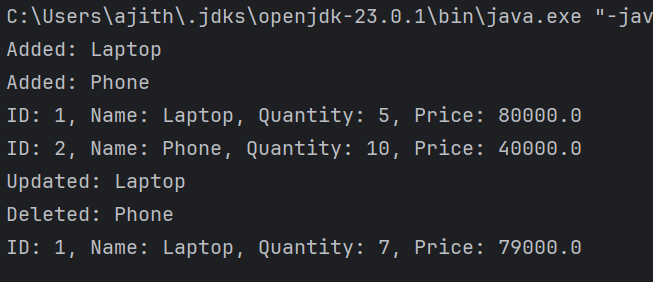
public class Product {  
 private int productId;  
 private String productName;  
 private int quantity;  
 private double price;  
  
 public Product(int productId, String productName, int quantity, double price) {  
 this.productId = productId;  
 this.productName = productName;  
 this.quantity = quantity;  
 this.price = price;  
 }  
  
 public int getProductId() {  
 return productId;  
 }  
  
 public String getProductName() {  
 return productName;  
 }  
  
 public int getQuantity() {  
 return quantity;  
 }  
  
 public double getPrice() {  
 return price;  
 }  
  
 public void setQuantity(int quantity) {  
 this.quantity = quantity;  
 }  
  
 public void setPrice(double price) {  
 this.price = price;  
 }  
  
 public String toString() {  
 return "ID: " + productId + ", Name: " + productName + ", Quantity: " + quantity + ", Price: " + price;  
 }  
}

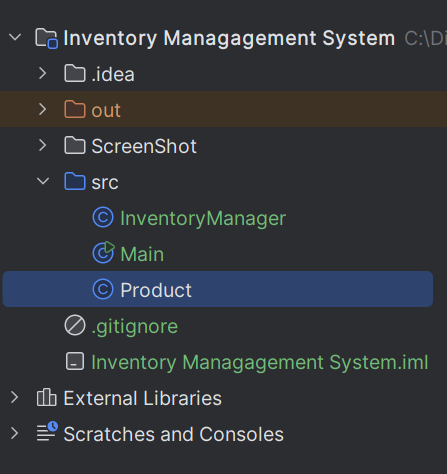
**InventoryManager.java**

import java.util.\*;  
  
public class InventoryManager {  
 private HashMap<Integer, Product> inventory = new HashMap<>();  
  
 public void addProduct(Product p) {  
 inventory.put(p.getProductId(), p);  
 System.*out*.println("Added: " + p.getProductName());  
 }  
  
 public void updateProduct(int id, int quantity, double price) {  
 if (inventory.containsKey(id)) {  
 Product p = inventory.get(id);  
 p.setQuantity(quantity);  
 p.setPrice(price);  
 System.*out*.println("Updated: " + p.getProductName());  
 } else {  
 System.*out*.println("Not found");  
 }  
 }  
  
 public void deleteProduct(int id) {  
 if (inventory.containsKey(id)) {  
 Product p = inventory.remove(id);  
 System.*out*.println("Deleted: " + p.getProductName());  
 } else {  
 System.*out*.println("Not found");  
 }  
 }  
  
 public void displayInventory() {  
 if (inventory.isEmpty()) {  
 System.*out*.println("Inventory empty");  
 } else {  
 for (Product p : inventory.values()) {  
 System.*out*.println(p);  
 }  
 }  
 }  
}

**Main.java**

public class Main {  
 public static void main(String[] args) {  
 InventoryManager im = new InventoryManager();  
  
 Product p1 = new Product(1, "Laptop", 5, 80000);  
 Product p2 = new Product(2, "Phone", 10, 40000);  
  
 im.addProduct(p1);  
 im.addProduct(p2);  
  
 im.displayInventory();  
  
 im.updateProduct(1, 7, 79000);  
 im.deleteProduct(2);  
  
 im.displayInventory();  
 }  
}

**Output**



**Exercise 2: E-commerce Platform Search Function**

**Product.java**

public class Product {  
 int productId;  
 String productName;  
 String category;  
  
 Product(int productId, String productName, String category) {  
 this.productId = productId;  
 this.productName = productName;  
 this.category = category;  
 }  
  
 public String toString() {  
 return productId + " " + productName + " " + category;  
 }  
}

**SearchFunction.java**

public class SearchFunction {  
  
 static int linearSearch(Product[] arr, String name) {  
 for (int i = 0; i < arr.length; i++) {  
 if (arr[i].productName.equals(name)) {  
 return i;  
 }  
 }  
 return -1;  
 }  
  
 static void bubbleSort(Product[] arr) {  
 for (int i = 0; i < arr.length - 1; i++) {  
 for (int j = 0; j < arr.length - i - 1; j++) {  
 if (arr[j].productName.compareTo(arr[j + 1].productName) > 0) {  
 Product temp = arr[j];  
 arr[j] = arr[j + 1];  
 arr[j + 1] = temp;  
 }  
 }  
 }  
 }  
  
 static int binarySearch(Product[] arr, String name) {  
 int low = 0;  
 int high = arr.length - 1;  
  
 while (low <= high) {  
 int mid = (low + high) / 2;  
 if (arr[mid].productName.equals(name)) {  
 return mid;  
 }  
 if (arr[mid].productName.compareTo(name) < 0) {  
 low = mid + 1;  
 } else {  
 high = mid - 1;  
 }  
 }  
 return -1;  
 }  
  
 public static void main(String[] args) {  
 Product[] arr = new Product[4];  
 arr[0] = new Product(1, "Laptop", "Electronics");  
 arr[1] = new Product(2, "Phone", "Electronics");  
 arr[2] = new Product(3, "Shoes", "Footwear");  
 arr[3] = new Product(4, "Watch", "Accessories");  
  
 int res1 = *linearSearch*(arr, "Phone");  
 if (res1 != -1) {  
 System.*out*.println("Linear found: " + arr[res1]);  
 } else {  
 System.*out*.println("Linear not found");  
 }  
  
 *bubbleSort*(arr);  
  
 int res2 = *binarySearch*(arr, "Phone");  
 if (res2 != -1) {  
 System.*out*.println("Binary found: " + arr[res2]);  
 } else {  
 System.*out*.println("Binary not found");  
 }  
 }  
}

**Output**

