Library.mkr Page 729 Thursday, November 6, 1997 10:16 AM

flag is `false`, delimiter characters serve to separate tokens. A token is a maximal sequence of consecutive characters that are not delimiters. If the flag is `true`, delimiter characters are considered to be tokens. A token is either one delimiter character or a maximal sequence of consecutive characters that are not delimiters.

Important Methods

`StringTokenizer( String str )`

Constructs a string tokenizer for the specified string. The tokenizer uses the default delimiter set, which is `" \t\n\r"`: the space character, the tab character, the newline character, and the carriage-return character.

`StringTokenizer( String str, String delim )`

Constructs a string tokenizer for the specified string. The characters in the `delim` argument are the delimiters for separating tokens.

`StringTokenizer( String str, String delim, boolean returnTokens )`

Constructs a string tokenizer for the specified string. The characters in the `delim` argument are the delimiters for separating tokens. If the `returnTokens` flag is `true`, then the delimiter characters are also returned as tokens. Each delimiter is returned as a string of length one. If the flag is `false`, the delimiter characters are skipped and serve only as separators between tokens.

`boolean hasMoreTokens( )`

Returns `true` if and only if there are more tokens available from this tokenizer's string.

`String nextToken( )`

Returns the next token from this string tokenizer. It throws a `NoSuchElementException` if there are no more tokens in this tokenizer's string.

`String nextToken( String delim )`

Returns the next token from this string tokenizer using `delim` as the delimiter set. The new delimiter set remains the default after this call. Throws a `NoSuchElementException` if there are no more tokens in this tokenizer's string.

`int countTokens( )`

Returns the number of tokens remaining in the string using the current delimiter set.