CS175

Introduction

In all our discussion, the values we have been using were all fixed by the programmer. This self-paced tutorial covers way to read input by the user from the keyboard using the java.util.Scanner class.

The **Java Scanner** class breaks the input into tokens using a delimiter that is whitespace by default. It provides many methods to read and parse various primitive values. Java Scanner class is widely used to parse text for string and primitive types using regular expression. As we have already discussed, classes have methods, so it is to the java Scanner class.

Commonly used methods of Scanner class

There is a list of commonly used Scanner class methods:

Method	Description
public String next()	it returns the next token from the scanner.
public String nextLine()	it moves the scanner position to the next line and returns the value as a string.
public byte nextByte()	it scans the next token as a byte.
public short nextShort()	it scans the next token as a short value.
public int nextInt()	it scans the next token as an int value.
public long nextLong()	it scans the next token as a long value.
public float nextFloat()	it scans the next token as a float value.
public double nextDouble()	it scans the next token as a double value.

To use these methods, a programmer has to incorporate the Scanner class into the program.

Java Scanner Example to get input from console

Let's see the simple example of the Java Scanner class which reads the int, string and double value as an input:

```
import java.util.Scanner;  //The class has methods to read inputs from users
class StudentRegister{  //this is your class
 public static void main(String args[]){  //main class
```

```
   Scanner sc=new Scanner(System.in);  //creates an object sc. In this case everything is standardized except for the name of an object.

   System.out.println("Enter your rollno");  //Asking the value from user
   int rollno=sc.nextInt();  //Reads the value typed by the user of type int
   System.out.println("Enter your name");  //Asking the value from user
   String name=sc.next();  //Reads the value typed by the user of type string
   System.out.println("Enter your Program of study");  //Asking the value from user
   double prog=sc.nextDouble();  //Reads the value typed by the user of type double
   System.out.println("Rollno:"+rollno+" name:"+name+" Program:"+prog);
   sc.close();  //explicitly call close() methods to free up resources
 }
}
```

Scanner object (in this **sc** object) uses input's nextInt, next(), nextDouble methods to obtain valus from the user at the keyboard. At each request the program waits for the user to type the value and press the Enter key to submit the number to the program.

Our program assumes that the user enters a valid value. If not, a runtime logic error will occur and the program will terminate. In the one of the upcoming modules is Exception Handling that presents a deeper Look on how to make your programs more robust by enabling them to handle such errors. This is also known as making your program *fault tolerant*.

REFERENCE:
(How to Program) Paul Deitel, Harvey Deitel-Java How to Program, Early Objects-Pearson Education (2015) **chapter 2, Section 2.5.6 Obtaining an int as Input from the User**

Review Questions

Exercise 01 (Leap year)

A leap year is a year with 366 days that has a 29th February in its calendar. Years whose division by 4 equals an integer are leap years except for years that are evenly divisible by 100 unless they are also evenly divisible by 400. Write a program that asks for a year and report on whether it is a leap year or not. Modify this code so that the program keeps asking for years, and compute its leap year property until the user input −1.

Exercise 02 (Displaying triangles)

Write a program that asks for an integer n, and write on the output a triangle as illustrated for the following example (with *n* = 5):

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```