Assignment 2

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

Write a Java program which:
(1) Prompts a user to enter student two numbers and symbol for operation such as '+', '-', '*', and '/' (use Scanner for input).
(2) Code uses nested if statement or switch to perform the operation on the two numbers
(3) If the provided symbol is valid, displays the input data along with the result of the calculation to the console. Otherwise displays error message

For example, for input: **7 8 +** the calculation is addition and the result will be 15

Note 1: To read a single character by Scanner, you can read in as string and then use the first character. For example: `scan.next().charAt(0)`
Note 2: Be careful you do not have integer division
Note 3: If user inputs invalid symbol, print an error message only

Test program:
A minimum of 5 test cases (one for each valid operation and one for invalid operation) 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
- Identifiers are written in upper CamelCase
- Class name starts with upper case letter and variables in lower case letter
- 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 4 test cases.

Here are a couple sample runs:
```
Enter two integer numbers separated by space: 7 6
Enter operation symbol (+, -, *, or /): +
Evaluation: 7 + 6 = 13.0
```

```
Enter two integer numbers separated by space: 5 6
Enter operation symbol (+, -, *, or /): (
Not valid operation symbol
```

Example test cases:

Input	Expected Output	Actual Output	Pass?
number 1 = 7 number 2 = 6 operation = +	Enter two integer numbers separated by space: 7 6 Enter operation symbol (+, -, *, or /): + Evaluation: 7 + 6 = 13.0	Enter two integer numbers separated by space: 7 6 Enter operation symbol (+, -, *, or /): + Evaluation: 7 + 6 = 13.0	Yes
Test Case 2			
Test Case 3			
Test Case 4			
Test Case 5			

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 2 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 almost correct prompts and captured input	Mistakes in prompts and/or capture of input	Missing or close to missing user input
Calculation	Correct or almost correct calculation	Mistakes in calculations	Missing or significantly incorrect calculation

Application output	Correct or almost correct output	Mistakes in output data or format	Missing or significantly incorrect output
Test Cases	Correct or almost correct test cases and test execution	Mistakes or incomplete test cases and execution	Missing or significantly incorrect test cases
Program documentation and style	Correct or almost correct program comments, identifiers, and screen captures	Mistakes or incomplete documentation and/or style	Missing or significantly incorrect documentation and/or style