



# Estácio

## **CSS3 – Introdução**



# CSS3 - Introdução

## Objetivos

- Identificar os elementos CSS3
- Empregar comandos do CSS3 em elementos HTML5.

# CSS3 - Introdução

## Definição

- Conforme W3C, Cascading Style Sheets (CSS) é um mecanismo simples para adicionar estilo (por exemplo, fontes, cores, espaçamento ) a documentos da Web. O CSS é um conjunto de comandos que alteram o comportamento dos elementos HTML 5
- O CSS formata a informação entregue pelo HTML. Essa informação pode ser qualquer coisa: imagem, texto, vídeo, áudio ou qualquer outro elemento criado

# Exemplo de utilização

Sintaxe: Uma regra CSS é composta de duas partes: o seletor e a declaração. A declaração compreende uma propriedade e um valor.

Onde:

seletor { propriedade: valor; }

Seletor → o alvo da regra CSS.

Propriedade → característica que você deseja modificar no elemento

Valor → valor referente a característica

Exemplo de utilização:

*h1, h2 {color: green }*

*h3 {color: black }*

# CSS3 - Seletores

- Seletores
  - O seletor representa uma estrutura. Essa estrutura é usada como uma condição para determinar quais elementos de um grupo de elementos serão formatados.
  - Tipos de seletores
    - Agrupamentos
      - Elementos separados por vírgula que herdam a mesma formatação
        - Ex.: `strong, em, h1 { color: red; }`
    - Encadeados ou Descendentes
      - Elementos contidos dentro de outros elementos
        - Ex.: `div p strong a { color: red; }`

# CSS3 - Seletores

- Seletores Complexos

PADRÃO	SIGNIFICADO	CSS
*	Seleciona qualquer elemento	2
E	Seleciona os elementos E	1
E.myclass	Elemento E com classe (atributo class) igual a myclass	1
E#myid	Elemento E com atributo id igual a myid	1
E > F	Seleciona os elementos E que são filhos diretos de F.	2
E + F	Seleciona um elemento F que precede imediatamente o elemento E.	2
E[atr]	Elemento E com um atributo específico.	2
E[atr="x"]	Elemento E que tenha um atributo com um valor específico igual a "x".	2
E[atr~="x"]	Elemento E com um atributo cujo valor é uma lista separada por espaços, sendo que um dos valores é "x".	2
E[atr^="x"]	Elemento E com um atributo cujo valor começa exatamente com string "x".	3
E[atr\$="x"]	Elemento E com um atributo cujo valor termina exatamente com a string "x"	3
E[atr*="x"]	Elemento E com um atributo cujo valor contenha a string "x".	3
E:enabled E:disabled	Seleciona um elemento E de interface que esteja habilitado ou desabilitado, como selects, checkbox, radio button etc.	3
E:checked	Seleciona elementos E que estão checados, como radio buttons e checkboxes.	3

Lista atualizada e completa → <http://www.w3.org/TR/css3-selectors/#selectors>

# CSS3 – Utilização no HTML

- Formas de inclusão no HTML
  - Estilo inline
    - Neste método, o CSS é aplicado diretamente no elemento utilizando o atributo **style**. É caracterizado por possuir uma implementação simples e direta, no entanto, este método dificulta a manutenção do código.
    - Deve ser usado para aplicar estilo a um elemento específico.

```
<h1 style="color:blue;margin-left:30px;">Isto é um cabeçalho</h1>
```

# CSS3 – Utilização no HTML

- Formas de inclusão no HTML
  - Estilo incorporado ou interno
    - Caracteriza-se por escrever o código CSS no utilizando o elemento **<style>** dentro do elemento **<head>**. Neste método as regras CSS estão localizadas no início do código HTML5.
    - No entanto, estas regras só poderão ser utilizadas dentro deste documento, impedindo o compartilhamento e a reutilização das regras.

```
<style>
    body {
        color: black;
        background: white; }
    em {
        font-style: normal;
        color: red; }
</style>
```



# CSS3 – Utilização no HTML

- Formas de inclusão no HTML
  - Estilo externo
    - As regras são escritas em um arquivo com extensão .css e carregadas no documento HTML5 com a utilização do elemento **<link>** dentro do elemento <head>
    - Desta maneira podemos alterar toda a apresentação de uma página substituindo apenas 1 arquivo .css
    - Dentro de um documento .css podemos importar outro documento .css escrevendo na primeira linha
      - @import "mystyle.css";

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

# CSS3 - Exercícios

Utilizando as 3 formas de declaração de estilos ( inline, interno e externo ) altere a cor do texto dos elementos `<h1>` e altere o comportamento da tag `<em>` para exibir um texto em negrito com cor vermelha.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <article>
      <header>
        <h1>Curso sobre HTML</h1>
        <h2>Como criar uma página </h2>
        <p>0 curso de <b>HTML</b><br>é muito
          <strong>importante</strong> </p>
      </header>
      <hr>
      <section>
        <h1>0 HTML 5</h1><p>Surgimento
          do <u><em>HTML 5</em></u></p>
      </section>
      <section><h1>Tags</h1>
        <p>Comandos
          de criação de páginas.</p>
      </section>
      <hr><hr>
      <footer><nav>
        <p><a href="/credits.html">Créditos</a> </p>
        <p><a href="/tos.html">Termos Legais</a> </p>
        <p><a href="/index.html">Mais Informações</a>
          </p>
      </nav></footer>
    </article>
    <br><br><br><br>
    <aside> <small>Estude na Estácio !!!</small> </aside>
  </body>
</html>
```

# CSS3 - Exercícios

- Altere a cor do texto dos links de azul para verde
- Altere a cor do texto de qualquer `<h1>` dentro de `<section>` para vermelho.
- Altere a cor do texto do elemento `<h1>` dentro do 1º elemento `<section>`
- Altere a cor do texto do último link para azul

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <article>
      <header>
        <h1>Curso sobre HTML</h1>
        <h2>Como criar uma página </h2>
        <p>O curso de <b>HTML</b><br>é muito
          <strong>importante</strong> </p>
      </header>
      <hr>
      <section>
        <h1>O HTML 5</h1><p>Surgimento
          do <u><em>HTML 5</em></u></p>
      </section>
      <section><h1>Tags</h1>
        <p>Comandos
          de criação de páginas.</p>
      </section>
      <hr><hr>
      <footer><nav>
        <p><a href="/credits.html">Créditos</a> </p>
        <p><a href="/tos.html">Termos Legais</a> </p>
        <p><a href="/index.html">Mais Informações</a>
          </p>
      </nav></footer>
    </article>
    <br><br><br><br>
    <aside> <small>Estude na Estácio !!!</small> </aside>
  </body>
</html>
```

# Propriedades tipográficas e fontes

Uso da propriedade font-family pode receber seu valor com ou sem aspas.

- com aspas: nome do arquivo de fonte a ser utilizado
- sem aspas: família da fonte

Padrão: texto em um tipo "serif"

- Eg.: "Times" e "Times New Roman"

Alterando a família de fonte:

```
h1 { font-family: serif; }
```

```
h2 { font-family: sans-serif; }
```

```
p { font-family: monospace; }
```

```
body {  
    font-family: "Arial", "Helvetica", sans-serif;  
}
```

# Alinhamento e decoração

Uma das propriedades mais simples, porém muito utilizada, é a que diz respeito ao alinhamento de texto: a propriedade `text-align`.

```
p { text-align: right; }
```

É possível configurar também uma série de espaçamentos de texto com o CSS:

```
p {  
  line-height: 3px; /* tamanho da altura de cada linha */  
  letter-spacing: 3px; /* tamanho do espaço entre cada letra */  
  word-spacing: 5px; /* tamanho do espaço entre cada palavra */  
  text-indent: 30px; /* tamanho da margem da primeira linha do texto */  
}
```

# Imagem de fundo

A propriedade background-image permite indicar um arquivo de imagem para ser exibido ao fundo do elemento. Por exemplo:

```
h1 {  
    background-image: url(sobre-background.jpg);  
}
```

Com essa declaração, o navegador vai requisitar um arquivo sobre-background.jpg, que deve estar na mesma pasta do arquivo CSS onde consta essa declaração.

# Bordas

As propriedades do CSS para definirmos as bordas de um elemento nos apresentam uma série de opções. Podemos, para cada borda de um elemento, determinar sua cor, seu estilo de exibição e sua largura. Por exemplo:

```
body {  
    border-color: red;  
    border-style: solid;  
    border-width: 1px;  
}
```

# Espaçamento e margem

Utilizamos a propriedade padding para espaçamento e margin para margem. Vejamos cada uma e como elas diferem entre si.

## Padding

A propriedade padding é utilizada para definir uma margem interna em alguns elementos (por margem interna queremos dizer a distância entre o limite do elemento, sua borda, e seu respectivo conteúdo) e tem as subpropriedades listadas a seguir:

- padding-top
- padding-right
- padding-bottom
- padding-left

Essas propriedades aplicam uma distância entre o limite do elemento e seu conteúdo acima, à direita, abaixo e à esquerda.





# Padding

```
p { padding: 10px; }
```

```
p { padding: 10px 15px; }
```

```
p { padding: 10px 20px 15px; }
```

```
p { padding: 10px 20px 15px 5px; }
```

# Espaçamento e margem

## Margin

A propriedade margin é bem parecida com a propriedade padding, exceto que ela adiciona espaço após o limite do elemento, ou seja, é um espaçamento além do elemento em si. Além das subpropriedades listadas a seguir, há a shorthand property margin que se comporta da mesma maneira que a shorthand property do padding vista no tópico anterior.

- margin-top
- margin-right
- margin-bottom
- margin-left

```
p { margin: 0 auto; }
```

# Dimensões

É possível determinar as dimensões de um elemento, por exemplo:

```
p {  
    background-color: red;  
    height: 300px;  
    width: 300px;  
}
```

Todos os parágrafos do nosso HTML ocuparão 300 pixels de largura e de altura, com cor de fundo vermelha.

# CSS

## Posicionamento e Transições

# Position x Float

As propriedades abaixo determinam como um box será posicionado no modelo visual de formatação de um documento.

## *position e float*

As recomendações do W3C (CSS 2.1) prevêm a existência de três esquemas de posicionamento:

- **normal**: os elementos são apresentados um após o outro de cima para baixo;
- **flutuante**: o elemento é deslocado para uma nova posição e seu espaço original preenchido por outro elemento, na medida do possível;
- **absoluto**: definido pela regra position com valores absolute e fixed. O box é removido do fluxo normal sem impactar nos elementos posteriores.

# Position

Define o método de posicionamento utilizado por um elemento.

- **static:** (padrão) segue o fluxo normal da página, não é afetado pelas regras: *top, right, bottom, left*.
- **relative:** o elemento é posicionado relativamente a posição normal e pode ser ajustado a partir da posição normal pelas regras: *top, right, bottom, left.position*
- **fixed:** o elemento é posicionado relativo ao viewport (janela de exibição), ficará no mesmo lugar mesmo se a página for rolada.
- **absolute:** o elemento é posicionado relativamente ao elemento ancestral mais próximo posicionado, se não houver, posicionado em relação ao body e acompanhará o rolamento da página.

# Exercício

# Float

Define se um elemento deve flutuar, podendo receber os seguintes valores:

- **none**: (padrão) o elemento não irá flutuar
- **left**: o elemento irá flutuar a esquerda
- **right**: o elemento irá flutuar a direita

Importante: elementos após elementos flutuantes irão fluir em torno dele. Caso deseje, para evitar este comportamento, use a propriedade:

- **clear**: proíbe a flutuação dos lados do elemento  
valores possíveis: *left*, *right*, *both*



# Exercício

# Overflow

Define o comportamento se o conteúdo extrapola o espaço de um elemento.

- **visible:** (padrão) o conteúdo não é cortado e pode ser renderizado para fora da caixa de conteúdo.
- **hidden:** o conteúdo é cortado e nenhuma barra de rolagem é exibida.
- **scroll:** uma barra de rolagem é adicionada para permitir a visualização.

# Exercício

# Transições

Permitem alterar os valores de propriedades suavemente (de um valor para outro), durante um determinado período.  
Normalmente utilizada em conjunto com o seletor *:hover*

Para criar um efeito de transição precisa-se especificar pelo menos duas propriedades:

- a propriedade que irá receber o efeito
- a duração do efeito

## Propriedades

- **transition-property**: aceita propriedades separadas por vírgula; ex.: *div { transition-property: width, height; }*
- **transition-duration**: define o tempo de duração do efeito de transição; ex.: *div { transition-duration: 5s; }*
- **transition**: define as propriedades em uma única linha; ex.: *div { transition: width 2s; }*

# Exercício

# Transições - propriedades

Propriedade	Descrição
transition	Define todas as 4 propriedades em um única declaração
transition-delay	Define um atraso em segundos para iniciar a transição
transition-duration	Define a duração da transição
transition-property	Define a propriedade CSS que
transition-timing-function	Define a curva de velocidade do efeito de transição

# Posicionamento - propriedades

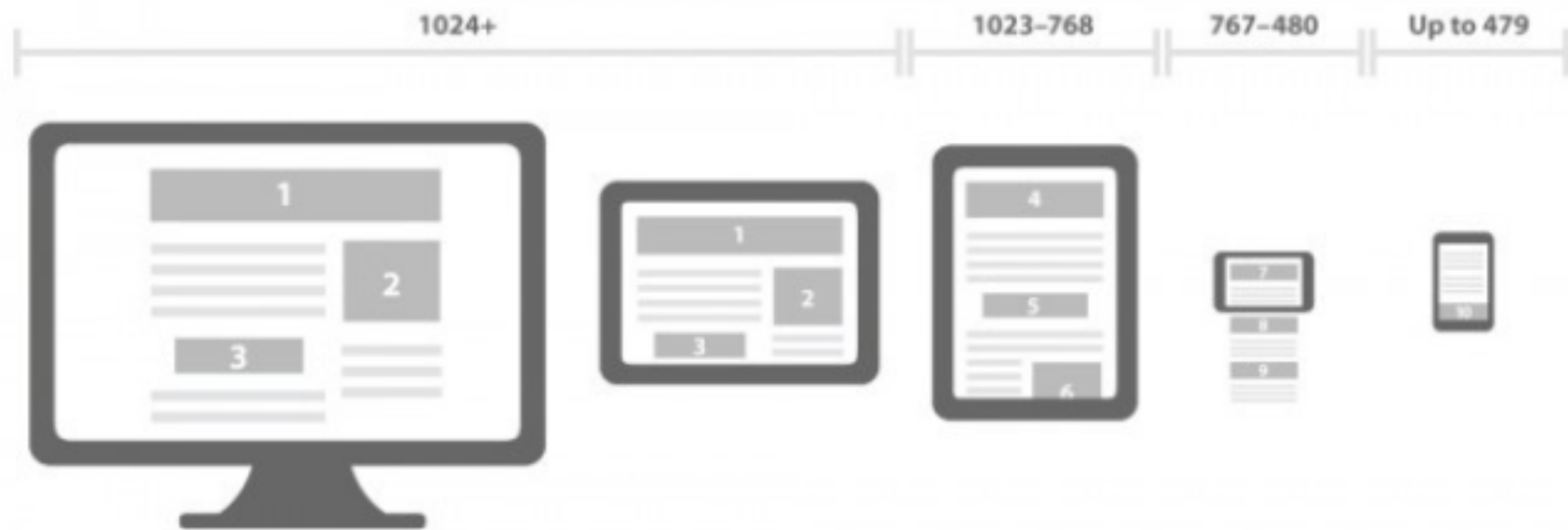
Propriedade	Descrição
clear	Proíbe flutuação nos lados de um elemento
float	Define a forma de flutuação
overflow	Define o que acontece se o conteúdo sobrepõe outro elemento
overflow-x	Define o que fazer se sobrepor nos lados esquerdo e direito
overflow-y	Define o que fazer se sobrepor nos lados cima e baixo
position	Define a forma de posicionamento do elemento

# Web Design Responsivo



# Web Design Responsivo

Técnica de criar páginas para a web que se adaptem a diferentes resoluções, telas, dispositivos, sem precisar criar uma página específica para cada situação, mantendo sempre a acessibilidade e a usabilidade da página.



# Web Design Responsivo

- Página Web com “boa” apresentação em qualquer dispositivo
- Páginas Web podem ser visualizadas em diferentes dispositivos: laptops, tablets, smartphones, ...
- A página Web deve adaptar-se para cada tipo de dispositivo  
Utiliza apenas HTML e CSS

# Motivação

Estado	Celular/Tablet	Computador
Sergipe	28,9%	19,3%
Pará	41,2%	17,3%
Roraima	32%	17,2%
Amapá	43%	11,9%
Amazonas	39,6%	11,1%

Fonte: IBGE 2013

# Viewport

- Área visível de uma página Web: dimensões variam com o dispositivo
- Antes dos tablets e smartphones era comum termos páginas com elementos de tamanhos fixos
- Navegando nestas páginas em dispositivos móveis não era uma experiência agradável
- Então naqueles dispositivos os navegadores reduziam toda a página para caber na tela, simulando no dispositivo uma largura maior.

# Viewport

- HTML5 introduziu um método para deixar web designers ter controle sobre o viewport , através da tag `< meta >`

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

- **width=device-width** → define a largura da página para seguir a largura do dispositivo
- **initial-scale=1.0** → define o zoom inicial para 1.0, ou seja, o dispositivo não irá reduzir o zoom para alterar a apresentação.

# Viewport

## Sem controle do viewport

```
<!DOCTYPE html>
<html>
<head><meta charset="UTF-8"></head>
<body>
  
<p>
  Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue dui dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum.
</p>
</body>
</html>
```

## Com controle do viewport

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
  img { max-width:100%; }
</style>
</head>
<body>
  
<p>
  Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue dui dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum.
</p>
</body>
</html>
```



## Referências de Estudo

- Apostila css-web.pdf
- Material da Estácio
- <http://www.w3c.br/Home/WebHome>
- <http://www.w3schools.com/css/>