

Assignment 1

Before attempting this project, be sure you have completed all of the reading assignments, nongraded exercises, discussions, and assignments to date.

Write a Java program which:

- (1) Prompts a user to enter customer id, unit price in this format (e.g. 3.75), quantity (as whole number), product description, and discount in this format (e.g., .10) (use Scanner for input).
- (2) Calculates the customer's overall order total before and after discount.
- (3) Displays the input data along with the order total before and after the discount to the console.

To calculate order total **before** discount: (unit price multiplied by quantity)

To calculate order total **after** discount: (unit price multiplied by quantity) multiplied by 1 minus (unit price multiplied by quantity) multiplied by discount

- **Example 1:** For this example, given unit price of 5.00 with a quantity of 2, and the discount of .10: $(5.00 * 2) = 10.00$ **order total before discount** and $(5.00 * 2) * 1 - (5.00 * 2) * .10 = 9.00$ **order total after discount**
- **Example 2:** For this example, given unit price of 7.00 with a quantity of 3, and the discount of .15: $(7.00 * 3) = 21.00$ **order total before discount** and $(7.00 * 3) * 1 - (7.00 * 3) * .15 = 17.85$ **order total after discount**
- **Example 3:** For this example, given unit price of 12.50 with a quantity of 2, and the discount of .10: $(12.50 * 2) = 25.00$ **order total before discount** and $(12.50 * 2) * 1 - (12.50 * 2) * .10 = 22.50$ **order total after discount**

Summary of the Examples:

<u>Example #</u>	<u>Unit Price</u>	<u>Quantity</u>	<u>Discount</u>	<u>Total Price (Before Discount)</u>	<u>Total Price (After Discount)</u>
1	5.0	2	.10	10.00	9.00
2	7.0	3	.15	21.00	17.85
3	12.50	2	.10	25.00	22.50

Test program:

A minimum of 3 test cases should be supplied in the form of table with columns indicating the input values, expected output, actual output and if the test case passed or failed. This table should contain 4 columns with appropriate labels and a row for each test case. An example template is shown below. Note that the actual output should be the actual results you receive when running your program and applying the input for the test record.

Make sure your Java program is using the recommended style such as:

- Javadoc comment up front with your name as author, date, and brief purpose of the program
- Comments for variables and blocks of code to describe major functionality
- Meaningful variable names and prompts
- Class names are written in upper CamelCase
- Constants are written in All Capitals
- Use proper spacing and empty lines to make code human readable

Capture execution:

You should capture and label screen captures associated with compiling your code and running each of your 3 test cases.

Here is a sample run:

```
Enter customer id: 4225
Enter unit price in decimal format (e.g. 3.5): 5.0
Enter quantity: 2
Enter product description: Whole Wheat Bread
Enter discount in decimal format (e.g. .05): .10
```

```
ORDER DATA:
Customer id: 4225
Unit Price: 5.00
Quantity:2
Product Description: Whole Wheat Bread
Discount: .10
```

```
Order total BEFORE discount: 10.00
Order total AFTER discount: 9.00
```

Example test cases:

Input	Expected Output	Actual Output	Pass?
custid: 4225 price: 5.00 quantity: 2 description: Whole Wheat Bread discount: .10	ORDER DATA: Customer ID: 4225 Price: 5.00 Quantity: 2 Product Description: Whole Wheat Bread Discount: .10 Order Total BEFORE Discount: 10.00 Order Total AFTER Discount: 9.00	ORDER DATA: Customer ID: 4225 Price: 5.00 Quantity: 2 Product Description: Whole Wheat Bread Discount: .10 Order Total BEFORE Discount: 10.00 Order Total AFTER Discount: 9.00	Yes

Test Case 2	To be completed	To be completed	
Test Case 3	To be completed	To be completed	

Submission requirements

Deliverables include Java program (.java) and a single Word (or PDF) document. The Java and Word/PDF files should be named appropriately for the assignment (as indicated in the SubmissionRequirements document).

The word (or PDF) document should include screen captures showing the successful compiling and running of each of the test cases. Each screen capture should be properly labeled clearly indicated what the screen capture represents. The test cases table should be included in your Word or PDF document and properly labeled as well.

Submit your files to Assignment 1 submission area no later than the due date listed in your online classroom.

Grading Rubric:

The following grading rubric will be used to determine your grade:

Attribute	Level (15-20 points)	Level (5-15 points)	Level 0 (0 - 5 points)
User input	Correct or one incorrect prompt and captured input	Two mistakes in prompts and/or capture of input	Three or more missing essential elements for user input
Calculation	Correct or one mistake in calculation	Two mistakes in calculations	Three or more missing or significantly incorrect calculations
Application output	Correct or one mistake in output	Two mistakes in output data or format	Three or more missing or significantly incorrect output
Test Cases	Correct or one incorrect test case and/or test execution	Two incorrect or incomplete test cases and/or test execution	Three or more missing or significantly incorrect or incomplete test cases
Program documentation and style	Correct or one missing program comment, identifier, and/or screen capture	Two incorrect or incomplete documentation and/or style elements	Three or more missing or significantly incorrect documentation and/or style elements