



INSTITUTO FEDERAL
Rio Grande do Sul

Desenvolvimento de Web II

Aula 03

Professora: Andrenizia A. Eluan da Rosa

E-mail: andrenizia.rosa@erechim.ifrs.edu.br





JavaScript: Sintaxe, variáveis, constantes e tipos de dados



Variáveis

- Em JavaScript podemos declarar variáveis de duas formas atualmente, com **let** ou **var**, sendo let a forma recomendada.
- **let** permite declarar variáveis que existem apenas no escopo no qual desejamos utilizá-las. Esse comportamento é diferente quando utilizamos var, que cria variáveis globais e que podem ser acessadas em qualquer local no arquivo ou função onde foram declaradas.

Diferença entre **var** e **let**



INSTITUTO FEDERAL
Rio Grande do Sul

A principal diferença são as regras de escopo. Variáveis declaradas pela palavra-chave **var** têm escopo para o **corpo da função imediata** (daí o escopo da função) enquanto variáveis **let** têm escopo para o **bloco delimitador imediato denotado por {}** (daí o escopo do bloco).



```
function exemplo() {  
  if (true) {  
    var x = 1; // Variável var com escopo de função  
    let y = 2; // Variável let com escopo de bloco  
  }  
}
```

```
console.log(x); // 1 - Acessível pois "x" é global dentro  
da função  
console.log(y); // Erro! "y" não é acessível fora do bloco  
"if"  
}  
exemplo();
```



Redeclaração: **var** permitirá que você declare novamente a mesma variável no mesmo escopo enquanto **let** gera um **SyntaxError**.

```
var foo = "foo1";  
var foo = "foo2"; //sem problemas, 'foo1' é trocada por 'foo2'.
```

```
let bar = "bar1";  
let bar = "bar2"; //erro, bar já declarada
```

Variáveis

- É importante saber diferenciar entre o comportamento de **let** e **var**.
- Supondo que desejamos incluir em um script o **cálculo da média de preços de alguns produtos**, dada uma determinada condição. Considere que ao fazer isso o programador nomeie e variável **media** dentro do **if** como **total** por engano, assim como no código a seguir:

```
1  var precos = [ 1200.8, 345.8, 2543.98 ];
2
3  var total = // Total dos valores produtos no array precos
4  ...
5  var incluirMedia = true;
6  ...
7  if(incluirMedia) {
8      var total = total/precos.length;
9
10     // Continuo a lógica
11 }
```

Uma vez que declaramos ambas as variáveis com **var**, tendo elas o mesmo identificador, **total**, a lógica de execução do script estará errada porque para o interpretador ambas são a mesma variável.

Dessa forma, o valor atribuído à variável **total** utilizada dentro do **if** sobrescreverá aquele dado a ela em sua declaração, algumas linhas acima.



Sintaxe

```
1 | let | var [ identificador ] [ = valor | expressão ]
```

- **identificador:** nome da variável
- **valor:** valor inicial da variável, que pode ser o resultado de uma expressão (opcional)

OBS: As variáveis declaradas com **let** dentro de blocos **if** ou **for** não podem ser acessadas de fora destes blocos.

Tipos de dados



INSTITUTO FEDERAL
Rio Grande do Sul

No javascript não existe uma tipagem explícita de dados na declaração de variáveis;

Apesar de não existir uma tipagem explícita de dados no Javascript, a linguagem **interpreta alguns tipos de dados**, divididos em três categorias:

Tipos de dados Primitivos;

Number, string, boolean

Tipos de dados especiais (undefined e Null**);**

Tipos de dados Compostos (por referência).

Date, Array, i

Exemplos



INSTITUTO FEDERAL
Rio Grande do Sul

```
1 let tipoNumber = 12
2 let tipoString1 = 'Texto entre aspas simples'
3 let tipoString2 = "Texto entre aspas duplas"
4 let tipoNull = null
5 let tipoUndefined = undefined
6 let tipoObject = { nome : 'José da Silva', idade : 18 }
7 let tipoBoolean = true || false
```

Constantes

- O valor de uma constante não pode ser modificado. No JavaScript declaramos constantes com a palavra reservada **const**

Sintaxe:

```
1 | const [ identificador ] = [ valor | expressão ]
```

- **identificador:** nome da constante
- **valor:** valor inicial que pode ser o resultado de uma expressão (opcional)

Exemplo

Podemos atribuir a uma constante o resultado de uma expressão:

```
1 | const peso = 80.5
2 | const altura = 1.80
3 | const imc = peso / (altura * altura)
4 |
5 | console.log(imc)
```

É comum usarmos **constantes** para evitar que um valor seja sobrescrito ao longo do código.

Caixas de Diálogo ou de Mensagem



INSTITUTO FEDERAL
Rio Grande do Sul

- Agora vamos conhecer um pouco mais sobre a caixa de diálogo usando o recurso **alert**.
- Vamos ver o artigo no link abaixo que explica como funciona

<https://www.devmedia.com.br/alert-em-javascript/37208>

