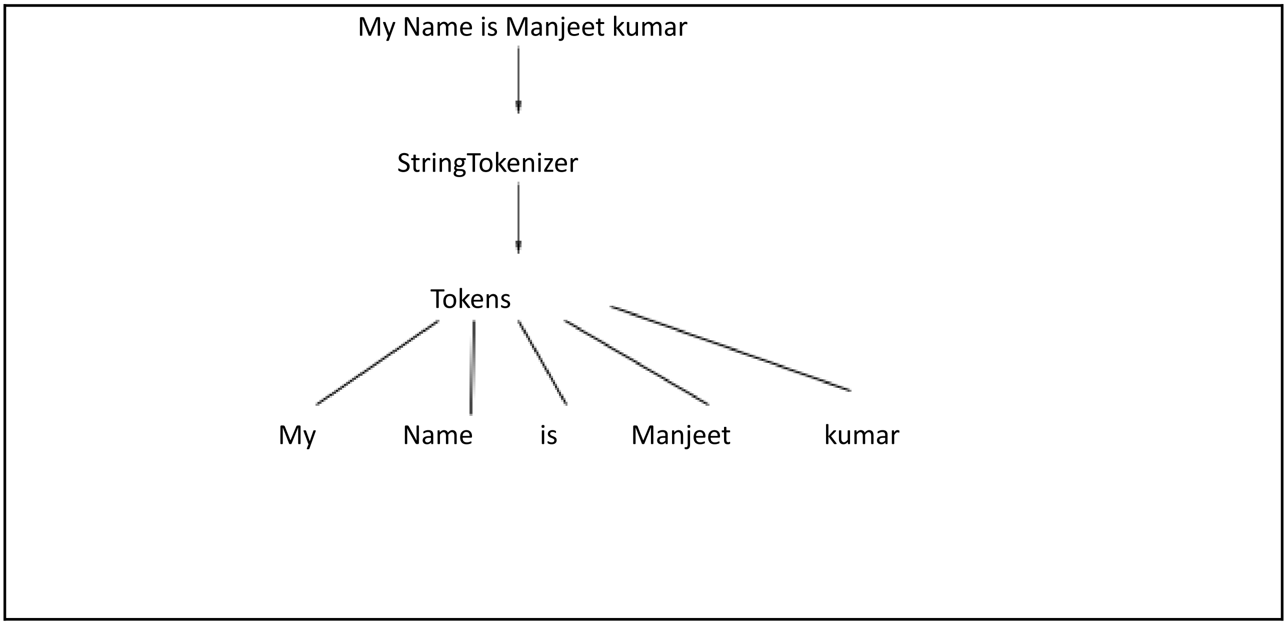
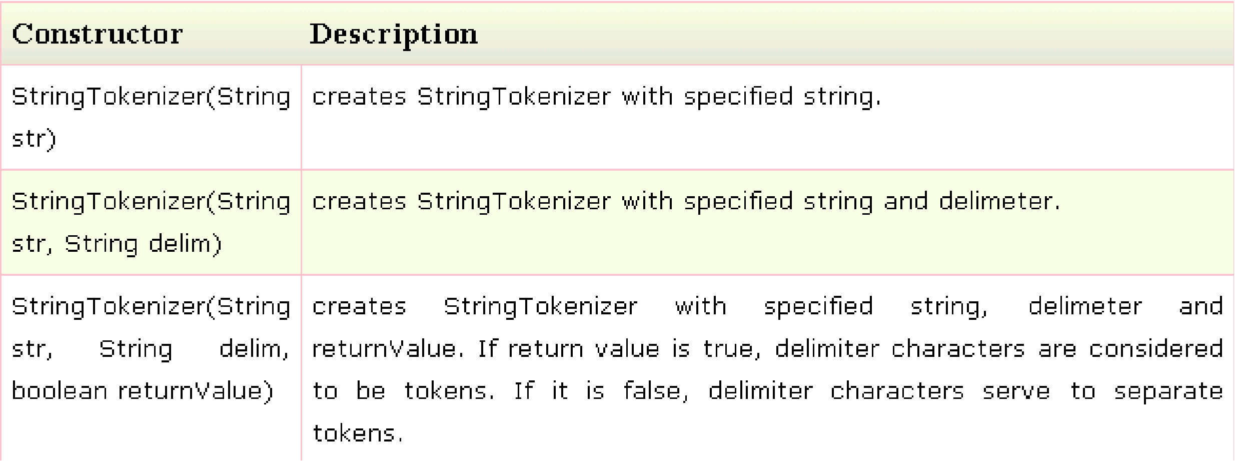
StringTokenizer in Java

A StringTokenizer class is a class present in the **java. util** package and it is used to break a String into tokens. In other words, we can split a sentence into its words and perform various operations like counting the number of tokens or breaking a sentence into tokens.



Constructors of StringTokenizer class

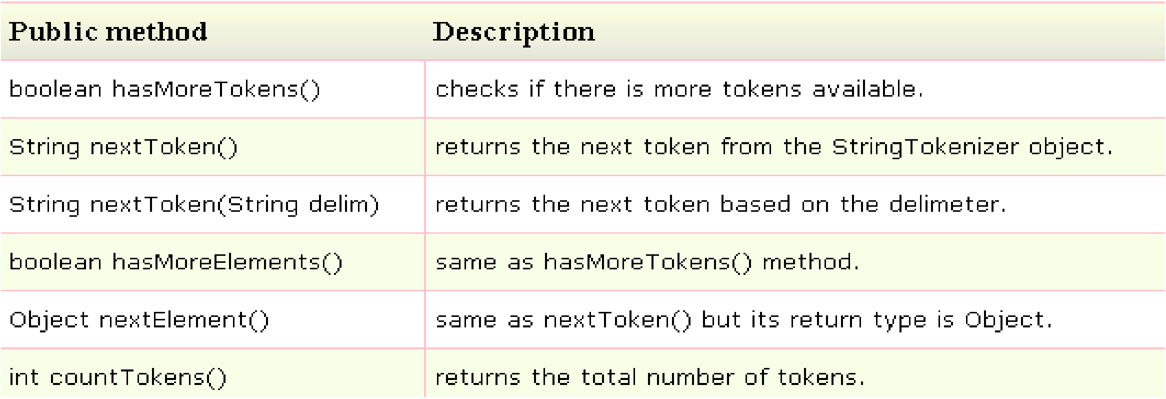
There are 3 constructors defined in the StringTokenizer class.



Constructor	Description
StringTokenizer(String str)	creates StringTokenizer with specified string.
StringTokenizer(String str, String delim)	creates StringTokenizer with specified string and delimeter.
StringTokenizer(String str, String delim, boolean returnValue)	creates StringTokenizer with specified string, delimeter and returnValue. If return value is true, delimiter characters are considered to be tokens. If it is false, delimiter characters serve to separate tokens.

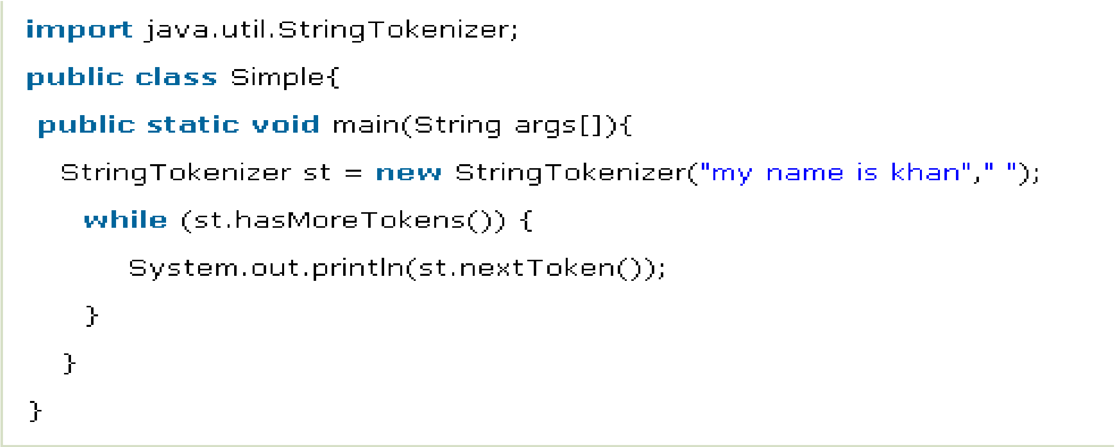
Methods of StringTokenizer class

The 6 useful methods of StringTokenizer class are as follows:

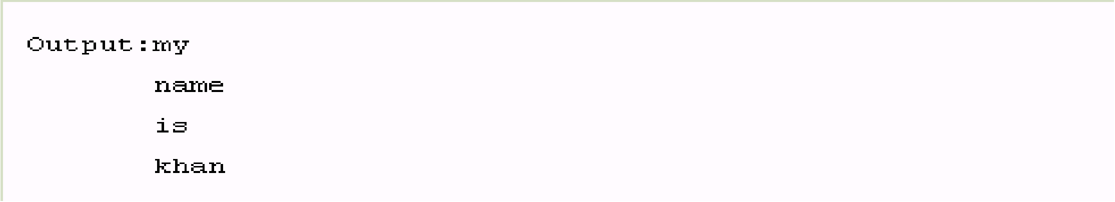


Public method	Description
boolean hasMoreTokens()	checks if there is more tokens available.
String nextToken()	returns the next token from the StringTokenizer object.
String nextToken(String delim)	returns the next token based on the delimeter.
boolean hasMoreElements()	same as hasMoreTokens() method.
Object nextElement()	same as nextToken() but its return type is Object.
int countTokens()	returns the total number of tokens.

1. **example of StringTokenizer class that tokenizes a string "my name is khan" on the basis of whitespace.**

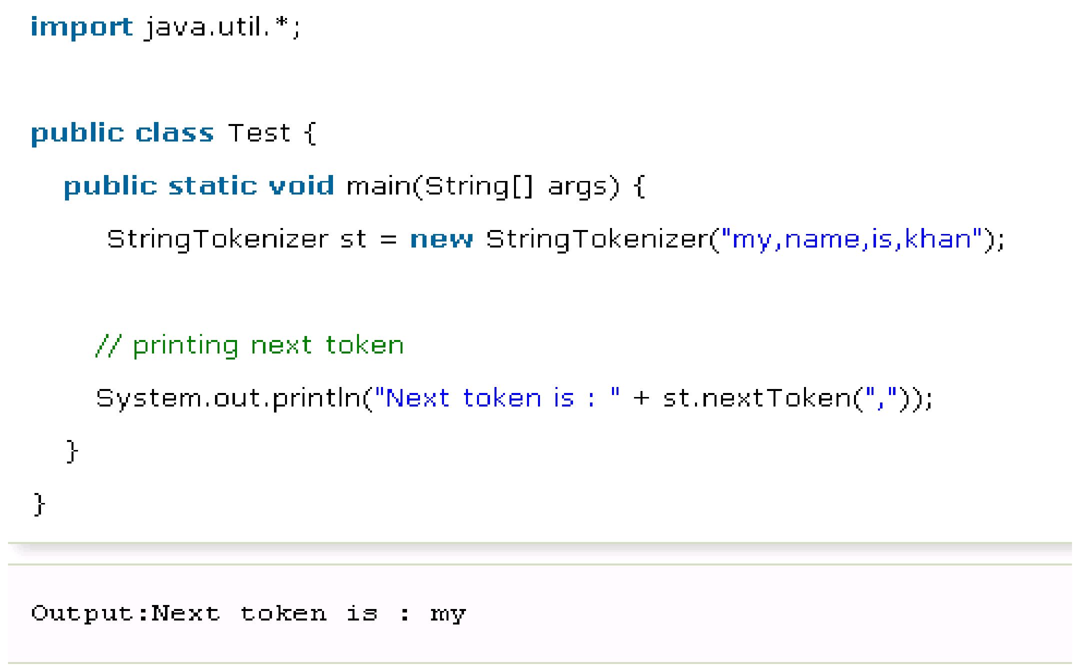


```
import java.util.StringTokenizer;
public class Simple{
 public static void main(String args[]){
   StringTokenizer st = new StringTokenizer("my name is khan"," ");
     while (st.hasMoreTokens()) {
         System.out.println(st.nextToken());
     }
   }
}
```



```
Output:my
       name
       is
       khan
```

2. **Example of nextToken(String delim) method of StringTokenizer class**



```
import java.util.*;

public class Test {
   public static void main(String[] args) {
       StringTokenizer st = new StringTokenizer("my,name,is,khan");

      // printing next token
      System.out.println("Next token is : " + st.nextToken(","));
   }
}
```

```
Output:Next token is : my
```

```
import java.util.*;

class demo{

 public static void main(String args[])

{

  StringTokenizer st = new StringTokenizer("my:= java; name:= Manjeet; is:= khan;","=;");

    while (st.hasMoreTokens()) {

                            String key=st.nextToken();

                            String val=st.nextToken();

        System.out.println(key+"\t"+val);

}

}
```

o/p:

```
my:                    Java
```

```
name:          Manjeet

is:            Khan
```

```
StringTokenizer str_arr= new StringTokenizer("Lets; practice; Java; StringTokenizer",";",false);



        while(str_arr.hasMoreElements())

            System.out.println(""+str_arr.nextToken());



        // Counting the tokens

        int count = str_arr.countTokens();

        System.out.println("Total number of Tokens: " + count);







// Print the tokens

        for (int i = 0; i < count; i++)

            System.out.println("token at [" + i + "] : "

                               + str_arr.nextToken());
```

```
import java.util.*;
class demo{
 public static void main(String args[])
{
Date dt=new Date();
System.out.println("System Date:"+dt);

Calendar c=Calendar.getInstance();
            //int d=;
System.out.println("Tody's Date :"+c.get(Calendar.DATE));
System.out.println("Tody's Month :"+c.get(Calendar.MONTH));
System.out.println("Tody's Year :"+c.get(Calendar.YEAR));
System.out.println("System Date:");
System.out.println(c.get(Calendar.MONTH)+"/"+c.get(Calendar.DATE)+"/"+c.get(Calendar.YEAR));

System.out.println(c.get(Calendar.HOUR)+":"+c.get(Calendar.MINUTE)+":"+c.get(Calendar.SECOND));

GregorianCalendar gc=new GregorianCalendar();

System.out.println(gc.get(Calendar.MONTH)+"/"+gc.get(Calendar.DATE)+"/"+gc.get(Calendar.YEAR));

int x=gc.get(Calendar.YEAR);

if(gc.isLeapYear(x))
System.out.println("Leap Year");
else
System.out.println("Not a Leap Year");
}
```