AMAZON KINDLE ANALYSER

IIHT

Time To Complete: 3 hrs

CONTENTS

1 PROBLEM STATEMENT

The Amazon Kindle Analyzer Application allows users to perform not only CRUD (Create, Read, Update, Delete) operations and search functionalities in different criterias on books, shelves, user profiles, and user ratings but also provides the analytical operations like getting suggested shelves as per user's genre, get list of all trending books, get list of all purchased books and many more for analysis and viewing.

2 BUSINESS REQUIREMENTS:

Screen Name	Console input screen
Problem Statement	1. User needs to enter into the application. 2. The user should be able to do the particular operations 3. The console should display the menu 1) create a new user 2) update a user 3) get details of a user 4) create a new book 5) update a book 6) get details of a book 7) create a new shelf 8) update a shelf 9) get details of a shelf 10) create user rating 11) search for a book 12) get books suggestions by genre 13) get insights of most read books by age 14) get book suggestions by rating 15) show trending books 16) get list of purchased books 17) get percentage of books purchased after reading their sample 18) exit

3 IMPLEMENTATION/FUNCTIONAL REQUIREMENTS

3.1 CODE QUALITY/OPTIMIZATIONS

1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

3.2 TEMPLATE CODE STRUCTURE

PACKAGE: COM.AMAZONKINDLEANALYSERAPPLICATION

Resources

Class/Interface	Description	Status
AmazonKindleAnalyserApplication.java(class)	This represents bootstrap class i.e class with Main method, that shall contain all console interaction with the user.	Partially implemented

PACKAGE: COM.AMAZONKINDLEANALYSERAPPLICATION.MODEL

Resources

Class/Interface	Description	Status
Book.java(class)	This represents entity class for Book	Partially Implemented
Shelf.java(class)	This represents entity class for Shelf	Partially Implemented
User.java(class)	This represents entity class for User	Partially Implemented
UserRating.java(class)	This represents entity class for UserRating	Partially Implemented

PACKAGE: COM.AMAZONKINDLEANALYSERAPPLICATION.REPOSITORY

Resources

Class/Interface	Description	Status
BookDAO.java(interface)	This is an interface containing declaration of DAO method	Already Implemented
BookDAOImpl.java(class)	This is an implementation class for DAO methods. Contains empty method bodies, where logic needs to written by test taker	Partially Implemented
ShelfDAO.java(interface)	This is an interface containing declaration of DAO method	Already Implemented
ShelfDAOImpl.java(class)	This is an implementation class for DAO methods. Contains empty method bodies, where logic needs to written by test taker	Partially Implemented
UserDAO.java(interface)	This is an interface containing declaration of DAO method	Already Implemented
UserDAOImpl.java(class)	This is an implementation class for DAO methods. Contains empty	Partially Implemented

	method bodies, where logic needs to written by test taker	
UserRatingDAO.java(interface)	**This is an interface containing declaration of DAO method**	**Already Implemented**
UserRatingDAOImpl.java(class)	**This is an implementation class for DAO methods. Contains empty method bodies, where logic needs to written by test taker**	**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 the Application menu (Three horizontal lines at left top) -> Terminal ->New Terminal.**
3. **To build your project use command:**
   mvn clean package -Dmaven.test.skip
4. **This editor Auto Saves the code.**
5. **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.**
6. **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.**

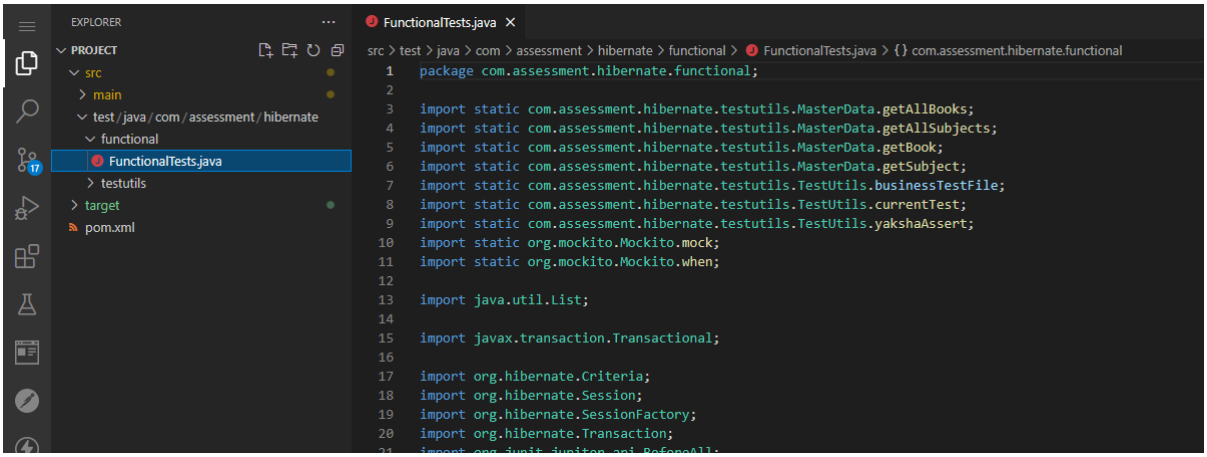
7. **Default credentials for MySQL:**
   a. **Username: root**
   b. **Password: pass@word1**

8. **To login to mysql instance: Open new terminal and use following command:**
   a. **sudo systemctl enable mysql**
   b. **mysql -u root -p**
   The last command will ask for password which is 'pass@word1'

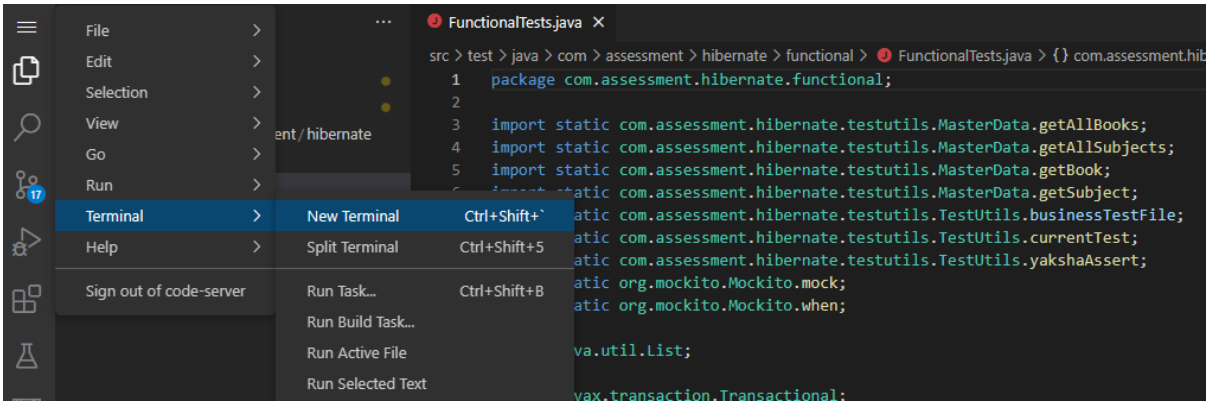
9. **These are time bound assessments. The timer would stop if you logout (Save & Exit) and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.**
10. **To run your project use command:**

    mvn clean install exec:java -Dexec.mainClass="com.amazonkindleanalyserapplication.AmazonKindleAnalyserApplication"
11. **To test your project, use the command**

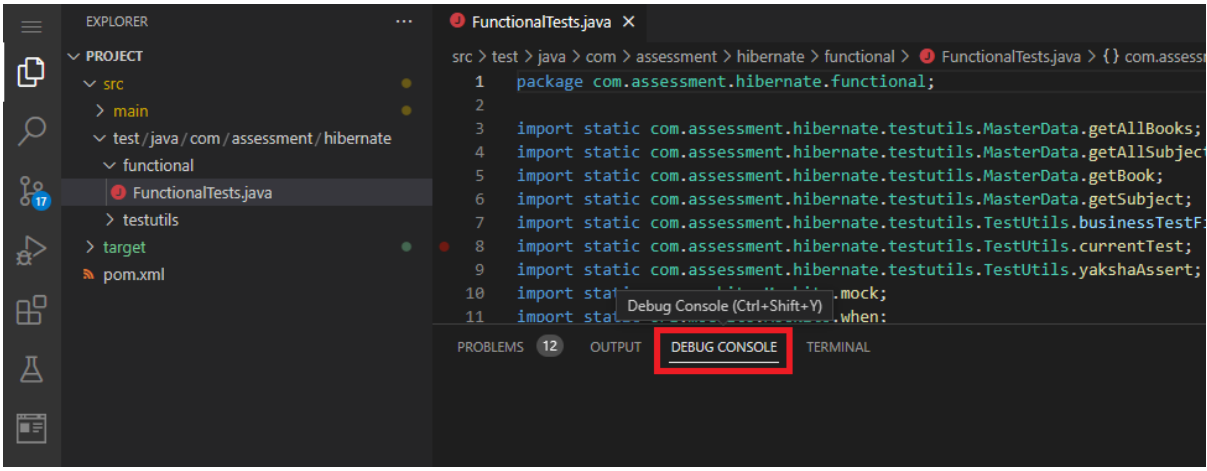
a. **Open FunctionalTests.java file in editor**



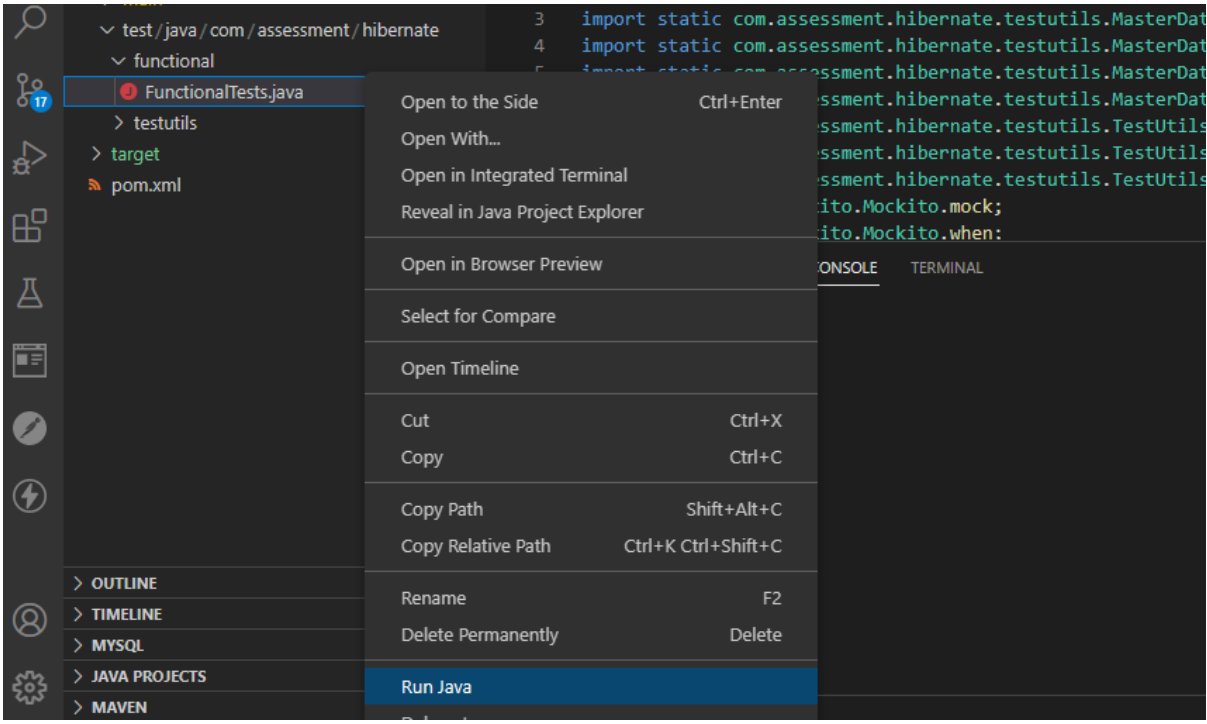
b. **Open a new Terminal**



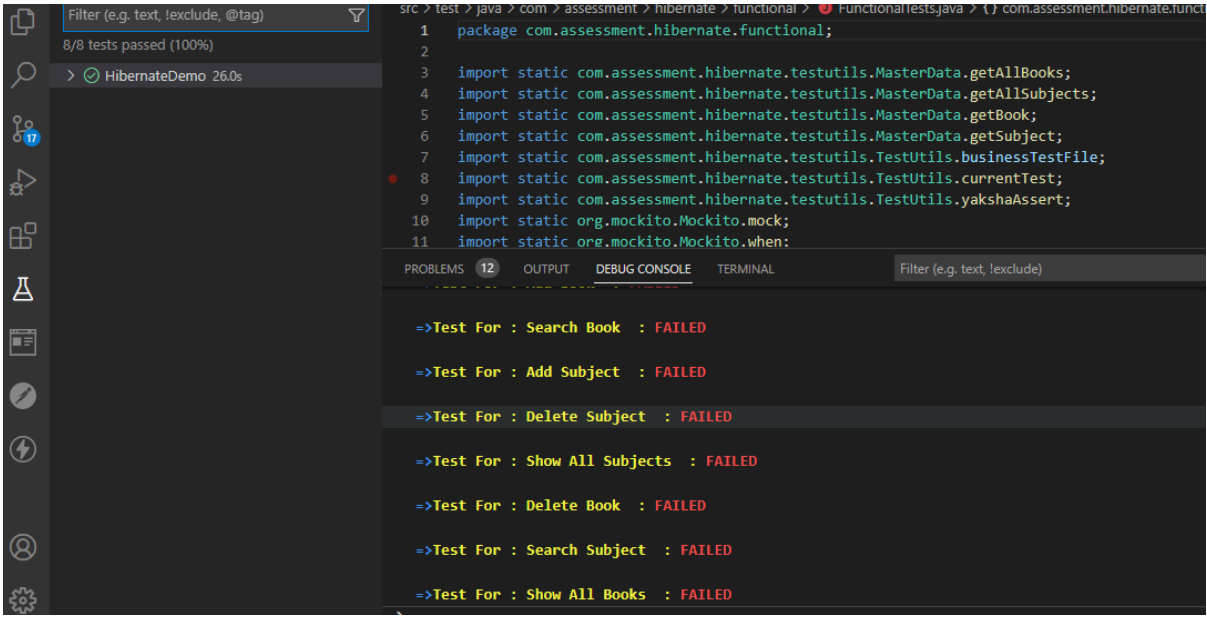
c. **Go to Debug Console Tab**



d. **Right click on FunctionalTests.java file and select option Run Java**



e. **This will launch the test cases and status of the same can be viewed in Debug Console**



12. **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.**