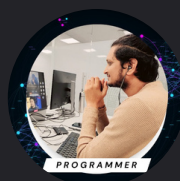


# Challenging Javascript Logic Quizzes



JS



Varun Sharma



```
console.log(018 - 015);  
console.log("018" - "015");
```

### ***Explanation:***

1/ An octal number is a number in JavaScript that has a leading zero. 018 is regarded as a decimal number, nevertheless, because it is an incorrect octal number. 015 is 13 in octal notation.  $018 - 015 = 18 - 13 = 5$  as a result.

2/ The second `console.log("018" - "015");` will result in **3**. When subtracting strings that can be parsed as numbers, JavaScript will convert them to numbers first.

```
console.log(3 > 2 > 1);
```

### ***Explanation:***

In first  $(3 > 2)$ , comparison evaluates to true because 3 is greater than 2. Now, true is implicitly converted to a number for the second comparison.

In JavaScript, true is converted to 1. So, the comparison effectively becomes  $1 > 1$ , which evaluates to **false**.



```
console.log(('b' + 'a' + + 'a' + 'a').toLowerCase());
```

### ***Explanation:***

This will log "**banana**". Here's why:

- 'b' + 'a' is 'ba'.
- + 'a' is NaN because 'a' cannot be converted to a number.
- 'ba' + NaN + 'a' results in 'baNaNNa'.
- Calling .toLowerCase() on 'baNaNNa' gives 'banana'.





```
const numbers = [33, 2, 8];  
numbers.sort();  
console.log(numbers[1])
```

### ***Explanation:***

This will log **33**. The `sort()` method sorts the array elements as strings by default. Hence, the array becomes `[2, 33, 8]`.



# Drop Your Views In Comments



# Follow For More Post Like This