Lesson 4: Basic Class and Object – Scanner Class

Class and Object:

Class:

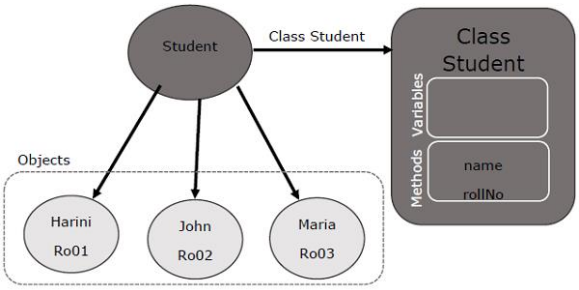
A class is the basic program unit in Java language. You can also call a class a module. A class can be used as a collection of functions (methods), data variable, and constants.

A class is also a template for creation of objects.

Object:

An object is the basic unit of a run-time data and program entity.

The relationship of class and object can be denoted as the following figure:



In this figure, Student is a class. It is written in a file named: Student.java and it is compiled to a bytecode file Student.class. When this class is loaded to memory, it is located in a global data pool (as the right hand side data area). And this class template can be used to create objects (instance) such as Harini, John or Maria. These individual student data are called object (instance). These exist at run-time.

It you print a object directly, you will see the hashcode (kind of like its memory address).

```
Student s = new Student();
System.out.println(s);
```

It may print out:
Student@35df45a3

Where Student is the class name and 35df45a3 is the hash code that is the address for the location of the memory where the object data are stored (or can be found.)

We will now consider how to input from the keyboard the three data types…. *int*, *double*, and *String* using Scanner class.

Inputting an integer:

Use the *nextInt* method to input an **integer** from the keyboard:

```
import java.io.*; //see "Imports necessary" on next page
import java.util.*;
public class Tester{
    public static void main( String args[] ){
        Scanner kbReader = new Scanner(System.in); //see "Mysterious
        //objects" on next page
        System.out.print("Enter your integer here. "); //enter 3001
        int i = kbReader.nextInt( );
        System.out.println(3 * i); //prints 9003
    }
}
```

Inputting a *double*:

Use the *nextDouble* method to input a ***double*** from the keyboard:

```
import java.io.*;
import java.util.*;
public class Tester{
    public static void main( String args[] ){
        Scanner kbReader = new Scanner(System.in);
        System.out.print("Enter your decimal number here. "); //1000.5
        double d = kbReader.nextDouble( );
        System.out.println( 3 * d ); //prints 3001.5
    }
}
```

Inputting a *String*:

Use the *next* method to input a ***String*** from the keyboard:

```
import java.io.*;
import java.util.*;
public class Tester{
    public static void main( String args[] ){
        Scanner kbReader = new Scanner(System.in);
        System.out.print("Enter your String here. "); //Enter One Two
        String s = kbReader.next( ); //inputs up to first white space
        System.out.println( "This is the first part of the String,… " + s);
        s = kbReader.next( );
        System.out.println( "This is the next part of the String,… " + s);
    }
}
```

Output would be as shown below:

```
Enter your String here. One Two
This is first part of the String,... One
This is next part of the String,... Two
```

Multiple inputs:

In a similar way *nextInt( )* and *nextDouble( )* can be used multiple times to parse data

input from the keyboard. For example, if **34 88 192 18** is input from the keyboard, then *nextInt( )* can be applied four times to access these four integers separated by white space.

Inputting an entire line of text:

Inputting a ***String*** (it could contain spaces) from the keyboard using *nextLine( )*:

```
import java.io.*;
import java.util.*;
public class Tester{
        public static void main( String args[] ){
                Scanner kbReader = new Scanner(System.in);
                System.out.print("Enter your String here. "); //Enter One Two
                String s= kbReader.nextLine( );
                System.out.println( "This is my string,… " + s);
        }
}
```

Output would be as shown below:

```
Enter your String here. One Two
This is my string,... One Two
```

Imports necessary:

We must **import** two classes,*….java.io.* and java.util.** that provide methods for inputting integers, *double*s, and *String*s. See Appendix I for more on the meaning of "importing".

Mysterious objects:

In the above three examples we used the following code:

```
Scanner kbReader = new Scanner(System.in);
```

It simply creates the keyboard reader **object** (we arbitrarily named it *kbReader*) that provides access to the *nextInt( ), nextDouble( ), next( ),* and *nextLine( )* methods. For now just accept the necessity of all this…it will all be explained later.

The *Scanner* class used here to create our keyboard reader object only applies to1.5.0_xx or higher versions of Java. For older versions, see Appendix M for an alternate way to obtain keyboard input.

An anomaly:

Using a **single** *Scanner* object, the methods *nextInt( ), nextDouble( ), next( ),* and *nextLine( )* may be used in **any sequence** with the following exception:

It is not permissible to follow *nextInt( )* or *nextDouble( )* with *nextLine( )*. If it is necessary to do this, then a new *Scanner* object must be constructed for use with *nextLine( )* and any subsequent inputs.