

Overview: In this module, you have studied how to use conditionals and logical structures to create branches or "forks in the code." In Stepping Stone Lab Three, you will put this emerging knowledge into practice by developing a branching structure for the recipe manager program.

Be sure to review the Stepping Stone Three guidelines before beginning this lab. The completed code (.java file) from Stepping Stone Lab Two is included in this module to use as a reference for comparison of your work in Module Three as well as moving forward. This code serves a number of functions:

- It allows you to review your own submitted code.
- You may utilize it as a useful foundation for Stepping Stone Three as well as other assignments to come.
- Together with instructor feedback on your submitted code, you are building your own final project application.

Prompt: In Stepping Stone Three, you will write a short application that uses conditionals to create a "forked" branching structure for the recipe manager.

Whenever a program accepts user input, it is best practice to be sure the input is what you as the programmer expect. As you continue to develop the recipe manager, you will need to be able to validate user input and ensure that the user enters values that are valid. In this lab, you will write a short program that first, tests that the input is numerical, then checks that the number is within a specific range, and checks that the maximum number of cups of our main ingredient is 100 and that the minimum number of cups is 1.

This application uses the Scanner class to accept a number between 1 and 100 from the user. The Scanner class is useful for parsing primitive values, including numbers.

Specifically, you will create a branching structure that leads to the following output:

If the number entered is between 1 and 100 (inclusive), the application will display a message that says, "is a valid number of cups!".

However, if the number entered is not between 1 and 100 (inclusive), error messages will be displayed to inform the user that the entry does not fit the expected range: "is not a valid number of cups!" and "Please enter another number of cups between 1 and 100.".

Guidelines for Submission: This assignment should be submitted as a Java file.

Extending This Lab for Your Final Project

For your final project, adapt your Ingredient Java file to include **data-type validation** steps for each of the variables in the class:

- ingredientName (String)
- ingredientAmount (float)
- unitMeasurement (String)
- Number of calories (double)