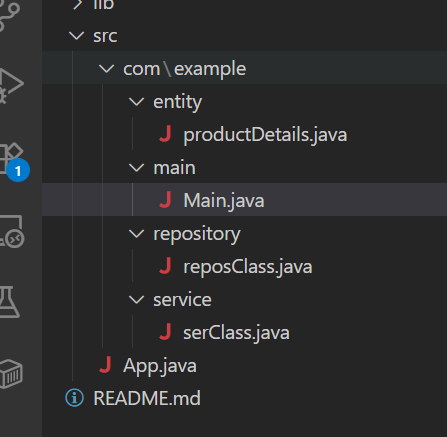
**Inventory Management System**

### **Suitable Data Structures**

| **Data Structure** | **Pros** | **Use Case** |
| --- | --- | --- |
| **HashMap** (Dictionary) | Fast lookup (O(1) average), update, delete by key | Ideal when accessing products by productId |
| **ArrayList** | Ordered list, fast access by index | Useful if order matters and few deletions |
| **LinkedList** | Easy insertion/deletion at ends | Less efficient for searching |
| **TreeMap** | Sorted keys, moderate speed (O(log n)) | When sorted access is required |

✅ **Best choice**: HashMap<Integer, Product> – fast access using productId.

**CODE:**



productDetails.java

package com.example.entity;

public class productDetails {

private int productId;

private String productName;

private int quantity;

private double price;

public productDetails(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 void setProductId(int productId) {

this.productId = productId;

}

public String getProductName() {

return productName;

}

public void setProductName(String productName) {

this.productName = productName;

}

public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

public double getPrice() {

return price;

}

public void setPrice(double price) {

this.price = price;

}

@Override

public String toString() {

return "productDetails [productId=" + productId + ", productName=" + productName + ", quantity=" + quantity

+ ", price=" + price + "]";

}

}

main.java

package com.example.main;

import java.util.Scanner;

import com.example.service.serClass;

public class Main {

private static Scanner sc = new Scanner(System.in);

private static Main main = new Main();

private static serClass ser = new serClass();

public static void main(String args[]) {

System.out.println();

System.out.println("INVENTORY MANAGEMENT SYSTEM : ");

System.out.println();

boolean flag = true;

while (flag) {

System.out.println("1. Add Product Details");

System.out.println("2. Update details");

System.out.println("3. Delete a Product");

System.out.println("4. Exit");

System.out.print("Choose an Option : ");

int option = sc.nextInt();

System.out.println();

sc.nextLine();

switch (option) {

case 1:

main.addProduct();

break;

case 2:

main.updateProduct();

break;

case 3:

System.out.print("Enter a product id to delete = ");

int id = sc.nextInt();

sc.nextLine();

System.out.println();

ser.deleteProduct(id);

break;

case 4:

System.out.println("Successfully Exited...");

flag = false;

break;

default:

System.out.println("Invalid Option.");

break;

}

System.out.println();

System.out.println(

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println();

}

}

private void addProduct() {

System.out.print("Enter the product id = ");

int id = sc.nextInt();

sc.nextLine();

System.out.println();

System.out.print("Enter the product Name = ");

String name = sc.nextLine();

System.out.println();

System.out.print("Enter the number of quantity = ");

int quantity = sc.nextInt();

sc.nextLine();

System.out.println();

System.out.print("Enter thr price of the product = ");

double price = sc.nextDouble();

sc.nextLine();

System.out.println();

ser.addProduct(id, name, quantity, price);

}

private void updateProduct() {

System.out.print("Enter the product id to update= ");

int id = sc.nextInt();

sc.nextLine();

System.out.println();

System.out.print("Enter the product Name to update= ");

String name = sc.nextLine();

System.out.println();

System.out.print("Enter the number of quantity to update= ");

int quantity = sc.nextInt();

sc.nextLine();

System.out.println();

System.out.print("Enter thr price of the product to update= ");

double price = sc.nextDouble();

sc.nextLine();

System.out.println();

ser.updateProduct(id, name, quantity, price);

}

}

repoClass.java

package com.example.repository;

import java.util.HashMap;

import com.example.entity.productDetails;

public class reposClass {

private static HashMap<Integer, productDetails> map = new HashMap<>();

private static reposClass repo = new reposClass();

public void addProduct(int id, String name, int quantity, double price) {

if (map.containsKey(id)) {

System.out.println("Product already exists");

} else {

productDetails pro = new productDetails(id, name, quantity, price);

map.put(id, pro);

System.out.println("Product Added Successfully");

System.out.println();

repo.displayAll();

}

}

public void updateProduct(int id, String name, int quantity, double price) {

if (map.containsKey(id)) {

productDetails pro = map.get(id);

pro.setProductName(name);

pro.setQuantity(quantity);

pro.setPrice(price);

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

System.out.println();

repo.displayAll();

} else {

System.out.println("Invalid product id");

System.out.println();

}

}

public void deleteProduct(int id)

{

if(map.containsKey(id))

{

map.remove(id);

System.out.println("Deleted Successfully");

System.out.println();

repo.displayAll();

}

else

{

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

}

}

private void displayAll() {

System.out.println("Inventory : ");

System.out.println("ID \t PRO\_NAME \t QUANTITY \t PRICE");

System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

for (productDetails pro : map.values()) {

System.out.println(pro.getProductId() + "\t" + pro.getProductName() + "\t\t" + pro.getQuantity() + "\t\t"

+ pro.getPrice());

System.out.println("------------------------------------------------------------------------");

}

System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

}

serClass.java

package com.example.service;

import com.example.repository.reposClass;

public class serClass {

private static reposClass repo = new reposClass();

public void addProduct(int id,String name,int quantity,double price)

{

repo.addProduct(id,name,quantity,price);

}

public void updateProduct(int id,String name,int quantity,double price)

{

repo.updateProduct(id, name, quantity, price);

}

public void deleteProduct(int id)

{

repo.deleteProduct(id);

}

}

### **Time Complexity**

| **Operation** | **Time Complexity** | **Notes** |
| --- | --- | --- |
| **Add** | O(1) average | Uses put() in HashMap |
| **Update** | O(1) average | Uses get() and set values |
| **Delete** | O(1) average | Uses remove() |
| **Display** | O(n) | Loop over values |

## **Output :**

