# Lab: Iterators and Comparators

Problems for exercises and homework for the ["Java OOP Advanced" course @ SoftUni.](https://softuni.bg/courses/java-oop-advanced)

You can check your solutions here: <https://judge.softuni.bg/Contests/523/Iterators-and-Comparators-Lab>

## Book

Create a class **Book** from **UML diagram** below:

|  |  |
| --- | --- |
| **Book** | |
| - | title: String |
| - | year: int |
| - | authors: List<String> |
| - | setTitle(String) |
| - | setYear(String) |
| - | setAuthors(String…) |
| + | getTitle(): String |
| + | getYear(): int |
| + | getAuthors(): List<String> |

You can use only **one constructor**. Authors can be **anonymous, one or many**.

### Examples

|  |
| --- |
| Main.java |
| **public static void** main(String[] args) {  Book bookOne = **new** Book(**"Animal Farm"**, 2003, **"George Orwell"**);  Book bookThree = **new** Book(**"The Documents in the Case"**, 2002);  Book bookTwo = **new** Book(**"The Documents in the Case"**, 1930, **"Dorothy Sayers"**, **"Robert Eustace"**);  List<Book> books = **new** ArrayList<>();  books.add(bookOne);  books.add(bookTwo);  books.add(bookThree);  } |

### Solution



## Library

Create a class **Library** from **UML diagram** below:

|  |  |
| --- | --- |
| **<<Iterable<Book>>>**  **Library** | |
| - | books: Book[] |
| + | iterator(): Iterator<Book> |

Create a **nested class** **LibIterator** from **UML diagram** below:

|  |  |
| --- | --- |
| **<<Iterator<Book>>>**  **LibIterator** | |
| - | counter: int |
| + | hasNext(): boolean |
| + | next(): Book |

### Examples

|  |
| --- |
| Main.java |
| **public static void** main(String[] args) {  Book bookOne = **new** Book(**"Animal Farm"**, 2003, **"George Orwell"**);  Book bookThree = **new** Book(**"The Documents in the Case"**, 2002);  Book bookTwo = **new** Book(**"The Documents in the Case"**, 1930, **"Dorothy Sayers"**, **"Robert Eustace"**);   Library library = **new** Library<>(bookOne, bookTwo, bookThree);   **for** (Book book : library) {  System.***out***.println(book.getTitle());  } } |

### Solution



## Comparable Book

Expand **Book** from **UML diagram** below:

|  |  |
| --- | --- |
| **<<Comparable<Book>>>**  **Book** | |
| - | title: String |
| - | year: int |
| - | authors: List<String> |
| - | setTitle(String) |
| - | setYear(String) |
| - | setAuthors(String…) |
| + | getTitle(): String |
| + | getYear(): int |
| + | getAuthors(): List<String> |
| + | **compareTo(Book): int** |

You can use only **one constructor**. Authors can be **anonymous, one or many**.

### Examples

|  |
| --- |
| Main.java |
| **public static void** main(String[] args) {  Book bookOne = **new** Book(**"Animal Farm"**, 2003, **"George Orwell"**);  Book bookThree = **new** Book(**"The Documents in the Case"**, 2002);  Book bookTwo = **new** Book(**"The Documents in the Case"**, 1930, **"Dorothy Sayers"**, **"Robert Eustace"**);   **if** (bookOne.compareTo(bookTwo) > 0) {  System.***out***.println(String.*format*(**"%s is before %s"**, bookOne, bookTwo));  } **else if** (bookOne.compareTo(bookTwo) < 0) {  System.***out***.println(String.*format*(**"%s is before %s"**, bookTwo, bookOne));  } **else** {  System.***out***.println(**"Book are equal"**);  }  } |

## Book Comparator

Create a class **BookComparator** from **UML diagram** below:

|  |  |
| --- | --- |
| **<<Comparator<Book>>>**  **BookComparator** | |
| + | **compare(Book, Book): int** |

**BookComparator** have to **compare** two books by:

1. Book title
2. Year of publishing a book

### Examples

|  |
| --- |
| Main.java |
| **public static void** main(String[] args) {  Book bookOne = **new** Book(**"Animal Farm"**, 2003, **"George Orwell"**);  Book bookThree = **new** Book(**"The Documents in the Case"**, 2002);  Book bookTwo = **new** Book(**"The Documents in the Case"**, 1930, **"Dorothy Sayers"**, **"Robert Eustace"**);   List<Book> books = **new** ArrayList<>();  books.add(bookOne);  books.add(bookTwo);  books.add(bookThree);   books.sort(**new** BookComparator());   **for** (Book book : books) {  System.***out***.println(book.getTitle() + book.getYear());  } } |