

# DECISIONS, DECISIONS, DECISIONS

## THE GAS STATION PROGRAM

Your assignment is create a program that allows users to purchase gas and gives them the option of purchasing a car wash. Your gas station program must offer three grades of gas: BRONZE (\$1.31), SILVER (\$1.34), and GOLD (\$1.37).

Once users select the type of gas they wish to purchase, the user will need to enter the amount of litres in a text box. Users will then need to determine whether they want to purchase a car wash by clicking on one of two radio buttons (YES or NO). The price of a car wash is \$9.99 if the user purchases less than 25 litres of gas or \$7.99 if the user purchases 25 litres or more.

Once all the information has been entered and the user clicks the **CALCULATE** button, your program must calculate (and output) the subtotal, car wash total, HST and the grand total.

Your program output should look something like this:

The image displays two screenshots of a Windows application titled "Sunoco Gas Station".

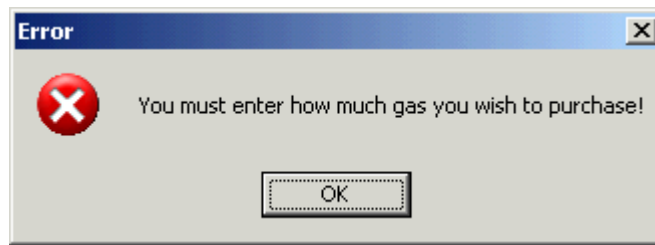
**Left Screenshot (Initial State):**

- Select grade:** Three radio buttons are present: Bronze (selected), Silver, and Gold.
- Price buttons:** Three buttons show prices: 1.31 (Bronze), 1.34 (Silver), and 1.37 (Gold).
- FUEL COST:** Text box contains "1.31".
- TOTAL LITRES:** Empty text box.
- Message:** "Save \$2 on a car wash when you purchase 25 litres or more!".
- Would you like a car wash today?:** Radio buttons for Yes and No are both unselected.
- Total Cost:** Four empty text boxes for SUBTOTAL, CAR WASH, HST, and TOTAL.
- Buttons:** CALCULATE, CLEAR, and EXIT.

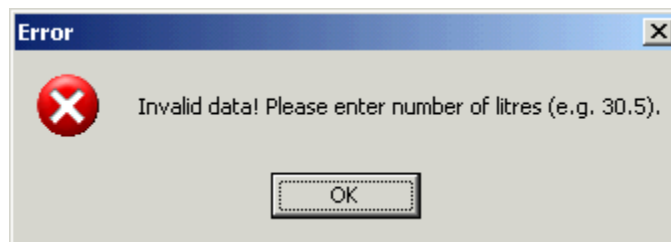
**Right Screenshot (After Calculation):**

- Select grade:** Bronze, Silver, and Gold (selected).
- Price buttons:** 1.31, 1.34, and 1.37.
- FUEL COST:** Text box contains "1.37".
- TOTAL LITRES:** Text box contains "25".
- Message:** "Save \$2 on a car wash when you purchase 25 litres or more!".
- Would you like a car wash today?:** Radio button for Yes is selected, No is unselected.
- Total Cost:** SUBTOTAL: \$34.25, CAR WASH: \$7.99, HST: \$1.04, TOTAL: \$43.28.
- Buttons:** CALCULATE, CLEAR, and EXIT.

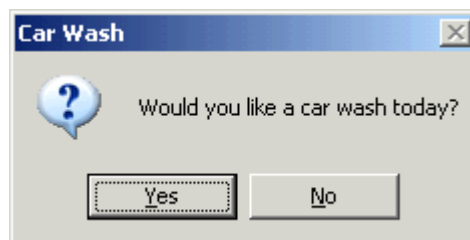
For your **CALCULATE** button, you will need to include a **Try-Catch** statement that handles the exception that would be thrown if the user enters something in the text box that cannot be converted into a double value. If the user does not enter any information in the text box, the following message should appear:



If, on the other hand, the user enters the number of litres but the amount the user enters cannot be converted to a double value, the following message should appear:



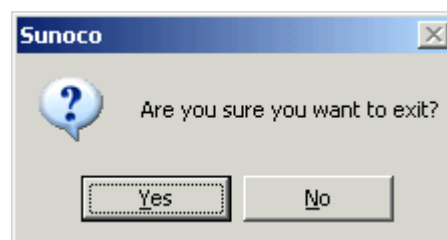
Once the user enters valid data into the text box, the program should check whether or not the user wants a car wash. If the user did not select either YES or NO, the following message should appear:



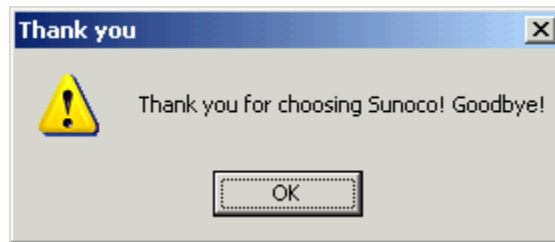
If the user clicks YES, the YES radio button should be clicked. Otherwise, the NO radio button should be clicked. The totals should then be outputted in the corresponding labels.

When the user clicks the CLEAR button, the totals should be cleared, the amount of litres should be cleared and the bronze radio button should be selected.

When the user clicks the EXIT button, the following message should be displayed:



If the user selects YES, the following message should be outputted before exiting the application:



Here's the pseudo code which should help you as you're writing the code for the **CALCULATE** button:

Store the fuel cost that the user selects

Try

    Store the amount of litres that the user enters in the text box

    If no car wash radio buttons are checked then

        Ask user if he/she wants a car wash

        If the user selects yes then

            Check the YES radio button

        Else

            Check the NO radio button

        End If

    End If

    If the user wants a car wash and purchased 25 or more litres of gas then

        Make car wash total equal 7.99

    Else if the user wants a car wash and purchased less than 25 litres of gas then

        Make car wash total equal 9.99

    Else

        Make car wash total equal 0.00

    End If

    Calculate subtotal, hst, and grand total

    Format and output totals to the corresponding labels

Catch the exception that is thrown if the user's input cannot be converted into a double value

    If the user enters nothing in the text box then

        Output message indicating that the user needs to enter amount of litres

    Else

        Output message indicating that the user has entered invalid data

    End If

    Set focus back to text box, select contents of the text box, and set back colour of the text box to red

Once you have completed the assignment, save the program in your COMPLETED ASSIGNMENTS folder.

# GAS STATION PROGRAM RUBRIC

NAME: \_\_\_\_\_

TOTAL: / 40

CATEGORY	CRITERIA	LEVEL 1 50 – 59%	LEVEL 2 60 – 69%	LEVEL 3 70 – 79%	LEVEL 4 80 – 100%	MARK
<b>Knowledge and Understanding</b>	Demonstrates an understanding of how to write a program(s) that uses decision structures and that handles exceptions	<ul style="list-style-type: none"> <li>Demonstrates a limited understanding of how to use decision structures and exception handling</li> </ul> <p style="text-align: center;"><b>5.0-5.9</b></p>	<ul style="list-style-type: none"> <li>Demonstrates some understanding of how to use decision structures and exception handling</li> </ul> <p style="text-align: center;"><b>6.0-6.9</b></p>	<ul style="list-style-type: none"> <li>Demonstrates considerable understanding of how to use decision structures and exception handling</li> </ul> <p style="text-align: center;"><b>7.0-7.9</b></p>	<ul style="list-style-type: none"> <li>Demonstrates thorough understanding of how to use decision structures and exception handling</li> </ul> <p style="text-align: center;"><b>8.0-10</b></p>	<b>/10</b>
<b>Thinking</b>	<p>The program meets all the required specifications</p> <p>Validates program to ensure the program produces correct results</p>	<ul style="list-style-type: none"> <li>Program meets a limited number of the required specifications</li> <li>Validates program with limited success</li> </ul> <p style="text-align: center;"><b>5.0-5.9</b></p>	<ul style="list-style-type: none"> <li>Program meets some of the required specifications</li> <li>Validates program with some success</li> </ul> <p style="text-align: center;"><b>6.0-6.9</b></p>	<ul style="list-style-type: none"> <li>Program meets most of the required specifications</li> <li>Validates program with considerable success</li> </ul> <p style="text-align: center;"><b>7.0-7.9</b></p>	<ul style="list-style-type: none"> <li>Program meets all of the required specifications</li> <li>Validates program with great success</li> </ul> <p style="text-align: center;"><b>8.0-10</b></p>	<b>/10</b>
<b>Communication</b>	Provides internal documentation that clearly explains program logic	<ul style="list-style-type: none"> <li>Documents program logic with limited success</li> </ul> <p style="text-align: center;"><b>5.0-5.9</b></p>	<ul style="list-style-type: none"> <li>Documents program logic with some success</li> </ul> <p style="text-align: center;"><b>6.0-6.9</b></p>	<ul style="list-style-type: none"> <li>Documents program logic with considerable success</li> </ul> <p style="text-align: center;"><b>7.0-7.9</b></p>	<ul style="list-style-type: none"> <li>Documents program logic with great success</li> </ul> <p style="text-align: center;"><b>8.0-10</b></p>	<b>/10</b>
<b>Application</b>	Effectively applies programming skills and knowledge of Visual Basic to create a program	<ul style="list-style-type: none"> <li>Applies programming knowledge and skills with limited success</li> </ul> <p style="text-align: center;"><b>5.0-5.9</b></p>	<ul style="list-style-type: none"> <li>Applies programming knowledge and skills with some success</li> </ul> <p style="text-align: center;"><b>6.0-6.9</b></p>	<ul style="list-style-type: none"> <li>Applies programming knowledge and skills with considerable success</li> </ul> <p style="text-align: center;"><b>7.0-7.9</b></p>	<ul style="list-style-type: none"> <li>Applies programming knowledge and skills with great success</li> </ul> <p style="text-align: center;"><b>8.0-10</b></p>	<b>/10</b>

**CURRICULUM EXPECTATIONS THAT ARE COVERED IN THIS ASSIGNMENT:**

- B2.1 Use a visual problem-solving model to plan the content of a program.
- B2.2 Use variables, expressions, and assignment statements to store and manipulate numbers and text in a program.
- B2.3 Write keyboard input and screen output statements that conform to program specifications.
- B2.4 Write a program that includes a decision structure for two or more choices.
- B3.1 Write clear and maintainable code using proper programming standards.
- B3.2 Write clear and maintainable internal documentation to a specific set of standards.
- B3.3 Use a tracing technique to understand program flow and to identify and correct logic and run-time errors in a computer program.
- B3.4 Demonstrate the ability to validate a computer program using test cases.