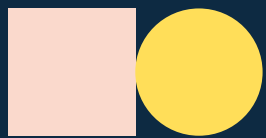


SightSoft Solutions Presents...



sightsoft.org



Complete JAVASCRIPT CHEATSHEET

#JavaScript Conditionals | Part 2

@SightSoft Solutions



one

sightsoft.org

if Statement

if Statement

```
const isMailSent = true;

if (isMailSent) {
  console.log('Mail sent to recipient');
}
```

two

sightsoft.org

Ternary Operator

Ternary Operator

```
var x=1;
```

```
// => true
```

```
result = (x == 1) ? true : false;
```

Operators - 1

Operators

```
true || false;           // true
10 > 5 || 10 > 20;        // true
false || false;          // false
10 > 100 || 10 > 20;      // false
```

Logical Operator &&

```
true && true;             // true
1 > 2 && 2 > 1;            // false
true && false;             // false
4 === 4 && 3 > 1;          // true
```

Logical Operator !

```
let lateToWork = true;
let oppositeValue = !lateToWork;

// => false
console.log(oppositeValue);
```

Operators - 2

Comparison Operators

```
1 > 3           // false
3 > 1           // true
250 >= 250      // true
1 === 1         // true
1 === 2         // false
1 === '1'       // false
```

Nullish coalescing operator ??

```
null ?? 'I win';           // 'I win'
undefined ?? 'Me too';      // 'Me too'

false ?? 'I lose'          // false
0 ?? 'I lose again'        // 0
'' ?? 'Damn it'            // ''
```

sightsoft.org

else if

else if

```
const size = 10;

if (size > 100) {
  console.log('Big');
} else if (size > 20) {
  console.log('Medium');
} else if (size > 4) {
  console.log('Small');
} else {
  console.log('Tiny');
}

// Print: Small
```

sightsoft.org

switch Statement

switch Statement

```
const food = 'salad';

switch (food) {
  case 'oyster':
    console.log('The taste of the sea');
    break;
  case 'pizza':
    console.log('A delicious pie');
    break;
  default:
    console.log('Enjoy your meal');
}
```

sightsoft.org

== VS ===

== VS ===

```
0 == false    // true
0 === false   // false, different type
1 == "1"      // true,  automatic type conversion
1 === "1"     // false, different type
null == undefined // true
null === undefined // false
'0' == false   // true
'0' === false  // false
```

The == just check the value, === check both the value and the type.