

## G- Fact 28 (Widening Primitive Conversion -Java)

Here is a small code snippet given. Try to Guess the output

```
public class Test
{
    public static void main(String[] args)
    {
        System.out.print("Y" + "O");
        System.out.print('L' + 'O');
    }
}
```

[Run on IDE](#)

*At first glance, we expect “YOLO” to be printed.*

### Actual Output:

“YO155”.

### Explanation:

When we use double quotes, the text is treated as a string and “YO” is printed, but when we use single quotes, the characters ‘L’ and ‘O’ are converted to int. This is called widening primitive conversion. After conversion to integer, the numbers are added ( ‘L’ is 76 and ‘O’ is 79) and 155 is printed.

Now try to guess the output of following:

```
public class Test
{
    public static void main(String[] args)
    {
        System.out.print("Y" + "O");
        System.out.print('L');
        System.out.print('O');
    }
}
```

[Run on IDE](#)

Output: YOLO

**Explanation:** This will now print “YOLO” instead of “YO7679”.

It is because the widening primitive conversion happens only when '+' operator is present.

Widening primitive conversion is applied to convert either or both operands as specified by the following rules. The result of adding Java chars, shorts or bytes is an **int**:

- If either operand is of type double, the other is converted to double.
- Otherwise, if either operand is of type float, the other is converted to float.
- Otherwise, if either operand is of type long, the other is converted to long.
- Otherwise, **both operands are converted to type int**

Reference: <http://docs.oracle.com/javase/specs/jls/se7/html/jls-5.html#jls-5.6.2>

This article is contributed by **Anurag Rai**. If you like GeeksforGeeks and would like to contribute, you can also write an article and mail your article to [contribute@geeksforgeeks.org](mailto:contribute@geeksforgeeks.org). See your article appearing on the GeeksforGeeks main page and help other Geeks.

**[Company Wise Coding Practice](#)**  
**[Topic Wise Coding Practice](#)**

---

Viewed using [Just Read](#)