

## Programming Assignment Unit 6

Godfrey Ouma

University of the People

CS 1103: Programming 2

Vikas Thada

March 15, 2024

```
package cs;

import java.util.*;

//Custom exception class for item not found
class ItemNotFoundException extends Exception {
    public ItemNotFoundException(String message) {
        super(message);
    }
}

//Generic LibraryItem class
class LibraryItem<T> {
    private String title;
    private String author;
    private int itemID;

    public LibraryItem(String title, String author, int itemID) {
        this.title = title;
        this.author = author;
        this.itemID = itemID;
    }

    //Getters and setters
    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {
        this.title = title;
    }

    public String getAuthor() {
        return author;
    }

    public void setAuthor(String author) {
        this.author = author;
    }

    public int getItemID() {
        return itemID;
    }

    @Override
    public String toString() {
        return "LibraryItem{" +
            "title='" + title + '\'' +
            ", author='" + author + '\'' +
            ", itemID=" + itemID +
            '}';
    }
}

//Generic Catalog Class
class Catalog<T> extends LibraryItem<?>> {
    private List<T> items;

    public Catalog() {
```

```

        items = new ArrayList<>();
    }

    //Method to add a new library item
    public void addItem(T item) {
        items.add(item);
        System.out.println("Item added to catalog: " + item);
    }

    //Method to remove a library item
    public void removeItem(T item) throws ItemNotFoundException {
        if (items.contains(item)) {
            items.remove(item);
            System.out.println("Item removed from catalog: " + item);
        } else {
            throw new ItemNotFoundException("Item not found in catalog.");
        }
    }

    //Method to retrieve item details
    public void getItemDetails() {
        for (T item : items) {
            System.out.println(item);
        }
    }
}

public class LibraryCatalog {

    public static void main(String[] args) {
        //Creating a catalog for LibraryItem objects
        Catalog<LibraryItem<?>> catalog = new Catalog<>();

        //Adding new LibraryItems to the Catalog
        LibraryItem<Book> book1 = new LibraryItem<>("Java Programming", "John Smith",
101);
        LibraryItem<DVD> dvd1 = new LibraryItem<>("The Matrix", "Lana Wachowski", 201);
        LibraryItem<Magazine> magazine1 = new LibraryItem<>("National Geographic",
"Various Authors", 301);

        catalog.addItem(book1);
        catalog.addItem(dvd1);
        catalog.addItem(magazine1);

        try {
            //Attempt to remove an item from catalog
            catalog.removeItem(dvd1);

            //Trying to remove a non-existent item
            catalog.removeItem(new LibraryItem<>("Non-existent Item", "Unknown", 999));
        } catch (ItemNotFoundException e) {
            System.out.println("Error: " + e.getMessage());
        }

        //Retrieving item details
        System.out.println("Catalog items: ");
        catalog.getItemDetails();
    }
}

```

```
//Example subclasses representing different types of library items
class Book extends LibraryItem<Book> {
    public Book(String title, String author, int itemID) {
        super(title, author, itemID);
    }
}

class DVD extends LibraryItem<DVD> {
    public DVD(String title, String author, int itemID) {
        super(title, author, itemID);
    }
}

class Magazine extends LibraryItem<Magazine> {
    public Magazine(String title, String author, int itemID) {
        super(title, author, itemID);
    }
}
```

## Console

```
Item added to catalog: LibraryItem{title='Java Programming', author='John Smith', itemID=101}
Item added to catalog: LibraryItem{title='The Matrix', author='Lana Wachowski', itemID=201}
Item added to catalog: LibraryItem{title='National Geographic', author='Various Authors', itemID=301}
Item removed from catalog: LibraryItem{title='The Matrix', author='Lana Wachowski', itemID=201}
Error: Item not found in catalog.
Catalog items:
LibraryItem{title='Java Programming', author='John Smith', itemID=101}
LibraryItem{title='National Geographic', author='Various Authors', itemID=301}
```