

JAVA.LANG.BYTE.PARSEBYTE METHOD

http://www.tutorialspoint.com/java/lang/byte_parsebyte.htm

Copyright © tutorialspoint.com

Description

The **java.lang.Byte.parseByte** parses the string argument as a signed decimal byte. The characters in the string must all be decimal digits, except that the first character may be an ASCII minus sign '-' '\u002D' to indicate a negative value or an ASCII plus sign '+' '\u002B' to indicate a positive value.

The resulting byte value is returned, exactly as if the argument and the radix 10 were given as arguments to the `parseByte` *java.lang.String, int* method.

Declaration

Following is the declaration for **java.lang.Byte.parseByte** method

```
public static byte parseByte(String s) throws NumberFormatException
```

Parameters

- **s** - a String containing the byte representation to be parsed

Return Value

This method returns the byte value represented by the argument in decimal.

Exception

- **NumberFormatException** - if the string does not contain a parsable byte.

Example

The following example shows the usage of `lang.Byte.parseByte` method.

```
package com.tutorialspoint;

import java.lang.*;

public class ByteDemo {

    public static void main(String[] args) {

        // create 2 byte primitives bt1, bt2
        byte bt1, bt2;

        // create and assign values to String's s1, s2
        String s1 = "+123";
        String s2 = "-123";

        /**
         * static method is called using class name.
         * assign parseByte result on s1, s2 to bt1, bt2
         */
        bt1 = Byte.parseByte(s1);
        bt2 = Byte.parseByte(s2);

        String str1 = "Parse byte value of " + s1 + " is " + bt1;
        String str2 = "Parse byte value of " + s2 + " is " + bt2;

        // print bt1, bt2 values
        System.out.println( str1 );
        System.out.println( str2 );
    }
}
```

```
}
```

Let us compile and run the above program, this will produce the following result:

```
Parse byte value of +123 is 123  
Parse byte value of -123 is -123
```

```
Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js
```