|  |  |  |
| --- | --- | --- |
|  | Name | Shubham Goel |
|  | Roll No | 2019130015 |
| **OOP Lab** | Batch | A |
| Exp 9: Collections | Date | 18 October 2021 |
|  | Branch | COMPS |

**Aim:** Implement Collections

**Theory**

Collection

The **Collection in Java** is a framework that provides an architecture to store and manipulate the group of objects.

Java Collections can achieve all the operations that you perform on a data such as searching, sorting, insertion, manipulation, and deletion.

Java Collection means a single unit of objects. Java Collection framework provides many interfaces (Set, List, Queue, Deque) and classes ([ArrayList](https://www.javatpoint.com/java-arraylist), Vector, [LinkedList](https://www.javatpoint.com/java-linkedlist), [PriorityQueue](https://www.javatpoint.com/java-priorityqueue), HashSet, LinkedHashSet, TreeSet).

A Collection represents a single unit of objects, i.e., a group.

Framework in Java

* It provides readymade architecture.
* It represents a set of classes and interfaces.
* It is optional.



**Code:**

import java.util.\*;

import java.util.ArrayList;

import java.util.Collection;

import java.util.HashMap;

import java.util.List;

import java.util.Scanner;

//custom exception

class InvalidPrice extends Exception {

    public InvalidPrice(String str) {

        super(str);

    }

}

// Base class

class User {

    String name;

    int age;

    void printUserStamp() {

        System.out.println("");

        System.out.println("Inside the user class");

        System.out.println("");

    }

    User() {

        this.name = "a\_user";

        this.age = 100;

    }

    void printUserInfo() {

    };

    void inputNameAge() {

        // printUserStamp();

        Scanner scanner\_obj = new Scanner(System.in); // Create a Scanner object

        System.out.println("Enter the name of the user");

        this.name = scanner\_obj.nextLine();

        System.out.println("Enter the age");

        this.age = scanner\_obj.nextInt();

    }

    void printUserDetails() {

        System.out.println(this.name);

        System.out.println(this.age);

    }

}

class Cart {

    // Product cartArr[] = new Product[100];

    Collection<Product> cartArr = new ArrayList<Product>();

    int firstFreeIndex;

    void printCartDetails() {

        for (Product i : cartArr) {

            System.out.println("Product in cart " + i.name);

        }

    }

    void addProductToCart(Product productToAdd) {

        // this.cartArr[this.firstFreeIndex++] = productToAdd;

        this.cartArr.add(productToAdd);

    }

    void buyProductsInCart() {

        // this.cartArr[this.firstFreeIndex++] = productToAdd;

        for (Product i : cartArr) {

            System.out.println("User bought " + i.name);

        }

    }

}

class Buyer extends User {

    String address;

    String creditCardID;

    Cart cart;

    Buyer(String name) {

        this.cart = new Cart();

        this.name = name;

    }

    void printBuyerInfo() {

        printUserDetails();

        System.out.println(this.address);

        System.out.println(this.creditCardID);

    }

    void printBuyerStamp() {

        System.out.println("");

        System.out.println("Inside the buyer class");

        System.out.println("");

    }

    void inputBuyerDetails() {

        // printBuyerStamp();

        inputNameAge();

        Scanner scan\_obj = new Scanner(System.in); // Create a Scanner object

        System.out.println("Enter address for the user");

        this.address = scan\_obj.nextLine();

        System.out.println("Enter the credit card id");

        this.creditCardID = scan\_obj.nextLine();

        // scan\_obj.close();

    }

}

// composition

class Product {

    String name;

    int price;

    int quantity;

    // static Product allProducts[] = new Product[400];

    // collection has no index

    static List<Product> allProducts = new ArrayList<Product>();

    static int lastProductLoc = 0;

    static void addToAllProducts(Product toAdd) {

        // allProducts[lastProductLoc++] = toAdd;

        allProducts.add(toAdd);

    }

    Product(String name, int price) {

        this.name = name;

        this.price = price;

        this.quantity = 10;

        addToAllProducts(this);

    }

    static void printAllProducts() {

        int i = 0;

        for (Product cur\_product : allProducts) {

            System.out.println("Index = " + i);

            System.out.println("price = " + cur\_product.price);

            System.out.println("name = " + cur\_product.name);

            System.out.println();

            i++;

        }

    }

    void printProductDetails() {

        System.out.println("price = " + this.price);

        System.out.println("name = " + this.name);

        System.out.println();

    }

}

public class Main {

    // validate the price

    static void validatePrice(int price) throws InvalidPrice {

        if (price < 0) {

            throw new InvalidPrice("The price should be positive");

        } else {

            System.out.println("The price is valid");

        }

    }

    public static void main(String args[]) {

        Buyer b = new Buyer("second");

        Product pa = new Product("One plus X", 29999);

        Product pb = new Product("Echo dot", 4999);

        Product pc = new Product("BOAT smart watch", 2999);

        Product pd = new Product("Mi home security camera", 2999);

        Product pe = new Product("Fire stick", 3999);

        System.out.println("All the products are");

        Product.printAllProducts();

        // collection of integer and string

        Collection<Object> allProductOrdered = new ArrayList<>();

        System.out.println("Enter products u want");

        System.out.println("Enter -1 to exit");

        for (int i = 0; i < 100; i++) {

            Scanner scanner\_obj = new Scanner(System.in); // Create a Scanner object

            int cart\_loc = scanner\_obj.nextInt();

            if (cart\_loc == -1) {

                break;

            }

            Product productOrderedByUser = Product.allProducts.get(cart\_loc);

            b.cart.addProductToCart(productOrderedByUser); // use list index

            System.out.println("Product " + productOrderedByUser.name + " added to cart");

            // string and int added into the same ArrayList

            allProductOrdered.add(productOrderedByUser.name);

            allProductOrdered.add(productOrderedByUser.price);

        }

        int total = 0;

        boolean isName = true;

        System.out.println("Total Products bought on the website");

        for (Object cur\_order : allProductOrdered) {

            if (isName) {

                System.out.println("The product name:" + cur\_order);

            } else {

                int price = (int) cur\_order; // cast to the integer

                System.out.println("The price:" + price);

                total += price;

            }

            isName = !isName;

        }

        System.out.println("The money spend on the website " + total);

        // b.cart.printCartDetails();

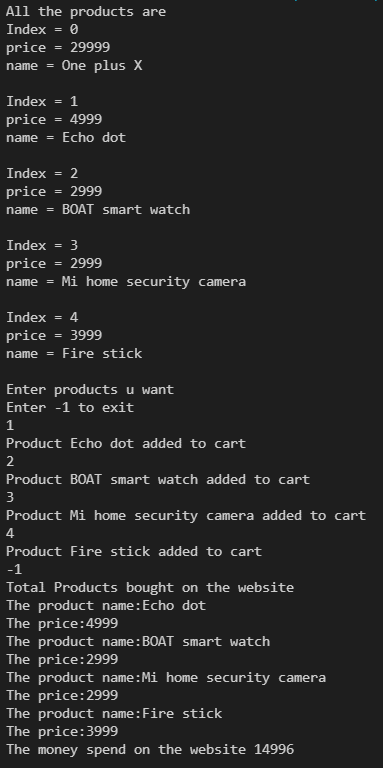
        // b.cart.buyProductsInCart();

        System.out.println();

    }

}

**Output:**



**Conclusion**

I have learnt and implemented collections in JAVA.