



**INSTITUTO FEDERAL**

Catarinense

Campus Camboriú

# Desenvolvimento Web I

**Professor: Euclides Paim**

*euclides.paim@ifc.edu.br*



**INSTITUTO FEDERAL**

Catarinense

Campus Camboriú

# HTML Responsivo

**Professor: Euclides Paim**

*[euclides.paim@ifc.edu.br](mailto:euclides.paim@ifc.edu.br)*



# HTML Responsivo

## Introdução

Design Responsivo é o conjunto de técnicas para melhorar a experiência do usuário independentemente do dispositivo que ele esteja utilizando. É uma técnica de estruturação HTML e CSS, que consiste em adaptar o site ao browser do usuário sem que seja necessário definir várias folhas de estilos específicas para cada resolução. Esse tipo de design teve origem em 2010 com o objetivo de melhorar a experiência de navegação dos usuários de páginas web.



# HTML Responsivo

## Viewport

O **Viewport** (ou janela de visualização) é toda a área de visível do seu site. É a área branca da janela quando você abre o browser. O *viewport* sempre vai ter o tamanho da janela. Mas a forma como os elementos são renderizados vai depender bastante do dispositivo. A janela varia de acordo com o dispositivo, e será menor em um telefone celular do que em uma tela de computador.

Antes de tablets e telefones celulares, páginas web foram projetados apenas para telas de computador, e era comum páginas da web terem um design estático e um tamanho fixo. As primeiras adaptações para corrigir a aparência nesses dispositivos reduzia toda a página web para caber na tela. Essa abordagem corrige o problema rapidamente, mas não evita que desenvolvedores web tenham que analisar o comportamento em diferentes dispositivos.



# HTML Responsivo

## Viewport

O elemento **«meta» viewport** surgiu para que os desenvolvedores tivessem controle sobre o *viewport*. Portanto se você quer suas páginas web se comportem como responsivas uma das primeiras coisas que você deve se preocupar é inserir a **«meta» viewport** entre as *tags* de abertura e fechamento do elemento **«head»**. Uma configuração bastante utilizada é a que veremos se seguir.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

No código a tag **meta** recebe o atributo **name** com valor **viewport** que determina que teremos o controle da janela de visualização. O atributo **content** recebe parâmetros que podem determinar a largura do *viewport*, a escala inicial para zoom, o valor máximo e mínimo de escala e até impossibilitar o usuário de efetuar zoom.



# HTML Responsivo

## Ferramentas de Desenvolvedor

Use as Ferramentas de desenvolvedor do Google Chrome para verificar *tags*. As Ferramentas de desenvolvedor do Google Chrome permitem analisar rapidamente o conteúdo/recursos de uma página da Web.

Como acessar as ferramentas para desenvolvedores:

- Clique com o botão direito do mouse em uma página e selecione "Inspecionar elemento". O código HTML do elemento clicado será exibido.
- Selecione Visualização > Desenvolvedor > Ferramentas para desenvolvedores.
- Use o atalho "Alt + Command + I



# HTML Responsivo

## Viewport

"A iniciativa da Internet móvel é importante, informações devem ser igualmente disponíveis em qualquer dispositivo" -- Tim Berners-Lee



# HTML + CSS

## Introdução CSS

O que significa CSS?

- **CSS** significa **Cascading Style Sheets**
- CSS descreve **como elementos HTML devem ser exibidos na tela**
- CSS **economiza um monte de trabalho**. Ele pode controlar o layout de várias páginas da Web de uma só vez
- Folhas de estilo externas são armazenados em **arquivos CSS**

CSS é usado para definir estilos para suas páginas da web, incluindo o design, layout e as variações na exibição para diferentes dispositivos e tamanhos de tela.





# HTML + CSS

## Sintaxe CSS

Seletor

h1

Bloco de Declaração

```
{color:blue; font-size:12px;}
```

Propriedade Valor

Propriedade

Valor

O conjunto de regras CSS consiste em um **Seletor** e um **Bloco de Declaração**. Os pontos seletor para o elemento HTML que deseja estilo. O bloco de declaração contém uma ou mais declarações separadas por um ponto e vírgula. Cada declaração inclui um nome de propriedade CSS e um valor, separados por dois pontos. A declaração CSS sempre termina com um ponto e vírgula, e blocos de declaração são cercadas por chaves.



# HTML + CSS

## Sintaxe CSS

Exemplo:

```
p {  
  color: red;  
  text-align: center;  
}
```

Exemplo detalhado:

- **p** é um **seletor** no CSS (ele aponta para o elemento HTML que você quer estilizar: <p>).
- **color** é uma propriedade, e **red** é o valor da propriedade.
- **text-align** é uma propriedade, e **center** é o valor da propriedade.



# HTML + CSS

## Seletores CSS

Seletores CSS são usados para "encontrar" (ou selecionar) os elementos HTML que você deseja aplicar estilo.

Podemos dividir seletores CSS em cinco categorias:

- **Seletores simples** (seleciona elementos com base no nome, id, classe);
- **Seletores combinadores** (seleciona elementos com base em uma relação específica entre eles);
- **Seletores pseudo-classe** (seleciona elementos com base em um determinado estado);
- **Seletores pseudo-elementos** (seleciona e estiliza apenas uma parte de um elemento);
- **Seletores de atributos** (seleciona elementos com base em um atributo ou valor de atributo);



# HTML + CSS

## Seletores CSS

O ***Element Selector*** seleciona elementos HTML com base no nome do elemento Exemplo:

```
p {  
    color: red;  
    text-align: center;  
}
```

Aqui todos os elementos **<p>** na página vão ser alinhados ao centro e com texto em vermelho.



# HTML + CSS

## Seletores CSS

O ***Id selector*** usa o atributo **id** de um elemento HTML para selecionar um elemento específico. A id de um elemento é única dentro de uma página, portanto o seletor de id é usado para selecionar um elemento único. Para selecionar um elemento identificado por uma **id**, escreva um caracter *hash* (#), seguido pela id do elemento.

```
#para1 {  
    color: red;  
    text-align: center;  
}
```

Obs.: O nome de uma id não pode começar com números.



# HTML + CSS

## Seletores CSS

O ***Class selector*** seleciona elementos HTML com um atributo de classe específico. Para selecionar elementos com uma classe específica, escreva o caractere ponto (.), seguido pelo nome da classe. No exemplo, todos os elementos HTML com `class = "center"` serão vermelhos e alinhados ao centro:

```
.center {  
    color: red;  
    text-align: center;  
}
```



# HTML + CSS

## Seletores CSS

Também podemos especificar que apenas elementos HTML específicos devem ser afetados pela classe. Exemplo:

```
p.center {  
    color: red;  
    text-align: center;  
}
```

Obs.: Neste exemplo apenas os elementos **<p>** da `class="center"` vão ser alinhados



# HTML + CSS

## Seletores CSS

O ***Universal selector*** (\*) seleciona todos os elementos HTML em uma página. A regra CSS abaixo vai afetar todos os elementos HTML na página.

```
* {  
    color: blue;  
    text-align: center;  
}
```





# HTML + CSS

## Seletores CSS

O ***Grouping selector*** (agrupamento de seletores) seleciona todos os elementos HTML com as mesmas definições de estilo. No código CSS a seguir, (o **h1**, **h2**, e elementos **p** possuem as mesmas instruções de estilo):

```
h1, h2, p {  
    color: blue;  
    text-align: center;  
}
```

Obs.: ao agrupar seletores, cada elemento deve estar separado por vírgula.



# HTML + CSS

## Inserindo instruções CSS

Quando um navegador lê uma folha de estilo, ele irá formatar o documento HTML de acordo com as informações na folha de estilo.

Existem três formas de aplicar CSS em um documento HTML:

- **CSS externo**
- **CSS interno**
- **CSS inline**



# HTML + CSS

## Inserindo instruções CSS

**CSS externo:** com uma folha de estilo externa, você pode mudar a aparência de um site inteiro, alterando apenas um arquivo. Cada página HTML deve incluir uma referência ao arquivo de folha de estilo externo dentro do elemento `<link>`, dentro da seção `<head>`. Exemplo:

```
<link rel="stylesheet" type="text/css" href="mystyle.css">
```

# HTML + CSS

## Inserindo instruções CSS

Uma folha de estilo externa pode ser escrito em qualquer editor de texto, e deve ser salvo com uma extensão **.css**. O arquivo **.css** externo não deve conter quaisquer tags HTML.

```
body {  
    background-color: lightblue;  
}
```

```
h1 {  
    color: navy;  
    margin-left: 20px;  
}
```

Aqui está como o arquivo ***mystyle.css*** do exemplo anterior se pareceria.

# HTML + CSS

## Inserindo instruções CSS

***CSS interno:*** uma folha de estilo interna pode ser usada se uma página HTML simples tem um estilo único. O estilo interno é definido dentro do elemento `<style>`, dentro da seção de `<head>`.

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    background-color: linen;
}

h1 {
    color: maroon;
    margin-left: 40px;
}
</style>
</head>
<body>
```

# HTML + CSS

## Inserindo instruções CSS

**CSS inline:** um estilo *inline* pode ser usado para aplicar um estilo único para um único elemento. Para usar estilos *inline*, adicione o atributo de estilo ao elemento relevante. O atributo de estilo pode conter qualquer propriedade CSS.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1 style="color:blue;text-align:center;">This is a heading</h1>
```

```
<p style="color:red;">This is a paragraph.</p>
```

```
</body>
```

```
</html>
```



# HTML + CSS

## Prioridade de Estilos

Qual estilo será usado quando existir mais de um estilo definido para um elemento HTML?

Todos os estilos em uma página serão escalados (cascata) em uma folha de estilos "virtual" seguindo as regras a seguir, onde a número um possui maior prioridade:

1. Estilo *inline* (dentro de um elemento HTML)
2. Estilos externos e internos (na seção <head>)
3. Padronização do navegador

Obs.: Dessa forma, um estilo inline tem a mais alta prioridade, e vai sobrescrever estilos externos, internos e os definidos pelo navegador.



- *HTML* respopnsivo
  - *Viewport*
- Ferramentas de desenvolvedor
- CSS
  - *Element Selector*
  - *Id Selector*
  - *Class Selector*
  - *Universal Selector*
  - *Grouping Selector*
- CSS externo
- CSS interno
- *CSS inline*
- Prioridade de estilos





# HTML + CSS

## Exercícios

1. Crie um arquivo HTML novo e adicione uma folha de estilos externa seguindo o padrão "*css/estilo.css*".
2. Defina uma cor de background: "azul" utilizando uma folha de estilos interna.
3. Defina uma cor de background: "vermelha" utilizando estilo *inline*.
4. Remova os estilos inseridos deixando apenas o estilo externo.



# Referências

## **Referências Básicas**

Livro NIEDERAUER, Juliano. Desenvolvendo websites com PHP: aprenda a criar websites dinâmicos e interativos com PHP e bancos de dados. 2. ed. rev. e atual. São Paulo: Novatec, 2011.

Livro SILVA, Maurício Samy. HTML 5: a linguagem de marcação que revolucionou a web. São Paulo: Novatec, 2011.

Livro SILVA, Maurício Samy. CSS3: desenvolva aplicações web profissionais com uso dos poderosos recursos de estilização das CSS3. São Paulo: Novatec, 2012.

## **Referências Complementares**

Livro DEITEL, Paul J. Ajax,. Rich Internet applications e desenvolvimento Web para programadores. . Pearson Prentice Hall. 2009

Livro DALL'OGGIO, Pablo. PHP: programando com orientação a objetos. . Novatec. 2009

Livro SOARES, Wallace. PHP 5: conceitos, programação e integração com banco de dados. . Érica. 2010

Livro SILVA, Maurício Samy. Criando sites com HTML: sites de alta qualidade com HTML e CSS. . Novatec. 2010

Livro FLANAGAN, David. o guia definitivo. . O'Reilly. 2012