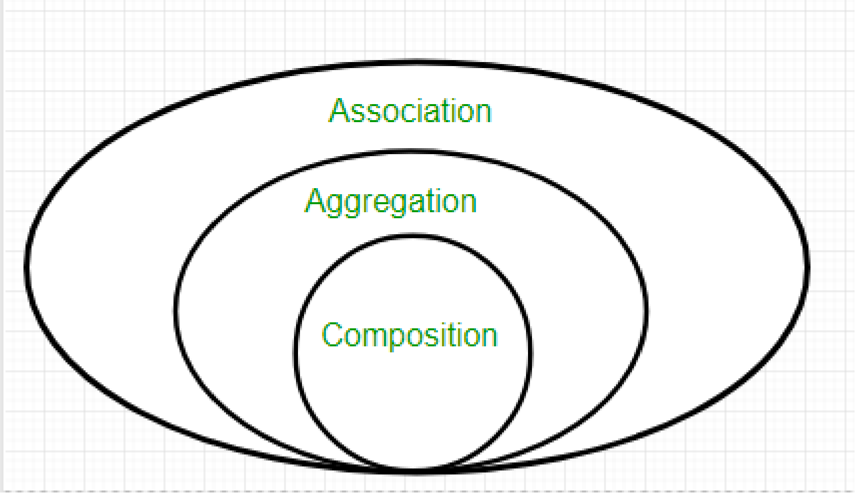
OOP Lab Exp 5: Implement Association	Name	Shubham Goel
	Roll No	2019130015
	Batch	A
	Date	18 October 2021
	Branch	COMPS

Aim: Implement Association

Theory

Association:

Association is a relation between two separate classes which establishes through their Objects. The object communicates to another object to use functionality and services provided by that object. Composition and Aggregation are the two forms of association.



Composition:

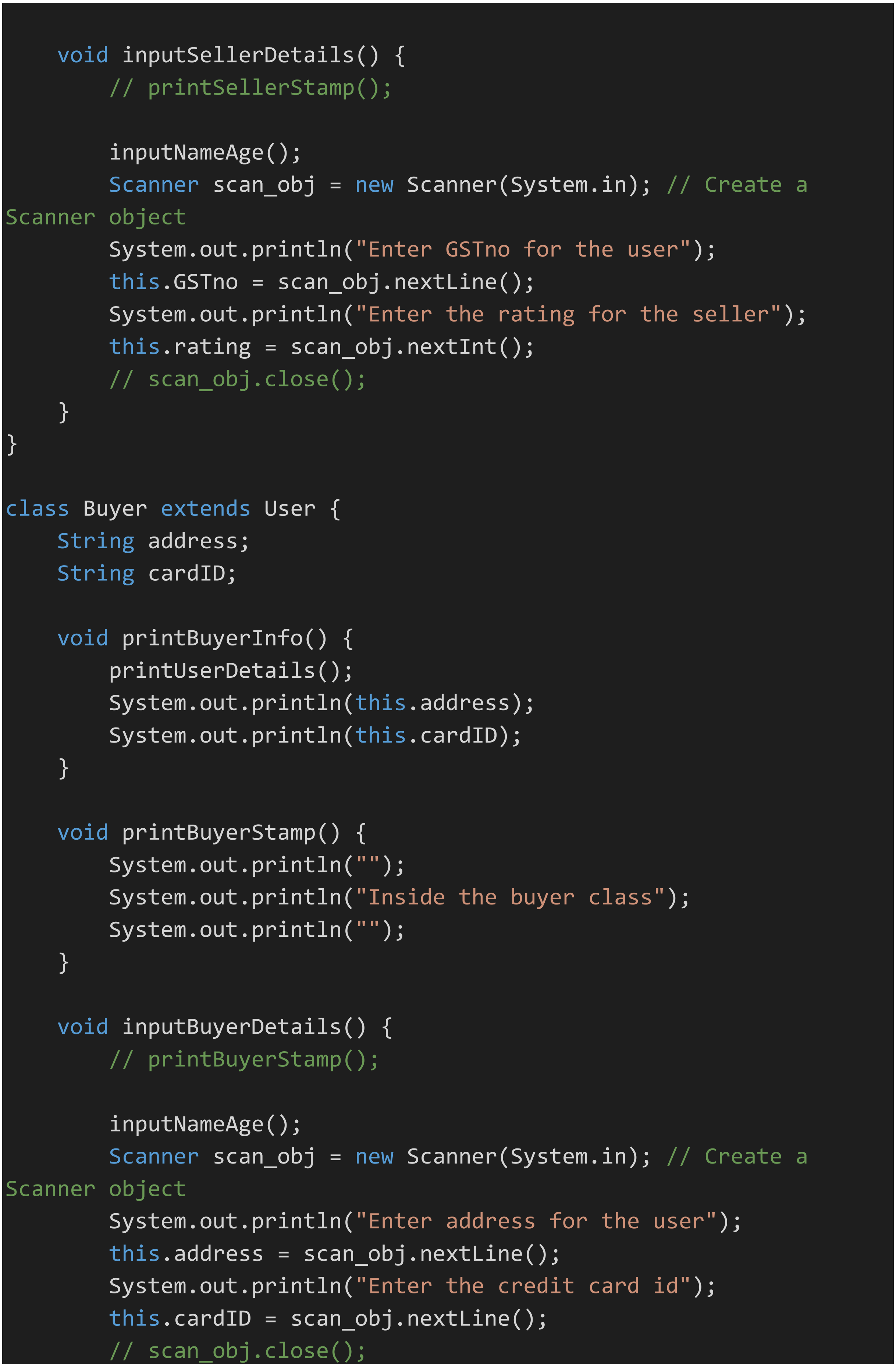
- It represents **part-of** relationship.
- In composition, both entities are dependent on each other.
- When there is a composition between two entities, the composed object **cannot exist** without the other entity.

Aggregation

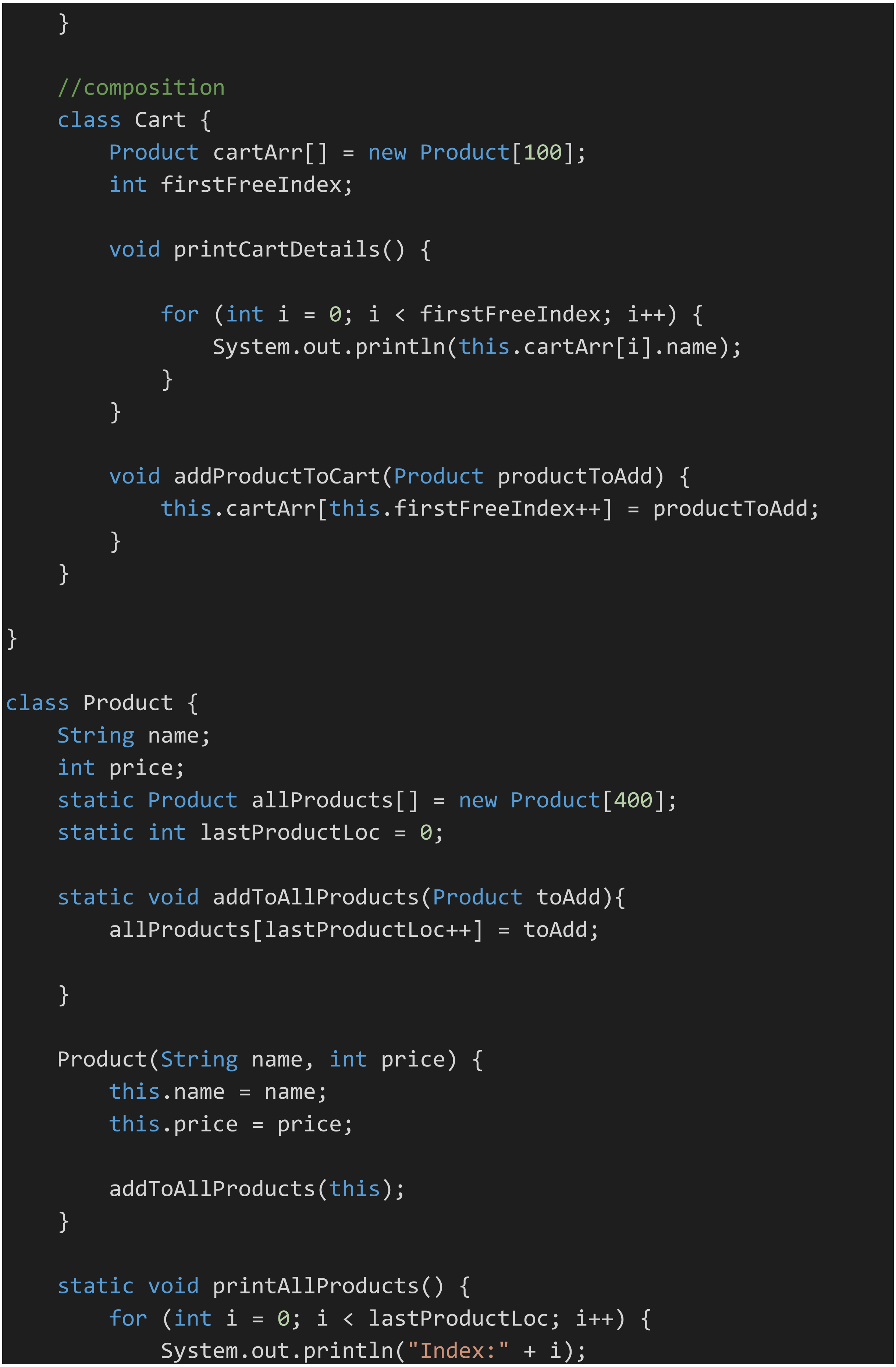
- It represents Has-A’s relationship.
- It is a **unidirectional association** i.e. a one-way relationship. For example, a department can have students but vice versa is not possible and thus unidirectional in nature.

Code:

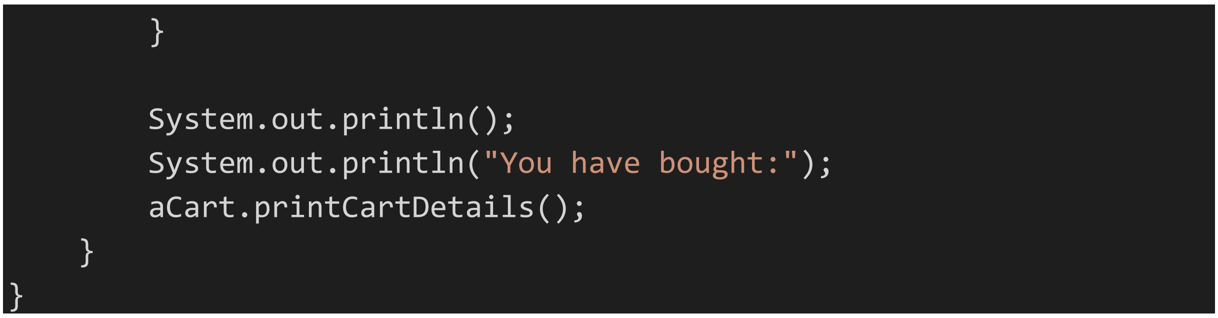
```java
import java.util.*;


// 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 Seller extends User{
    String GSTno;
    int rating;
    Product sellerProducts[] = new Product[100]; //aggregation
    int lastFreeIndex = 0;


    Seller() {
        this.GSTno = "123";
        this.rating = 3;
    }

    void addProduct(String name, int price) {
        Product addProduct = new Product(name, price);
        this.sellerProducts[this.lastFreeIndex++] = addProduct;
    }

    void printSellerProducts() {
        for (int i = 0; i < this.lastFreeIndex; i++) {
            System.out.println("Index:" + i);
            System.out.println("price = " +
this.sellerProducts[i].price);
            System.out.println("name = " +
this.sellerProducts[i].name);
            System.out.println();
        }
    }

    void printSellerInfo() {
        printUserDetails();
        System.out.println(this.GSTno);
        System.out.println(this.rating);
    }

    void printSellerStamp() {
        System.out.println("");
        System.out.println("Inside the Seller class");
        System.out.println("");
    }
```



```
    void inputSellerDetails() {
        // printSellerStamp();

        inputNameAge();
        Scanner scan_obj = new Scanner(System.in); // Create a
Scanner object
        System.out.println("Enter GSTno for the user");
        this.GSTno = scan_obj.nextLine();
        System.out.println("Enter the rating for the seller");
        this.rating = scan_obj.nextInt();
        // scan_obj.close();
    }
}

class Buyer extends User {
    String address;
    String cardID;

    void printBuyerInfo() {
        printUserDetails();
        System.out.println(this.address);
        System.out.println(this.cardID);
    }

    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.cardID = scan_obj.nextLine();
        // scan_obj.close();
```

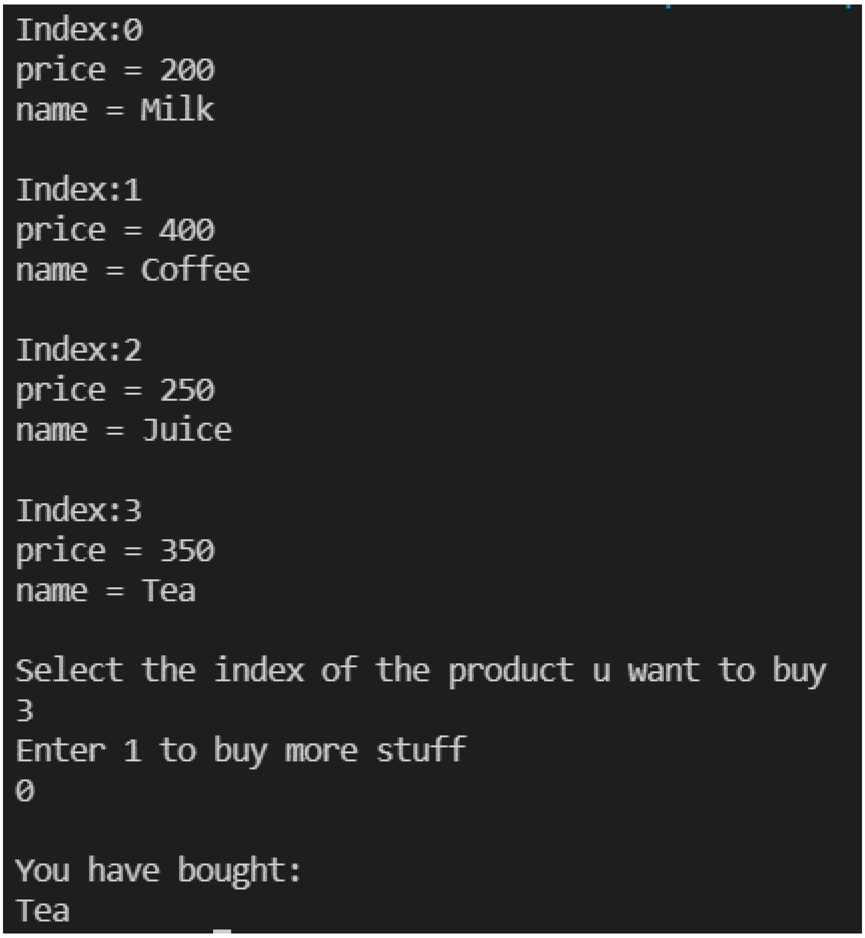


```
    }

    //composition
    class Cart {
        Product cartArr[] = new Product[100];
        int firstFreeIndex;

        void printCartDetails() {

            for (int i = 0; i < firstFreeIndex; i++) {
                System.out.println(this.cartArr[i].name);
            }
        }

        void addProductToCart(Product productToAdd) {
            this.cartArr[this.firstFreeIndex++] = productToAdd;
        }
    }

}

class Product {
    String name;
    int price;
    static Product allProducts[] = new Product[400];
    static int lastProductLoc = 0;

    static void addToAllProducts(Product toAdd){
        allProducts[lastProductLoc++] = toAdd;

    }

    Product(String name, int price) {
        this.name = name;
        this.price = price;

        addToAllProducts(this);
    }

    static void printAllProducts() {
        for (int i = 0; i < lastProductLoc; i++) {
            System.out.println("Index:" + i);
            System.out.println("price = " + allProducts[i].price);
            System.out.println("name = " + allProducts[i].name);
            System.out.println();
        }
    }

}



public class Main {
    public static void main(String args[]){
        Scanner scan_obj = new Scanner(System.in); // Create a
Scanner object


        Buyer aBuyer = new Buyer();
        // aBuyer.inputBuyerDetails();

        Buyer.Cart aCart = aBuyer.new Cart();

        Seller aSeller = new Seller();
        // aSeller.inputSellerDetails();

        aSeller.addProduct("Milk", 200);
        aSeller.addProduct("Coffee", 400);
        aSeller.addProduct("Juice", 250);
        aSeller.addProduct("Tea", 350);

        // Product a = new Product("name", 20);

        int buy_more = 1;
        while (buy_more == 1) {
            Product.printAllProducts();
            System.out.println("Select the index of the product u
want to buy");
            int product_loc = scan_obj.nextInt();

            aCart.addProductToCart(Product.allProducts[product_loc])
;
            System.out.println("Enter 1 to buy more stuff");
            buy_more = scan_obj.nextInt();
```



```
        }

        System.out.println();
        System.out.println("You have bought:");
        aCart.printCartDetails();
    }
}
```

Output:



```
Index:0
price = 200
name = Milk

Index:1
price = 400
name = Coffee

Index:2
price = 250
name = Juice

Index:3
price = 350
name = Tea

Select the index of the product u want to buy
3
Enter 1 to buy more stuff
0

You have bought:
Tea
```

Conclusion

I have learnt and implemented Association properties. I have also understood the difference between aggregation and composition.

References:

https://www.geeksforgeeks.org/association-composition-aggregation-java/