	Name	Shubham Goel
OOP Lab Exp: Generics	Roll No	2019130015
	Batch	A
	Date	22 Nov. 21
	Branch	COMPS

Aim: Implement Generics

Theory

Generics are a facility of generic programming that were added to the Java programming language in 2004. They were designed to extend Java's type system to allow "a type or method to operate on objects of various types while providing compile-time type safety". The aspect compile-time type safety was not fully achieved, since it was shown in 2016 that it is not guaranteed in all cases.

The Java collections framework supports generics to specify the type of objects stored in a collection instance.

Code:

Main.java

```java
import java.util.*;
import java.lang.*;

//custom exception
class InvalidPrice extends Exception {
    public InvalidPrice(String str) {
        super(str);
    }
}

// Base class
class User {
    String name;
    int age;

    void printUserStamp() {
```

```java
        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) {
```

```java
        // 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");
```

```java
            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++;
        }
```

```java
    }

    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("p1", 200);
        Product pb = new Product("p2", 200);
        Product pc = new Product("p3", 200);
        Product pd = new Product("p4", 200);
        Product pe = new Product("p5", 200);

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

        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;
            }
            b.cart.addProductToCart(Product.allProducts.get(cart_loc
)); // use list index
```

```
            System.out.println("Product " +
Product.allProducts.get(cart_loc).name + " added to cart");
        }

        b.cart.printCartDetails();
        b.cart.buyProductsInCart();
        System.out.println();


    }
}
```

Output:

```
All the products are
Index = 0
price = 200
name = p1

Index = 1
price = 200
name = p2

Index = 2
price = 200
name = p3

Index = 3
price = 200
name = p4

Index = 4
price = 200
name = p5
```

```
Enter products u want
Enter -1 to exit
1
Product p2 added to cart
3
Product p4 added to cart
2
Product p3 added to cart
1
Product p2 added to cart
-1
Product in cart p2
Product in cart p4
Product in cart p3
Product in cart p2
User bought p2
User bought p4
User bought p3
User bought p2
```

Conclusion

I have learnt and implemented Generics.

References:

https://en.wikipedia.org/wiki/Generics_in_Java