Reading Console Input

So far in this course all the data used by a program was defined inside the program. In practice, it is very useful…and necessary…to allow users to input some or all of the data for a program. One way to input data is to have a user type the data in from the keyboard, or console, and have a program read the data that was entered. To read input data from the console we use something called a **`Scanner`** object. At this point it's not important to understand what a **`Scanner`** object is…just what it does and how to use it. For now, just think of a `Scanner` object as a black box that knows how to read data from the console.

The following example illustrates how to use a `Scanner` to read an integer value from the console.

```
import java.util.Scanner;
public class ScannerDemo1
{
      public static void main( String [] args )
      {
            // Create a Scanner object
            Scanner input = new Scanner( System.in );

            // Prompt the user to enter data
            System.out.print( "Please enter an integer value: " );

            // Read the value entered
            int data = input.nextInt();

            // Display the value entered
            System.out.println( "You entered " + data );
      }
}
```

There are several things that should be noted in the above program. First, we must "import" the `Scanner` class. You will learn what importing means in another lecture, but for now think of it as telling the Java compiler where to find the `Scanner` black box. The import statement must be at the top of your source file and only comment lines can come before it. The second thing to note is that the program creates a `Scanner` object called `input`, and `input` will be used like a variable name to refer to the `Scanner` object. The name `System.in` is used to refer to the standard input device, which is the keyboard by default. The program then prompts the user to enter an integer value, then defines an integer variable called `data` that will be used to store the value entered. The `Scanner` method `nextInt()` is called to read an integer typed in from the keyboard and returns its value to the program. That's all there is to it. You may want to compile and run this code to get a feel for how it works.

Reading Console Input

The table below describes some of the frequently used `Scanner` methods.

Method	Function
`nextByte()`	Returns the value entered as a `byte` type
`nextShort()`	Returns the value entered as a `short` type.
`nextInt()`	Returns the value as an `int` type.
`nextLong()`	Returns the value entered as a `long` type.
`nextFloat()`	Returns the value entered as a `float` type.
`nextDouble()`	Returns the value entered as a `double` type.
`next()`	Returns the value entered as a `String` type. Reads the characters entered before a *whitespace* character. [1]
`nextLine()`	Returns a line of text (characters entered before an Enter key is pressed) and returns it as a `String` type.
`hasNext()`	Returns the `boolean` value `true` if there is more data to input and false otherwise. [2]

[1] A whitespace character is a blank space, a tab character, or a newline character.

[2] When reading console input a user signifies there is no more input data by pressing CTL-Z, and on Unix/Linux/Mac by pressing CTL-D.

Reading Console Input

Here's another sample program that demonstrates reading a series of data items from the keyboard. The program prompts the user to enter a set of integers and then calculates and displays the sum of the integers.

```
import java.util.Scanner;
public class ScannerDemo2
{
      public static void main( String [] args )
      {
            int data = 0;          // integer value to be input
            int total = 0;         // total sum of integers

            // create a Scanner object
            Scanner input = new Scanner( System.in );

            // prompt user to enter data
            System.out.println( "Enter 1 or more integers." );
            System.out.println( "To end input type: " );
            System.out.println( "CTL-Z (for Windows)" );
            System.out.println( "CTL-D (for Mac/Linux)" );

            // read data until user indicates done
            while ( input.hasNext() )
            {
                  data = input.nextInt();
                  total += data;
            }

            // Display the output
            System.out.println( "The sum is " + total );

      }
}
```