

Programação Visual e Autoria Web

Cascading Style Sheet

Universidade Federal do Rio Grande do Norte

Introdução

- Nas aulas anteriores construimos páginas para web **sem formatação**.
- O máximo de formatação que conseguíamos eram os negritos dos headings <h1-6> e os itálicos de alguns elementos como ênfase .
- Nas versões anteriores do HTML os desenvolvedores podiam misturar código HTML com formatação.
- Essa possibilidade criava um **emaranhado de código** misturando conteúdo e formatação dificultando **reúso e modificações**.

Cascading Style Sheets

- É uma linguagem de folha de estilo usada para descrever a apresentação de um documento escrito em HTML ou XML (ou dialetos como SVG, MathML ou XHTML).
- Descreve como os elementos devem ser renderizados na tela, no papel, na fala ou em outras mídias.
- É padronizado em navegadores da Web de acordo com a especificação W3C.
 - CSS nível 1 – obsoleto
 - CSS nível 2.1 – corrigia erros do CSS 2
 - CSS nível 3 – baseia-se no CSS Nível 2 módulo por módulo, usando a especificação CSS2.1 como base. Cada módulo adiciona funcionalidade e/ou substitui parte da especificação CSS2.1
 - CSS nível 4 – Não há CSS Nível 4. Módulos independentes podem atingir o nível 4 ou além, mas o CSS não tem mais níveis.

Cascading Style Sheets

- O CSS adota uma estrutura simples para declaração de **propriedades e valores**.
- Uma propriedade é **separada de seu valor por um sinal de dois pontos “:”**.
- As propriedades são separadas entre si por um sinal de ponto-e-vírgula “;”.

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

Aplicando o CSS

- Existem três maneiras de se aplicar uma formatação CSS a um elemento HTML.
 - #1 Atributo Style do próprio elemento HTML
 - #2 Elemento `<style>` dentro do `<head>` do HTML
 - #3 Arquivo externo adicionado através do elemento `<link>` dentro do `<head>` do HTML

Atributo Style do próprio elemento HTML

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <title>Metal Contra as Nuvens</title>
  </head>
  <body>
    <h1 style="color:red; text-align: center">Metal Contra as Nuvens</h1>
  </body>
</html>
```

Elemento <style> dentro do <head> do HTML

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <title>Metal Contra as Nuvens</title>
    <style>
      h1 {
        color: red;
        text-align: center;
      }
    </style>
  </head>
  <body>
    <h1>Metal Contra as Nuvens</h1>
  </body>
</html>
```

← Seletor CSS, indica qual elemento será afetado.

Arquivo externo

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <title>Metal Contra as Nuvens</title>
    <link rel="stylesheet" href="estilo.css" />
  </head>
  <body>
    <h1>Metal Contra as Nuvens</h1>
  </body>
</html>
```

estilo.css

```
h1 {
  color: red;
  text-align: center;
}
```


Debate sobre as maneiras de uso do CSS

- Quais as vantagens e desvantagens de cada uma das maneiras de uso do CSS?
 - #1 Atributo Style do próprio elemento HTML
 - #2 Elemento <style> dentro do <head> do HTML
 - #3 Arquivo externo adicionado através do elemento <link> dentro do <head> do HTML

Seletores CSS

- Seletores CSS permitem selecionar e manipular elementos HTML.
- Os seletores são utilizados para localizar (ou selecionar) elementos com base no atributo HTML, id ou classe.
- Assim, podemos dizer que temos seletores de três tipos:
 - Elemento
 - id
 - Classe

Seletor de Elemento

- Seleciona elementos HTML com base em seu nome.
- São aplicados a todos os elementos HTML de mesmo nome.

```
h1 {  
    color: red;  
    text-align: center;  
}
```

Seletor de ID

- Seleciona elementos HTML com base no valor do atributo ID.
- Uso bastante restrito, pois segundo as regras do HTML os ids devem ser únicos em um documento.
- Necessário o uso da caractere # antes do id.

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

```
<p id="param">Esse elemento foi alterado pelo ID</p>
```

Seletor de Classe

- Seleciona elementos HTML de uma classe específica.
- As classes de um elemento podem ser atribuídas através do atributo class
 - No REACT vamos utilizado o atributo className.
- Necessário o uso do caractere “.” para definir o layout de uma classe.

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

```
<p class="vermelho">Texto 1</p>  
<p class="centralizado">Texto 2</p>  
<p class="vermelho centralizado">Texto 3</p>  
  
<p className='textodireita corvermelha'> Texto </p>
```

Seletor de Classe

- É possível indicar que apenas determinados elementos HTML irão ser modificados por uma classe.
- Isso é feito com a adição do nome do elemento HTML antes do ponto (.) do nome da classe.

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

```
<p class="destaque">Texto 1</p>  
<p class="centralizado">Texto 2</p>  
<p class="destaque centralizado">Texto 3</p>  
<article class="destaque">Texto 4</article>  
  
<p className='textodireita corvermelha'> Texto </p>
```

Seletor de Classe

- Também é possível adicionar múltiplas classes a um elemento.
- No atributo classe separamos as múltiplas classes por um espaço em branco.

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

```
<p class="destaque">Texto 1</p>  
<p class="centralizado">Texto 2</p>  
<p class="destaque centralizado">Texto 3</p>  
<article class="destaque">Texto 4</article>
```

Agrupando estilos

- É possível que alguns elementos compartilhem os mesmos estilos.
- Usamos o caractere vírgula “,” para indicar quais seletores estão compartilhando as mesmas propriedades de estilo.

```
article,.centralizado {  
    text-align: center;  
}  
.destaque {  
    color: blueviolet;  
}  
p.destaque {  
    color: red;  
}
```

```
<p class="destaque">Texto 1</p>  
<p class="centralizado">Texto 2</p>  
<p class="destaque centralizado">Texto 3</p>  
<article class="destaque">Texto 4</article>
```


Seletor em cascata

- Podemos selecionar elementos a partir de regras de cascata.
- Por exemplo, selecionar apenas elementos `<p>` que estejam dentro de `<article>`
- Para isso, utilizaremos o caractere espaço “ ” no seletor.

```
article h1 {  
    color: blue;  
}
```

```
<article><h1>Título</h1></article>  
<article>Texto normal</article>
```

Pseudo Classes

- Usada para definir a formatação em um estado especial de um elemento.
- Pode ser combinada com um seletor para adicionar um efeito a elementos existentes com base em seus estados.
- Existem várias pseudo classes, vamos ver apenas algumas. Para o restante consultar documentação.

Pseudo Classes

:hover

Usada para adicionar uma formatação específica quando o mouse está sobre o elemento.

```
.glow:hover {  
    background-color: aqua;  
}
```

```
<article class="glow">Texto normal</article>
```

Pseudo Classes

:active

Usada para adicionar uma formatação específica quando o usuário clica no elemento.

```
.glow:active {  
  background-color: aqua;  
}
```

```
<article class="glow">Texto normal</article>
```

Pseudo Classes

:visited

Usada para adicionar uma formatação específica quando o usuário já visitou o link.

```
a {  
    text-decoration: none;  
}  
a:hover {  
    text-decoration: underline;  
    background-color: chartreuse;  
}  
a:visited {  
    background-color: brown;  
}
```

```
<a href="http://www.google.com">Link</a>
```

Bibliografia Básica

- DUCKETT, J. **Javascript e JQuery**: desenvolvimento de interfaces web interativas. Rio de Janeiro: Alta Bokks, 2016
- FLANAGAN, D. **JavaScript**: O Guia Definitivo. Bookman, 2012.
- ROGERS, Y.; SHARP, H.; PREECE, J. **Design de Interação: além da interação homem-computador**. Bookman, 2013.
- MEYER
- Web Content Accessibility Guidelines (WCAG) 2.1. Disponível em: <<https://www.w3.org/TR/WCAG21/>>
- <https://developer.mozilla.org/en-US/docs/Web/HTML/>
- <https://www.w3schools.com/tags/>
- <https://css-tricks.com/archives/>

Bibliografia Complementar

- AMARAL, L. G. **CSS Cascading Style Sheets**: guia de consulta rápida. 2. ed. São Paulo: Novatec, c2006.
- KAWANO, W. **Crie aplicativos Web com HTML, CSS, JavaScript, PHP, PostgreSQL, Bootstrap, AngularJS e Laravel**. Rio de Janeiro: Ciência Moderna, 2016.
- PUREWAL, S. **Aprendendo a Desenvolver Aplicações Web**. Novatec, 2014.
- TONSIG, S. L. **Aplicações na nuvem**: como construir com HTML5, javaScript, CSS, PHP e MYSQL. Rio de Janeiro: Ciência moderna, 2012.
- USABILIDADE.com. Disponível em <<http://www.usabilidade.com/>>
- TASK-Centered User Interface Design: A Practical Introduction. Disponível em <<http://hcibib.org/tcuid/>>