



Prof. Taciano Balardin
tacianobalardin@gmail.com



E-MAIL:

taciano@ulbra.edu.br



SITE DA DISCIPLINA:

<http://www.taciano.pro.br/>

SENHA:

[@fpro](#)

CSS

Introdução

Conceitos Básicos

FUNDAMENTOS PROFISSIONAIS

AULA 13

Qual o problema com o HTML?

- **Desejamos criar uma aplicação Web, quais seriam os passos?**
 1. Definimos o conteúdo que vamos querer exibir;
 2. Definimos a estruturação do conteúdo na tela;
 3. Fazemos uso de tags para definirmos a hierarquia do conteúdo;
 4. Definimos a formatação a ser adotada (cores, tamanhos de fonte, espaçamentos, etc).
- **Qual o problema desta abordagem?**
 - Estamos unindo o conteúdo a ser exibido com a sua formatação!
... considere um site que todos os anos altera o seu layout

CSS: A solução

- **Separa a marcação do HTML da apresentação do site;**
- HTML marca e estrutura textos, cabeçalhos, parágrafos, links, botões, formulários, imagens e demais elementos da pagina;
- Portanto, a responsabilidade do HTML é apenas estruturar o conteúdo, enquanto que o **visual** é de **responsabilidade do CSS**;
- **CSS** define cores, posicionamento na tela, estilos de linhas, bordas e tudo mais relacionado a apresentação.

CSS: A solução

No início era uma **confusão!**

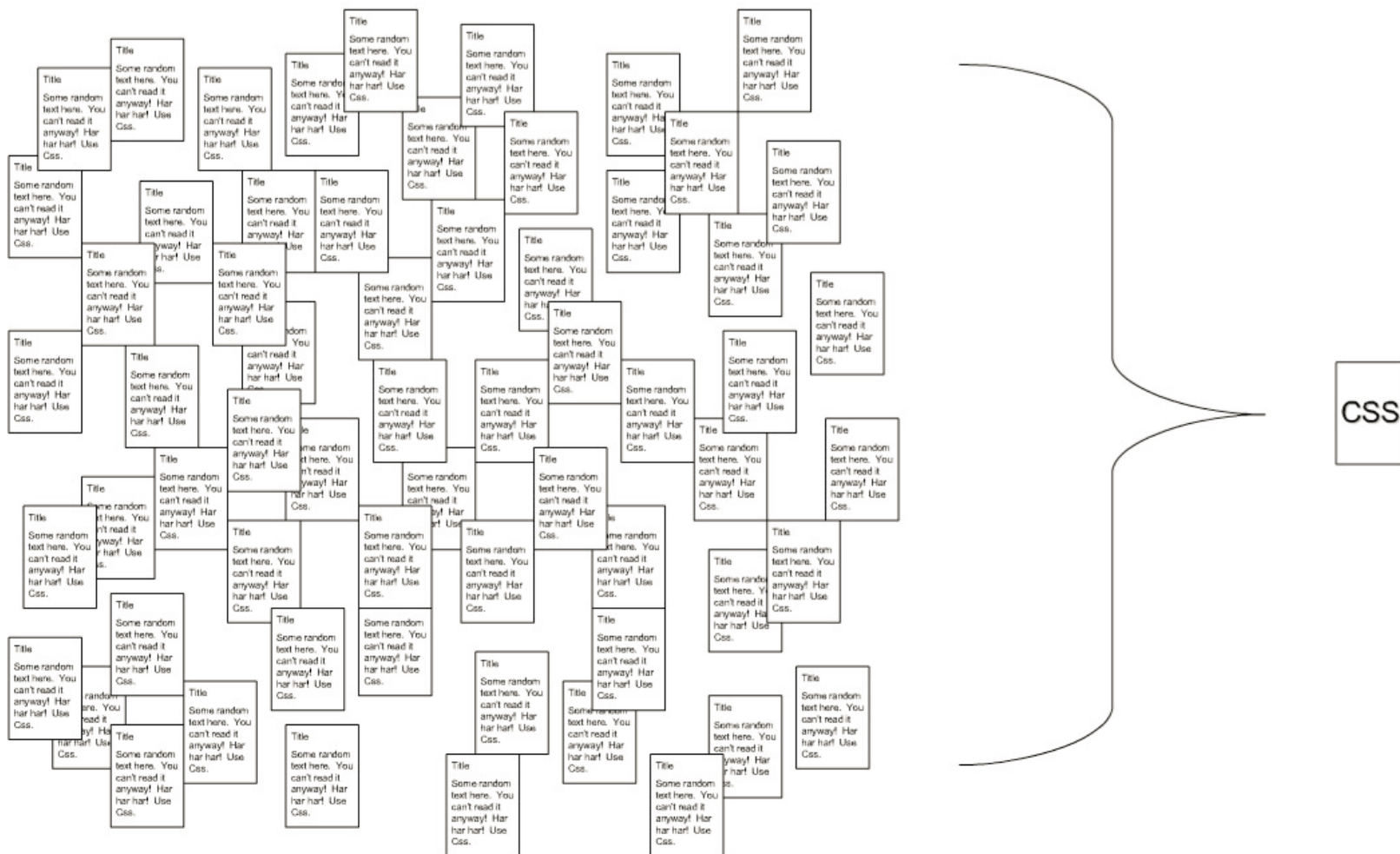


HTML
Styles...
JavaScript

CSS: A solução

Atualmente, organização em “camadas”

CSS: A solução



CSS

CSS: O que é?

- Acrônimo de **C**ascading **S**tyle **S**heets;
- Define como os elementos HTML serão exibidos;
- Podem ser definidos de três formas:
 - **Inline:**
 - Para modificações totalmente localizadas.
 - **Incorporada:**
 - Para modificações específicas de uma página.
 - **Externa:**
 - Uso geral.



Estilos *Inline*

- Estilos *Inline* são definidos dentro das *tags*;
- A maioria dos **benefícios são perdidos**:
 - Mistura conteúdo e apresentação;
 - Sempre que a página for carregada, CSS é relido.
- Utiliza o parâmetro **style**.

Exemplo:

```
<p style="color: #000000; margin: 5px;">
```

Parágrafo na cor preta com margem de 5 pixels.

Estilos Incorporados

- Estilos Incorporados são definidos **no documento**.
 - Boa parte dos benefícios são perdidos;
 - Mistura conteúdo e apresentação (menos que *inline*).
- Sempre que a página for carregada o CSS é relido.
- Utiliza as tags `<style>` e `</style>`

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <style>
```

```
      p { color: #000000; margin: 5px; }
```

```
    </style>
```

```
  </head>
```

```
  <body></body>
```

```
</html>
```

Estilos Incorporados

```
<head>
<style>
  body {
    background: #000000;
    url(imagens/fundo.jpg);
  }
  h3 {
    color: #0000FF;
  }
  p {
    margin-left: 20px;
  }
</style>
</head>
```



Tudo que estiver aqui dentro será considerado definição de estilo

Estilos Externos

- Arquivo .CSS com os estilos:
 - Separa conteúdo e apresentação;
 - Uso de um mesmo CSS para várias páginas;
 - Carregamento de novas páginas reaproveita CSS já carregado.
- Sintaxe:
 - Uso da tag **<link>** dentro da seção **<head>...</head>**

Estilos Externos

...

<head>

...

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

...

</head>

...

A regra CSS e sua sintaxe

- Uma regra CSS é uma declaração que segue uma **sintaxe própria** e que **define como será aplicado estilo a um ou mais elementos HTML**;
- Uma regra CSS compõe-se de três partes:
 - um **seletor**, uma **propriedade** e um **valor**.
- E tem a seguinte sintaxe:

seletor { **propriedade**: **valor**; }

Definindo os Elementos

- **Seletores:** é o elemento HTML para qual a regra será válida:
 - Pode ser uma **tag**, uma **classe**, ou por uma **ID**, etc;
 - Por exemplo: p, h1, form, .minhaclasse, #menu, etc).
- **Propriedade:** é o atributo do elemento HTML ao qual será aplicada a regra (por exemplo: font, color, background, etc);
- **Valor:** é a característica específica a ser assumida pela propriedade.

A atribuição dos valores às propriedades é feito através do operador : e não do operador =

Exemplos

/* Elementos p e body customizados: */

```
p {  
    font-size: 12px;  
}
```

```
body {  
    color: #000000;  
    background: #FFFFFF;  
    font-weight: bold;  
}
```

/* Alterar a fonte para o conteúdo do elemento h3: */

```
h3 {  
    font-family: Tahoma, Arial, "Comic Sans MS";  
}
```

Agrupamento de Seletores

- Muitas vezes precisamos definir a mesma regra para vários seletores, deste modo, é necessário **agrupar** todos os elementos em uma única regra.

/* Exemplo de **agrupamento de seletores**: */

h1 , h2 , h3 , h4 , h5 , h6 {

color: #00FF00;

}

O seletor class

- Até agora vimos a aplicação de estilos **apenas para as tags**.
- E se eu quiser que um parágrafo seja de uma cor/fonte/tamanho e o outro de outra?
 - **Com estilos apenas em tags não é possível definir diferentes estilos ao mesmo tempo.**
- Nestes casos utiliza-se os seletores **class** e **id**.

O seletor class

- O seletor **class** permite que você defina seus próprios identificadores que são utilizados nas regras de estilo.

```
elemento.minhaclasse {  
    propriedade: valor;  
}
```

```
p.corUm {  
    color: #000;  
}
```

```
p.corDois {  
    color: #0ff;  
}
```

Usando o seletor class

```
/* CSS */
```

```
p.corUm {  
    color: #000;  
}  
  
p.corDois {  
    color: #0ff;  
}
```

```
<!-- HTML -->
```

```
<p class="corUm">este parágrafo será na cor preta.</p>
```

```
<p class="corDois">este parágrafo será na cor azul.</p>
```

Estendendo o seletor class

- Ao criar uma classe você talvez queira que ela seja aplicável a qualquer elemento HTML.;
- Neste caso basta que se omita o nome do elemento antes da classe.

```
.corTres {  
    color: #f00;  
}
```

<p class="corTres"> este conteúdo terá cor verde </p>

 este conteúdo terá cor verde </p>

O seletor id

- O seletor ID difere do seletor class, por ser ÚNICO.
- Um seletor ID de determinado nome só pode ser aplicado a **UM** e **somente UM** elemento HTML dentro do documento.

```
#meuID {  
    propriedade: valor;  
}
```

```
p#cor {  
    color: #00f;  
}
```

Usando o seletor id

```
p#cor {  
    color: #00f;  
}
```

<p id="cor"> parágrafo na cor vermelha </p>

id x class - quando usar?

- O seletor id deve ser usado quando **apenas 1 elemento** deverá ser alterado.
- O elemento class será utilizado quando **diversos** elementos serão customizados.