# Introduction

The purpose of this assignment is to demonstrate the extent of knowledge acquired on the topic of Software Development over the weeks. For this task, I chose to design a product inventory system to store product information, as well as design an inventory system that would generate a report of the stock at hand and its monetary value.

# Inventory System

To achieve this purpose, the system would have to be comprised of 2 classes:

* Product class to store individual product info.
* Inventory class to generate a report based on product stored in memory.

Originally, the system was to be built with only these 2 classes; however for the purpose of efficient coupling and cohesion, that is independency between these 2 classes among other things, the system was designed with a Main class, which would perform the following:

* A *Public Static Void Main* method, for interaction with the user via console.
* Creating an object to hold the info concerning the products.
* Generating of a financial report on screen.

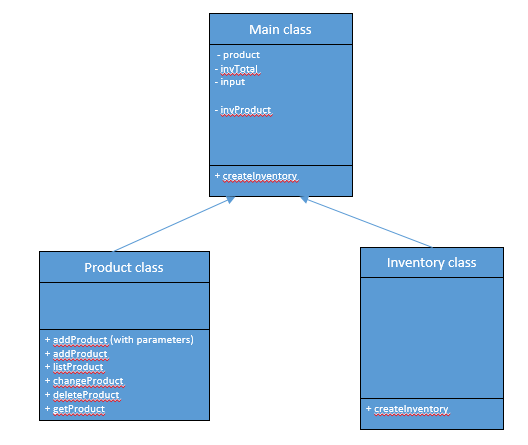


Figure : Class diagram of system

## Main class

The Main class is the class which serves as interface between the system and the user. It has a few variables declared in it, which are inherited by the other 2 classes. Such variables are:

**product**: *hashmap* to store the product info with a key (unique identifier) and value. The key is the product ID which is unique to every product, while the value contains the name, quantity, price of the product.

**invTotal**: imports the product *hashmap* to generate the total value of each product, as well as calculate the total value of the inventory.

**scanner**: enables the user to input data in the system.

All these variables are declared as *protected*, which means that they are accessible from any class within the package.

The Main class also contains a *public static void main* method, which creates a new object of Product class, and uses the functions from the latter to perform the different methods within that class.

It also has a method to generate a report, using the *invTotal* variable.

For the user to perform the different tasks required, there is a sample list of products loaded at the beginning. Then a loop is used to ensure that the user can perform tasks, until he chooses to quit the system.

## The Product class

Its function is to inherit the product variable (from the Main class), and use it to add, delete, change, or list products. Since it is using the variables from the Main class to perform the different methods within it, no other variable has to be declared in the class. However a few variables are created within some methods, such as in the addProduct method, where a list is declared and used to store the name, quantity, and price of a product. That list is then added as the value of the product, while the product ID is used as the key. Therefore using the key (product ID), the information concerning the product can be retrieved when needed.

## The Inventory class

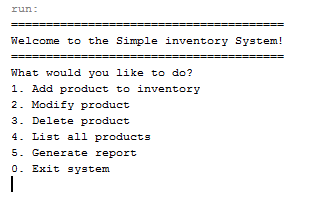
It imports the product variable, and converts the quantity field and price field back into integer and float, and multiplies these 2 values to get the total value of 1 product. Then it uses the latter to increment the value of the total stock. It also presents the data in a formatted display on the console screen.

However since the Main class has the same method in it, it is overridden.

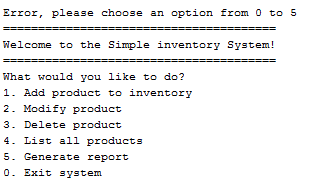
# Testing

I tried to introduce a few Junit testing, but unfortunately since the system mainly uses void methods, there are no return values to compare with using the *assert* method.

**Running the system**



**Inputting a number out of range**



# Conclusion

As demonstrated with the running of the program, it works. However it is to be noted that a few more things could have been implemented in the system, such as a GUI for better looks and interactivity with the user, as well as more error handling. This would also ensure that the system could keep on working even though erroneous data was inputted.

As long as the user inputs the right information, the system responds well.

# Appendix

## The Main Class

/\*

\* This is assignment 2 for Software Development.

\* This class contains the main method to launch the program.

\* It also create the object to store products and inventory

\* and call methods from the product and inventory class.

\*/

package IT6260\_Assignment2\_2804582;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.Map;

import java.util.Scanner;

/\*\*

\*

\* @author : Francois Gilles Mindiel

\* ID number: 2804582

\*/

public class Main {

// declaring variables to store product info.

protected HashMap<Integer, ArrayList<String>> product = new HashMap<>();

protected float invTotal;

protected Scanner scanner = new Scanner(System.in); // to get user input

public static void main(String[] args) {

Product invProduct = new Product(); // creating object of Product class

Scanner input = new Scanner(System.in);

boolean quit = false;

//populating inventory with sample info

invProduct.addProduct(45, "Shapes", 15, 2.99);

invProduct.addProduct(25, "Tea", 100, 1.20);

invProduct.addProduct(53, "coffee", 20, 6.99);

invProduct.addProduct(34, "sugar", 10, 0.99);

invProduct.addProduct(2, "cream", 70, 3.49);

invProduct.addProduct(4, "sprite", 50, 3);

//creates a welcome screen which comes back until user wishes to quit

while (quit != true){

//creating a dialog on console screen to get user input

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

System.out.println("Welcome to the Simple inventory System!");

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

System.out.println("What would you like to do?");

System.out.println("1. Add product to inventory\n"

+ "2. Modify product\n"

+ "3. Delete product\n"

+ "4. List all products\n"

+ "5. Generate report\n"

+ "0. Exit system");

int choice = input.nextInt(); // to get user option using integer

if (choice == 1) {

invProduct.clearScreen();

invProduct.addProduct();

}

if (choice == 2) {

invProduct.clearScreen();

invProduct.changeProduct();

}

if (choice == 3) {

invProduct.clearScreen();

invProduct.deleteProduct();

}

if (choice == 4) {

invProduct.clearScreen();

invProduct.listProduct();

}

if (choice == 5) {

invProduct.clearScreen();

invProduct.createInventory();

}

if (choice == 0) {

quit = true;

}

if ((choice <0) || (choice >5)) { // else option was always printing error message...

invProduct.clearScreen();

System.out.println("Error, please choose an option from 0 to 5");

}

}

// when exiting welcome screen!

invProduct.clearScreen();

System.out.println("Thank you for your visit. Goodbye");

}

public void clearScreen(){

for (int i = 0; i < 50; i++) { // to "clear" console screen

System.out.println("");

}

}

public void createInventory(){

System.out.println("Product ID"+"\t"+"Product Name"+"\t\t\t"+"Quantity"+"\t"+"Price"+"\t\t"+"Total Value");

for (Map.Entry<Integer, ArrayList<String>> entry : product.entrySet()) {

Integer key = entry.getKey();

ArrayList value = entry.getValue();

// extracts String data "quantity" and "price" and converts into float. Then multiplies both numbers

float itemTotal = (Integer.parseInt(String.valueOf(value.get(1))))\*(Float.parseFloat(String.valueOf(value.get(2))));

value.add(itemTotal); // adds it to array, for display of inventory.

System.out.println(key+"\t\t"+value.get(0)+"\t\t\t\t"+value.get(1)+"\t\t$ "+value.get(2)+"\t\t$ "+value.get(3));

invTotal = invTotal + itemTotal;

}

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

System.out.println("\t\t\tINVENTORY VALUE:\t\t\t$" + invTotal);

}

}

## The Product Class

/\*

\* This is assignment 2 for Software Development.

\* This class creates a hashmap to store product info.

\*/

package IT6260\_Assignment2\_2804582;

// importing the java utility packages needed for array lists, hashmaps and maps.

import java.util.ArrayList;

import java.util.HashMap;

import java.util.Map;

/\*\*

\*

\* @author : Francois Gilles Mindiel

\* ID number: 2804582

\*/

public class Product extends Main{

/\*\*

\*

\* @param id

\* @param name

\* @param qty

\* @param price

\*/

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

/\*

method used to add product using parameters. Useful to populate the inventory

\*/

ArrayList productInfo = new ArrayList(); // stores name, quantity and price of each product

int pdctID = id; // only integer number allowed

String pdctName = name;

int pdctQty = qty; // quantity can only be whole number

double pdctPrice = price; // since price has cents

//populate the list

productInfo.add(pdctName);

productInfo.add(pdctQty);

productInfo.add(pdctPrice);

product.put(pdctID,productInfo); // to create hashmap, with id as key

}

public void addProduct(){ // gets input from user

/\*

this method gets user to input the different values to store product info

and store them in hashmap

\*/

ArrayList productInfo = new ArrayList();

System.out.println("Enter Product ID:");

int pdctID = scanner.nextInt();

System.out.println("Enter product name:");

String pdctName = scanner.next();

System.out.println("Enter Product quantity:");

int pdctQty = scanner.nextInt();

System.out.println("Enter product price:");

double pdctPrice = scanner.nextFloat();

//populate the list

productInfo.add(pdctName);

productInfo.add(pdctQty);

productInfo.add(pdctPrice);

try {

product.put(pdctID,productInfo); // to create hashmap, with id as key

System.out.println("Product added to inventory");

} catch (Exception e) {

System.out.println("Error. Please check that input is correct");

}

}

public void listProduct(){

System.out.println("Product ID"+"\t"+"Product Name"+"\t\t\t"+"Quantity"+"\t"+"Price");

for (Map.Entry<Integer, ArrayList<String>> entry : product.entrySet()) {

Integer key = entry.getKey();

ArrayList value = entry.getValue();

System.out.println(key+"\t\t"+value.get(0)+"\t\t\t\t"+value.get(1)+"\t\t$ "+value.get(2));

}

}

public void changeProduct(){

/\*

used to modify a product. it selects a product first. then gets user input stored

in a different variable (temp), deletes the product, and adds the "new" product,

with same product key as previously back into the list.

\*/

System.out.println("enter pdct id number:");

int id = scanner.nextInt();

String answer = scanner.next();

ArrayList temp = product.get(id);

System.out.println("Product: "+temp.get(0));

System.out.println("");

System.out.println("Enter new name:");

String newName = scanner.next();

System.out.println("Enter new quantity:");

int newQty = scanner.nextInt();

System.out.println("Enter new price:");

double newPrice = scanner.nextDouble();

System.out.println("Are you sure you want to modify product?");

System.out.println("yes or no");

if ("yes".equals(answer)) {

temp.clear();

temp.add(newName);

temp.add(newQty);

temp.add(newPrice);

product.remove(id);

product.put(id, temp);

//Product info changed!!!

System.out.println(product.get(id));

if ("no".equals(answer)) {

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

}

}

}

public void deleteProduct(){

System.out.println("enter pdct id number:");

int id = scanner.nextInt();

String answer = scanner.next();

ArrayList temp = product.get(id);

System.out.println("Product: "+temp.get(0));

System.out.println("Are you sure you want to delete product?");

System.out.println("yes or no");

if ("yes".equals(answer)) {

product.remove(id);

System.out.println("Product removed from inventory");

}

if ("no".equals(answer)) {

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

}

}

public HashMap getProduct() {

return product;

}

}

## The Inventory class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package IT6260\_Assignment2\_2804582;

import java.util.ArrayList;

import java.util.Map;

/\*\*

\*

\* @author : Francois Gilles Mindiel

\* ID number: 2804582

\*/

public class Inventory extends Main{

@Override

public void createInventory(){

System.out.println("Product ID"+"\t"+"Product Name"+"\t\t\t"+"Quantity"+"\t"+"Price"+"\t\t"+"Total Value");

for (Map.Entry<Integer, ArrayList<String>> entry : product.entrySet()) {

Integer key = entry.getKey();

ArrayList value = entry.getValue();

// extracts String data "quantity" and "price" and converts into float. Then multiplies both numbers

float itemTotal = (Integer.parseInt(String.valueOf(value.get(1))))\*(Float.parseFloat(String.valueOf(value.get(2))));

value.add(itemTotal); // adds it to array, for display of inventory.

System.out.println(key+"\t\t"+value.get(0)+"\t\t\t\t"+value.get(1)+"\t\t$ "+value.get(2)+"\t\t$ "+value.get(3));

invTotal = invTotal + itemTotal; // stores total value of stock

}

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

System.out.println("\t\t\tINVENTORY VALUE:\t\t\t$" + invTotal);

}

}