JAVA.LANG.CHARACTER.VALUEOF METHOD

Description

The **java.lang.Character.valueOf***charc* returns a Character instance representing the specified char value. If a new Character instance is not required, this method should generally be used in preference to the constructor Character*char*, as this method is likely to yield significantly better space and time performance by caching frequently requested values.

This method will always cache values in the range '\u0000' to '\u007F', inclusive, and may cache other values outside of this range.

Declaration

Following is the declaration for **java.lang.Character.valueOf** method

```
public static Character valueOf(char c)
```

Parameters

- **c** - a char value

Return Value

This method returns a Character instance representing c.

Exception

- **NA**

Example

The following example shows the usage of lang.Character.valueOf method.

```
package com.tutorialspoint;

import java.lang.*;

public class CharacterDemo {

   public static void main(String[] args) {

      // create 2 Character objects c1, c2
      Character c1, c2;

      // create 2 char primitives and assign values
      char ch1 = 'i';
      char ch2 = 65;

      // assign Character values of ch1, ch2 to c1, c2
      c1 = Character.valueOf(ch1);
      c2 = Character.valueOf(ch2);

      String str1 = "Character value of " + ch1 + " is " + c1;
      String str2 = "Character value of " + ch2 + " is " + c2;

      // print c1, c2 values
      System.out.println( str1 );
      System.out.println( str2 );
   }
}
```

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

```
Character value of i is i
Character value of A is A
```