Orcrist Watch(---RETIRED---)

Grade settings: Maximum grade: 100
Disable external file upload, paste and drop external content: Yes
Based on: Orcrist Watch(---RETIRED---)
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

Orcrist is one of the famous watch brands. They establish their brand all over the city. They now wanted to automate the process of calculating the cost of watch for the customers. The manager intimates a software developer to help in their process. You being a software developer, develop a Java program based on the requirement.

Component Specification: OrderInfo

Type(Class)	Attributes	Methods
OrderInfo	String orderId String customerName String watchModel int quantity	Appropriate getter, setters and a four-argument c provided as a part of code skeleton.

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

Type(Class)	Methods	Responsibilities
UserInterface	public static OrderInfo **extractDetails**(String orderDetails)	This method accepts orderDetails separated by c argument and should extract the properties of th from the argument. Set these values to the Order return this object

Functional Requirement2: Calculate the watch cost to be paid by the customer.

Type(Class)	Methods	Responsibilities
OrderInfo	public double **calculateWatchCost**()	This method is used to calculate the total cost to be paid customer. The watchModel and the cost per watch is given in the ta ***Condition***: • ***watchModel*** *is case-insensitive.*

		• *The **watchModel** must be either one of **[OrcristVox2, OrcristVox1, OrcristTrx1000** else return -1.* • *The **quantity** must be **greater than zero**, else retu*

watchModel	Cost per watch
OrcristVox2	5200
OrcristVox1	4600
OrcristTrx1000	3200
OrcristRvs2	2500

Formula to calculate the watch cost:

Total cost = quantity * Cost per watch

[find the cost based on the watchModel from the above table]

If the quantity is 2 and watchModel is "OrcristVox2",

Total cost = 2 * 5200 = 10400.0

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

Note:

- 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 1:

Enter the order details

W001:James:OrcristVox1:3

Sample Output 1:

Order Id : W001

Customer Name : James

Watch Model : OrcristVox1

Quantity : 3

Total cost to be paid : 13800.0

Sample Input 2:

Enter the order details

W002:John:OrcristVox3:3

Sample Output 2:

Invalid details

Explanation: As the watchModel is not one among the mentioned watchModel, the given detail is considered as invalid.

Sample Input 3:

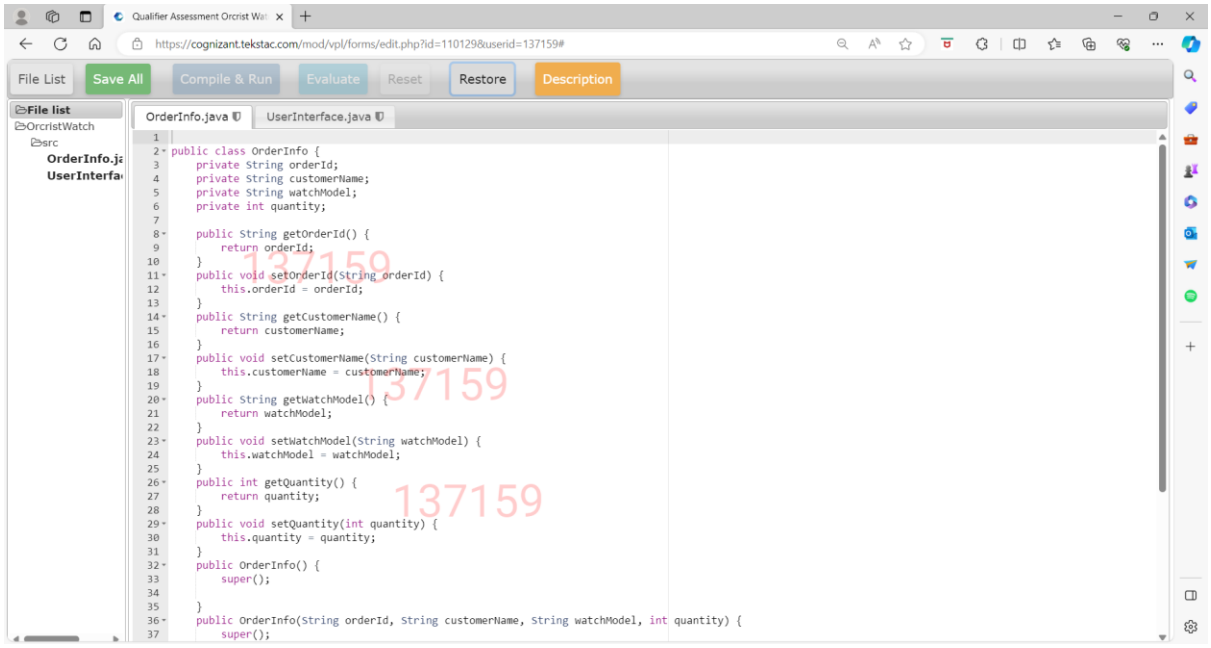
Enter the order details

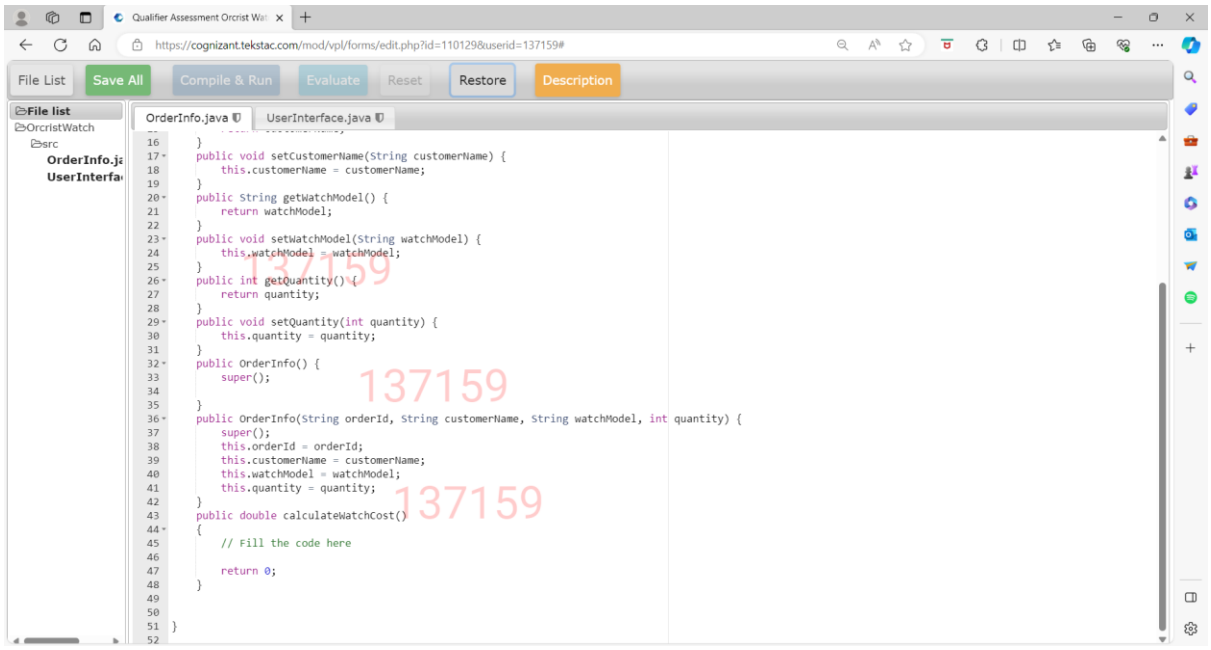
W003:Dave:OrcristVox1:-6

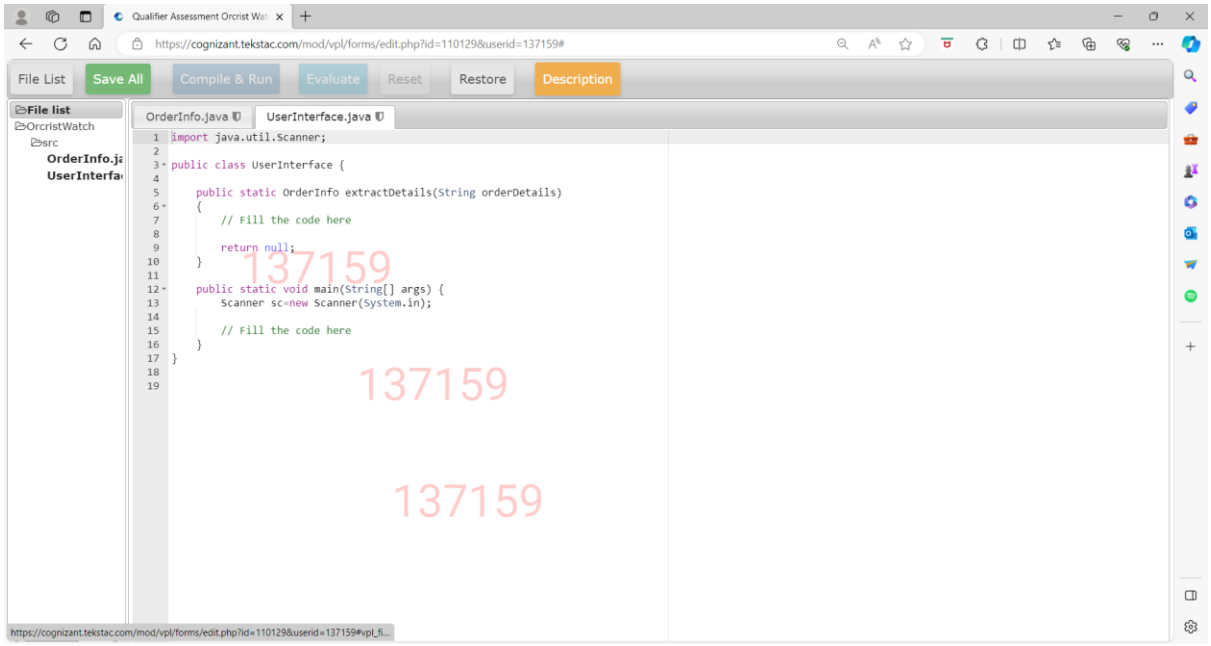
Sample Output 3:

Invalid details

Explanation: As quantity is less than zero, the given detail is considered as invalid


Qualifier Assessment Orcrist Wat
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=1101298&userid=137159#
File List
Save All
Compile & Run
Evaluate
Reset
Restore
Description
File list
OrcristWatch
src
OrderInfo.ja
UserInterfa
OrderInfo.java
UserInterface.java
public class OrderInfo {
private String orderId;
private String customerName;
private String watchModel;
private int quantity;
public String getOrderId() {
return orderId;
}
public void setOrderId(String orderId) {
this.orderId = orderId;
}
public String getCustomerName() {
return customerName;
}
public void setCustomerName(String customerName) {
this.customerName = customerName;
}
public String getWatchModel() {
return watchModel;
}
public void setWatchModel(String watchModel) {
this.watchModel = watchModel;
}
public int getQuantity() {
return quantity;
}
public void setQuantity(int quantity) {
this.quantity = quantity;
}
public OrderInfo() {
super();
}
public OrderInfo(String orderId, String customerName, String watchModel, int quantity) {
super();
137159
137159
137159


Qualifier Assessment Orcrist Wat
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=1101298&userid=137159#
File List
Save All
Compile & Run
Evaluate
Reset
Restore
Description
File list
OrcristWatch
src
OrderInfo.ja
UserInterfa
OrderInfo.java
UserInterface.java
}
public void setCustomerName(String customerName) {
this.customerName = customerName;
}
public String getWatchModel() {
return watchModel;
}
public void setWatchModel(String watchModel) {
this.watchModel = watchModel;
}
public int getQuantity() {
return quantity;
}
public void setQuantity(int quantity) {
this.quantity = quantity;
}
public OrderInfo() {
super();
}
public OrderInfo(String orderId, String customerName, String watchModel, int quantity) {
super();
this.orderId = orderId;
this.customerName = customerName;
this.watchModel = watchModel;
this.quantity = quantity;
}
public double calculateWatchCost()
{
// Fill the code here
return 0;
}
}
137159
137159
137159


Qualifier Assessment Orcrist Wat
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=110129&userid=137159#
File List
Save All
Compile & Run
Evaluate
Reset
Restore
Description
File list
OrcristWatch
src
OrderInfo.ja
UserInterfa
OrderInfo.java
UserInterface.java
1 import java.util.Scanner;
2
3 public class UserInterface {
4
5 public static OrderInfo extractDetails(String orderDetails)
6 {
7 // Fill the code here
8
9 return null;
10 }
11
12 public static void main(String[] args) {
13 Scanner sc=new Scanner(System.in);
14
15 // Fill the code here
16 }
17 }
18
19
137159
137159
137159
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=110129&userid=137159#vpl_fi...