Object-Oriented Programming (CS F213)

Module II: Arrays and Strings in Java

CS F213 RL 8.5: StringTokenizer class in Java

BITS Pilani

Dr. Pankaj Vyas
Department of Computer Science, BITS-Pilani, Pilani Campus

CS F213 RL 8.5 : Topics

- StringTokenizer class in Java

StringTokenizer class

innovate achieve lead

- Used for Parsing a Formatted Input String and defined in java.util package
- Parsing : Division of text into set of discrete parts known as tokens
- Token convey a semantic meaning and represents a block of text
- Individual tokens are separated by Delimiters
- Delimiters can be specified by a delimiter String and each character in delimiter String is treated as a separate delimiter
- A provision is provided by which you can treat individual delimiters as tokens also
- Space , Tab, newline and carriage return <<Default Set of Delimiters>>

Example

String str = "CS F213, 1683, Object-Oriented Programming, Lecture-Section 1; MWF, 9, 6163, NAB";

If white space character is taken as delimiter

Number of Tokens = 11

Individual Tokens

{CS} {F213,} {1683,} {Object-Oriented} {Programming,} {Lecture-Section} {1;} {MWF,} {9,} {6163,} {NAB}

If Only Comma (,) is taken as delimiter:

Number of Tokens = 7

Individual Tokens

{CS F213} {1683} {Object-Oriented Programming} {Lecture-Section 1; MWF} {9} {6163} {NAB}

If Only Semicolon (;) is taken as delimiter

Number of Tokens = 2

Individual Tokens

{CS F213, 1683, Object-Oriented Programming, Lecture-Section 1} { MWF, 9, 6163, NAB}

If Both Comma(,) and Semicolon (;) are taken as delimiters:

Number of Tokens = 8

Individual Tokens

{CS F213} { 1683} { Object-Oriented Programming} { Lecture-Section 1} { MWF} { 9} { 6163} { NAB}

StringTokenizer : Constructors

innovate achieve lead

- StringTokenizer(String str)

1. << str> String to be tokenized.
2. No delimiter is specified so white space character will be used as delimiter.
3. Delimiters will not be considered as tokens

- StringTokenizer(String str, String delimiters)

1. << delimiters>> specify a delimiter String. Single or multiple characters can be specified as delimiters.
2. If the <<delimiters>> String is ":,;" then colon (:), comma (,) and semicolon(;) are separately used as delimiters
3. Each character in the <<delimiters>> String is separately treated as a delimiters.
4. By Default Delimiters will not be considered as tokens

- StringTokennizer(String str, String delimiters, boolean delimAsToken)

1. First Two parameters are same as previous constructor.
2. Third parameter delimAsToken indicates whether delimiters are to be taken as tokens or not.
3. Delimiters will be considered as tokens if delimAsToken is true otherwise Delimiters will not be considered as tokens

StringTokenizer : Importatnt Methods

- int countTokens() → Counts the number of tokens in StringTokenizer based upon delimiters
- boolean hasMoreTokens() / boolean hasMoreElements()
  - Helpful in parsing a String.
  - Returns true if there are one or more tokens left otherwise false.
- String nextToken()
  - Returns the next token in String form
  - Used in conjunction with hasMoreTokens() method
- Object nextElement() → Same as nextToken() but returns next Token in Object Form not in String Form

StringTokenizer: Example

innovate achieve lead

```
String str = "CS F213, 1683, Object-Oriented Programming, Lecture-Section 1; MWF, 9, 6163, NAB";

StringTokenizer strZ = new StringTokenizer(str); // Default Delimiter is white space

System.out.println("Number of Tokens:" + strZ.countTokens());

while(strZ.hasMoreTokens())
        System.out.println( strZ.nextToken());
```

Number of Tokens:11
CS
F213,
1683,
Object-Oriented
Programming,
Lecture-Section
1;
MWF,
9,
6163,
NAB

StringTokenizer: Example ….


```
String str = "CS F213, 1683, Object-Oriented Programming, Lecture-Section 1; MWF, 9, 6163, NAB";

StringTokenizer strZ = new StringTokenizer(str, ",");        // Comma (,) as Delimter

System.out.println("Number of Tokens:" + strZ.countTokens());

while(strZ.hasMoreTokens())
            System.out.println( strZ.nextToken());
```

Number of Tokens:7
CS F213
1683
Object-Oriented Programming
Lecture-Section 1; MWF
9
6163
NAB

StringTokenizer: Example ….


```
String str = "CS F213, 1683, Object-Oriented Programming, Lecture-Section 1; MWF, 9, 6163, NAB";

StringTokenizer strZ = new StringTokenizer(str, ",;");       // Comma (,) and Semi-colon(;)as Delimters

System.out.println("Number of Tokens:" + strZ.countTokens());

while(strZ.hasMoreTokens())
            System.out.println( strZ.nextToken());
```

Number of Tokens:8
CS F213
1683
Object-Oriented Programming
Lecture-Section 1
MWF
9
6163
NAB

StringTokenizer: Example ….


```
String str = "CS F213, 1683, Object-Oriented Programming, Lecture-Section 1; MWF, 9, 6163, NAB";

StringTokenizer strZ = new StringTokenizer(str, ”abc”);    // Characters ‘a’ , ‘b’ and ‘c’ as Delimiters

System.out.println("Number of Tokens:" + strZ.countTokens());

while(strZ.hasMoreTokens())
        System.out.println( strZ.nextToken());
```

Number of Tokens:6
CS F213, 1683, O
je
t-Oriented Progr
mming, Le
ture-Se
tion 1; MWF, 9, 6163, NAB

StringTokenizer: Example ….


```
String str = "CS F213, 1683, Object-Oriented Programming, Lecture-Section 1; MWF, 9, 6163, NAB";

StringTokenizer strZ = new StringTokenizer(str, ”abc”,true);
// Characters ‘a’ , ‘b’ and ‘c’ as Delimiters. Individual delimiters are also considered as tokens

System.out.println("Number of Tokens:" + strZ.countTokens());

while(strZ.hasMoreTokens())
        System.out.println( strZ.nextToken());
```

Number of Tokens:11
CS F213, 1683, O
b
je
c
t-Oriented Progr
a
mming, Le
c
ture-Se
c
tion 1; MWF, 9, 6163, NAB

Thank You