1.6 Casting and Ranges of Variables

User Input

`Scanner` allows us to get input from the user and incorporate it into our program. We'll always format our scanner like this (you'll learn later why):

```
import java.util.Scanner;    //import scanner class from Java library
public class Demo {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);       //in main method
    }
}
```

Scanner

When we use a `Scanner`, we usually read a String from a keyboard and store it as a variable.

We can think of variables as containers in memory. When we declare a variable, we tell the computer to allocate some space in memory for something. We tell it what the name of the thing will be, so it knows how to find it later, and what the type of the thing will be, so it knows how much space to set aside:

```
        String n;
        //This tells the computer to allocate space for a String and call
    that space ‘n’
```

Variables can change in programming, as they are just containers. When we first put a value in the container we initialize the variable. Later we can change the variable so it contains a different value. To either initialize or change the value of a variable we use the assignment operator, which is just an equals sign (=). The variable on the left of the equals sign is set to the value on the right of the equals sign.

Syntax

```
import java.util.Scanner;    //import scanner class from Java library
public class Demo {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);       //in main method
        String n;    //This tells the computer to allocate space for a String and call that space ‘n’
        n = scan.nextLine(); //Variable n now contains whatever the user types
        n = “John”; //Variable n now contains “John”
    }
}
```

When we want to print things out, we can combine literal Strings (things in quotation marks) and variables, for example:

```
System.out.println(“Hello ” + n + “ how are you?”);
```

If we want to do calculations, we need number input. To take number values as input from a `Scanner`, we need a different command:

```
Scanner scan = new Scanner(System.in);
int x = scan.nextInt(); //when a declaration and assignment
happens on one line, it is called initialization

double y = scan.nextDouble(); //you can do this on one line
or two; it does not matter.
```

For example, to create the variable "y", you can use the code above or you can write it:

```
double y;
y = scan.nextDouble();
```

Introduction to Casting

If you combine an **`int`** and a **`double`** with an operator `(+ - * / % )`, what is the result?

If you combine an **`int`** and a **`String`** with a ‘+’, what is the result?

If you combine a **`double`** and a **`String`** with a ‘+’ what is the result?

Promoting

In Java we call this promoting!

```
int + double = double

int + String = String

double + String = String
```

```
(int) 4.29      ⇒ 4
(double) 1      ⇒ 1.0
```

Note: `String` *is not a primitive type and cannot be cast. Java will convert* `int` *and* `double` *to strings when concatenated with a string.*

```
    “” + 4.892      ⇒ “4.892”
```