
System Requirements Specification Index

For

Book Info Processor- File Handling

Version 1.0

IIHT Pvt. Ltd.
fullstack@iiht.com

TABLE OF CONTENTS

1	Project Abstract	3
2	Common Constraints	3
3	Template Code Structure	3
3.1	Package: com.bookinfoapplication	3
4	Execution Steps to Follow	4

BOOK INFO PROCESSOR

System Requirements Specification

1 PROJECT ABSTRACT

Book Information Processor is a console-based application designed to manage and process information about books. The application utilizes file handling to read and write book data, allowing users to input book details, perform data operations, and find books with the highest number of pages.

2 COMMON CONSTRAINTS

1. Take console input for the book in sequence title, author, page like xyz, abc, 100.
2. Store the books data into List.
3. In performDataOperation(), we must update the page count of each book by multiplying it with 2.
4. That list data should be saved into a "books.txt" file in the same directory.
5. We must return the author name of such a book whose page count is highest from findBookWithHighestPages().

3 TEMPLATE CODE STRUCTURE

3.1 PACKAGE: COM.BOOKINFOAPPLICATION

Resources

Class/Interface	Description	Status
Book (class)	<ul style="list-style-type: none">• This class contains all the properties of the Book class.	Already implemented.
BookInfoProcessor(class)	<ul style="list-style-type: none">• This class contains all the methods which are used to write the business logic for the application• You can create any number of private methods in the class• You need to write the code in functions.	Partially implemented.

4 EXECUTION STEPS TO FOLLOW

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers, need to go to Application menu (Three horizontal lines at left top) □ Terminal □New Terminal.
3. This editor Auto Saves the code.
4. If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use CTRL+Shift+B-command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.
5. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.
6. To build your project use command:
mvn clean package -Dmaven.test.skip
7. To run your project use command:
mvn clean install exec:java
-Dexec.mainClass="com.bookinfoapplication.BookInfoProcessor"
8. To test your project test cases, use the command
mvn test
9. You need to use CTRL+Shift+B - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.