

Virtual Lecture Notes (Part 3)

So far, the use of conditionals has only been demonstrated with numeric data. Java has issues with evaluating **Strings** in **boolean** expressions that will be dealt with shortly; however, there is one common situation that will be helpful to know about now.

There are many programming situations in which flow of control is affected by the entry of a single keyboard character. For example, what do each of these user prompts have in common?

- Please press Q to quit
- Would you like instructions? (Y or N):
- Select P to Play, S to Stop

Each of these prompts asks the user to enter a single keyboard character in order to effect some action. You can probably imagine a **boolean** expression within a program to evaluate whether the requested input had been provided and then one or more conditional statements that alter the flow of control accordingly.

The following simple program illustrates one way to handle a situation requiring the entry of a single keyboard character. First examine the source code line-by-line to understand the purpose of the program.

```
import java.util.Scanner;
public class SingleCharacterInput
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.print("Do you live in Florida (Y/N)? ");
        String liveInFlorida = in.next();
        char resident = liveInFlorida.charAt(0);

        boolean isResident = resident == 'Y';
        if(isResident)
            System.out.println("Status: Florida Resident");
        else
            System.out.println("Status: Non-Resident");
    }
}
```

Line <10> parses the user input from a **String** into a **char** primitive data type, therefore the **boolean** expression in Line <12> can use a single character with a relational operator. In turn, the **boolean** variable in Line <14> alters the flow of control within the **if-else** code. The following code, however, would not work because **liveInFlorida** is a **String**.

```
boolean isResident = liveInFlorida == "Y";
```

This technique will come in handy on the next programming assignment, so type the code into a BlueJ project and observe how it works.