Book Shop(---RETIRED---)

Grade settings: Maximum grade: 100
Disable external file upload, paste and drop external content: Yes
Based on: Book Shop
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

Paul runs a bookshop on the main street. He wants to sell most of the books in his shop, so he plans to give a discount based on the published year. As a software developer help Paul to calculate the book cost.

Component Specification: Book

Type(Class)	Attributes	Methods
Book	String bookName String authorName int publishedYear double cost	Necessary getters, setters and constructors are provided as part of the code skeleton.

Functional Requirement 1: Extract the details of the Book and create an object of the Book class.

Type(Class)	Methods	Responsibilities
UserInterface	public static Book **extractDetails** (String bookDetails)	This method accepts **bookDetails** separated by colon as an argument and should extract the properties of the **Book** from the argument by parsing the **bookDetails**. Set these values to the **Book** object and return this object

Functional Requirement 2: Calculate the book cost based on the published year.

Type(Class)	Methods	Responsibilities
Book	public double **calculateBookCost**()	This method is used to calculate the book cost based on the published year. If the publishedYear is between **1900 and 1940** (both inclusive), discount is 25% of the cost. If the publishedYear is between **1941 and 1980** (both inclusive), discount is 35% of the cost. If the publishedYear is between **1981 and 2020** (both inclusive), discount is 15% of the cost. ***Condition:*** • *The **publishedYear** should be between 1900 to 2020, else return 0.* • *The **cost** should be greater than 0, else return 0.*

Formula to calculate book cost:

Book Cost = cost - discount

Example:

Enter the Book details

The Rainbow:Lawrance:1915:600 (If the publishedYear is between, 1900 to 1940, discount is 25% of the cost.)

Book Cost = 600 - (600*25)/100

Book Cost = 450.0 (Roundoff the result to one decimal place)

The main method in the UserInterface class is excluded from evaluation. You are free to write your own code in the main method to invoke the business methods to check its correctness.

Note:

- Roundoff the result to one decimal place.
- In the Sample Input / Output provided, the highlighted text in bold corresponds to the input given by the user and the rest of the text represents the output.
- Ensure to follow the object-oriented specifications provided in the question.
- Ensure to provide the names for classes, attributes, and methods as specified in the question.
- Adhere to the code template, if provided.

Sample Input / Output 1:

Enter the Book details

Invisible Man:Ralph Ellison:1952:500

Book Details

Book Name : Invisible Man

Author Name : Ralph Ellison

Published Year : 1952

Book Cost : 325.0

Sample Input / Output 2:

Enter the Book details

Lolita:Nabakov:1970:-200

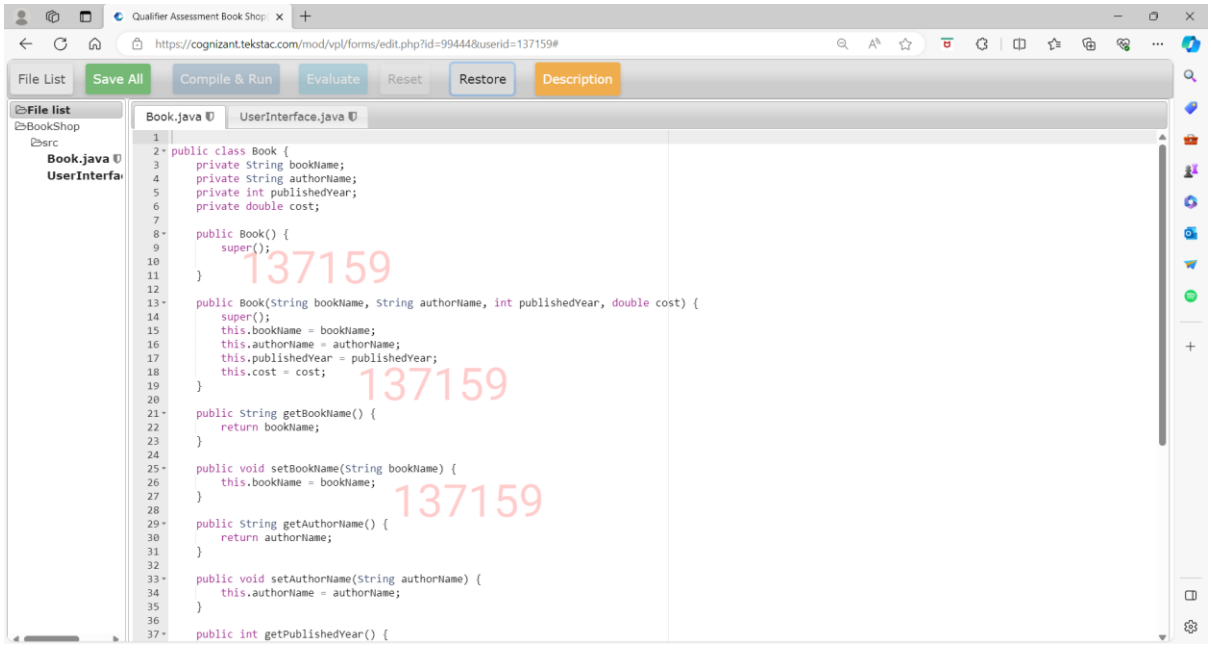
Invalid book details

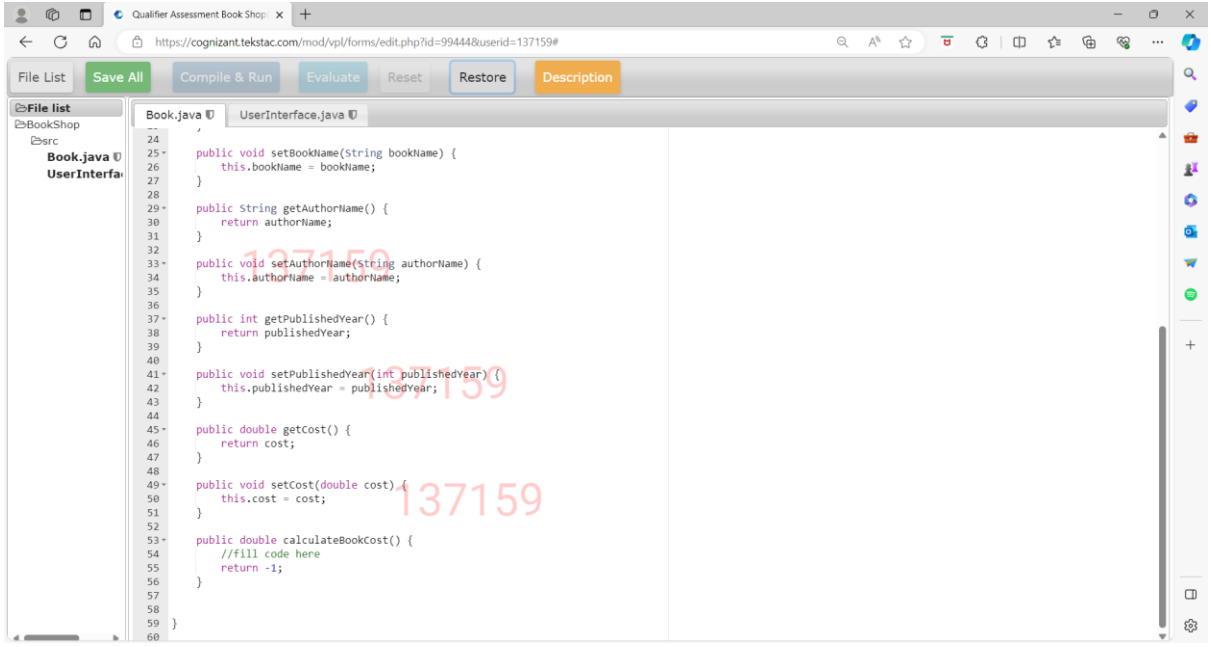
Sample Input / Output 3:

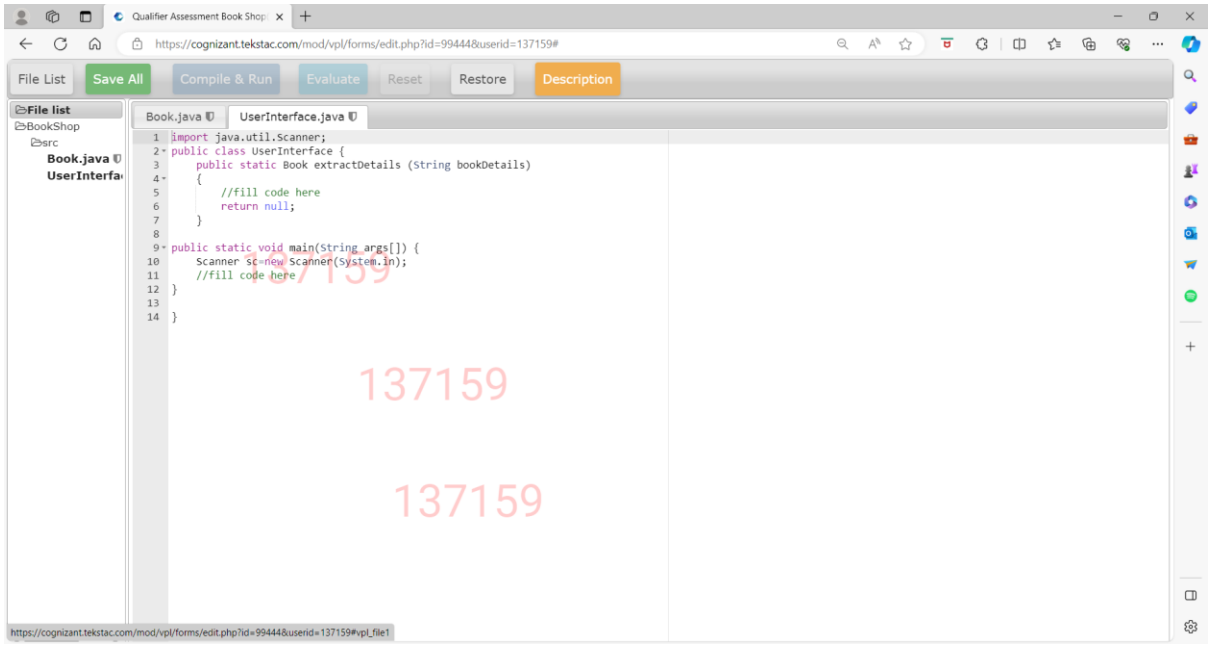
Enter the Book details

The Shining:Stephen:1870:600

Invalid book details






Qualifier Assessment Book Shop
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=99444&userid=137159#
File List
Save All
Compile & Run
Evaluate
Reset
Restore
Description
File list
BookShop
src
Book.java
UserInterfa
Book.java
UserInterface.java
import java.util.Scanner;
public class UserInterface {
public static Book extractDetails (String bookDetails)
{
//fill code here
return null;
}
public static void main(String args[]) {
Scanner sc=new Scanner(System.in);
//fill code here
}
}
137159
137159
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=99444&userid=137159#vpl_file1