

A LAB REPORT ON SUPERMARKET BILLING SYSTEM

JUNAID HOSSAIN | ID: 222008812 | CSE 124.1

SUMMER 2023 | AUGUST 18, 2023

INTRODUCTION:

The application I am trying to build is a **Supermarket Billing System** in java. It is a console-based application which has features to manage an inventory of products and calculate bills upon inputting the name and quantity of the product. In order to add proper functionality, on the user side of the application there will be two different roles a user can perform; one will be an administrative role and another will be a staff role. To perform a certain role, the user must input their username and their password to login, after login the user will be taken to their respective login page where they can choose between certain actions to perform depending if they are an administrator or a staff. The administrator can add products to the inventory, set their prices, set their initial quantity and additional option to update the price and quantity of a product. There will be an option to view the total list of inventory which will be shared between both administrator and staff. On the other side, the staff can view the total list of inventory and can conjure a bill when a product is purchased. In order to produce a bill, the staff will first choose the billing menu where they will be prompted to input the name and numbers of products purchased. Based on the retail price the admin has set, the program will show a bill to the customer with and without tax, where the taxed bill be deemed as the total bill. The program will be running indefinitely until the user chooses the exit option explicitly.

A diagram showing how the code will work is given below:

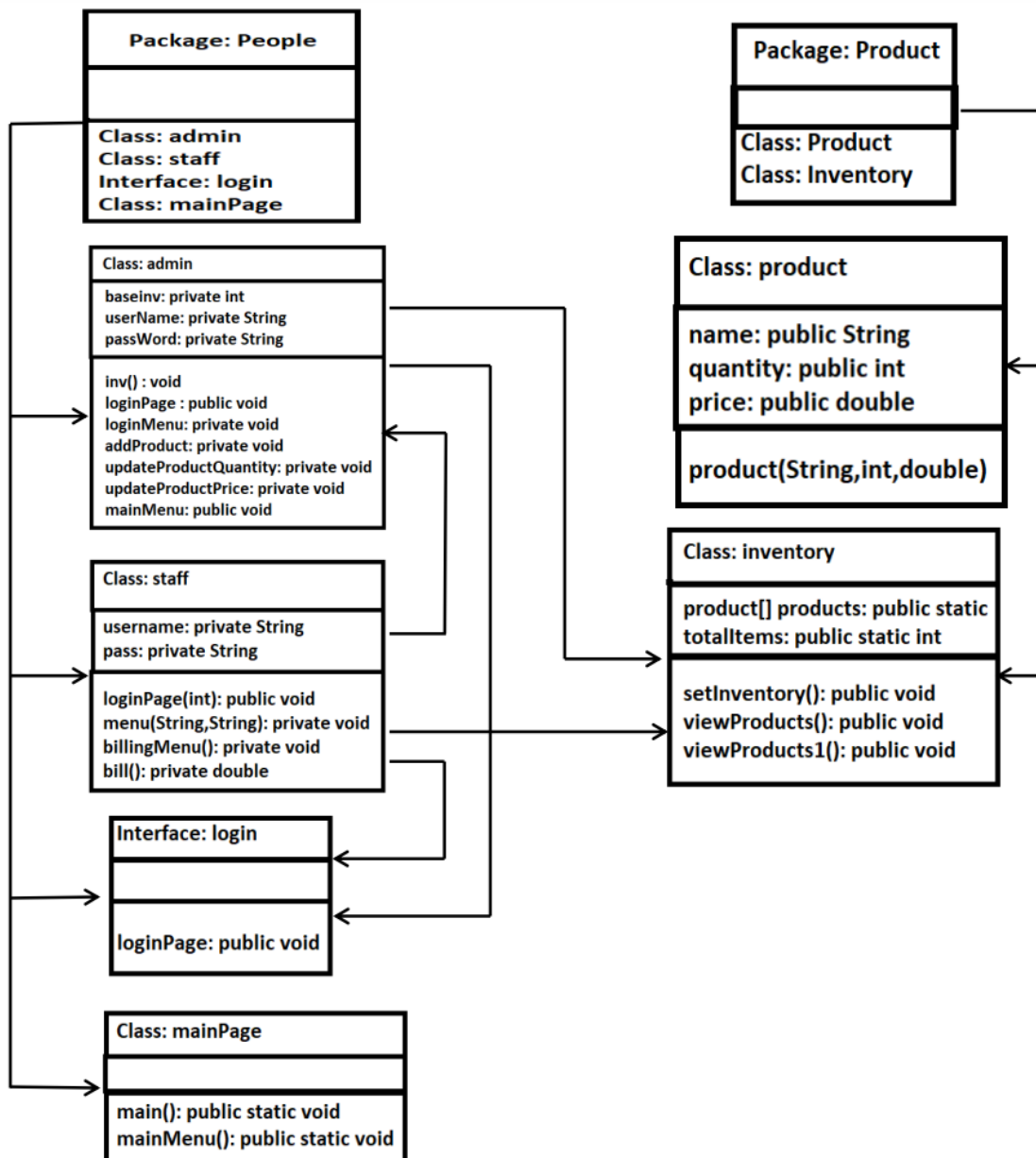


Fig 1 : Diagram showing the connection between classes and interfaces.

The Java code of the application:

```

package People;
import java.util.Scanner;

public class mainPage {
    public static void main(String[] args) {
        mainMenu();
    }
    public static void mainMenu() {
        Scanner s = new Scanner(System.in);
        while(true) {
            System.out.println("Main Menu : Choose one.\n"
                + "1. Amdin.\n"
                + "2. Staff.\n"
                + "3. Exit.");
            String c = s.next();

            while(true) {
                if(c.equals("1")) {
                    admin a1 = new admin();
                    a1.inv();
                    a1.loginPage();
                }else if(c.equals("2")) {
                    staff s1 = new staff();
                    s1.inv();
                    s1.loginPage(1);
                }else if(c.equals("3")) {
                    System.out.println("Program Exited.");
                    System.exit(0);
                }else {
                    System.out.println("Invalid Choice.");
                    break;
                }
            }
        }
    }
}

```

```

package People;
import java.util.Scanner;
import Product.*;
public class admin extends inventory implements login {
    Scanner s = new Scanner(System.in);
    private int baseInv=0;
    private String userName = "Admin";
    private String passWord = "password";
    void inv() {
        while(baseInv==0) {
            setInventory();
            baseInv++;
        }
    }

    public void loginPage() {
        while(true) {
            System.out.println("Enter your Admin username : ");
            String username = s.next();
            System.out.println("Enter your password : ");
            String pass = s.next();

            if(username.equalsIgnoreCase(userName) && pass.equals(passWord)) {
                while(true) {
                    loginMenu();
                }
            } else {
                System.out.println("Wrong username or password");
            }
        }
    }

    private void loginMenu() {
        int c=0;
        do {
            System.out.println("Choose one : \n"
                + "1. Add product\n"
                + "2. Update product quantity\n"
                + "3. Update product price\n"
                + "4. View products\n"
                + "5. Exit to Main Menu.");
            c = s.nextInt();
            if(c==1) {
                addProduct();
            }
            else if(c==2) {
                updateQuantity();
            }
            else if(c==3) {
                updatePrice();
            }
            else if(c==4) {
                viewProducts1();
            }
        } while(c != 5);
    }
}

```

```

    }
    else if(c==5) {
        mainMenu();
    }
    else {
        System.out.println("Invalid choice.");
    }
}while(true);
}

private void addProduct() {
    System.out.println("Enter the product name: ");
    String name = s.next();
    System.out.println("Enter the product quantity: ");
    int quantity = s.nextInt();
    System.out.println("Enter the product retail price: ");
    int price = s.nextInt();
    products[totalItems] = new product(name, quantity, price);
    totalItems++;
}

private void updateQuantity() {
    System.out.println("Enter the product name: ");
    String name = s.next();
    for (int i = 0; i <= totalItems; i++) {
        if(i==totalItems) {
            System.out.println("Product Not Available\n");
        }
        else if (products[i].name.equalsIgnoreCase(name)) {
            System.out.println("Enter the new product quantity: ");
            int quantity = s.nextInt();
            products[i].quantity = quantity;
            break;
        }
    }
}

private void updatePrice() {
    System.out.println("Enter the product name: ");
    String name = s.next();
    for (int i = 0; i <= totalItems; i++) {
        if(i==totalItems) {
            System.out.println("Product Not Available\n");
        }
        else if (products[i].name.equalsIgnoreCase(name)) {
            System.out.println("Enter the new product price: ");
            int price = s.nextInt();
            products[i].price=price;
            break;
        }
    }
}

public void mainMenu() {

```

```

        while(true) {
            System.out.println("Main Menu : Choose one.\n"
                + "1. Amdin.\n"
                + "2. Staff.\n"
                + "3. Exit.");
            String c = s.next();

            while(true) {
                if(c.equals("1")) {
                    loginPage();
                }else if(c.equals("2")) {
                    staff s1 = new staff();
                    s1.loginPage(1);
                }else if(c.equals("3")) {
                    System.out.println("Program Exited.");
                    System.exit(0);
                }else {
                    System.out.println("Invalid Choice.");
                    break;
                }
            }
        }
    }
}

```

```

package People;
import java.util.Scanner;
public class staff extends admin {
    Scanner s = new Scanner(System.in);

    private String username="Staff";
    private String pass="password";

    public void loginPage(int a) {
        while(true) {
            System.out.println("Enter your Staff username : ");
            String u = s.next();
            System.out.println("Enter your password : ");
            String p = s.next();
            menu(u,p);
        }
    }

    private void menu(String u,String p) {
        if(u.equalsIgnoreCase(username) && p.equals(pass)) {
            while(true) {
                System.out.println("Staff Menu : Choose one.\n"
                    + "1. Billing Menu\n"
                    + "2. View Inventory\n"

```

```

        + "3. Exit to Main Menu.");
        int c = s.nextInt();
        if(c==1) {
            billingMenu();
        }else if(c==2) {
            viewProducts1();
        }
        else if(c==3) {
            mainMenu();
        }else {
            System.out.println("Invalid Choice.");
        }
    }
}
}
}

private void billingMenu() {
    while(true) {
        System.out.println("Enter item name or type "+"\"fin\""+
to exit to Staff Menu : ");
        String itemName = s.next();
        for(int i=0;i<totalItems;i++) {
            if (itemName.equalsIgnoreCase(products[i].name)) {
                System.out.println("Enter item quantity: ");
                int itemQuantity = s.nextInt();
                if(itemQuantity>products[i].quantity) {
                    System.out.println("Not Enough Inventory.");
                }else {
                    products[i].quantity-=itemQuantity;
                    System.out.println("Total bill :
"+bill(itemQuantity,products[i].price)+"\n");
                }
            }else if(itemName.equalsIgnoreCase("fin")) {
                menu(username,pass);
            }
        }
    }
}

private double bill(int itemQ,double itemP) {
    double bill = 0;
    bill=itemQ*itemP;
    System.out.println("Bill : "+bill);
    System.out.println("\nTax : "+bill*.15+"\n");
    bill+=(bill*.15);
    return bill;
}
}
}

```



```
package People;
public interface login {
    public void loginPage();
}
```

```
package Product;
public class product {
    public String name;
    public int quantity;
    public double price;

    public product(String name, int quantity, double price) {
this.name = name;
this.quantity = quantity;
this.price = price;
}
}
```

```
package Product;
public class inventory {
    public static product[] products = new product[100];
    public static int totalItems = 0;

    public void setInventory() {
products[totalItems] = new product("eggs",100,12);
totalItems++;
        products[totalItems] = new product("soap",60,22.50);
totalItems++;
        products[totalItems] = new product("shampoo",30,80);
totalItems++;
        products[totalItems] = new product("chips",50,15);
totalItems++;
        products[totalItems] = new product("milk",12,90);
totalItems++;
    }

    public void viewProducts() {
        for(int i=0;i<totalItems;i++) {
            System.out.println("\nName of the item:
"+products[i].name+"\n\nQuantity: "+ products[i].quantity +"\n\nPrice: "+
products[i].price+"\n");
        }
    }

    public void viewProducts1() {
        System.out.println("Product Name | Quantity | Retail Price\n");
    }
}
```

```
for(int i=0;i<totalItems;i++) {  
    System.out.println(products[i].name+" | "+  
products[i].quantity+" | "+ products[i].price+"\n");  
}  
}
```

Explanation of the code:

1. Packages:

- The code is divided between two packages: “People” and “Product”.
- “People” package contains classes related to users (Admin and Staff).
- “Product” package contains classes related to product and inventory management.

2. Product package:

- The product package contains three classes and one interface. They are **mainPage** which contains the main() method, **admin**, **staff** and one interface named **login** which is implemented in both admin and staff class.
- The **admin** class represents the administrator role. It has methods to manage the inventory, such as adding products, updating individual product quantities and updating prices. The loginPage() method allows the admin to log in, and the loginMenu() method provides various management options. The admin can navigate between the main menu and the admin-specific actions.
- The **staff** class represents the staff role, which has limited privileges compared to the **admin**. The loginPage(int a) method allows the staff to log in. The menu(String u, String p) method provides the staff with options to perform billing or view inventory. The staff can navigate between the staff menu and the main menu.
- The **mainPage** class contains the main method main() which initiates the program. It also has a mainMenu() method that can show the main menu options to choose between Admin, Staff and Exit.

- The interface **login** has a loginPage() method that is implemented in admin and staff classes.

3. Product package:

- The class **product** defines attributes in an individual product, such as price, quantity and its name.
- The **inventory** class is the used for managing the inventory. It has an array named products, which handle an array of product objects. This array is used to store product data and represent the information of every product.
- The **inventory** class also has two methods named viewProducts() and viewProducts1(), which can display all the products available in different visual styles.
- A method setInventory() is also used in the **inventory** class to occupy the inventory with some initial products.
- The **inventory** class is the parent class of both **admin** and **staff** class as both classes inherits the methods and attributes to display the inventory. The **admin** class gets extra functionality to edit items within the inventory.

The program uses few pillars of Object Oriented Programming such as inheritance, encapsulation and polymorphism. It also divides up features into separate objects and classes which make the program easy to maintain and expand upon.