**Topic 5: Console App**

Top of Form

Bottom of Form

**Content**

**Setting up for Module 4 with Unit Tests**

[Setting up for Module 4 with Unit Tests](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286912_1)

https://youtu.be/J6pxDxheiyE

**Running Unit Tests-All Tests Passing**

[Running Unit Tests-All Tests Passing](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286912_1)

https://youtu.be/v6IpHLiuPzA

[**Console App: Decision w/Unit Test**](https://dmacc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_7286928_1&course_id=_102593_1&group_id=&mode=view)

[UnitTestMealSchedule.cs](https://dmacc.blackboard.com/bbcswebdav/pid-7286928-dt-content-rid-101466152_1/xid-101466152_1)

Deciding on where or what to eat can be difficult. Fans of the Big Bang Theory may know the character Sheldon prefers to stick to his schedule.

Write a program that prompts the user for the day (Monday-Thursday) and meal (lunch or dinner). Then prints the food choice. Create an new solution called Module4. It's okay to keep your class Program.cs so the Unit Tests will work.

|  |  |
| --- | --- |
| Lunch | * + Monday: VeggieBurger and Fries   + Tuesday: Chili and cornbread   + Wednesday: Pad Thai   + Thursday: Baked Potato |
| Dinner | * + Monday: Lasagna   + Tuesday: Pizza   + Wednesday: Soup and Salad   + Thursday: Spaghetti |

* + Write a method DecideMeal()
    - Accepts the day (Monday, Tuesday, Wednesday, Thursday) and meal (lunch, dinner)
    - Uses nested if statements that decides the day checks and the meal.
    - Returns the meal
    - NOTE: must be named DecideMeal for Unit Tests.
  + In Main()
    - Prompt user for day
    - Prompt user for meal
    - Call DecideMeal(), sending the necessary variables, storing the return value in a variable mealString
    - Print the day, time and the meal (the variable mealString)
  + Sample Manual Test:
  + Make sure to handle upper/lower case input for days. If the user misspells a day or meal time or inputs invalid input, have them eat ice cream!

|  |  |
| --- | --- |
| What day is it?  Monday  What time meal is it? (lunch/dinner)  lunch  Monday is veggieburger and fries for lunch. | What day is it?  thursday  What time meal is it? (lunch/dinner)  dinner  Thursday is spaghetti for dinner. |

* + Use the attached Unit Tests to test your code.
    - For the xUnit Unit tests to work as written, name your solution Module4 leave Program.cs and include public in front of class Program and DecideMeal method.
    - To the Module4 Solution, Add a project TestModule4 and include the attached Unit Tests, UnitTestMealSchedule. You can delete the UnitTest1.cs file.
    - Don't forget to Add Project Reference in Dependencies on the the Unit Test Project TestModule4. Right click on Dependencies--> Add Project Reference... then check Module4

Graphical user interface, text, application

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

For this project, zip your entire Solution so that both projects (xUnit and Module4) (See Course Success Toolkit for help). In your Program.cs file with the Academic Honesty Header included.

Make sure to review the rubric. This requires nested if statements.

This is 15 points.

Extra Credit: Add Friday, Saturday, Sunday and accompanying Unit Tests. (up to 5 points) Be sure to write a Comment (in the Comment Box) that you attempted Extra Credit.