



@tiacademybrasil

JAVASCRIPT

AULA 1 | Prof. Marcelo

ABORDAGEM DO CURSO

- Tipos de dados
- Variáveis e constantes
- Operadores
- Funções
- Estrutura condicionais
- Manipulação de html/css

HISTÓRICO DO JAVASCRIPT

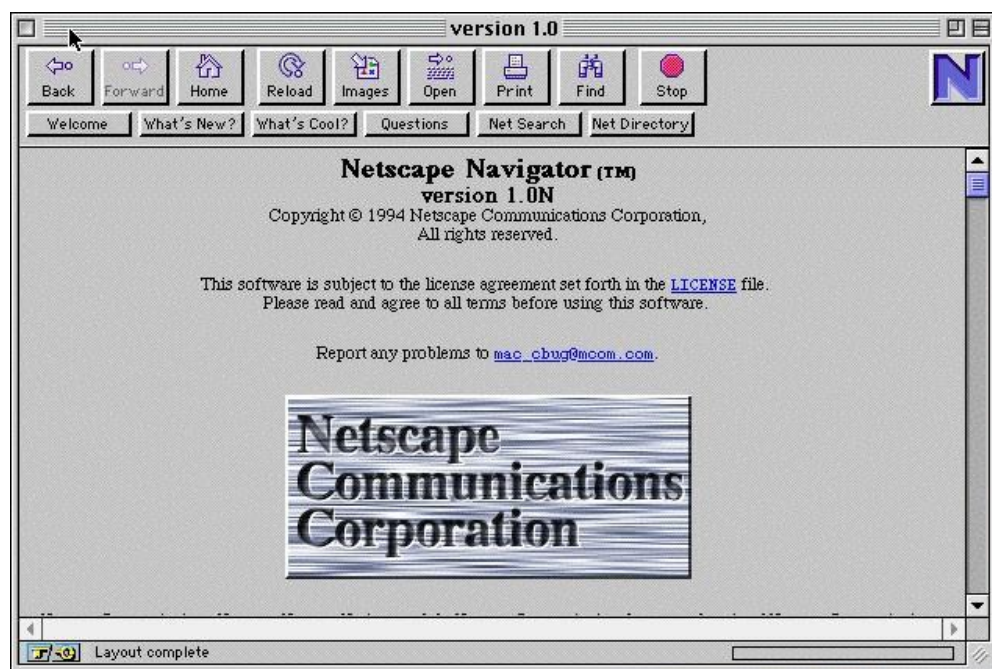
O **JavaScript** é uma linguagem de scripting. Uma linguagem de scripting é normalmente definida como um tipo de linguagem de programação que permite ao programador controlar uma ou mais aplicações de terceiros.

Com o **JavaScript**, podemos controlar vários comportamentos dos navegadores por meio de código que são enviados para uma página HTML.

A yellow square containing the letters 'JS' in a bold, black, sans-serif font.

HISTÓRICO DO JAVASCRIPT

O JS foi desenvolvido na Netscape na década de 90 com a intenção de aplicar dinamismo às páginas html.



JS

O que é o JavaScript

Importa dizer que basicamente a semelhaça entre **Java** e **JavaScript** está no nome. O JavaScript não é nenhuma implementação do Java.

O JavaScript é uma linguagem de programação que opera no lado do cliente, portanto é uma linguagem **client-side** (lado do cliente).

O que é o JavaScript

O JavaScript a princípio é uma linguagem interpretada. O responsável por interpretar as instruções que estão em **JS é o próprio navegador.**

Os navegadores possuem um engine, um motorzinho, responsável por interpretar o código.

Como escrever um código JS

O código JS pode ser implementado juntamente com a escrita HTML entre as tags **<script>...</script>** basicamente em qualquer local do código.

Um código JS em geral é codificado **em um arquivo com a extensão .js**.

Por padrão como já é sabido os arquivos possuem um nome e sua extensão, por uma questão de padrão adote nomes simples em **letras minúsculas**.

Arquivos JS

Pode-se também escrever **nomes de arquivos**, sempre em minúsculo, separados por **hífen** “-” ou o **sublinhado** “_”.

Ex.: “arquivo-de-script.js

OU

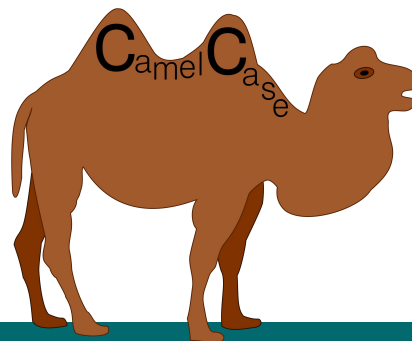
Ex.: “arquivo_de_script.js

Arquivos JS

Você pode adotar o padrão **CAMELCASE** para nomes e arquivos.

O padrão **camelcase** implica em iniciar o primeiro nome em minúscula e o nome seguinte com a letra inicial em maiúscula, **sempre sem espaços entre as palavras**.

ex.: **arquivoTeste.js**



Este padrão é adotado principalmente para declaração de variáveis.

Regras para codificar

A maioria das linguagens para programação possuem uma estrutura de linguagem natural ou também chamada de formal, isto é, compreendida pelo intelecto humano.

Os termos que são usados para construir um algoritmo, normalmente estão descritos na língua inglesa.

Regras para codificar - sintaxe e semântica

A maioria das linguagens para programação possuem uma estrutura de linguagem **natural** e a linguagem **formal**, isto é, compreendida pelo intelecto humano.

Os termos que são usados para construir um algoritmo, normalmente em uma linguagem formal.

Regras para codificar - sintaxe e semântica

A sintaxe de uma linguagem de programação consiste nas regras de codificação ou escrita de uma determinada linguagem.

Toda linguagem possui uma estrutura de palavras que irão compor o código e este código será combinado com um conjunto de outras palavras ou símbolos. As linguagens possuem seu próprio mecanismo para entender a estrutura de algoritmo.

Atribuição de variável em js:

```
var i = 1;
```

Atribuição de variável em Pascal:

```
var i := 1;
```

A sintaxe de uma linguagem também prevê o uso de “palavras reservadas” que fazem parte do seu próprio dicionário de palavras.

Regras para codificar - sintaxe e semântica

Já a semântica preocupa-se em uma “redação legível” para o código. Esta legibilidade de código permite que outros desenvolvedores entendam que operação aquele código irá realizar além de permitir que o motor de interpretação ou compilação “entender” o código.

Regras para codificar - sintaxe e semântica

A sintaxe de uma linguagem de programação consiste nas regras de codificação ou escrita de uma determinada linguagem.

Toda linguagem possui uma estrutura de palavras que irão compor o código e este código será combinado com um conjunto de outras palavras ou símbolos. As linguagens possuem seu próprio mecanismo para entender a estrutura de algoritmo.

Atribuição de variável em js:

```
var i = 1;
```

Atribuição de variável em Pascal:

```
var i := 1;
```

Palavras reservadas

Palavras reservadas são instruções e ou comandos da própria linguagem que não podem ser usadas fora do seu contexto. Alguns exemplos:

- new
- return
- if
- else
- switch
- typeof
- var
- for
- while

Tipos de dados

Variáveis são “espaços” ou “alocações” na memória de algum tipo de dado que pode ser manipulado em algum momento da execução do programa.

Number: É um tipo de dados que pode representar um número inteiro positivo ou negativo e também números decimais “ponto flutuante” (float).

Booleano: São valores que assumem apenas dois estados, VERDADEIRO ou FALSO. Pode ser representado apenas um bit 1 ou 0.

String: É uma sequência de um ou mais de caracteres delimitados entre aspas simples ou duplas.

Undefined: Classifica “*valor*” (variável ou sentença) não definida ou indefinido.

Null: Classifica “*valor*” como inexistente.

Declaração de variáveis

Variáveis são “espaços” ou “alocações” na memória de algum tipo de dado que pode ser manipulado em algum momento da execução do programa.

Declaração de variável em JS:

```
var idade = 21;
```

Atenção: O JavaScript é case-sensitive (diferenciação de minúscula e maiúscula). Ex.: **idade** não é o mesmo que **Idade**.

Pode-se usar camelcase para definir uma variável, mas não inicie variáveis com números.

JS

