JAVA FUNDAMENTALS SECTION-07 PART-2

M.Poli reddy

192325056

Steps for Customizing the Inventory Software

1. Create the DVD and CD Subclasses

- a. Create the DVD subclass with additional fields for movie length, age rating, and film studio.
- b. Implement a constructor in DVD that initializes all fields, calling super to pass values to the Product constructor.
- c. Create getters and setters for the DVD fields.
- d. Similarly, create the CD subclass with fields for artist, number of songs, and record label.
- e. Implement a constructor in CD that initializes all fields, calling super to pass values to the Product constructor.
- f. Create getters and setters for the CD fields.

2. Override Methods in DVD and CD

- a. Override the getInventoryValue method in DVD to include a 5% restocking fee.
- b. Override the toString method in both DVD and CD to display subclass-specific information.

3. Modify the ProductTester Class

- a. Create a method addCDToInventory to handle adding CDs, including prompts for CD-specific fields and array indexing.
- b. Update the existing addToInventory method or create a new addDVDToInventory method for DVDs.
- c. Modify the addInventory method to include a menu for choosing between CD and DVD, and call the appropriate add method based on user input.

- d. Ensure the addToInventory method in Product stops adding stock for discontinued products.

FULL JAVA CODE

```
import java.util.Scanner;


// Base Product Class
class Product {
    private String name;
    private double price;
    private int quantityInStock;
    private int itemNumber;
    private boolean isDiscontinued;


    public Product(String name, double price, int quantityInStock, int itemNumber, boolean isDiscontinued) {
        this.name = name;
        this.price = price;
        this.quantityInStock = quantityInStock;
        this.itemNumber = itemNumber;
        this.isDiscontinued = isDiscontinued;
    }


    public double calculateStockValue() {
        return price * quantityInStock;
    }
```

```
    public void setDiscontinued(boolean discontinued) {
        this.isDiscontinued = discontinued;
    }

    public boolean isDiscontinued() {
        return isDiscontinued;
    }

    @Override
    public String toString() {
        return "Item Number: " + itemNumber + "\n" +
                "Name: " + name + "\n" +
                "Quantity in stock: " + quantityInStock + "\n" +
                "Price: " + price + "\n" +
                "Stock Value: " + calculateStockValue() + "\n" +
                "Product Status: " + (isDiscontinued ? "Discontinued" :
"Active") + "\n";
    }

    // Getters and Setters
    public String getName() { return name; }
    public void setName(String name) { this.name = name; }
    public double getPrice() { return price; }
    public void setPrice(double price) { this.price = price; }
```

```
    public int getQuantityInStock() { return quantityInStock; }
    public void setQuantityInStock(int quantityInStock) {
this.quantityInStock = quantityInStock; }
    public int getItemNumber() { return itemNumber; }
    public void setItemNumber(int itemNumber) { this.itemNumber =
itemNumber; }
}

// DVD Subclass
class DVD extends Product {
    private int length;
    private int ageRating;
    private String filmStudio;

    public DVD(String name, double price, int quantityInStock, int
itemNumber, boolean isDiscontinued,
            int length, int ageRating, String filmStudio) {
        super(name, price, quantityInStock, itemNumber,
isDiscontinued);
        this.length = length;
        this.ageRating = ageRating;
        this.filmStudio = filmStudio;
    }

    @Override
    public double calculateStockValue() {
```

```
        return super.calculateStockValue() * 1.05; // 5% restocking fee
    }

    @Override
    public String toString() {
        return super.toString() +
                "Movie Length: " + length + " minutes\n" +
                "Age Rating: " + ageRating + "\n" +
                "Film Studio: " + filmStudio + "\n";
    }

    // Getters and Setters
    public int getLength() { return length; }
    public void setLength(int length) { this.length = length; }
    public int getAgeRating() { return ageRating; }
    public void setAgeRating(int ageRating) { this.ageRating =
ageRating; }
    public String getFilmStudio() { return filmStudio; }
    public void setFilmStudio(String filmStudio) { this.filmStudio =
filmStudio; }
}

// CD Subclass
class CD extends Product {
    private String artist;
```

```
    private int numberOfSongs;
    private String label;

    public CD(String name, double price, int quantityInStock, int
itemNumber, boolean isDiscontinued,
            String artist, int numberOfSongs, String label) {
        super(name, price, quantityInStock, itemNumber,
isDiscontinued);
        this.artist = artist;
        this.numberOfSongs = numberOfSongs;
        this.label = label;
    }

    @Override
    public String toString() {
        return super.toString() +
                "Artist: " + artist + "\n" +
                "Songs on Album: " + numberOfSongs + "\n" +
                "Record Label: " + label + "\n";
    }

    // Getters and Setters
    public String getArtist() { return artist; }
    public void setArtist(String artist) { this.artist = artist; }
    public int getNumberOfSongs() { return numberOfSongs; }
```

```
    public void setNumberOfSongs(int numberOfSongs) {
this.numberOfSongs = numberOfSongs; }
    public String getLabel() { return label; }
    public void setLabel(String label) { this.label = label; }
}

// ProductTester Class
public class ProductTester {
    private static Product[] products = new Product[100];
    private static int productIndex = 0;

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        while (true) {
            System.out.println("1: Add CD");
            System.out.println("2: Add DVD");
            System.out.println("3: Exit");
            System.out.print("Please enter the product type: ");
            int stockChoice = scanner.nextInt();
            scanner.nextLine();  // Clear the buffer

            if (stockChoice == 1) {
                if (productIndex >= products.length) {
```

```
                System.out.println("Inventory full, cannot add more
products.");
                break;
            }
            addCDToInventory(productIndex, scanner);
        } else if (stockChoice == 2) {
            if (productIndex >= products.length) {
                System.out.println("Inventory full, cannot add more
products.");
                break;
            }
            addDVDToInventory(productIndex, scanner);
        } else if (stockChoice == 3) {
            break;
        } else {
            System.out.println("Only numbers 1 or 2 allowed!");
            continue;
        }

        productIndex++;
    }

    // Display all products
    for (Product p : products) {
        if (p != null) {
```

```
                System.out.println(p);
            }
        }

        scanner.close();
    }

    private static void addCDToInventory(int i, Scanner scanner) {
        System.out.print("Please enter the CD name: ");
        String name = scanner.nextLine();
        System.out.print("Please enter the artist name: ");
        String artist = scanner.nextLine();
        System.out.print("Please enter the record label name: ");
        String label = scanner.nextLine();
        System.out.print("Please enter the number of songs: ");
        int numberOfSongs = scanner.nextInt();
        System.out.print("Please enter the quantity of stock for this
product: ");
        int quantity = scanner.nextInt();
        System.out.print("Please enter the price for this product: ");
        double price = scanner.nextDouble();
        System.out.print("Please enter the item number: ");
        int itemNumber = scanner.nextInt();
        System.out.print("Is this product discontinued? (true/false): ");
        boolean isDiscontinued = scanner.nextBoolean();
```

```
        scanner.nextLine();  // Clear the buffer

        products[i] = new CD(name, price, quantity, itemNumber,
isDiscontinued, artist, numberOfSongs, label);
    }

    private static void addDVDToInventory(int i, Scanner scanner) {
        System.out.print("Please enter the DVD name: ");
        String name = scanner.nextLine();
        System.out.print("Please enter the film studio name: ");
        String filmStudio = scanner.nextLine();
        System.out.print("Please enter the age rating: ");
        int ageRating = scanner.nextInt();
        System.out.print("Please enter the length in minutes: ");
        int length = scanner.nextInt();
        System.out.print("Please enter the quantity of stock for this
product: ");
        int quantity = scanner.nextInt();
        System.out.print("Please enter the price for this product: ");
        double price = scanner.nextDouble();
        System.out.print("Please enter the item number: ");
        int itemNumber = scanner.nextInt();
        System.out.print("Is this product discontinued? (true/false): ");
        boolean isDiscontinued = scanner.nextBoolean();
        scanner.nextLine();  // Clear the buffer
```

```
        products[i] = new DVD(name, price, quantity, itemNumber,
isDiscontinued, length, ageRating, filmStudio);
    }
}
```

OUTPUT:

Output

```
1: Add CD
2: Add DVD
3: Exit
Please enter the product type: 1
Please enter the CD name: Blue
Please enter the artist name: Da Ba Dee
Please enter the record label name: Mars Attax FM
Please enter the number of songs: 10
Please enter the quantity of stock for this product: 50
Please enter the price for this product: 9.0
Please enter the item number: 2
Is this product discontinued? (true/false): false
1: Add CD
2: Add DVD
3: Exit
Please enter the product type: 3
Item Number: 2
Name: Blue
Quantity in stock: 50
Price: 9.0
Stock Value: 450.0
Product Status: Active
Artist: Da Ba Dee
Songs on Album: 10
Record Label: Mars Attax FM
```