



## Aula 01 - Criar uma página HTML do zero

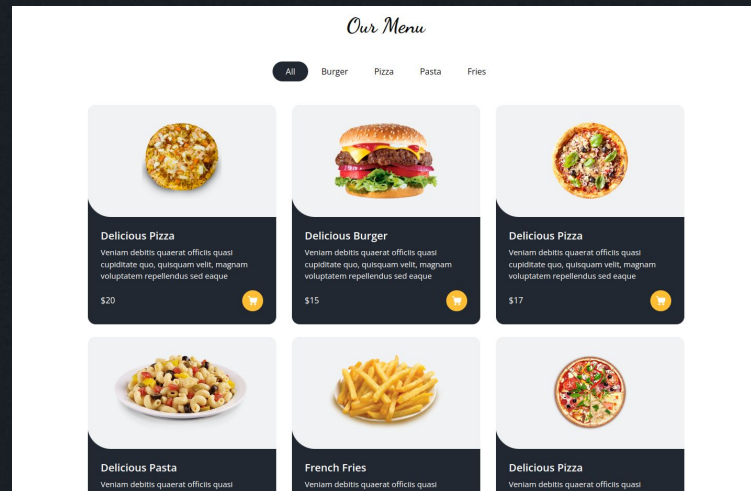
---

Front-End Estático - Ada

# Objetivos do módulo

## O que vamos fazer?

- Lista de exercícios no portal.
- Vamos construir um **Catálogo** estático com HTML e CSS.
- <https://html.design/>
- <https://html.design/demo/feane/index.html>

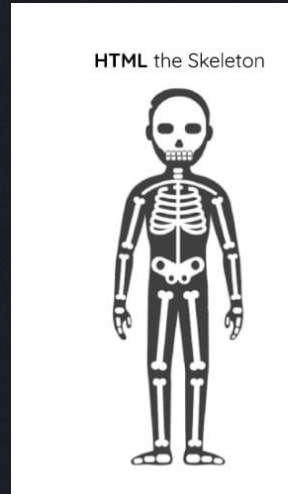


## Recapitulando...

### O que é HTML?

HTML é uma linguagem de marcação utilizada na construção de páginas na Web. Documentos HTML podem ser interpretados por navegadores.

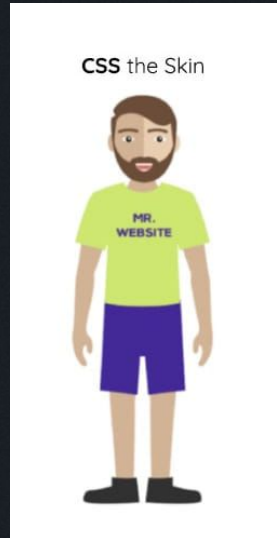
O HTML foi criado em 1991, por Tim Berners-Lee, no CERN (European Council for Nuclear Research) na Suíça.





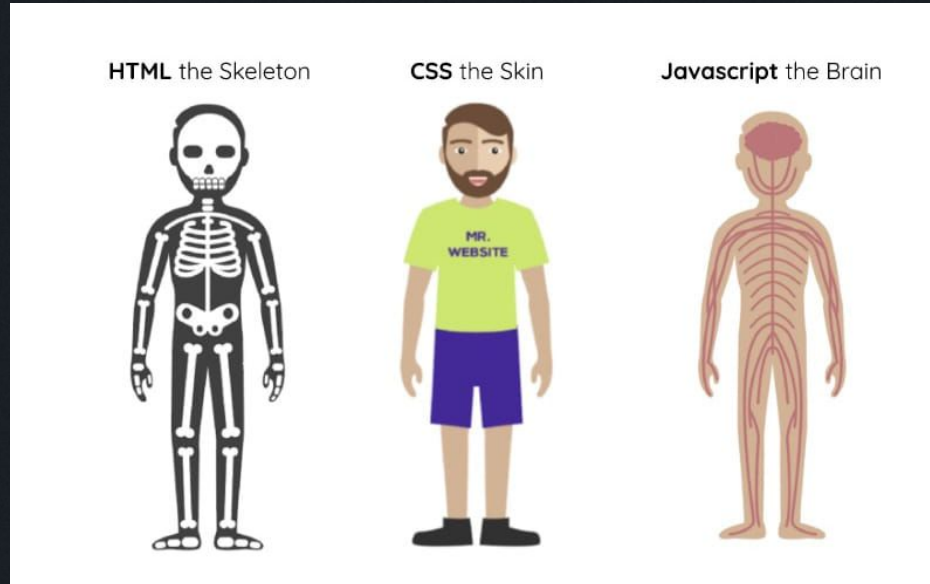
## O que é CSS?

Cascading Style Sheets é um mecanismo para adicionar estilos a uma página web, aplicado diretamente nas tags HTML ou ficar contido dentro das tags `<style>`. Também é possível adicionar estilos adicionando um link para um arquivo CSS que contém os estilos.



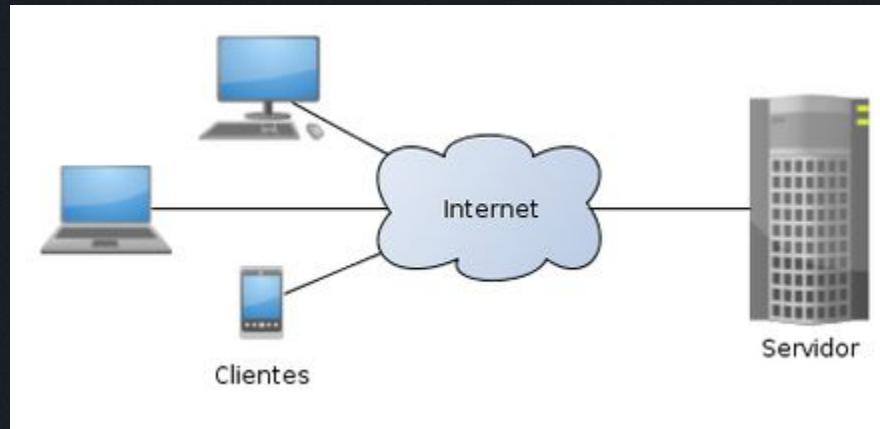
## E o JavaScript?

JavaScript (JS) é uma linguagem de programação interpretada estruturada, com tipagem dinâmica fraca. Juntamente com HTML e CSS, o JavaScript é uma das três principais tecnologias da World Wide Web.



## Modelo cliente-servidor

O modelo cliente-servidor (em inglês client/server model), em computação, é uma estrutura de aplicação distribuída que distribui as tarefas e cargas de trabalho entre os fornecedores de um recurso ou serviço, designados como servidores, e os requerentes dos serviços, designados como clientes.



# HTML

## Primeiro HTML

```
index.html x
<!DOCTYPE html>
<html>
<head>
  <title>Título da sua página</title>
</head>
<body>
  <h1>Hello World! :)</h1>
  <p>Seu primeiro hello world em HTML</p>
</body>
</html>
```





## Aula 02 - Tags HTML e introdução ao CSS

---

Front-End Estático - Ada



## Principais TAGs

| Tag      | O que é   |
|----------|---|
| <html>   | Estrutura HTML  |
| <head>   | Define os metadados do documento HTML                 |
| <titulo> | Define o título da página                             |
| <body>   | Define o corpo / conteúdo do HTML                     |
| <p>      | Parágrafo   |
| <h1>     | Título 1  |
| <h2>     | Título 2  |
| <header> | Cabeçalho da página                                   |
| <main>   | Define o conteúdo principal da página                 |
| <footer> | Rodapé da página                                      |
| <strong> | Importância   |
| <input>  | Entrada de dados                                      |
| <pre>    | Texto pré-formatado                                   |
| <br>     | Quebra de linha única                                 |
| <div>    | Divisão   |
| <a>      | Tag âncora para um link                               |
| <img>    | Imagem  |
| <area>   | Área de um mapa de imagem                             |
| <ul>     | Lista não ordenada                                    |
| <ol>     | Lista ordenada  |
| <li>     | Lista   |
| <dl>     | Lista de descrição                                    |
| <dt>     | Termo na lista de descrição                           |
| <dd>     | Definição/descrição de um termo na lista de descrição |

## Formatação das TAGs

ABERTURA ATRIBUTO

VALOR

`<a href="http://www.gcfaprendelivre.org" >`

GCFAprendeLivre } CONTEÚDO

FECHAMENTO

`</a>`

## Lista Ordenada

```
<ol>  
  <li>Item 1</li>  
  <li>Item 2</li>  
  <li>Item 3</li>  
</ol>
```

### Lista ordenada:

1. Item 1
2. Item 2
3. Item 3

## Lista Não Ordenada

```
<ul>  
  <li>Item 1</li>  
  <li>Item 2</li>  
  <li>Item 3</li>  
</ul>
```

### **Lista não ordenada:**

- Item 1
- Item 2
- Item 3



## Table

The diagram illustrates the HTML structure of a table with the following code and annotations:

```
<table>
  <tr>
    <th>Nome</th>
    <th>Sobrenome</th>
  </tr>
  <tr>
    <td>Fábio</td>
    <td>dos Reis</td>
  </tr>
</table>
```

Annotations:

- <table>**: Abre tabela
- <tr>**: Cria uma linha
- <th>Nome</th>** and **<th>Sobrenome</th>**: Criar cabeçalhos
- </tr>**: Fecha a linha
- <tr>**: Cria outra linha
- <td>Fábio</td>** and **<td>dos Reis</td>**: Criar células (colunas)
- </tr>**: Fecha segunda linha
- </table>**: Fecha a tabela

## HTML + CSS

Existem três formas de adicionar CSS em sua página HTML.

- propriedade style do elemento HTML
- elemento style (<style></style>)
- carregar um arquivo css usando: <link rel='stylesheet' type='text/css'>

## Selectors in CSS

### Element Selector

```
h2 {  
  color: #c70039 ;  
}
```

### Universal Selector

```
* {  
  color: #c70039 ;  
}
```

### ID Selector

```
#content {  
  color: #6E4253;  
  font-size: 15px;  
}
```

### Class Selector

```
.main {  
  margin-top: 10px  
  margin-bottom: 10px  
}
```

## Seletores

```
/* unvisited link */
a:link {
    color: #FF0000;
}

/* visited link */
a:visited {
    color: #00FF00;
}

/* mouse over link */
a:hover {
    color: #FF00FF;
}

/* selected link */
a:active {
    color: #0000FF;
}
```



## Atividade

## Relatório Financeiro de Fluxo de Caixa

Situação: Ambos, Intervalo: de: 02/2019 a 01/2020

Conta: CONTA PRINCIPAL

| Saldo Inicial:<br>14.801,43 | Fevereiro / 19 |           | Março / 19 |           |
|-----------------------------|----------------|-----------|------------|-----------|
|                             | PREV           | REAL      | PREV       | REAL      |
| Receitas                    | 10.500,00      | 5.600,00  | 10.900,00  | 0,00      |
| Despesas                    | 1.050,00       | 50,00     | 1.000,00   | 0,00      |
| Saldo mês                   | 9.450,00       | 5.550,00  | 9.900,00   | 0,00      |
| Saldo Acumulado             | 24.251,43      | 20.351,43 | 34.151,43  | 20.351,43 |



## Aula 03 - Formulários e verbos HTTP

---

Front-End Estático - Ada

## Estrutura de um formulário

CONTACT

Name:

E-mail:

Message:

```
<form action="https://nocodeform.io/f/65269da43ded9fa7c2ac6abd" method="post">
  <ul>
    <li>
      <label for="firstname">First Name:</label>
      <input type="text" id="firstname" name="firstname" />
    </li>
    <li>
      <label for="lastname">Last name:</label>
      <input type="text" id="lastname" name="lastname"></input>
    </li>
    <li>
      <label for="email">Email:</label>
      <input type="email" id="email" name="email" />
    </li>
    <li>
      <button type="submit">Submit</button>
    </li>
  </ul>
</form>
```

label + for  
Input + type + id + name



## Tipos de input

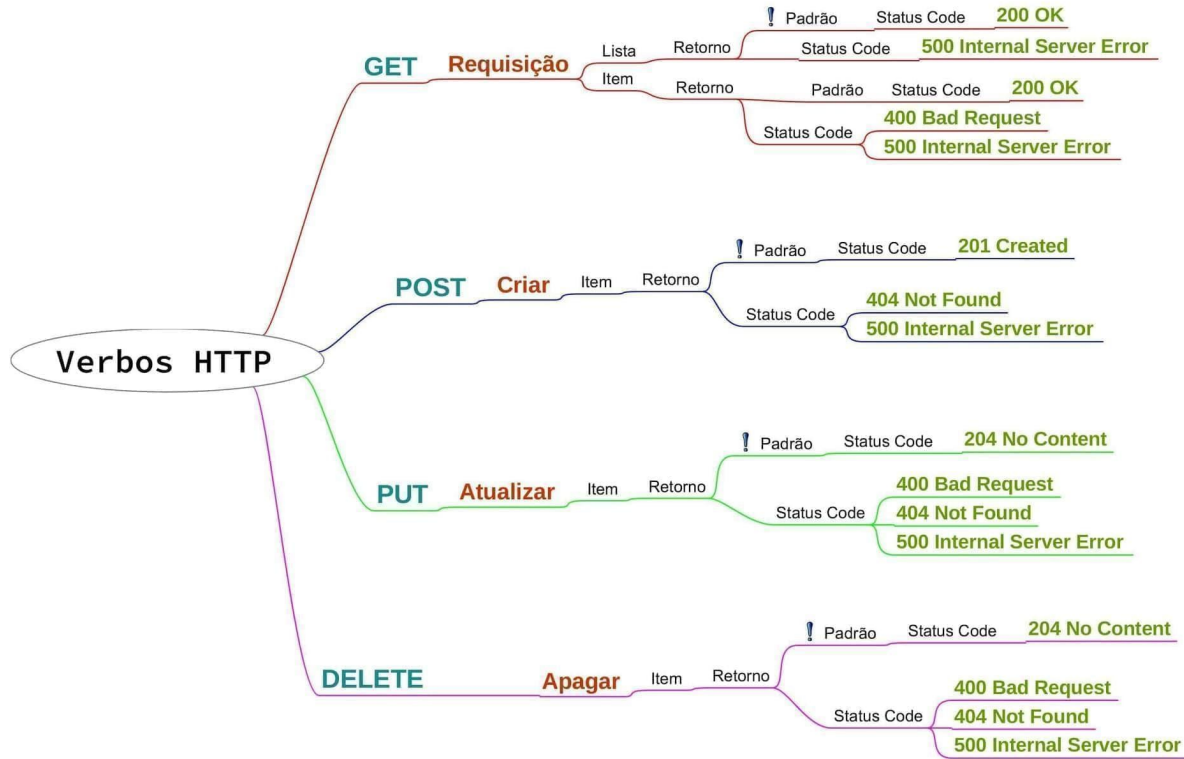
### HTML Input Types

Here are the different input types you can use in HTML:

- `<input type="button">`
- `<input type="checkbox">`
- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="email">`
- `<input type="file">`
- `<input type="hidden">`
- `<input type="image">`
- `<input type="month">`
- `<input type="number">`
- `<input type="password">`
- `<input type="radio">`
- `<input type="range">`
- `<input type="reset">`
- `<input type="search">`
- `<input type="submit">`
- `<input type="tel">`
- `<input type="text">`
- `<input type="time">`
- `<input type="url">`
- `<input type="week">`



## Verbos HTTP



## Validações

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Formulário HTML</title>
</head>
<body>
  <form action="http://localhost/meusite/login.php" method="POST">
    <label for="user">Usuário:</label>
    <input type="text" id="user" name="user" required
      ..... oninvalid="this.setCustomValidity('Preencha o nome do usuário')"
      ..... onvalid="this.setCustomValidity('')"
    /><br><br>

    <label for="password">Senha:</label>
    <input type="password" id="password" name="password" required
      oninvalid="this.setCustomValidity('Preencha a senha')"
      onvalid="this.setCustomValidity('')"
    /><br><br>

    <label for="submit">Enviar:</label>
    <input type="submit" id="submit" value="Enviar" /><br><br>
  </form>
</body>
</html>
```

## Atividade 2

- Criar um formulário;
- Utilizar labels com **for** para cada **input**;
- Utilizar as propriedades **id** e **name**;
- Utilizar pelo menos **5** tipos de inputs diferentes;
- Deverá ter validações de campos antes de enviar o formulário;
- Utilizar uma API para registrar os dados;
- Recuperar os dados da API e listar em uma tabela.

<https://crudcrud.com/>

<https://nocodeform.io/>



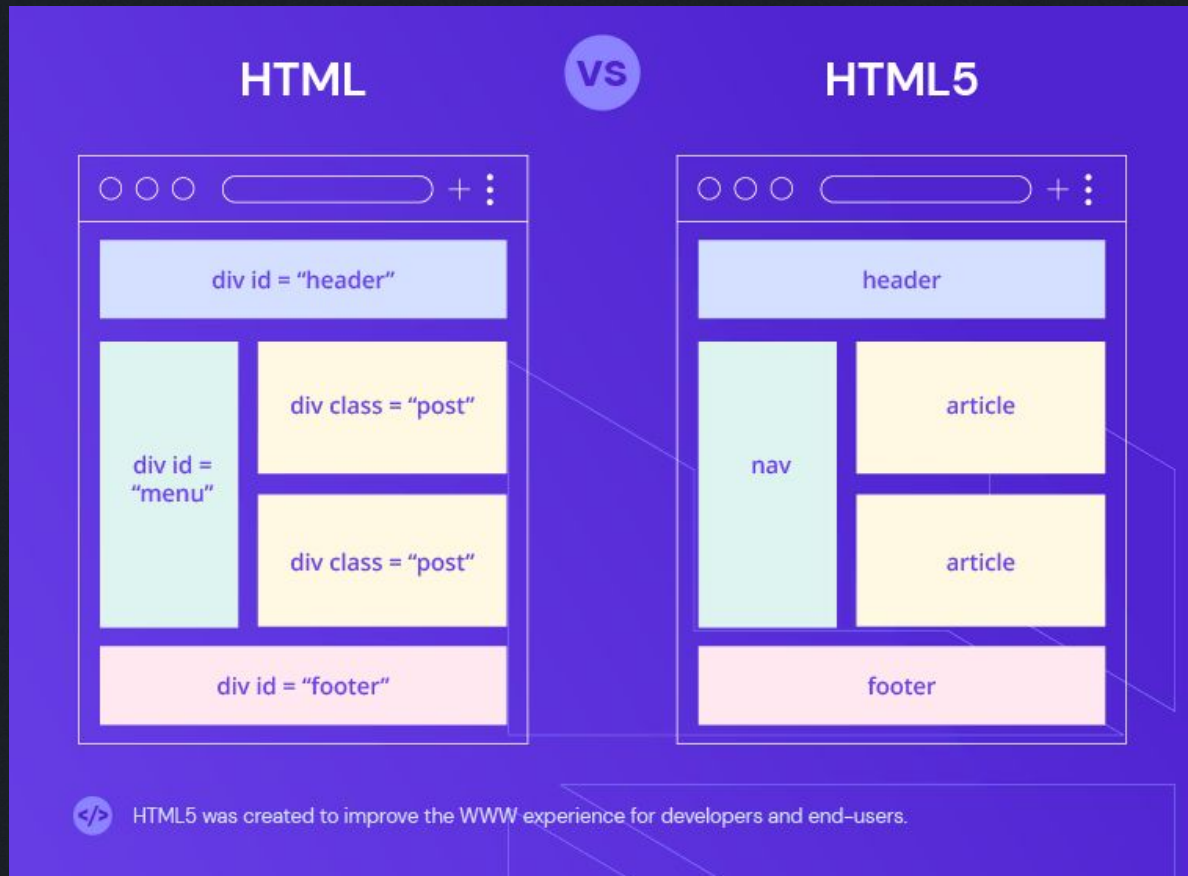


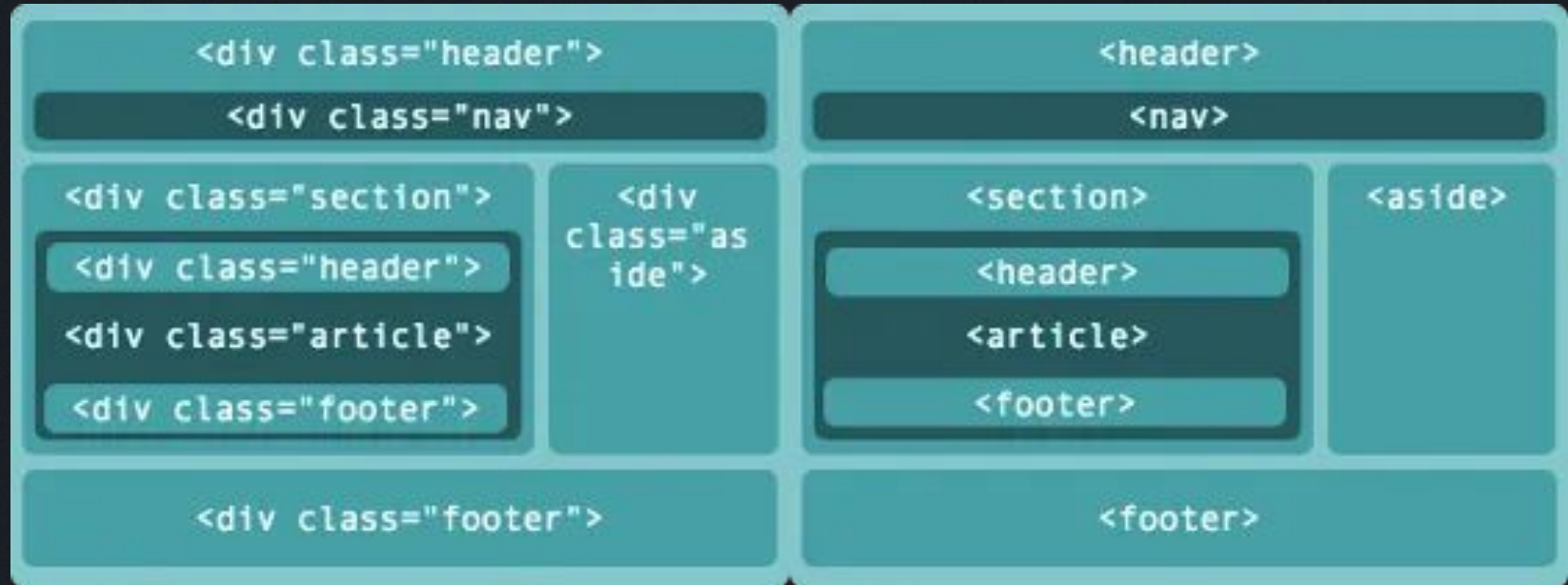
## Aula 04 - HTML para SEO e acessibilidade

---

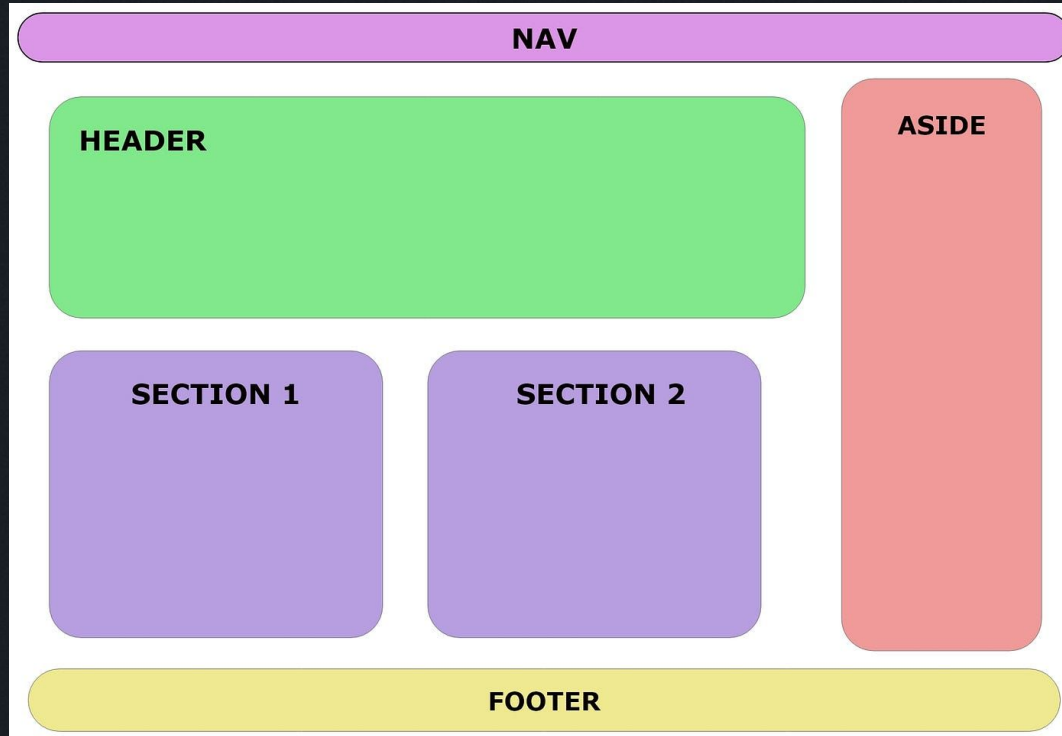
Front-End Estático - Ada







## Semântica



<https://developer.mozilla.org/pt-BR/docs/Glossary/Semantics>

# HTML

## DIV com significados

A semântica explica as divisões da página de uma forma que fique mais claro o que cada parte tem como função.

Os elementos semânticos mais utilizados são:

- `main` representa o conteúdo principal do documento.
- `section` representa uma subdivisão do documento dentro do `main`.
- `article` representa um conteúdo independente, como o post de um blog.
- `aside` representa conteúdo "lateral" como menus ou área de anúncios (ads) em um blog/site.
- `header` representa uma área, geralmente acima do conteúdo `main`, onde situa-se comumente a navegação.
- `footer` representa informações de "rodapé de página", que geralmente apresenta um mapa do site.
- `nav` representa um componente de navegação com links ou botões para as diversas páginas.
- `figure` representa um pequeno conteúdo à parte do todo, como uma imagem ou vídeo. Pode usar `figcaption` como legenda.
- `template` representa um conteúdo HTML que será utilizado por JavaScript. Ele, por padrão, não aparece na página.



### Projeto

Fazer um cardápio/catálogo digital;

- A página deve possuir um <**título**> e uma <**descrição**> explicando do que se trata;
  - Todo produto deste cardápio/catálogo deve possuir uma <**imagem**>, um <**nome**> e uma <**descrição**>;
  - É necessário ter um <**header**> e <**footer**> com <**navegação**>;
  - Deve existir uma <**tabela**> listando os produtos;
  - Deve possuir um <**formulário**> para cadastro;
  - A página deve possuir alguma estilização
  - O site deve estar otimizado para <**SEO**>, <**acessibilidade**> e <**responsivo**>.
- 
- **EXTRA**
  - Site inteiro feito com **Bootstrap**.

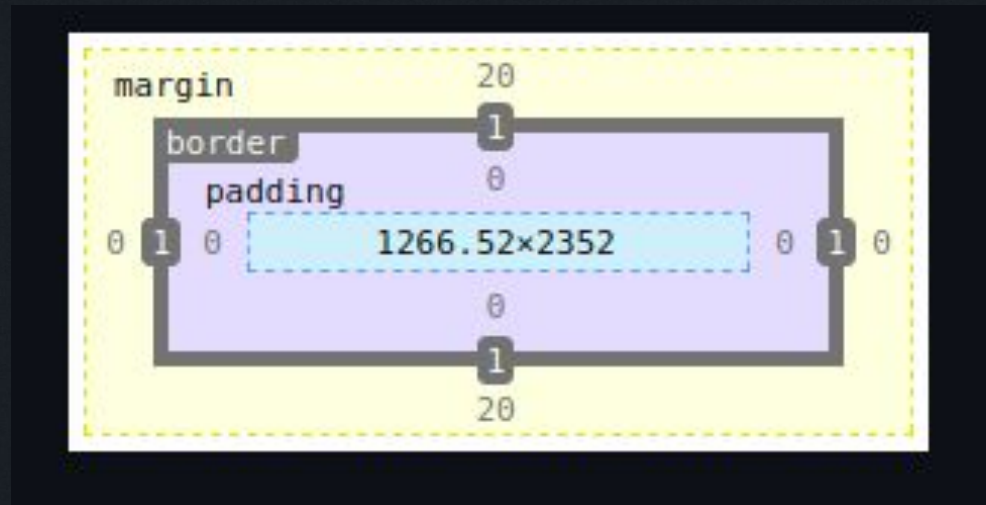


## Aula 05 - Flexbox, Position, Margin e responsividade

---

Front-End Estático - Ada

## Margin, padding e border





## Flexbox

- Para utilizar o flexbox devemos alterar o display do container, para flex usando o css: **display: flex;**
- **flex-direction** define como os elementos filhos serão organizados.

Os valores possíveis de flex-direction são:

- **row:** em linha (um ao lado do outro) da esquerda para direita
- **row-reverse:** em linha (um ao lado do outro) da direita para esquerda
- **column:** em coluna (um abaixo do outro) de cima para baixo
- **column-reverse:** em coluna (um abaixo do outro) de baixo para cima



## Alinhamento de conteúdo

- **justify-content** define como os elementos são espaçados na orientação do eixo principal.
- **align-items** define como os elementos são espaçados na perpendicular ao eixo principal.

Sendo assim, se **flex-direction** é **row**, **justify-content** define o posicionamento **horizontal** e **align-items** define o posicionamento **vertical**.

No entanto, se **flex-direction** for alterado para **column**, **justify-content** define o posicionamento **vertical** e **align-items** define o posicionamento **horizontal**.

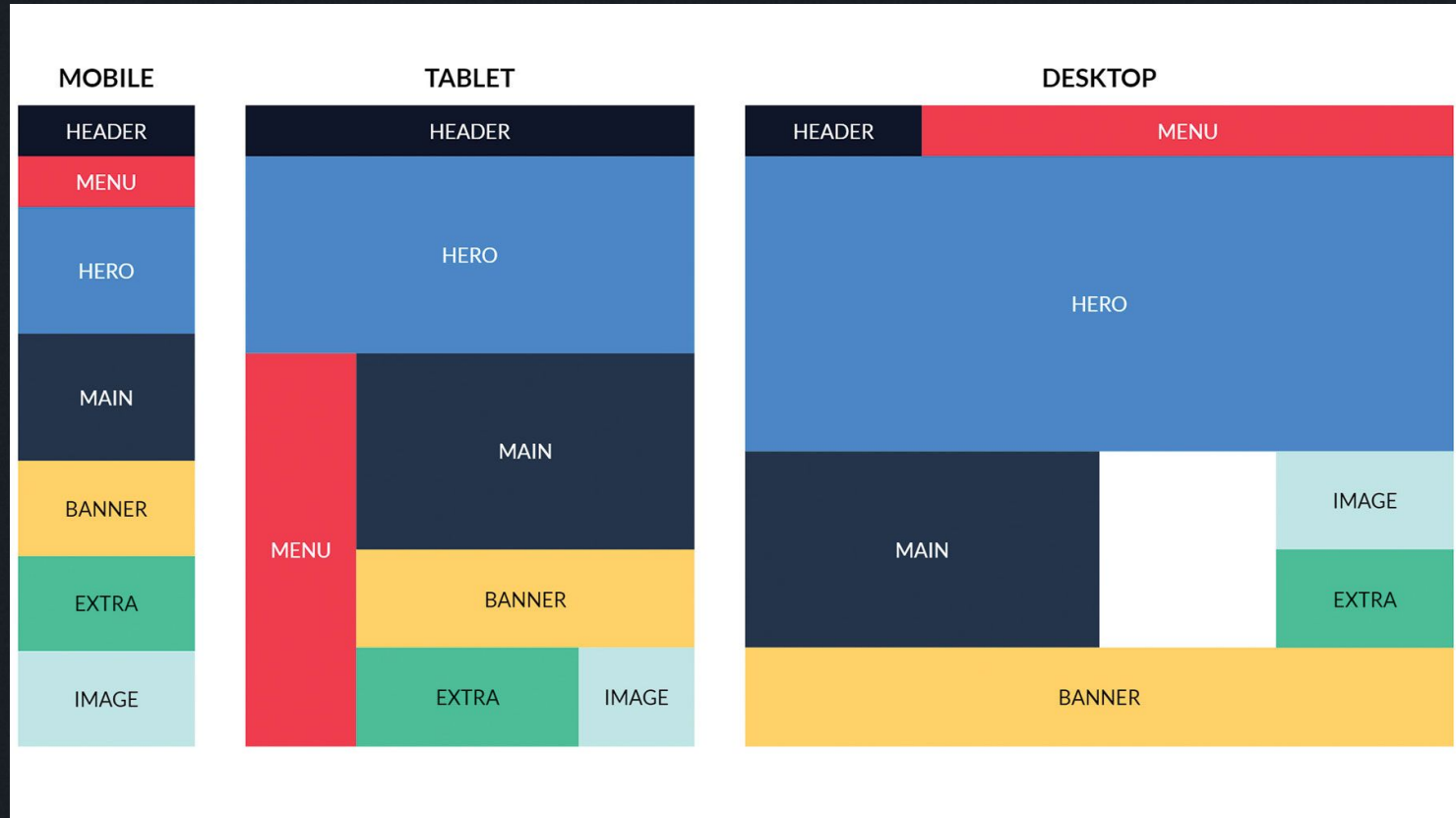


## Aula 06 - Grid layout e Bootstrap

---

Front-End Estático - Ada

## Grid Layout



## Grid Layout



[https://firefox-source-docs.mozilla.org/devtools-user/page\\_inspector/how\\_to/examine\\_grid\\_layouts/index.html](https://firefox-source-docs.mozilla.org/devtools-user/page_inspector/how_to/examine_grid_layouts/index.html)



## Media Queries

- Podem conter 1 ou mais expressões.
- Usam opções lógicas para determinar verdadeiro ou falso;
- Pode ser usado via link ou dentro de um stylesheet;

Exemplo via link:

### HTML

```
<!-- CSS media query em um elemento de link -->  
<link rel="stylesheet" media="(max-width: 800px)" href="example.css" />
```

Exemplo via CSS:

### CSS

```
/* CSS media query dentro de um stylesheet */  
@media (max-width: 600px) {  
  .facet_sidebar {  
    display: none;  
  }  
}
```

## Bootstrap

O Bootstrap foi criado no Twitter em meados de 2010 por um designer e um desenvolvedor, o Bootstrap se tornou um dos mais populares frameworks front-end e projetos de código aberto no mundo.

[https://www.w3schools.com/bootstrap4/bootstrap\\_ref\\_all\\_classes.asp](https://www.w3schools.com/bootstrap4/bootstrap_ref_all_classes.asp)

<https://bootstrapshuffle.com/pt/classes>

## Bootstrap Grid

- Distribuição do conteúdo em 12 colunas
- Totalmente responsivo



## Classes + usadas

- **container** - largura máxima da tela de 1140px
- **container-fluid** - largura de 100% em todos dispositivos
- **m** - margin
- **p** - padding
- **t** - top
- **b** - bottom
- **l** - left
- **r** - right

Os tamanhos variam de 0 a 5.

- **text-center** - alinhamento do texto ao centro
- **border** - estilo borda
- **bg-warning** - cor background
- **text-success** - cor texto



## Classes + usadas

- **btn** - forma
- **btn-primary** - cor
- **btn-block** - largura 100%

Podem ser usados com a tag `<a>`, `<button>` e `<input>` do tipo button, submit e reset.

- **form-control** - formulário

Tamanhos de colunas no grid:

- **col-xs** - extra small
- **col-sm** - small
- **col-md** - medium
- **col-lg** - large
- **col-xl** - extra large

Hierarquia - **container**, **row**, **col**

Obrig.ada