Chapter 1: Introduction

Introduction

> **"What distinguishes a computer from a simple calculator is its ability to make decisions."**
>
> — from *Computer Architecture* by
>
> David A. Patterson and
>
> John L. Hennessy

> **"When you come to a fork in the road, take it."**
>
> — Yogi Berra

I'm going to try to move a little more slowly in this lesson. We'll still learn some new concepts, but we'll also reuse a lot of what we learned last time. That way, by the time we're done with this lesson, last lesson's material will be that much more familiar.

We're going to continue building our temperature conversion program. Last time we got it to the point where it accepted a Fahrenheit temperature and displayed its equivalent in Celsius and Kelvin.

In this lesson we'll add the ability to enter a temperature in any of the three scales and display it in the other two. In order to do this, we'll use one of the most useful features in any programming language: *decision-making*.

Let's go back to part of the program design from Lesson 5. I described more of what our program needs to do in steps 1, 2, and 9:

> 1. Ask the user for a temperature type (K for Kelvin, C for Celsius, F for Fahrenheit, and Q to quit).
>
> 2. Read the user's entry from the keyboard.
>
> . . .
>
> 9. Convert the value from the indicated temperature scale to the other two scales using the formulas from the assumptions.

We'll deal with the Q entry in the next lesson, when we set the program up to repeat its actions. For now we'll focus on the three temperature types.

To do conversions correctly, we'll have to decide which scale the user entered so we know which two others to convert to. In other words, if the user enters K for the temperature type, we need to convert to Celsius and Fahrenheit; if the user enters C, we need to convert to Kelvin and Fahrenheit; and if the user enters F, we need to convert to Celsius and Kelvin.

As you can see, these steps have a common form. Each checks to see if a particular condition is true, and if so it performs a specific set of actions. Every decision in a computer program takes this form. When the condition is true, the program takes one action. When the condition is false, the program takes a different action or no action at all.

This lesson is all about how decisions happen in Java. So let's get to it!