

**ECMAScript 6**

**JS**

# DATA TYPES



JavaScript distinguishes  
between **numbers**, **strings**,  
and `true` or `false` values  
known as **Booleans**.



1

1

NUMBERS

0.75

NO QUOTES



1

NUMBERS

0.75

NO QUOTES

2

1

NUMBERS

0.75

NO QUOTES

2

STRINGS

'Hi Ivy!'

ENCLOSED IN QUOTES  
WHICH CAN BE SINGLE  
OR DOUBLE QUOTES,  
BUT MUST MATCH



1

NUMBERS

0.75

NO QUOTES

2

STRINGS

‘Hi Ivy!’

ENCLOSED IN QUOTES  
WHICH CAN BE SINGLE  
OR DOUBLE QUOTES,  
BUT MUST MATCH

3



1

NUMBERS

0.75

NO QUOTES

2

STRINGS

‘Hi Ivy!’

ENCLOSED IN QUOTES  
WHICH CAN BE SINGLE  
OR DOUBLE QUOTES,  
BUT MUST MATCH

3

BOOLEAN

true

EITHER TRUE OR FALSE



# Definir Variáveis

## Numbers



```
var anoAtual; // definir variavel  
    let dataNascimento, idade, x , y ;  
    let outroAno = 2000;  
  
    anoAtual = 2018;  
    idade = 25;
```



# operadores

```
//usar operadores numéricos
```

```
dataNascimento = anoAtual - idade;
```

```
x = 15;
```

```
y = 7;
```

```
msg = x + y ;// 22 - adicao
```

```
msg = x - y ;// 8 - subtracao
```

```
msg = x * y ;// 105 - mult
```

```
msg = x / y ;// 2.1428.... - div
```

```
msg = x % y ; // 1 - modulo, resto da divisao
```

```
// incrementar ou decrementar
```

```
x = x + 1;
```

```
x += 1;
```

```
x += 4;
```

```
x ++;
```

```
x --;
```

```
// ordem das operacoes
```

```
2 + 3 * 2; // 8
```

```
(2 + 3) * 2; // 10
```

# strings




```
/* TIPO DE DADOS - STRINGS */  
let nome = "João";  
let apelido = 'Goncalves';  
  
let nomeCompleto = "Sr. " + nome + "<br /> " + apelido;  
  
// ES6 - template Literal, multiline string (backsticks)  
let outroNome = ` Sr. ${nome} <br /> ${apelido} `;
```

# Strings



```
msg = apelido.length;  
msg = apelido.toUpperCase( );  
msg = apelido.toLowerCase( );  
msg = apelido.charAt(2);  
msg = apelido.indexOf( 'n' );  
...  
// ver referencias
```

# Estruturas de Decisão - IF



```
// Estruturas de decisão - IF/ELSE

//operadores de comparacao, > , < , >= ,<= , == , !=
let saldo = 300;

if (saldo > 500)
    msg = "Comprar uma prenda....";

if (saldo > 500) {
    //sera executado se a condicao for verdadeira
    msg = "Comprar uma prenda....";
} else {
    // so sera executado se condicao for falsa
    msg = "Comprar uma prendinha....";
}
```

# IF / ELSE



```
if (saldo > 750) {  
    //sera executado se a condicao for verdadeira  
    msg = "Comprar uma prenda....";  
} else if (saldo > 500) {  
    // senao, se .....  
    msg = "Comprar uma prendinha....";  
} else {  
    msg = 'vai mas é trabalhar.....';  
}
```

# and & or



```
saldo = 400;
let mes = 'janeiro';

// combinar condicoes com E ( && ), e OU ( || )

if ( (saldo>500) && (mes=='janeiro') ){
    // com &&, ambas as condicoes tem que ser verdadeiras
    msg = 'O saldo é maior que 500, e estamos em Janeiro';
} else {
    msg = 'Ou o saldo não é maior que 500 ou não estamos em Janeiro';
}

if ( (saldo>500) || (mes=='janeiros') ){
    // com ||, basta uma condicao que ser verdadeiras
    msg = 'O saldo é maior que 500 ou estamos em Janeiro';
} else {
    msg = 'Ou o saldo não é maior que 500 e não estamos em Janeiro';
}
```



# switch

```
// SWITCH
//let lingua = 'fr';

let lingua = prompt('qual a lingua (pt,en,fr)');

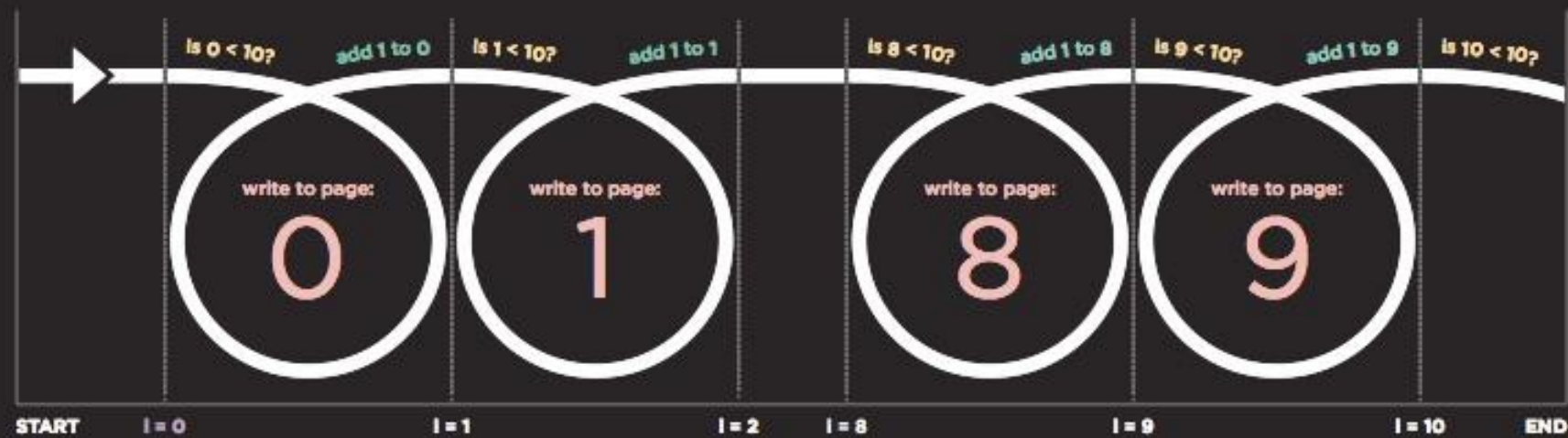
switch (lingua) {
  case 'pt' : msg = 'B0m Dia';
             break;
  case 'en' : msg = 'Good Morning';
             break;
  case 'fr' : msg = 'Bonjour';
             break;
  default: msg = 'Nao conheco essa lingua...';
}
```

# Exercícios:

- Crie um algoritmo que pergunte (prompt) as variáveis **nome** e **apelido**, e escreva no documento, o nome seguido do apelido.
- Com a ajuda do método **prompt** pergunte por dois números ao utilizador, apresente no documento o resultado da soma, subtracção, multiplicação e divisão. *(usar casting: **Number(prompt('pergunta'))**, pra converter Strings em Numbers)*
- Pergunte pelo raio de uma circunferência, apresente no documento o valor do seu raio, e da sua área, tomando como valor para  $\pi=3.14$  (ou usar a constante **Math.PI**).
- Perguntar por dois números, e por um operador numérico (+, -, \*, /), apresentar o resultado consoante o operador introduzido.

# LOOPING

```
for (var i = 0; i < 10; i++) {  
  document.write(i);  
}
```



The first time the loop is run, the variable `i` (the counter) is assigned a value of zero.

Every time the loop is run, the condition is checked. Is the variable `i` less than 10?

Then the code inside the loop (the statements between the curly brackets) is run.

The variable `i` can be used inside the loop. Here it is used to write a number to the page.

When the statements have finished, the variable `i` is incremented by 1.

When the condition is no longer true, the loop ends. The script moves to the next line of code.

# Repetição FOR

```
/* ESTRUTURAS DE REPTICA0, FOR, While, DO While */
```

```
for ( let i=0 ; i < 10 ; i++ ){  
    msg += i + "<br />";  
}
```

```
let i=10;  
for ( i ; i > 0 ; i-=2 ){  
    msg += i + "<br />";  
}
```

# while



```
let i = 20;

while ( i < 10){
    msg += i + "<br />";
    i++;
}

do {
    msg += i + "<br />";
    i++;
} while (i < 10);
```

# Exercícios:

1 - pergunte por dois numeros e apresente:

- a) a soma de todos os numeros intermédios, indicando no final quantos numeros foram somados
- b) apresenta a lista de numeros pares entre dois numeros inseridos
- c) apresenta a lista de numeros impares entre dois numeros inseridos por ordem decrescente

2 - Apresente os resultados de uma "tabuada" de multiplicação de um numero qualquer pedido ao utilizador.

3 - Jogo da adivinha:

- a) crie um numero inteiro aleatorio entre 0(zero) e vinte(20)(utilize o metodo `Math.random()` e `Math.round()`)
- b) Pergunte ao utilizador para adivinhar esse numero
- c) enquanto não acertar no numero devolva informação se o numero inserido é inferior ou superior ao numero a adivinhar
- d) Após acertar no valor, indique qual o numero, e quantas tentativas utilizou para adivinhar o numero.

# ARRAYS



An **array** is a special type of variable. It doesn't just store one value; it stores a list of values.





```
colors = [ ];
```



```
colors = ['pink'];
```



```
colors = ['pink', 'yellow'];
```



```
colors = ['pink', 'yellow', 'green'];
```



```
colors = ['pink', 'yellow', 'green'];
```

```
colors[ ];
```



```
colors = ['pink', 'yellow', 'green'];
```

```
colors[0];
```



```
colors = ['pink', 'yellow', 'green'];
```

```
colors[1];
```



```
colors = ['pink', 'yellow', 'green'];
```

```
colors[2];
```





Exemplos...





```
/* dados Complexos - ARRAYS (listas) */  
let formandos = [];  
formandos[0] = "Pedro";  
formandos[1] = "Ricardo";  
formandos[2] = "José";  
  
let formandas = ["Susana", "Carla", "Lena"];  
  
let outros = new Array(123, 'jose', true);
```



```
// propriedades e métodos
formandas.push( 'Margarida' );
formandas.unshift( 'Joana' );
formandas.shift( );
formandas.pop( );

let turma = formandas.concat(formandos);

turma.sort( );
turma.reverse( );

for ( let i=0 ; i < turma.length ; i++){
    //msg += `formando ${i+1}: ${turma[i]} <br />`;
    msg += "Aluno: "+ turma[i] + "<br />";
}
```

# Exercícios

- 1 - peça através do método prompt nomes de alunos ,guarde esses nomes numa array, apresente os mesmos ordenados por ordem crescente (pergunte primeiro pelo numero de alunos).
- 2 - Crie um script que pergunte pelo nome de um aluno, numero de disciplinas que tem, por cada disciplina pergunta pela nota que obteve, no final apresente em formato HTML legível, nome do aluno, lista de disciplinas e notas respectivas, assim média final.
- 3 - ao exercício anterior acrescente se o aluno ficou aprovado ou reprovado caso tenha tido uma média inferior ou superior a 10.
- 4 (fim de semana) , ao jogo da adivinha, acrescentar todas as tentativas erradas na pagina.