

Aula 6

Fundamentos de Desenvolvimento de *Softwares*

Prof. Leonardo Gomes

Conversa Inicial

Fundamentos de desenvolvimento de *software*

- ▀ História do JavaScript
- ▀ Configurações
- ▀ Sintaxe
- ▀ Principais comandos
- ▀ JSON

História do JavaScript

História do JavaScript

- ▀ Desenvolvida em 1995 por Brendan Eich
- ▀ Conexões 28.8 kbps faziam dos *client-side scripts* aliados inestimáveis
- ▀ Desenvolvida para a Netscape

História do JavaScript

- ▀ Longa história de nomes e versões
 - C-minus-minus (Cmm)
 - Scriptease
 - EspressoPages (primeiras versões com suporte web)
 - Mocha
 - LiveScript
 - JavaScript!

História do JavaScript

- Jscript (clone feito para o Internet Explorer)
- A existência de muitas versões de JavaScript era um problema
- European Computer Manufacturers Association (ECMA) – 1997
- ECMAScript é a base para as implementações de Javascript hoje

JavaScript

- Linguagem responsável pelo comportamento das páginas *web*
- Simples
- Baseada em C

Configurações

Acessar e modificar conteúdo HTML

- JavaScript aceita aspas simples e duplas

```
<p id="demo">Modificando conteúdo HTML</p>
```

```
<button onclick="document.getElementById('demo').innerHTML = 'Olá JavaScript!'">
Clique Aqui!
</button>
```

Onde colocar seu código JavaScript

- `<script> </script>` Tag
- Podemos ter diversas `<script>`
- Tanto `<head>` quanto `<body>`
- Executam na ordem em que aparecem
- Uma boa prática é colocar no final do código para evitar lentidão

Arquivo externo

- Separa o HTML e o código
 - Deixa ambos mais fáceis de serem lidos
 - JavaScripts em cache aceleram o carregamento de páginas
 - Podem ser utilizados em diversas páginas distintas
 - Podem ser armazenados local ou externamente
- ```
<script src="meuScript.js"></script>
```

## Sintaxe

## Modos de escrita

- **innerHTML**
  - Método comum
- **document.write()**
  - Perigoso, útil para testes
- **window.alert()**
  - Para pessoas horríveis
- **console.log()**
  - Bom para depuração, aperte F12 para modo *debug*

## Sintaxe

- **Muito semelhante ao C**

- Expressões
- Comentários
- *Case sensitive*
- Strings

- **Função**

```
function myFunction() {
 document.getElementById("demo1").innerHTML = "Olá!";
 document.getElementById("demo2").innerHTML = "Como está?";
}
- Declaração de variáveis:
var x = 6;
```

## Array

```
var pessoas = ["Mario", "Luigi", "Peach"];
```

- Aceita múltiplos tipos
- .length
- .push

## Debug

- **No Chrome, F12 entra no modo *developer***

```
var x = 15 * 5;
debugger;
document.getElementById("demo").innerHTML = x;
```

## Principais comandos

## Eventos

- ***onchange***
  - Quando o elemento HTML muda
- ***onclick***
  - Quando o usuário clica no elemento HTML
- ***onmouseover***
  - Quando o usuário move o *mouse* sobre o elemento HTML

## Eventos

- ***onmouseout***
  - Quando o usuário retira o cursor do *mouse* do objeto
- ***onkeydown***
  - Quando o usuário aperta um botão do teclado
- ***onload***
  - Quando o navegador termina de carregar o elemento HTML, usado principalmente na *tag body*

## JSON

## JSON

- ***JavaScript Object Notation***
- JSON é uma sintaxe para armazenar e trocar dados
- JSON é texto escrito com a notação do JavaScript

## JSON Acessar

```
var pessoa = {"nome":"Mario", "idade":31, "pais":"Reino Cogumelo"};
pessoa.nome;
pessoa["nome"];
```

## JSON

```
{"funcionarios":[
 { "nome":"Liu", "sobrenome":"Kang" },
 { "nome":"Shao", "sobrenome":"Kahn" },
 { "nome":"Peach", "sobrenome":"Toad" }
]}
```

## JSON Stringfy Parse

```
var formatoString = JSON.stringify(obj);
var obj = JSON.parse('{ "nome": "Sonya", "idade": 30, "cidade": "New York" }');
```

## XML

```
<funcionarios>
 <funcionario>
 <nome>Liu</nome> <sobrenome>Kang</sobrenome>
 </funcionario>
 <funcionario>
 <nome>Shao</nome><sobrenome>Kahn</sobrenome>
 </funcionario>
</funcionarios>
```