```
a -= b; // subtraction and reassignment shorthand
a *= b; // multiplication and reassignment shorthand
a /= b; // division and reassignment shorthand
a %= b; // modulo and reassignment shorthand
a ++; // increment by one
a --; // decrement by one
Infinity; // result of 1/0
-Infinity; // result of -1/0
Nan; // Not a Number, result of 0/0

// ---------- BOOLEAN ----------

c = true;
d = false;

// ---------- STRINGS ----------
    // single ('') or double ("") quotes are both accepted

e = 'this is a string';
f = "this is also a string";

// STRING METHODS

// STRING CONCATENATION
    // + for string concatentation
    // works for diff data types besides strings

i = "Hello " + "World"; // "Hello World"
j = "Hello " + 1 + " World"; // "Hello 1 World"
k = "Hello " + ["World ", 100]; // "Hello World 100"

// STRING SLICING
    // .charAt()
    // .substring()

"This is a string".charAt(0); // returns "T"
"Hello world!".substr(0,5); // returns "Hello"

// STRING LENGTH
    // .length

"Hello".length; // returns 5

// ---------- OTHERS ----------

l = null; // indicates deliberate non-value
m = undefined; // indicates a value is currently absent but intended to be filled later'

// false, null, undefined, NaN, 0, "" evaluate to false
// everything else evaluates to true

// ---------- LOGICAL OPERATORS ----------

g = !true; // false
h = !false; // true
1 === 1; // COMPLETE EQUALITY of value and type
1 !== 1; // COMPLETE INEQUALITY of value and type
1 == "1"; // PARTIAL EQUALITY of value
null != undefined; // PARTIAL EQUALITY of value

// COMPARISON OPERATOR

1 < 10; // true
1 > 10; // false
2 <= 2; // true
2 >= 2; // true
```

Data structures

- array
- object

```
// ---------- ARRAY ----------
    // Javascript arrays allow storing multiple values of different data types together with []

var myArray = ["Hello", 45, true];
```