

## Lab worksheet 3: Object Oriented Concepts

1. Imagine you are building a library system using Java to manage books and other library items. The system will involve the creation of classes for **BorrowableItems**, **Book**, **Person**, and **Student**. These classes will help you showcase OOP principles.

Create an abstract class called **BorrowableItems** with an abstract method **displayInfo()**.

Create a class named **Book** as a subclass of the **BorrowableItems** class with the following private attributes:

- **title** (String)
- **author** (String)
- **ISBN** (String)
- **available** (boolean)

Provide public getter and setter methods for these attributes.

Include a constructor to initialize the attributes.

Implement the **displayInfo()** method that displays information about the book.

Create a **Person** class with the following private attribute:

- **name** (String).

Create a constructor to set the **name**.

Create a **Student** class that inherits from the **Person** class.

Create a **Library** class with the following private attribute:

- **BorrowableItemsList** (Array or another suitable data structure to store **BorrowableItems** objects)

Implement methods to:

- Add library items to the collection.
- Check out library items (mark them as available or unavailable).
- List all available library items in the library.
- Display information about the library, including the number of available items.

Create instances of the **Book** class representing library books. Use the following book information:

- Title: "Seena Maragatham", Author: "Sujatha Thilaka", ISBN: "9789553023975"
- Title: "Nuramakadya Bauthika Nuladanayakshanamaala", Author: "Seynamasasekaka", ISBN: "9789553548721"
- Title: "Island of a Thousand Mirrors", Author: "Nayomi Munaweera", ISBN: "9781616953623"

## Lab worksheet 3: Object Oriented Concepts

Create an instance of the **Library** class to manage the library system.

Add the created books to the library's collection using the **addLibraryItem** method of the **Library** class. For each book, call **addLibraryItem** to include them in the library's inventory.

Display information about the library, including the total number of items, by calling the **displayLibraryInfo** method.

List all available library items in the library using the **listAvailableItems** method of the **Library** class.

Demonstrate the process of checking out a library item by calling the **checkoutItem** method. For example, check out the book with the title "Island of a Thousand Mirrors" by setting its availability to false.

List available items in the library again using the **listAvailableItems** method. Observe that the book you checked out is now marked as unavailable.

### BorrowableItems.java

```
abstract class BorrowableItems {  
    public abstract void displayInfo();  
}
```

### Book.java

```
class Book extends BorrowableItems {  
    private String title;  
    private String author;  
    private String ISBN;  
    private boolean available;  
  
    public Book(String title, String author, String ISBN) {  
        this.title = title;  
        this.author = author;  
        this.ISBN = ISBN;  
        this.available = true;  
    }  
  
    public String getTitle() {  
        return title;  
    }  
  
    public void setTitle(String title) {  
        this.title = title;  
    }  
}
```

## Lab worksheet 3: Object Oriented Concepts

```
}

public String getAuthor() {
    return author;
}

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

public String getISBN() {
    return ISBN;
}

public void setISBN(String ISBN) {
    this.ISBN = ISBN;
}

public boolean isAvailable() {
    return available;
}

public void setAvailable(boolean available) {
    this.available = available;
}

@Override
public void displayInfo() {
    System.out.println("Title: " + title);
    System.out.println("Author: " + author);
    System.out.println("ISBN: " + ISBN);
    System.out.println("Available: " + available);
}
}
```

### Person.java

```
class Person {
    private String name;

    public Person(String name) {
        this.name = name;
    }

    public String getName() {
```

## Lab worksheet 3: Object Oriented Concepts

```
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}
```

### Student.java

```
class Student extends Person {
    public Student(String name) {
        super(name);
    }
}
```

### Library.java

```
import java.util.ArrayList;
import java.util.List;

class Library {
    private List<BorrowableItems> borrowableItemsList = new
ArrayList<>();

    public void addLibraryItem(BorrowableItems item) {
        borrowableItemsList.add(item);
    }

    public void checkoutItem(String title) {
        for (BorrowableItems item : borrowableItemsList) {
            if (item instanceof Book) {
                Book book = (Book) item;
                if (book.getTitle().equals(title) &&
book.isAvailable()) {
                    book.setAvailable(false);
                    System.out.println("Checked out: " +
title);

                    return;
                }
            }
        }
        System.out.println("Book not found or already checked
out: " + title);
    }
}
```

## Lab worksheet 3: Object Oriented Concepts

```
public void listAvailableItems() {
    System.out.println("Available items in the library:");
    for (BorrowableItems item : borrowableItemsList) {
        if (item instanceof Book) {
            Book book = (Book) item;
            if (book.isAvailable()) {
                book.displayInfo();
            }
        }
    }
}

public void displayLibraryInfo() {
    System.out.println("Total number of items in the
library: " + borrowableItemsList.size());
}
}
```

### LibrarySystem.java

```
public class LibrarySystem {
    public static void main(String[] args) {
        // Create instances of the Book class representing
        library books
        Book book1 = new Book("Seena Maragatham", "Sujatha
Thilaka", "9789553023975");
        Book book2 = new Book("Nuramakadya Bauthika
Nuladanayakshanamaala", "Seynamasasekaka", "9789553548721");
        Book book3 = new Book("Island of a Thousand Mirrors",
"Nayomi Munaweera", "9781616953623");

        // Create an instance of the Library class to manage
the library system
        Library library = new Library();

        // Add the created books to the library's collection
        library.addLibraryItem(book1);
        library.addLibraryItem(book2);
        library.addLibraryItem(book3);

        // Display information about the library
        library.displayLibraryInfo();

        // List all available library items
        library.listAvailableItems();
    }
}
```

### Lab worksheet 3: Object Oriented Concepts

```
// Check out a book
library.checkoutItem("Island of a Thousand Mirrors");

// List available items in the library again
library.listAvailableItems();
}
}
```