Lesson 72 JS True & False

In JavaScript, a boolean value is one that can either be **TRUE** or **FALSE**. If you need to know “yes” or “no” about something, then you would want to use the **boolean** function. It sounds extremely simple, but **booleans** are used all the time in JavaScript programming, and they are extremely useful.

Example :

```
var YES = true;

var NO = false;


if(YES)

{

   alert("This code block will be executed");

}


if(NO)

{

   alert("This code block will not be executed");

}
```

The comparison expressions return **boolean** values to indicate whether the comparison is **true** or **false**. For example, the following expressions return boolean values :

```
var a = 10, b = 20;
```

```
var result = 1 > 2; // false

result = a < b; // true

result = a > b; // false

result = a + 20 > b + 5; // true
```

JavaScript provides the **Boolean()** function that converts other types to a boolean type. The value specified as the first parameter will be converted to a boolean value. The **Boolean()** will return true for any non-empty, non-zero, object, or array.

```
var a = 10, b = 20;

var b1 = Boolean('Hello'); // true

var b2 = Boolean('h'); // true

var b3 = Boolean(10); // true

var b4 = Boolean([]); // true

var b5 = Boolean(a + b); // true
```

```
var b1 = Boolean(''); // false

var b2 = Boolean(0); // false

var b3 = Boolean(null); // false
```

A ternary operator evaluates a condition and executes a block of code based on the condition :

Let's write a program to determine if a student passed or failed in the exam based on marks obtained.

```
// program to check pass or fail

let marks = prompt('Enter your marks :');

// check the condition
let result = (marks >= 40) ? 'pass' : 'fail';

console.log(`You ${result} the exam.`);
```

Output :

Enter your marks: 78

You pass the exam.