Chapter 5: Using else to Speed Processing Time

Using else to Speed Processing Time

In this lesson we've discussed about logical expressions and how to use them in a Java program to make decisions. Let me make just one more comment and suggested change.

The program we have works exactly as designed (so far . . . remember, we're not done yet!). But there's one more change that you can make that will improve its efficiency at run time.

Notice that every temperature type has to go through all three if statements in the main() method. Wouldn't it make more sense once we've processed a Fahrenheit temperature, for example, not to make it go through the checks for Celsius and Kelvin?

We can do that with the else part of the if structure. If we put the keyword else in front of each if after the first one, and then those if statements will execute only if all previous checks were false.

So if you look at my final version of the program later on in this chapter, you'll see the added else keywords. (I've highlighted them for you.) That way, if I enter a type of F, the first branch will process it and won't even check the others. But a K will fail the first two checks and be processed by the third.

The gain in efficiency is small in this case, but programmers usually use this technique when checking a list of values and doing different actions for each one.

You'll also see in the main() method that I've added a final else to display an error message if the temperature type is not one that I want to see. This is a common coding technique after a group of related if statements. It makes sense to have a final block that will execute if none of the expected conditions are met. We'll see more error handling like this in the next lesson.

Now try this game to see what you remember from this lesson. I won't keep track of your score—this is for fun.

Text equivalent start.

Instructions: Read the descriptions in the first column, and guess the code. Then read the second column for the answer.

Description	Answer
Angle brackets.	< >
Curly brackets.	{ }
A comparison operator that means "not equal to."	! =
A conditional operator that means "or."	||
An arithmetic operator called modulus or mod.	%

Description	Answer
This expression helps you speed up processing time.	else

Text equivalent stop.

How did you do? You'll see some of these symbols (though not the pictures) in this lesson's quiz.

Summary

Through this lesson, you made our program far more sophisticated than it was by adding decision-making elements. You used the if and else formats to give your statements branches—in other words, to allow for different conditions and exceptions. You wrote Boolean expressions, which return either true or false. And you used all this new information to make your program able to handle Celsius and Kelvin temperatures as well as Fahrenheit. Way to go!

Here are my final classes for this lesson:

Temperature Class

Hide answer

```
/**
 * Temperature converts temperature from Fahrenheit, Celsius,
 * or Kelvin scale to the other two scales.
 *
 * @author Merrill Hall
 * @version 2.0
 */
public class Temperature {
    private double degreesFahrenheit;  // Fahrenheit temperature
    private double degreesCelsius;  // Celsius temperature
    private double degreesKelvin;  // Kelvin temperature

    /**
     * This constructor for Temperature sets the temperature
     * values to the value from degrees, based on the type
     *
     * @param  type      temperature scale to use
     * @param  degrees   degrees Fahrenheit
     */
    public Temperature(String type, double degrees) {
        if (type.equalsIgnoreCase("C"))
            setDegreesCelsius(degrees);
        else if (type.equalsIgnoreCase("F"))
            setDegreesFahrenheit(degrees);
        else if (type.equalsIgnoreCase("K"))
            setDegreesKelvin(degrees);
    }

    /**
     * The setDegreesFahrenheit method sets the Fahrenheit temperature
     *
     * @param  degrees The Fahrenheit value to store
     */
    public void setDegreesFahrenheit(double degrees) {
        degreesFahrenheit = degrees; // set Fahrenheit value
        degreesCelsius    = (degreesFahrenheit - 32.0) * 5.0 / 9.0; // set Celsius
value
        degreesKelvin     = degreesCelsius + 273.15; // set Kelvin value
    }

    /**
     * The setDegreesCelsius method sets the Celsius temperature
```

```
     *
     * @param  degrees The Celsius value to store
     */
    public void setDegreesCelsius(double degrees) {
        degreesCelsius    = degrees;
        degreesFahrenheit = degreesCelsius * 9.0 / 5.0 + 32.0;
        degreesKelvin     = degreesCelsius + 273.15;
    }

    /**
     * The setDegreesKelvin method sets the Kelvin temperature
     *
     * @param  degrees The Kelvin value to store
     */
    public void setDegreesKelvin(double degrees) {
        degreesKelvin     = degrees;
        degreesCelsius    = degreesKelvin - 273.15;
        degreesFahrenheit = degreesCelsius * 9.0 / 5.0 + 32.0;
    }

    /**
     * getDegreesCelsius retrieves the Celsius temperature value
     *
     * @return a double value containing the Celsius temperature
     */
    public double getDegreesCelsius() {
        return degreesCelsius;
    }

    /**
     * getDegreesKelvin retrieves the Kelvin temperature value
     *
     * @return a double value containing the Kelvin temperature
     */
    public double getDegreesKelvin() {
        return degreesKelvin;
    }

    /**
     * getDegreesFahrenheit retrieves the Fahrenheit temperature value
     *
     * @return a double value containing the Fahrenheit temperature
```

```
     */
    public double getDegreesFahrenheit() {
        return degreesFahrenheit;
    }
}
```

TemperatureDriver Class

Hide answer

```
import java.util.Scanner;
/**
 * TemperatureDriver runs and tests the Temperature class.
 *
 * @author Merrill Hall
 * @version 2.0
 */
public class TemperatureDriver {
    /**
     * main() reads a temperature type and value, then
     * converts it to the other two temperature scales.
     */
    public static void main(String[] args) {
        double inputTemperature = 0.0;
        Scanner keyInput = new Scanner(System.in);
        Temperature t1;
        String temperatureType = "";

        System.out.print("Enter a temperature type (C for Celsius, " +
            "F for Fahrenheit, K for Kelvin): ");
        temperatureType = keyInput.next();
        System.out.print("Enter a temperature: ");
        inputTemperature = keyInput.nextDouble();
        t1 = new Temperature(temperatureType, inputTemperature);

        if (temperatureType.equalsIgnoreCase("F")) {
            System.out.println("You entered " + inputTemperature +
                " degrees Fahrenheit");
            System.out.println("which is " + t1.getDegreesCelsius() +
                " degrees Celsius");
            System.out.println("and " + t1.getDegreesKelvin() +
                " degrees Kelvin.");
        }
        else if (temperatureType.equalsIgnoreCase("C")) {
            System.out.println("You entered " + inputTemperature +
                " degrees Celsius");
            System.out.println("which is " + t1.getDegreesFahrenheit() +
                " degrees Fahrenheit");
            System.out.println("and " + t1.getDegreesKelvin() +
                " degrees Kelvin.");
        }
        else if (temperatureType.equalsIgnoreCase("K")) {
```

```
            System.out.println("You entered " + inputTemperature +
                " degrees Kelvin");
            System.out.println("which is " + t1.getDegreesCelsius() +
                " degrees Celsius");
            System.out.println("and " + t1.getDegreesFahrenheit() +
                " degrees Fahrenheit.");
        }
        else {
            System.out.println("You entered an invalid temperature type: " +
temperatureType);
            System.out.println("It must be \"C\", \"F\", or \"K\".  Please try the
program again.");
        }
    }
}
```

The image below shows what some successful runs look like. You can compare your results to mine and make sure things are working correctly.

Text equivalent start.

Terminal Window displaying example output:

Enter a temperature type (C for Celsius, F for Fahrenheit, K for Kelvin): c
Enter a temperature: 32
You entered 32.0 degrees Celsius
which is 89.6 degrees Fahrenheit
and 305.15 degrees Kelvin
Enter a temperature type (C for Celsius, F for Fahrenheit, K for Kelvin): f
Enter a temperature: 98.6
You entered 98.6 degrees Fahrenheit
which is 37.0 degrees Celsius
and 310.15 degrees Kelvin.
Enter a temperature type (C for Celsius, F for Fahrenheit, K for Kelvin): k
Enter a temperature: 310.15
You entered 310.15 degrees Kelvin
which is 37.0 degrees Celsius
and 98.6 degrees Fahrenheit.

Text equivalent stop.

In Lesson 7 we'll use our temperature conversion example one more time. I'll show you how to . . .

- Check for valid numeric values.

- Loop back to correct values that are in error.
- Make the program repeat actions using a loop.
- End the loops and program when the conditions are right.