Introduction to JAVA (Continued)

B001 — Integer Variables:

Sometimes in programming (often, actually) we want the program to "remember" a value for us, so we need to have a mechanism in place where we can temporarily store these values in memory. "Variables" satisfy this purpose for us perfectly.

When we store a variable in memory we have to tell the computer what "kind" of variable it is. The first common type of variable we will learn about is integer variables.

(Recall that an integer is a number that does not have a decimal place. 1, 6, 1243, and -6, are all integers. 2.0, -7.5, and 0.1 are not.)

To tell the computer what kind of variable we are using we have to "declare" it. The following is a "declaration" statement that declares an integer:

```
int age;
```

The int part of the statement tells the program that it is an integer variable. The age part of the statement tells the program that the variable age will be used to store an integer—in this case, the age of something or someone.

We next need to tell the computer what the age is. The simplest way to do this is to "assign" a value to age. This is done with an "assignment" statement:

```
age = 16;
```

An important thing about variables in java is that variable names are case-sensitive. This means that "age" is not the same as "Age" which is not the same as "AGE".

To output the variable to the screen we can still use out print() or println() statements. Try this program to see:

```
// The "B001" class.
import java.awt.*;
import hsa.Console;
public class B001
{
   static Console c;           // The output console
   public static void main (String[] args)
   {
      c = new Console ();
      int age;
      age = 15;
      c.println (age);
   } // main method
} // B001 class
```

Now this program is not very user-friendly. All it does is output "15" to the screen. Adapt the program so the output looks like this:

```
c.print ("Your age is ");
c.println (age);
```

That's a little nicer, but we can make it even more efficient by doing this:

```
c.print ("Your age is " + age);
```

Assignment:

Write a program that declares three variables A, B, and C. Set these variables to any integers you desire between 1 and 10, then print the value A + B × C.

Save as "B001.java".

B002 — Variable Reuse:

A huge benefit of variables is that they can be reused. For instance, in our previous program we set age to 15, then we printed the value of age. age can then be set to some other value to be used again.

Try the following test program:

```
// The "B002" class.
import java.awt.*;
import hsa.Console;
public class B002
{
   static Console c;           // The output console
   public static void main (String[] args)
   {
      c = new Console ();
      int age;
      age = 15;
      c.println ("Your age is " + age);
      age = 29;
      c.println ("Your age is now " + age);
   } // main method
} // B002 class
```

Assignment:

Rewrite program B001 so it reassigns new values to A, B, and C after they have been output with the original formula, then declares a new value D and outputs the value of A + B × C - D.

Be sure your output is "user-friendly" and written as efficiently as possible.

Save as "B002.java".

B003 — Real Variables:

Of course integers are not the only numbers we need to use in our programs, so we also need to have real numbers, or what java refers to as "double" variables.

Declaration of a double variable is similar to declaring an integer. We just substitute "double" for "int":

```
double height;
```

Assignment:

Enter the following test program:

```
// The "B003" class.
import java.awt.*;
import hsa.Console;
public class B003
{
   static Console c;           // The output console
   public static void main (String[] args)
   {
      c = new Console ();
      double height;
      height = 2.789;
      c.println ("The height is " + height + ".");
   } // main method
} // B003 class
```

Modify the program so that you also declare a variable named width. In your program, output the area of a rectangle and the area of a right-angled triangle given the values you assigned for width and height.

Be sure the output is "user-friendly" (in all cases from now on).

Save as "B003.java".

B004 — String Variables:

A string is a series of characters. String variables are declared and assigned just like the others, except "String" is used for the variable type and the variable must be enclosed in double quotes. Notice that "String" is capitalized, which is different than the others. The reason why will become apparent in future lessons when we deal with string manipulation.

```
String aString
aString = "Hello World!";
```

Write a program that has two variables: FName and LName. Assign your first name and last name to these variables, then print them in a user-friendly format on a single line.

Save as "B004.java".

B005 — Business Card with Variables:

Assignment:

Create a program that declares and assigns values to any variables that would be needed to create a business card. Be sure there is also a suitable variable for "Employee Number".

Print the business card neatly to the centre of the screen.

B006 — Variables with Math:

Assignment:

Create a program that declares and assigns values to any variables that would be needed to calculate the formula

$$Area\ (of\ a\ circle) = Pi \times r^2$$

Print the formula and the answer in a user-friendly format on the screen.

B007 — Boolean Variables:

Boolean variables are quite unique. They are not used to represent normal things like words or numbers; instead, they are used to represent one of two states—"true" or "false".

Boolean variables can be used to keep track of "status" kinds of things (called "flags"). Anytime you need to keep track of whether or not something happened (reach end of file yet?), or whether or not something is true (the name was entered correctly), you would use a boolean variable.

The declaration and assignment of boolean variables is just like the others, except we use the word "boolean".

```
boolean fileExists;
```

We will not be writing a program using a boolean variable at this time. Just keep it in mind for future programs.

B008 — Character Variables:

A character variable is declared as follows:

```
char c1;
```

We will not be writing a program using a character variable at this time. Just keep it in mind for future programs.