

Scalable Data Infrastructures - Code Exercise 03

Overview

This code exercise will give you a chance to work on your ability to use conditionals to determine the outcome of your application. You'll create a conditional that allows the user to select if he or she is a mobile or web degree student.

Instructions

It's always best to start with a problem analysis. While one is not required to be submitted for this code exercise, it would benefit you greatly to begin by examining the requirements and creating a bulleted list of the things you'll need to complete the code. You can then begin by creating a new solution with the following naming format: LastName_FirstName_CE03.

The first thing you'll do is ask the user if he or she is a student in the mobile degree program or the web degree program or if they are undecided. When creating your WriteLine, make sure you specifically indicate what the user must enter, such as Web or Mobile or Undecided. You may want to create a menu of choices and have the user enter 1 for Web or 2 for Mobile or 3 for Undecided. It is entirely up to you how you want them to select but you must include the three different choices and let the user know what they must enter.

You will then have a conditional that will determine the output of the code based on the user's input. So, if the user selects Web, one output will happen or else if the user selects Mobile, a different output will be sent to the console or if the user is undecided, a third option will be presented. What is output should give the user useful feedback in regards to their selection. You can decide what that output will be, but it must be meaningful to the user.

You should complete the assignment using a single conditional. That is, you should not have multiple "if" statements. Rather you should utilize "if" and "else if" to get full credit for the exercise.

Things to Consider

Look to your problem analysis to verify that you've included all the requirements of the problem. If you ask your instructor or a lab specialist for assistance on the code, he/she is going to ask to see your problem analysis first. If you didn't do that, we are going to tell you to complete that task before we will look at your code.

Make sure you're including comments at the top of the code to include your name, class and term, and the assignment name (Code Exercise 03) and that you have meaningful comments for each line of code in the project.

Finally, remember that you must compress the entire project folder for submission. Submitting only the Program.cs file or the .sln file will result in a 0 for the activity.

Rubric: Code Exercise 3							
Minimum Project Requirements							
These requirements must be satisfied before any points are awarded. Failing to meet these requirements will result in a zero (0) grade.							
1. Project must run when instructor compiles it. 2. The submission must be submitted in the proper format as defined in the FSO activity. 3. You will lose 5 points if the project does not follow the naming convention described in the activity's documentation.							
Topic	%	Excellent (100%)	Acceptable (80%)	Good (50%)	Fair (25%)	Poor (0%)	
Technical: Code							
Comments	10	Comments exist at the top of the code to include name, class and term, assignment, and date, and each line of the code is properly commented.	Missing the initial comments with name, class and term, assignment, and date, but the rest of the code is commented properly.	Missing up to four line comments, but some comments present.	Missing more than four comments	No comments in the code.	
Syntax	15	There are no syntax errors, including correct line and formatting according to the style taught.	There are no syntax errors, but the code does not follow the style taught.	Project code contains minor syntax errors but is easily fixed.	Project code contains more major syntax errors but are easily fixed.	Project code does not run.	
User Input	15	ReadLine is set up correctly with a corresponding WriteLine that contains descriptive text to indicate what the user must do.	ReadLine is set up correctly, but the text of the WriteLine is not descriptive.	ReadLine is set up correctly, but there is no WriteLine to indicate what the user must input.		ReadLine is not used or is not set up properly.	
Comparisons	25	All required comparisons are included and the proper comparison operators are used.	One required comparison is missing.	Missing more than one required comparison.	Incorrect use of comparison operators or comparisons do not affect the flow.	No comparisons are made in the project submitted.	
Conditional	25	A single conditional with multiple comparisons is used to determine the outcome of the application based on the user input.	Multiple conditional statements are used, but the application sends the correct output to the console.	Errors in the logic of the comparisons cause incorrect output.		No conditionals used in the application.	
Output	10	Output is meaningful to the user and correctly describes the outcome of the application.	Outputs exist but contain misspellings, punctuation or grammatical errors.	The outputs are not meaningful to the user.		No outputs are sent to the console.	