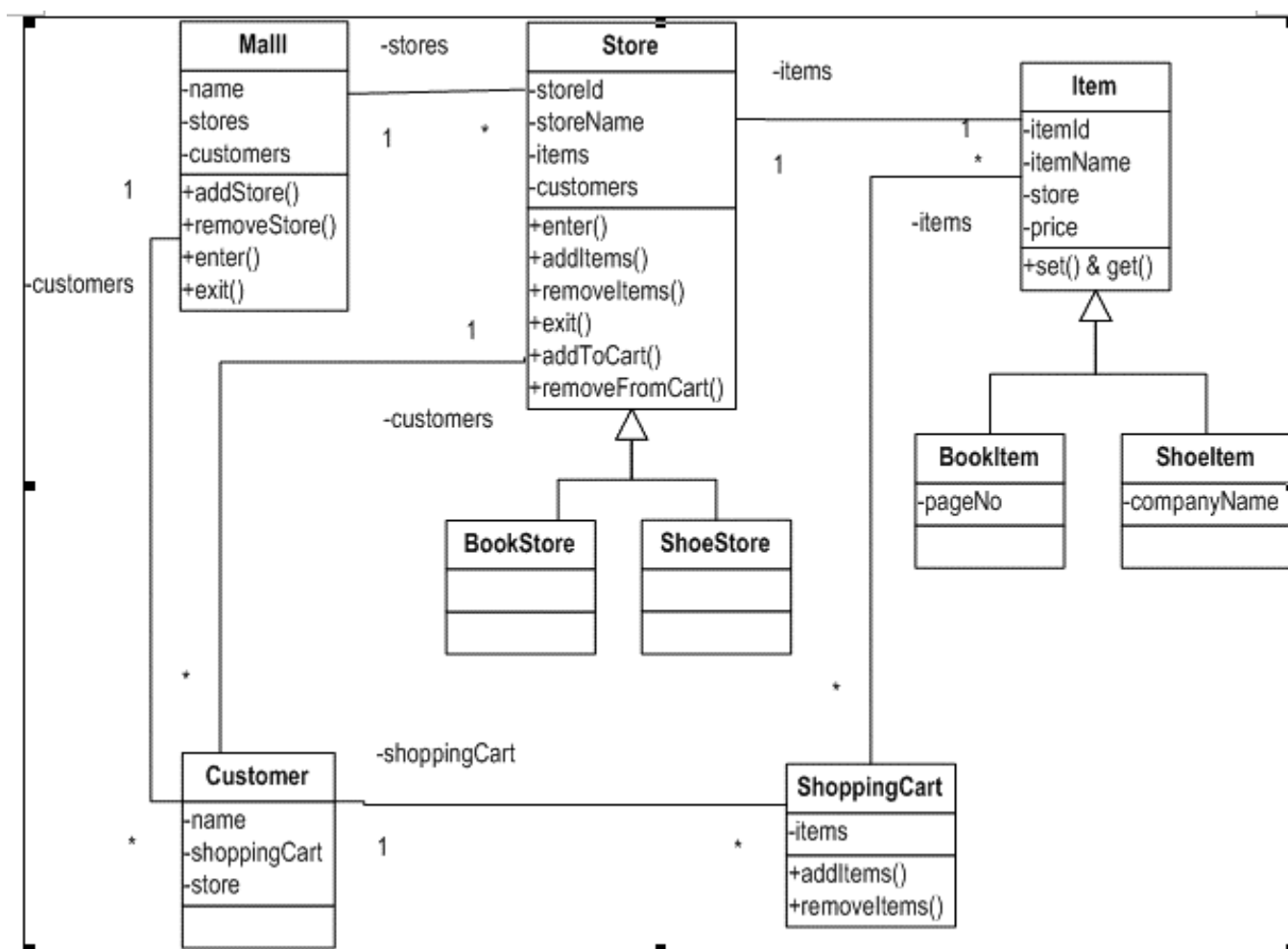


# Shopping Mall Project

## Project Description

In this project, you will have to implement a simple shopping mall–application for customers in Java.

The mall consists of stores and the customer enters the mall and can enter any store to buy items in the store and he pays for the items bought from stores in the mall before he exits the mall.



Here is a description of each class which you need to implement the project:

### ❖ class Mall :

The class mall performs the creation of a set of different stores, such as a book store, an shoe store, etc. The mall allows customers to enter stores for shopping.

#### • Data members :

- private String name (the name of the Mall )
- private List stores (list of stores in the Mall )
- private List customers(list of all customers in Mall )

- **Constructor :**
  - public Mall (String name);
- **Methods :**
  - public void addStore(Store s);
  - public void removeStore(Store s);
  - public void enter (Customer c);  
The customer c enters the Mall (not in the store )and gets an empty ShoppingCart
  - public void exit(Customer c)
    - The Customer should pay for the items in his ShoppingCart
    - The customer c exits the Mall after exiting the store if he is in any.

### ❖ **Class Store :**

the class store is abstract . It contains items for sale

- **Data members :**
  - private int storeId
  - private String storeName (the name of the store)
  - private List Items (the Items in the store)
  - private List customers (Customers in the store)
- **Constructor :**
  - public Store (int storeId, String name);
- **Methods :**
  - public void Enter(Customer c);  
The customer enters the store .He must be first in the Mall , not in an other store and not already in this store.
  - public void addItem(Item i )  
Adding the item in the store (in the list items in the store)
  - public void removeItem(Item i )  
Removing the item from the store (from the list items in the store)
  - public void Exit(Customer c );  
The customer exits the store if he is in this store and stays in the Mall.
  - public void addTotheCart(Customer c, Item i);  
The item i is added to the ShoppingCart of the customer c
  - public void removeFromtheCart(Customer c, Item i);  
The item i is removed from the ShoppingCart of the customer c

### ❖ **BookStore and ShoeStore:**

The class BookStore and the class ShoeStores extend the class Store . Each of them provides a constructor with parameter and they override the method addToCart .

### ❖ **Class Item :**

The class Items is abstract .

- **Data members :**
  - private int itemId
  - private String itemName (The name of the item)
  - private Store store ( the name of the store in which the item must be stored);
  - private float price;
- **Constructor:**
  - public Item(int itemId, Store store , String itemName , float price);
- **Methods:**
  - public set and get methods.

### ❖ **Class BookItem :**

the class BookItem inherits from Item

- **Data members :**

- int pageNo (the number of pages of the book) and including the data members of item

- **Constructor :**

- public BookItem(int itemId, BookStore store ,String itemName,float price, int pageNo);

- **Methods :**

- public set and get methods

### ❖ **Class ShoeItem:**

the class ShoeItem inherits from Item

- **Data members :**

- String companyName (the name of Company which has produced the shoe )  
and including the data members of item

- **Constructor :**

- public ShoeItem(int itemId, ShoeStore store , String itemName, float price, String companyName);

- **Methods :**

- public set and get methods

### ❖ **Class ShoppingCart:**

The cart which is given to the Customer when he enters the Mall

- **Data members :**

- List items ( list of items (shoes and books) which the customer buys in the Mall)

- **Constructor :**

- public ShoppingCart(); Creating an empty shopping cart

- **Methods :**

- public void addItem(Item i)

- adding the Item i in the shoppingCart (in the items of the shopping cart)

- public void removeItem(Item i)

- removing the Item i from the shoppingCart (from the items of the shopping cart)

### ❖ **Class Customer:**

- **Data members :**

- private String name (the name of Customer)

- private ShoppingCart shoppingCart (the shoppingCart of the customer)

- private Store store (the store in which the customer exists)

- **Constructor :**

- public Customer(String name);

- **Methods :**

- public get and set methods

### ❖ **Class MallTesting:**

The class MallTesting for testing your shopping mall and could be like the following:

```
public class TestingMall {  
    public static void main(String args[]){  
        Mall testingMall = new Mall("MY Mall");  
        BookStore bookStore1 = new BookStore(1,"BOOKSTORE1");  
        BookStore bookStore2 = new BookStore(2,"BOOKSTORE2");
```

```

ShoeStore shoeStore1 = new ShoeStore(3,"ShoeStore1");
ShoeStore shoeStore2 = new ShoeStore(4,"ShoeStore2");
BookItem book1 = new BookItem(1,bookStore1,"BOOK1",20.5f,200);
BookItem book2 = new BookItem(2,bookStore2,"BOOK2",50.8f,300);
BookItem book3 = new BookItem(3,bookStore2,"BOOK3",70.0f,500);
ShoeItem shoe1 = new ShoeItem(1,shoeStore1,"SHOE1",90.5f,"COMPANY1");
ShoeItem shoe2 = new ShoeItem(2,shoeStore2,"SHOE2",50.0f,"COMPANY2");
ShoeItem shoe3 = new ShoeItem(3,shoeStore2,"SHOE3",120.5f,"COMPANY3");
Customer customer1 = new Customer("Ahmed");
Customer customer2 = new Customer("Samir");
testingMall.addStore(bookStore1);
testingMall.addStore(bookStore2);
testingMall.addStore(shoeStore1);
testingMall.addStore(shoeStore2);
testingMall.enter(customer1);
bookStore1.enter(customer1);
bookStore1.addToCart(customer1,book1);
bookStore1.exit(customer1);
bookStore2.enter(customer1);
bookStore2.addToCart(customer1,book2);
bookStore2.addToCart(customer1,book3);
bookStore2.exit(customer1);
shoeStore1.enter(customer1);
shoeStore1.addToCart(customer1,shoe1);
shoeStore1.exit(customer1);
testingMall.exit(customer1); }}

```

### **Important Notes:**



**Submit in CD only** – CD includes your name, ID and Number Section in a Label form.



**You must submit the following :**

All source files + All class files.



**If a student copies** the Project of another student, he/she will be **assigned a zero grade** for the Project.



**The resulting implementation** will be judged according a brief **one-to-one discussion** with the student. **Maximum group project 2 Students**



**The date of Handover and discussion in same time.**

Group	Section	Discussion	Time
Female	طالبات - الاثنين	12/ 01/2008	11 - 14
Male	طلاب يوم الاثنين والثلاثاء	14 / 01/2008	11 - 14
Male	جميع طلاب يوم الخميس	16 / 01 /2008	11 - 14
Female	طالبات - الأربعاء	19 / 01 /2008	11 - 14



**Web Site:** [www.alazhar.edu.ps/staff/omar](http://www.alazhar.edu.ps/staff/omar) → نشاط

““ *Have Fun !* ““