Southern New Hampshire University

IT 511 Milestone One Guidelines and Rubric
The Ingredient Class

Overview: In your final project, you will create a program that will help you manage a collection of items. To complete this program, you will implement two classes: one for the main recipe item and one for the entire collection. If you decide to be more adventurous, you can make an additional class for the most important subcomponent of your main recipe item class.

Prompt: Your Ingredient class will model the details of individual ingredients in a recipe. Based on Stepping Stone Labs Two and Three, you will create an Ingredient class and give it the basic attributes: name, amount, unit of measure, and calories. Additionally, you will add code to validate the data type of the user input.

This Ingredient class will be modified for the submission of your final RecipeManager application; however, it should be functional code that accepts user input for each variable.

Specifically, the following **critical elements** of the final project must be addressed:

I. **Data Types:** Your Ingredient class should properly employ each of the following data types that meet the scenario’s requirements where necessary:
   A. Utilize numerical data types that represent quantitative values for variables and attributes in your class.
   B. Utilize strings that represent a sequence of characters needed as a value in your class.
   C. Utilize inline comments directed toward software engineers for the ongoing maintenance of your program that explain your choices of data types selected for your program.

II. **Algorithms and Control Structure:** Your final program should properly employ each of the following control structures as required or defined by the scenario where necessary:
   A. Utilize expressions or statements that carry out appropriate actions or that make appropriate changes to your program’s state as represented in your program’s variables.
   B. Employ the appropriate conditional control structures that enable choosing between options in your program.
   C. Utilize inline comments directed toward software engineers for the ongoing maintenance of your program that explain your choices of data types selected for your program.

Rubric

Guidelines for Submission: Your complete program should be submitted as a Java file of the project.

Critical Elements	Proficient (100%)	Needs Improvement (80%)	Not Evident (0%)	Value
Data Types: Numerical	Utilizes numerical data types that represent quantitative values for variables and attributes in the program, meeting the scenario’s requirements	Utilizes numerical data types that represent quantitative values for variables and attributes in the program, but use of data types is incomplete or illogical, contains inaccuracies, or lacks accordance with the scenario’s requirements	Does not utilize numerical data types that represent quantitative values for variables and attributes in the program	20
Data Types: Strings	Utilizes strings that represent a sequence of characters needed as a value in the program, meeting the scenario’s requirements	Utilizes strings that represent a sequence of characters needed as a value in the program, but use of strings is incomplete or illogical, contains inaccuracies, or lacks accordance with the scenario’s requirements	Does not utilize strings that represent a sequence of characters needed as a value in the program	20
Data Types: Inline Comments	Utilizes inline comments directed toward software engineers for the ongoing maintenance of the program that explain the choices of data types selected for the program	Utilizes inline comments that explain the choices of data types selected for the program but inline comments are incomplete or illogical, contain inaccuracies, or lack applicability toward software engineers for the ongoing maintenance of the program	Does not utilize inline comments that explain the choices of data types selected for the program	10
Algorithms and Control Structures: Expressions or Statements	Utilizes expressions or statements that carry out appropriate actions or that make appropriate changes to the program’s state as represented in the program’s variables and meet the scenario’s requirements	Utilizes expressions or statements that carry out actions or that make changes to the program’s state as represented in the program’s variables, but use of expressions or statements is incomplete or illogical, contains inaccuracies, or lacks accordance with the scenario’s requirements	Does not utilize expressions or statements that carry out actions or that make changes to the program’s state as represented in the program’s variables	20
Algorithms and Control Structures: Conditional Control Structures	Employs the appropriate conditional control structures, as the scenario defines, that enable choosing between options in the program	Employs the conditional control structures that enable choosing between options in the program, but use of conditional control structures is incomplete or illogical, contains inaccuracies, or lacks accordance with the scenario’s definition	Does not employ the conditional control structures that enable choosing between options in the program	20
Algorithms and Control Structures: Inline Comments	Utilizes inline comments directed toward software engineers for the ongoing maintenance of the program that explain how the use of algorithms and control structures appropriately addresses the scenario’s information management problem	Utilizes inline comments that explain how the use of algorithms and control structures addresses the scenario’s information management problem, but inline comments are incomplete or illogical, contain inaccuracies, or lack applicability toward software engineers for the ongoing maintenance of the program	Does not utilize inline comments that explain how the use of algorithms and control structures addresses the scenario’s information management problem	10
			Total	**100%**