

Lab 6: Interfaces for Iterators and Data Structures

Software provided:

- Classes and interfaces that represent a list of books, with iterators, and selectors, as well as samples of client code.

File lab6.java

- Classes and interfaces that represent a stack of books, as well as samples of client code.

File lab6-part2.java

The goals are

- Learn to use the interface as a tool to implement external views of lists of Objects
- Learn to program iterator-based traversal of structured data
- Learn to use interfaces to represent abstract data structure
- Quiz covers the complexities of Java

Details

- **A list of Objects and their traversals.**
 - Open the file lab6.java.
 - Draw by hand a class diagram that represents these classes. Work with the whole class. Explain to each other the meaning of various connections between classes and interfaces.
 - Add two of your favorite books (and their authors) to the examples and to one of the lists in the examples and add tests that use your examples.
 - Study with a partner the design of the method `filter` in the class `TestClass`.
 - Design the needed class that implements `IObj2Bool`, so you can produce a list of all books by a given author.

- Use a similar technique to design `orMap` method in the class `TestClass`. Test it with both classes that implement `IObj2Bool`.

- **Using an iterator in a loop**

- Study with a partner the design of the method `iterFilter` in the class `TestClass`.
- Produce a list of all books by a given author, using the class you designed earlier, and using the `iterFilter` method.
- Use a similar technique to design `orMap` method in the class `TestClass`. Test it with both classes that implement `IObj2Bool`.

- **Defining objects that encapsulate the behavior of a data structure**

- Study the code for the interface `IStack`. Think of how you could implement this interface, using a list of objects as the needed data.
- Study the code for the class `ListStack` that implements the `IStack` interface, and its use in `reverse` method in the class `TestClass`.
- Design several test cases for this method.

- **QUIZ**