

PLAYER'S SELECTION APPLICATION

IIHT

Time To Complete: 3 hrs

CONTENTS

1 Problem Statement	3
2 Business Requirements:	3
3 Implementation/Functional Requirements	3
3.1 Code Quality/Optimizations	3
3.2 Template Code Structure	4
a. Package: com.playersselectionapplication	4
b. Package: com.playersselectionapplication.model	4
c. Package: com.playersselectionapplication.repository	4
4 Execution Steps to Follow	5

1 PROBLEM STATEMENT

The Players Selection Application allows users to perform CRUD (Create, Read, Update, Delete) operations and search functionalities in different criterias on players and scores. Users can create new player profiles, record and update scores for each player, delete player entries, and retrieve player and score details for viewing.

2 BUSINESS REQUIREMENTS:

Screen Name	Console input screen
Problem Statement	<ol style="list-style-type: none">1. User needs to enter into the application.2. The user should be able to do the particular operations3. The console should display the menu<ol style="list-style-type: none">1) create a player2) create a score3) update player4) Update score5) delete player6) delete score7) show all players8) show all scores9) search players by name10) search players by domestic team11) get scores by player ID12) get average of last three scores for a player13) 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.PLAYERSSELECTIONAPPLICATION

Resources

Class/Interface	Description	Status
PlayersSelectionApplication.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.PLAYERSSELECTIONAPPLICATION.MODEL

Resources

Class/Interface	Description	Status
Player.java(class)	This represents entity class for Player	Partially Implemented
Score.java(class)	This represents entity class for Score	Partially Implemented

PACKAGE: COM.PLAYERSSELECTIONAPPLICATION.REPOSITORY

Resources

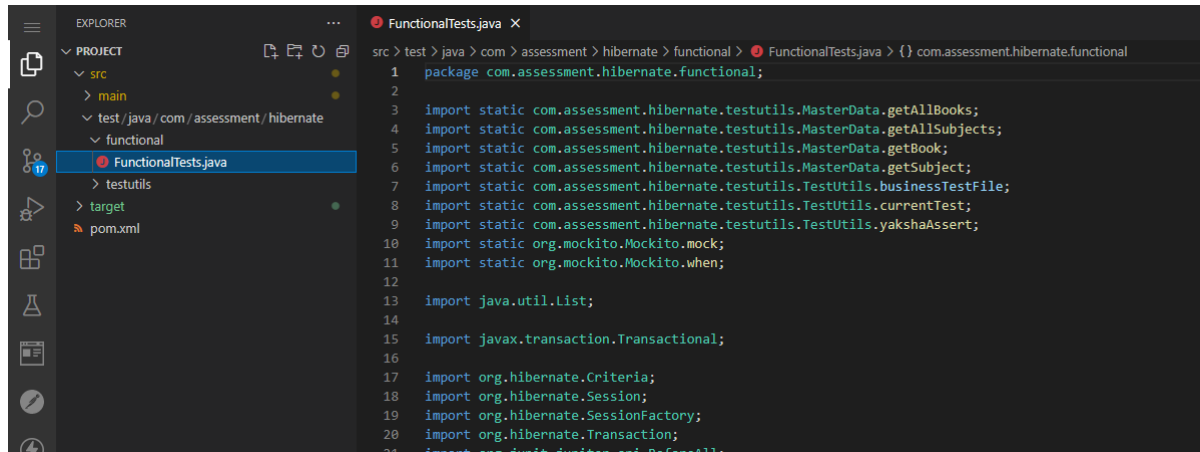
Class/Interface	Description	Status
PlayerDao.java(interface)	This is an interface containing declaration of DAO method	Already Implemented
PlayerDaoImpl.java(class)	This is an implementation class for DAO methods. Contains empty method bodies, where logic needs to written by test taker	Partially Implemented
ScoreDao.java(interface)	This is an interface containing declaration of DAO method	Already Implemented
ScoreDaoImpl.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

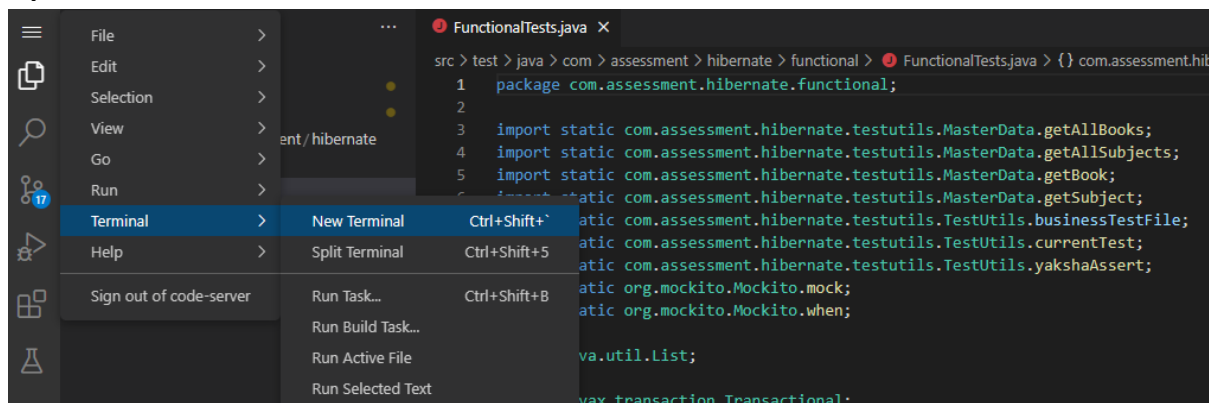
1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers need to go to the Application menu (Three horizontal lines at left top) -> Terminal ->New Terminal.
3. To build your project use command:
mvn clean 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.playersselectionapplication.PlayersSelectionApplication"

11. To test your project, use the command

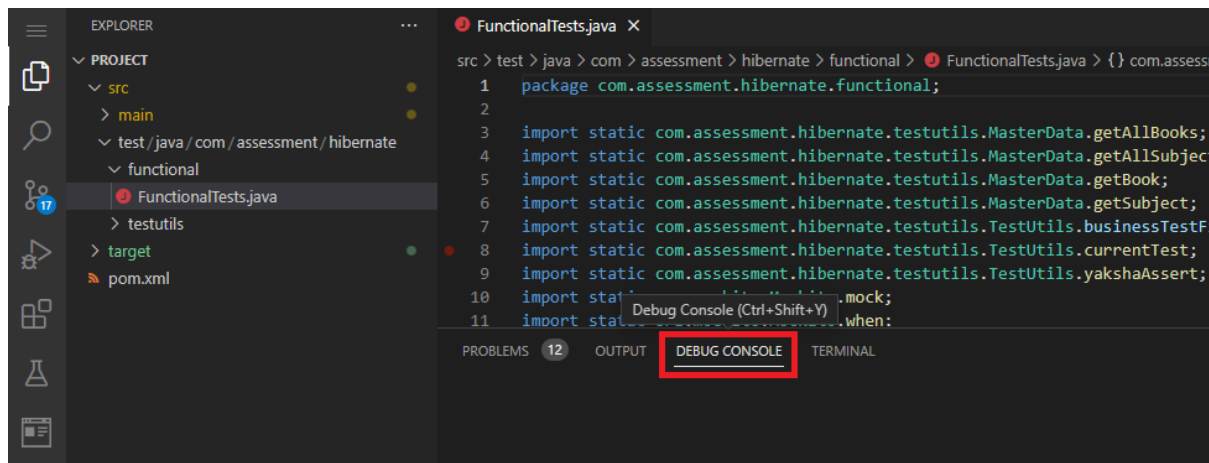
a. Open FunctionalTests.java file in editor



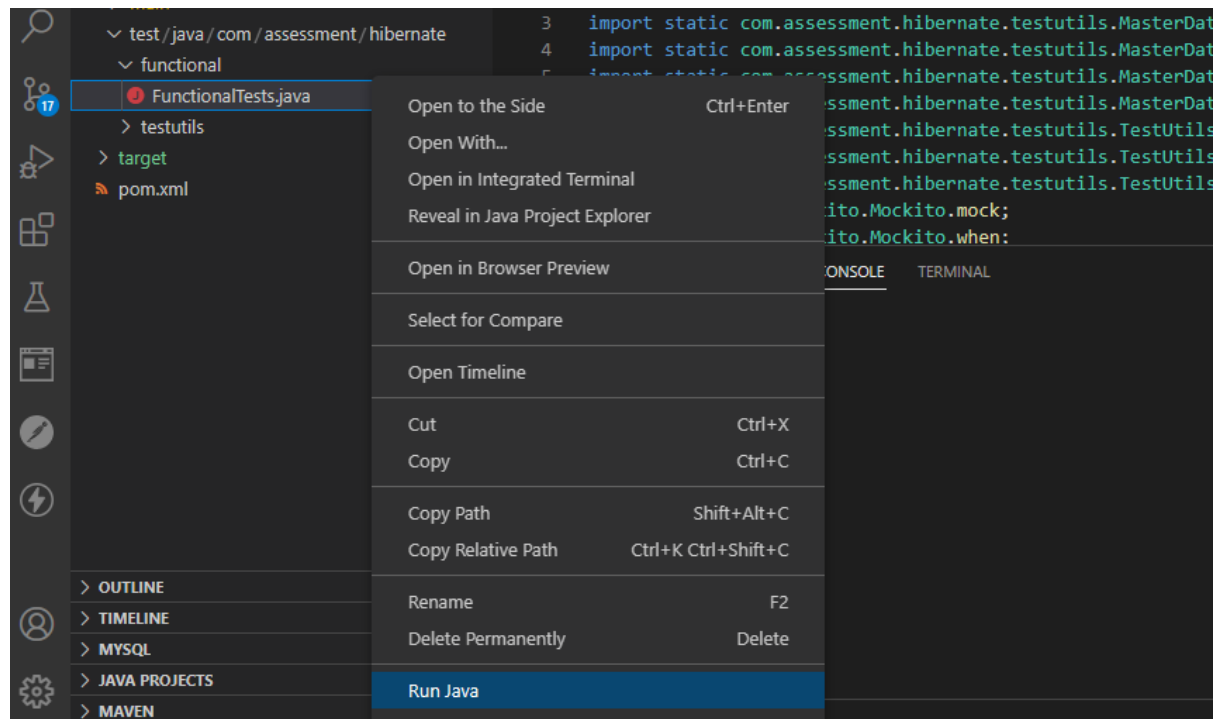
b. Open a new Terminal



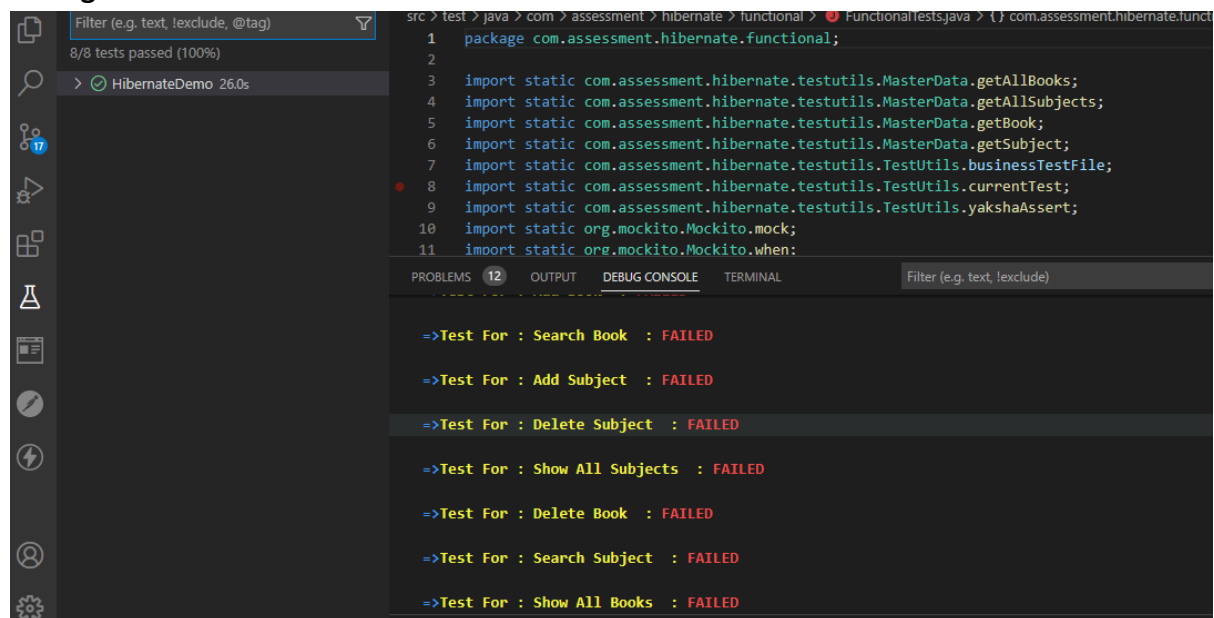
c. Go to Debug Console Tab



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.