



GOVERNO DO ESTADO
SÃO PAULO

Secretaria de Desenvolvimento
Econômico, Ciência, Tecnologia e Inovação

Faculdade de Tecnologia de Jales – “Professor José Camargo”

Padrões de Projetos de Sítios Internet I

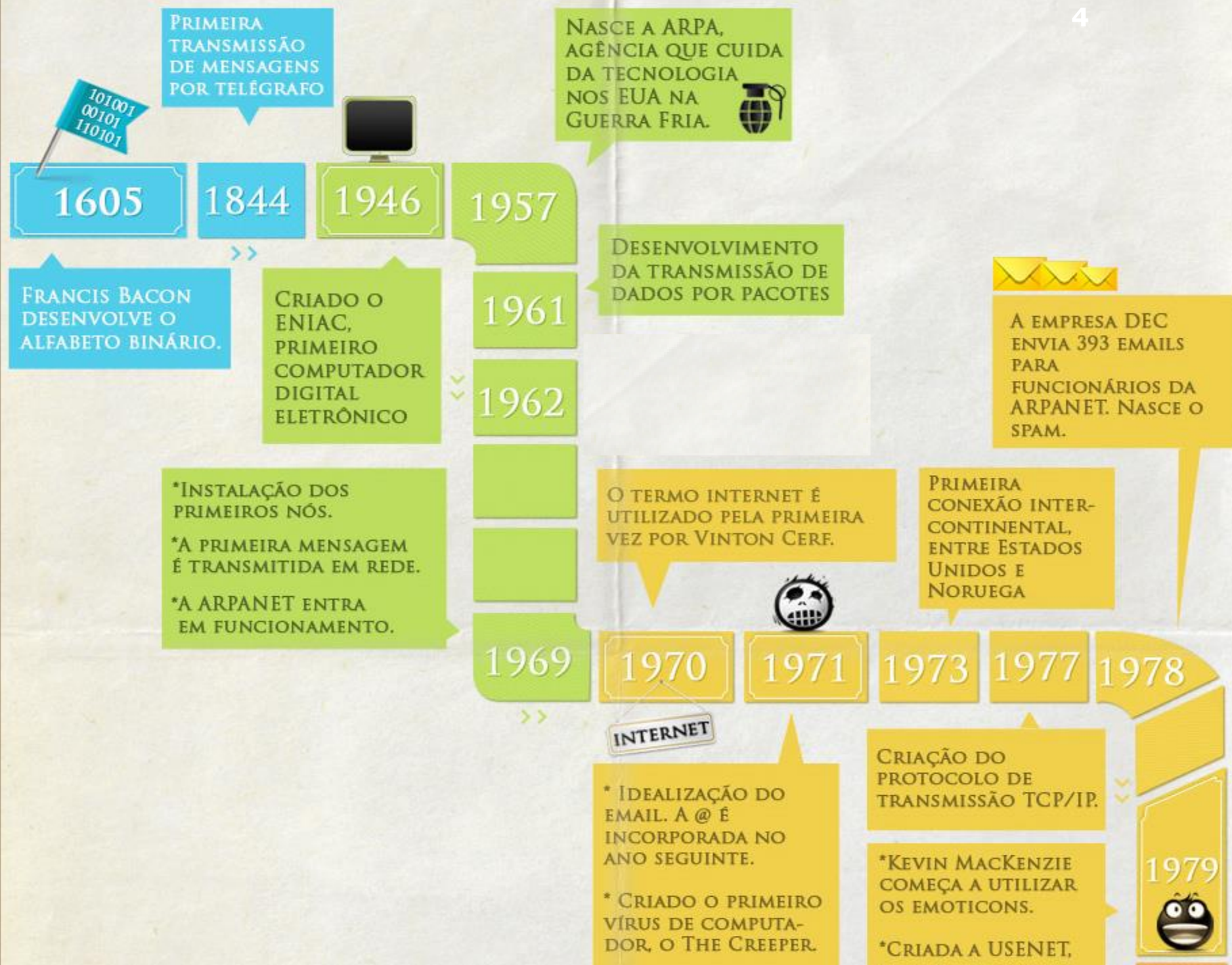
Prof. Alexandre Ap. Bernardes

alexandre.bernardes2@fatec.sp.gov.br

Objetivos:

- **Gerais:** Apresentar os fundamentos do HTML (*Hypertext Markup Language*) com ênfase na semântica do uso dos elementos e nos benefícios do emprego de padrões de codificação. Demonstrar uso do CSS (*Cascading Style Sheets*) para separar conteúdo e apresentação, de modo a aumentar velocidade do desenvolvimento e as habilidades de design.
- **Específicos:** Os estudantes deverão ser capazes de: construir páginas Web por meio do HTML e validá-las no validador do W3C, quando for o caso. Empregar o CSS para todas as apresentações. Criar uma estrutura de diretórios adequada para sítios Internet visando a facilidade de manutenção. Saber empregar FTP para transferir arquivos para o servidor. Construir formulários. Empregar um script fornecido. Criar um sítio completo com várias páginas com todos os elementos de design, HTML e CSS estudados no curso.

Internet
X
WEB





SISTEMA DE LOCALIZAÇÃO

1945 – 1989

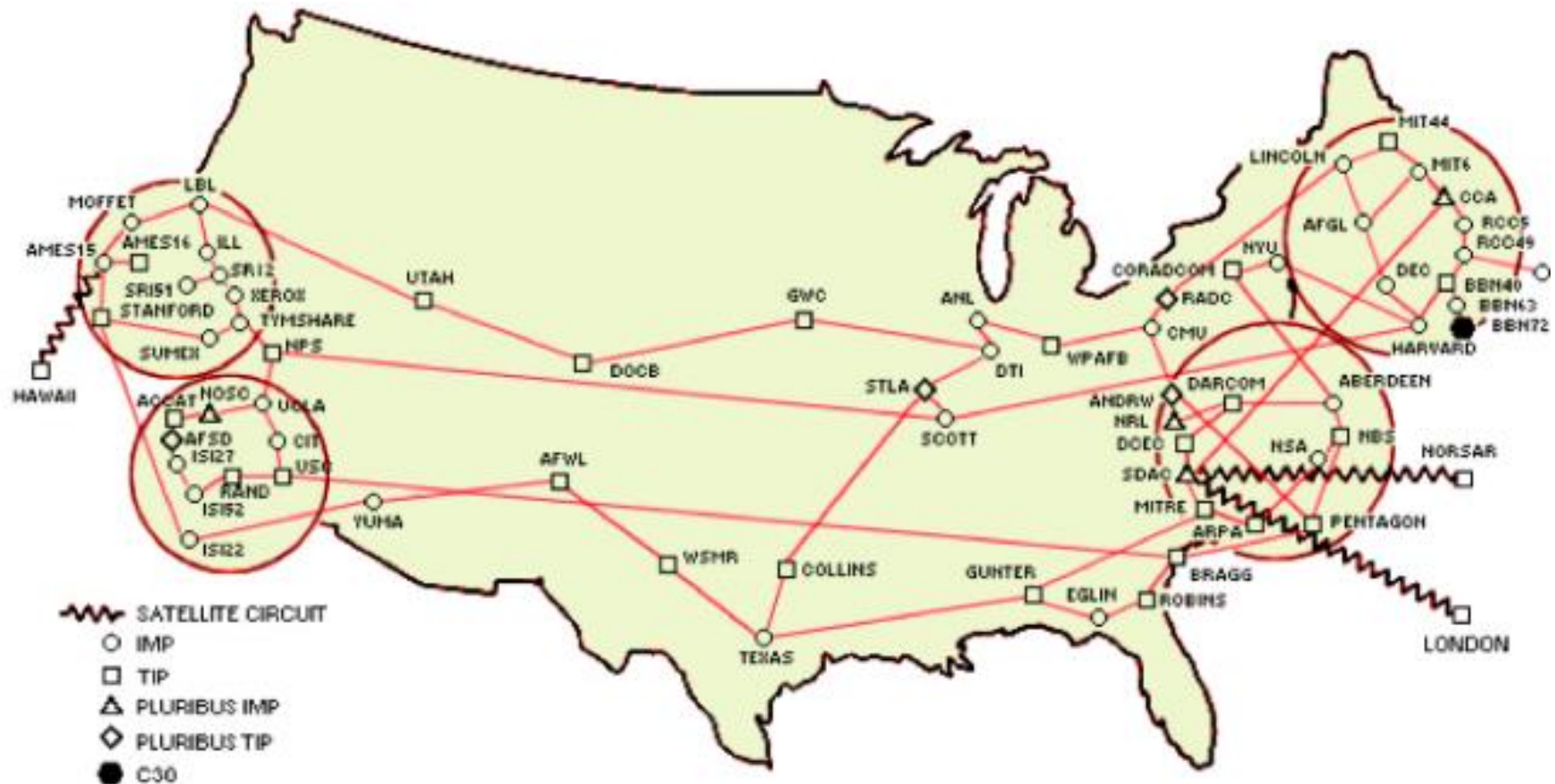
GUERRA FRIA





MAP 4 September 1971

ARPANET GEOGRAPHIC MAP, OCTOBER 1980



Como tudo começou?

- A Web foi criada em 1989, por Tim Berners-Lee, para ser um padrão de publicação e distribuição de textos científicos e acadêmicos.
- Atualmente Tim é diretor do W3C (*World Wide Web Consortium*), além de pesquisador e professor de Ciências da Computação.

CRIAÇÃO DA WEB



- **A WEB → CRIADA EM 1989**
- **TIM BERNERS-LEE**

HTML

HyperText Markup Language

(Linguagem de Marcação de Hipertexto)

É uma linguagem de marcação para produzir páginas Web.

HTML

- O HTML sofreu várias mudanças em suas diferentes versões, com o objetivo de estender a linguagem para que pudesse acompanhar a evolução da Web e das tecnologias nela inseridas.
- Versões do HTML:
 - HTML
 - HTML+
 - HTML 2.0
 - HTML 3.0
 - HTML 3.2
 - HTML 4.0
 - HTML 4.01
 - HTML 5 (versão em fase de desenvolvimento)

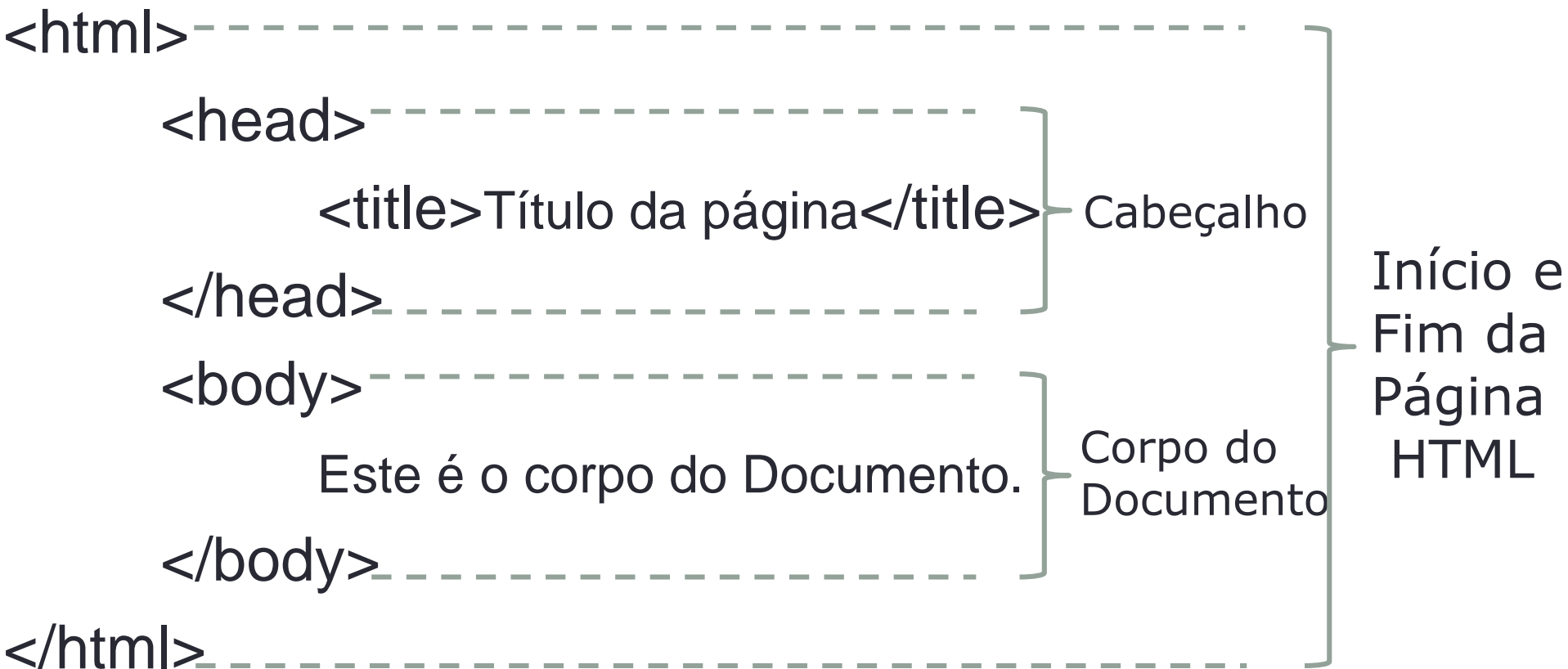
HTML

- Compõe a estrutura de uma página Web através de etiquetas (tags) e atributos.



<TAG ATRIBUTO1="" ATRIBUTO2=""> Texto </TAG>

Estrutura básica



Cabeçalho: contém informações sobre o documento.

Corpo do Documento: contém informações que serão exibidas pelo navegador, tais como parágrafos, imagens, listas, entre outros.

Cabeçalhos

- O HTML possui a etiqueta `<h>` que cria cabeçalhos na área de visualização das páginas.
- Existem seis tipos de cabeçalhos indo de `<h1>` a `<h6>`. Sendo `<h1>` de tamanho maior que `<h6>`.

Cabeçalhos

```
<html>
```

```
  <head>
```

```
    <title> Cabeçalhos </title>
```

```
  </head>
```

```
  <body>
```

```
    <h1> Cabeçalho de nível 1 </h1>
```

```
    <h2> Cabeçalho de nível 2 </h2>
```

```
    <h3> Cabeçalho de nível 3 </h3>
```

```
    <h4> Cabeçalho de nível 4 </h4>
```

```
    <h5> Cabeçalho de nível 5 </h5>
```

```
    <h6> Cabeçalho de nível 6 </h6>
```

```
  </body>
```

```
</html>
```

Como é visualizado no Browser

Cabeçalho de nível 1

Cabeçalho de nível 2

Cabeçalho de nível 3

Cabeçalho de nível 4

Cabeçalho de nível 5

Cabeçalho de nível 6

Cabeçalho

Possui o atributo **align** que faz o alinhamento do texto.

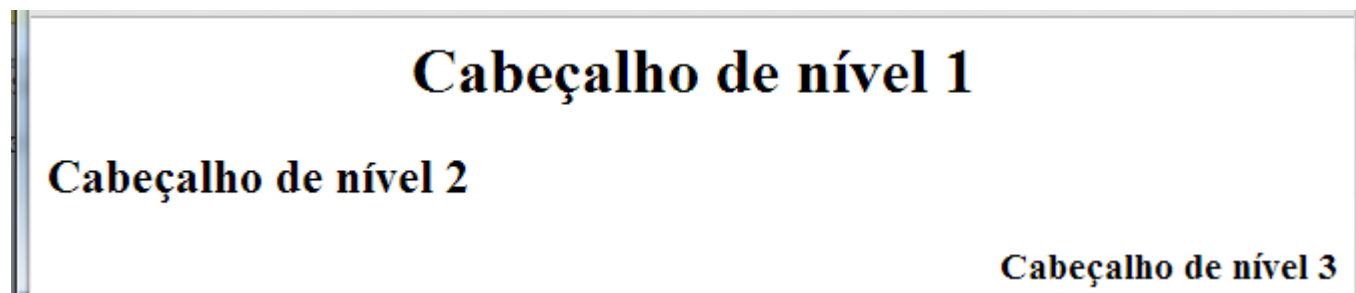
Tipos:

atributo	função
left	→ alinha o texto à esquerda
right	→ alinha o texto à direita
center	→ centraliza o texto
justify	→ justifica o texto

Exemplo

```
<html>  
  <head>  
    <title> Alinhamento de Cabeçalhos </title>  
  </head>  
  <body>  
    <h1 align="center"> Cabeçalho de nível 1 </h1>  
    <h2 align="left"> Cabeçalho de nível 2 </h2>  
    <h3 align="right"> Cabeçalho de nível 3 </h3>  
  </body>  
</html>
```

Visualização

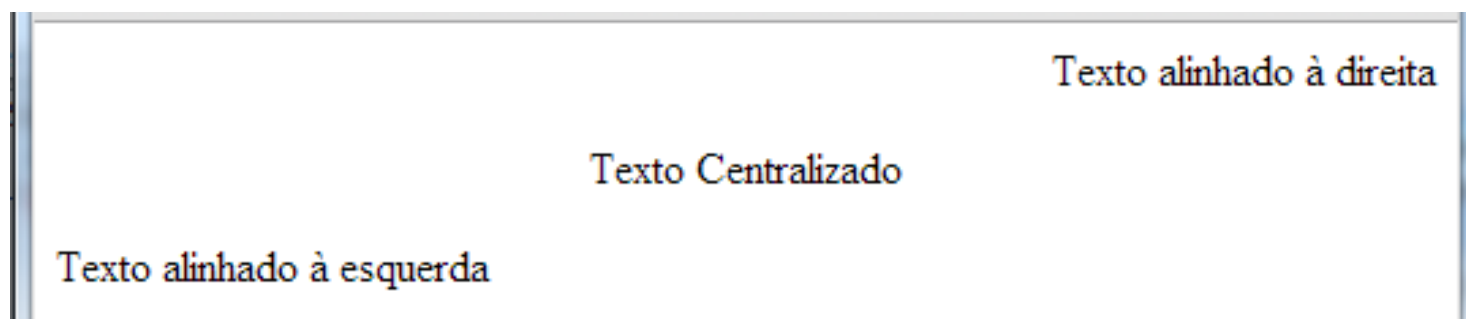


Parágrafos

- Existem duas etiquetas que permite uma quebra de linha, são elas: `
` e `<p>`.
- `
` Este elemento quebra uma linha em determinado ponto, dando continuidade ao texto em outra linha.
- `<p> ... </p>` Faz a separação entre os parágrafos. Assim como o comando `<h>`, ele também possui o atributo **align**.

Exemplo do elemento `<p>`

Visualização



Comando

`<p align="right">` Texto alinhado à direita `</p>`

`<p align="center">` Texto Centralizado `</p>`

`<p align="left">` Texto alinhado à esquerda `</p>`

Elemento ``

Altera o tamanho, cor e tipo da fonte.
Possui os atributos **size**, **color** e **face**.

atributo	função
size	→ tamanho da fonte
color	→ cor da fonte
face	→ tipo de fonte

Exemplo do elemento ``

Visualização

Esta frase está no tamanho 7 cor azul e tipo de letra Arial Black.

Comandos

``

Esta frase está no tamanho 7 cor azul e tipo de letra Arial Black.

``

ou

``

Esta frase está no tamanho 7 cor azul e tipo de letra Arial Black.

``

Formatação de Texto

elemento	semântica
strong	→ ênfase forte
<i>em</i>	→ itálico
<u>u</u>	→ sublinhado

Exemplo – formatação de texto

Visualização

formatacao.html

Texto com ênfase forte

Texto com ênfase

Texto com ênfase e ênfase forte

Comando

**** Texto com ênfase forte **** **
**

**** Texto com ênfase **** **
**

**** **** Texto com ênfase e ênfase forte **** ****

Lista não Ordenada - ``

Comando

Lista não ordenada

``

`` item 1 ``

`` item 2 ``

`` item 3 ``

`` item 4 ``

``

Visualização

Lista não ordenada

- item 1
- item 2
- item 3
- item 4

Lista Ordenada - ``

Visualização

Lista Ordenada

1. item 1
2. item 2
3. item 3
4. item 4

Comando

Lista Ordenada

``

`` item 1 ``

`` item 2 ``

`` item 3 ``

`` item 4 ``

``

Inserindo imagem – elemento ``

atributo

função

src

→ localização da imagem

alt

→ texto alternativo

height

→ altura de exibição da imagem

width

→ largura de exibição da imagem

Exemplo do elemento ``

Visualização

localização e nome da imagem → “imagens/logohtml.jpg”

Inserindo imagens



Comando

`<h1>` Inserindo imagens `</h1>`

``

Inserindo imagem de fundo

Exemplo do atributo `<background>`

Imagem

localização e nome da imagem → “imagens/fundo.jpg”



Comando

```
<body background="imagens/fundo.jpg">
```

Links

- A principal atração da internet é a criação de documentos com o conceito de hipertexto, ou seja, um documento que se vincula a outros documentos por meios de ligações especiais chamadas links ou hiperlinks.
- Com esse conceito, você pode criar documentos que façam referencias e permitam ao usuário acessar tais referencias não importando se elas estão em outra página da web, no seu micro ou em algum servidor da rede.

Links - `<a href>`

Comando

Lista de links

`<p>`

`` Página 1 `` `
`

`` Página 2 `` `
`

`` Página 3 `` `
`

`` Página 4 `` `
`

`</p>`

Visualização

Lista de links

[Página 1](#)

[Página 2](#)

[Página 3](#)

[Página 4](#)

Exercício

Web Semântica

Evoluindo a já clássica e conhecida “Web de documentos”, o W3C ajuda no desenvolvimento de tecnologias que darão suporte à “Web dos dados”, viabilizando pesquisas como num banco de dados.

O objetivo final da Web de dados é possibilitar com que computadores façam coisas mais úteis e com que o desenvolvimento de sistemas possa oferecer suporte a interações na rede. O termo “Web Semântica” refere-se à visão do W3C da Web dos Dados Linkados.

A Web Semântica dá às pessoas a capacidade de criarem repositórios de dados na Web, construírem vocabulários e escreverem regras para interoperarem com esses dados.

Fonte: www.w3c.br/Padroes/WebSemantica

Tabela <table>

elem./atri.	semântica/função
caption	→ legenda da tabela
tr	→ linha composta por célula de dados ou cabeçalho
th	→ célula contém informações de cabeçalho
td	→ célula de dados
border	→ definição em pixels da largura da borda em torno da tabela
height	→ altura para as colunas
width	→ largura para as colunas

Tabelas – exemplo 1

Visualização

primeiro exemplo

Coluna 1	Coluna 2
teste	teste
teste	teste

Comandos:

```
<table border="1">
```

```
  <caption>primeiro exemplo</caption>
```

```
  <tr><th> Coluna 1 </th><th> Coluna 2 </th></tr>
```

```
  <tr><td> teste </td><td> teste </td></tr>
```

```
  <tr><td> teste </td><td> teste </td></tr>
```

```
</table>
```


Tabela – agrupamento de células

Títulos compreendendo mais de uma coluna ou linha

atributo	função
rowspan	→ agrupamento vertical das células
colspan	→ agrupamento horizontal das células

Tabela – exemplo 2

Visualização

Segundo Exemplo

colunas 1 e 2	
teste	teste
teste	teste
3 linhas	uma linha
	duas linhas
	três linhas
teste	teste

Comandos:

```
<table border="1">  
  <caption>Segundo Exemplo</caption>  
  <tr><th colspan="2">colunas 1 e 2</th></tr>  
  <tr><td> teste </td><td> teste </td></tr>  
  <tr><td> teste </td><td> teste </td></tr>  
  <tr><td rowspan="3">3 linhas</td><td>uma linha</td></tr>  
  <tr><td> duas linhas </td></tr>  
  <tr><td> tres linhas </td></tr>  
  <tr><td> teste </td><td> teste </td></tr>  
</table>
```

Tabela - alinhamento

As linhas e células podem ter alinhamentos definidos através dos atributos:

atributo	função
align	→ alinhamento horizontal
valign	→ alinhamento vertical

Tabela - alinhamento

Tipos de alinhamento do texto na célula.

atributo	tipo	descrição
align	→ left	alinhamento à esquerda
	→ right	alinhamento à direita
	→ center	centraliza horizontalmente
	→ justify	justificado
valign	→ top	alinhamento na parte superior
	→ middle	centralizado verticalmente
	→ bottom	alinhamento na parte inferior

Tabela – exemplo 3

Visualização

Terceiro Exemplo

Alinhamento horizontal esquerdo e vertical superior
Alinhamento horizontal e vertical centralizado
Alinhamento horizontal direito e vertical inferior

Tabela – exemplo 3

comandos:

```
<table align="center" height="300" width="600" border="1">  
  <caption>Terceiro Exemplo</caption>  
  <tr><td align="left" valign="top">  
    Alinhamento horizontal esquerdo e vertical superior  
  </td>  
</tr>  
  <tr><td align="center" valign="middle">  
    Alinhamento horizontal no centro e vertical no meio  
  </td>  
</tr>  
  <tr><td align="right" valign="bottom">  
    Alinhamento horizontal direito e vertical inferior  
  </td>  
</tr>  
</table>
```

PADRÕES WEB

A Bagunça

- A Web cresce comercialmente, e veem a necessidade de publicação de conteúdo diagramado, como em revistas e jornais (algo para o qual o HTML não possuía recursos).
- Há então a adaptação da linguagem de forma equivocada, com uma preocupação unicamente visual, esquecendo da qualidade do código.

Com tudo isso...

- Cada arquivo de um site acaba sendo um incompreensível emaranhado de Tags com tabelas, formatações, scripts...
- Tudo muito maior que o necessário.
- Dessa forma, para que seja feita alguma alteração no site, é necessário alterar todos os arquivos...um a um!

O resultado de tudo isso...

Dificuldade de
Desenvolvimento
E Manutenção

Código sujo e
incompreensível

Falta de
padrão

spacer.gif

Formatação e
Estrutura juntas

<table>

Os Padrões Web

- Padrões *Web* ou *Web Standards* são um conjunto de normas, diretrizes, recomendações, notas, artigos, tutoriais e afins de caráter técnico, produzidos pelo W3C e destinados a orientar fabricantes, desenvolvedores e projetistas para o uso de práticas que possibilitem a criação de uma *Web* acessível a todos, independentemente dos dispositivos usados ou de suas necessidades especiais.

HTML5 - XHTML – HTML – CSS – DOM – XML ...

O que é W3C?

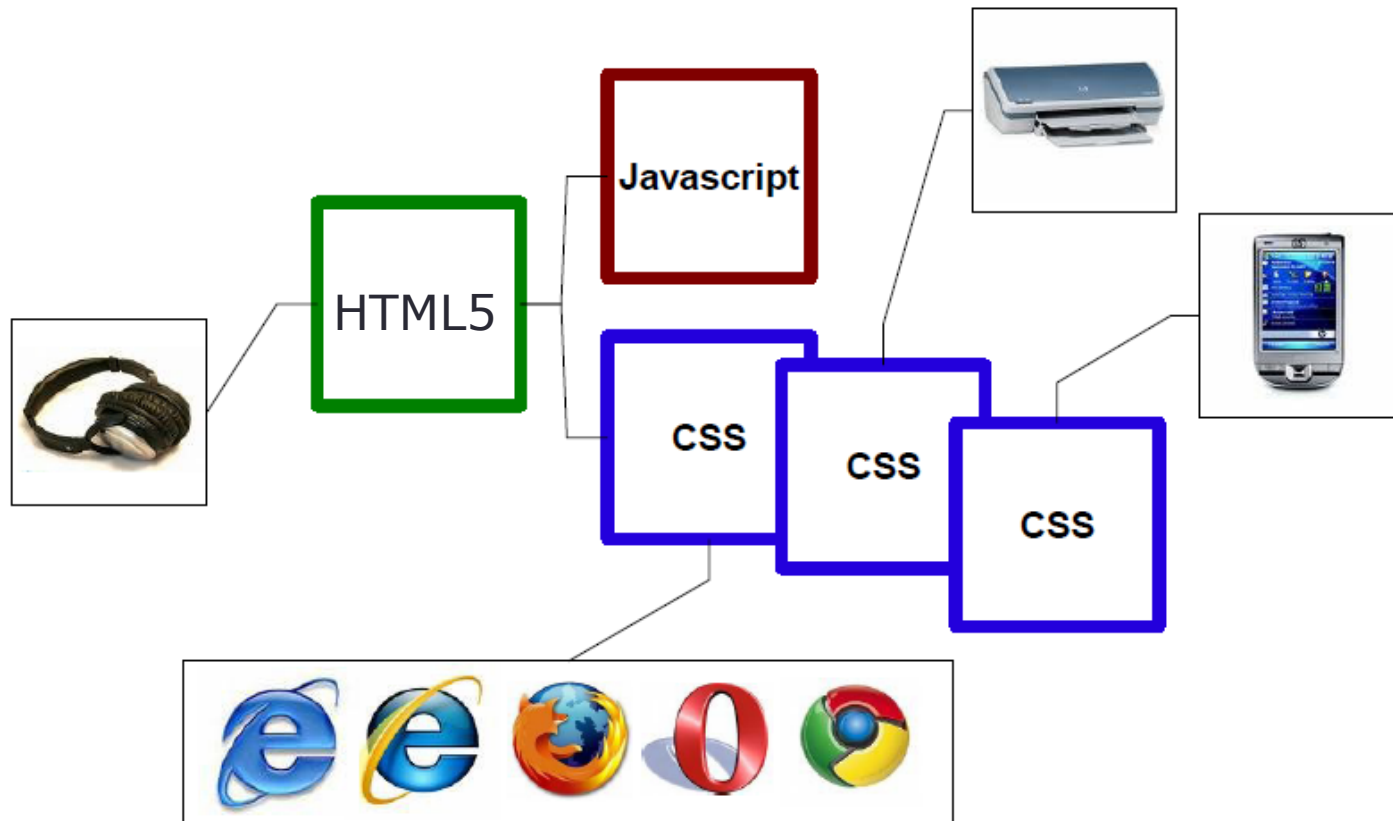
- W3C (**World Wide Web Consortium**) é um consórcio de empresas de tecnologia. Fundado por Tim Berners-Lee em 1994 para levar a Web ao seu potencial máximo.
- Desenvolve Recomendações abertas, até agora mais de 80 padrões...
- www.w3.org



Mudança de Conceito...

- Web dividida em três camadas:
 - Conteúdo (HTML5)
 - Apresentação (CSS)
 - Comportamento (Javascript)
- Devem ser desenvolvidas de forma independente, porém uma complementa a outra.

Mudança de Conceito...



Um Código Semântico

- Semântica refere-se ao estudo do significado.
- Quando utilizamos cada marcação para o que ela realmente foi criada, estamos construindo um “Código Semântico”.
- Parar de pensar em “isso vem aqui, isso vai ali...” e pensar na “Estrutura da Informação”
- Criar uma “Marcação com Significado”.



HTML 5

- HTML 5 é a nova versão do HTML4.
- Surgiu da necessidade de:
 - ser um diferencial para a semântica do código,
 - separação da estrutura do código com a formatação, e
 - princípios de acessibilidade.

HTML 5

- É uma cooperação entre a **W3C** (*World Wide Web Consortium*) e da **WHATWG** (*Web Hypertext Application Technology Working Group*).
- **W3C**: Consórcio Internacional, composto por empresas e organizações que visam desenvolver padrões para conteúdos web
- **WHATWG**: Grupo fundado por desenvolvedores da Mozilla, Opera e Apple em 2004, com a proposta de Atualizar o XHTML para a Versão 5 do HTML.

HTML 5

Principal Objetivo:

- Facilitar a manipulação do elemento possibilitando o desenvolvedor a modificar as características dos objetos de forma não intrusiva e de maneira que seja transparente para o usuário final.

Recomendações para utilização do HTML5

- Todas as tags devem ser escritas com letras minúsculas

- **Não recomendado:**

`<DIV><P>Aqui um texto!</P></DIV>`

- **Recomendado:**

`<div><p>Aqui um texto!</p></div>`

Recomendações para utilização do HTML5

- Tags devem estar convenientemente aninhadas

- **Não recomendado:**

```
<div><p>Aqui um <em>texto!</p></em></div>
```

- **Recomendado:**

```
<div><p>Aqui um <em>texto! </em></p></div>
```

Recomendações para utilização do HTML5

- Uso de tags de fechamento é obrigatório

- **Não recomendado:**

`<p>Um parágrafo.`

`Um item de lista.`

- **Recomendado:**

`<p>Um parágrafo.</p>`

`Um item de lista. `

Recomendações para utilização do HTML5

- Elementos vazios devem ser fechados
- **Não recomendado :**

`
`

``

- **recomendado:**

`
`

ou `
 </br>` (*não é de uso corrente*)

``

Diferenças entre HTML5 e HTML

- Valores dos atributos.

- **Não recomendado:**

`<td ROWSPAN=3>`

`<input checked>`

- **Recomendado:**

`<td rowspan="3">`

`<input checked="checked">`

Novos Recursos

- Novos Elementos `<header>`, `<article>`, `<section>`, `<nav>` e `<footer>` - estruturar o conteúdo;
- Novos Atributos para formulário que permite validação de dados;
- Elemento `<canvas>` - desenhos e gráficos;
- Elementos `<audio>` e `<video>` - reprodução de mídia;
- Novas APIs:
 - *Manipulação de conteúdo offline, geolocalização, acesso a banco de dados, validação formulário, manipulação de vídeo e áudio, criação de gráficos etc.*

Estrutura da página – HTML

```
1 <html>
2   <head>
3     <title>Título da página</title>
4   </head>
5   <body>
6     Este é o corpo do Texto.
7   </body>
8 </html>
```

Estrutura da página – XHTML 1.0

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
2     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">  
3 <html xmlns="http://www.w3.org/1999/xhtml">  
4   <head>  
5     <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>  
6     <title> Título da Página </title>  
7   </head>  
8   <body>  
9     Conteúdo da página  
10  </body>  
11 </html>
```

Estrutura da página – HTML 5

```
1  <!DOCTYPE HTML>
2  <html lang="pt-br">
3    <head>
4      <meta charset="UTF-8">
5      <link .../>
6      <script ... />
7      <title> Título da Página </title>
8    </head>
9    <body>
10     Conteúdo da página
11  </body>
12 </html>
```

Novos Elementos e Atributos

- Em versões anteriores do HTML não era possível diferenciar automaticamente o rodapé do cabeçalho.
- Essa diferenciação era percebida visualmente pelo layout pronto.
- As tags utilizadas poderiam ser iguais e não haviam padrões para nomenclatura de IDs e Classes.

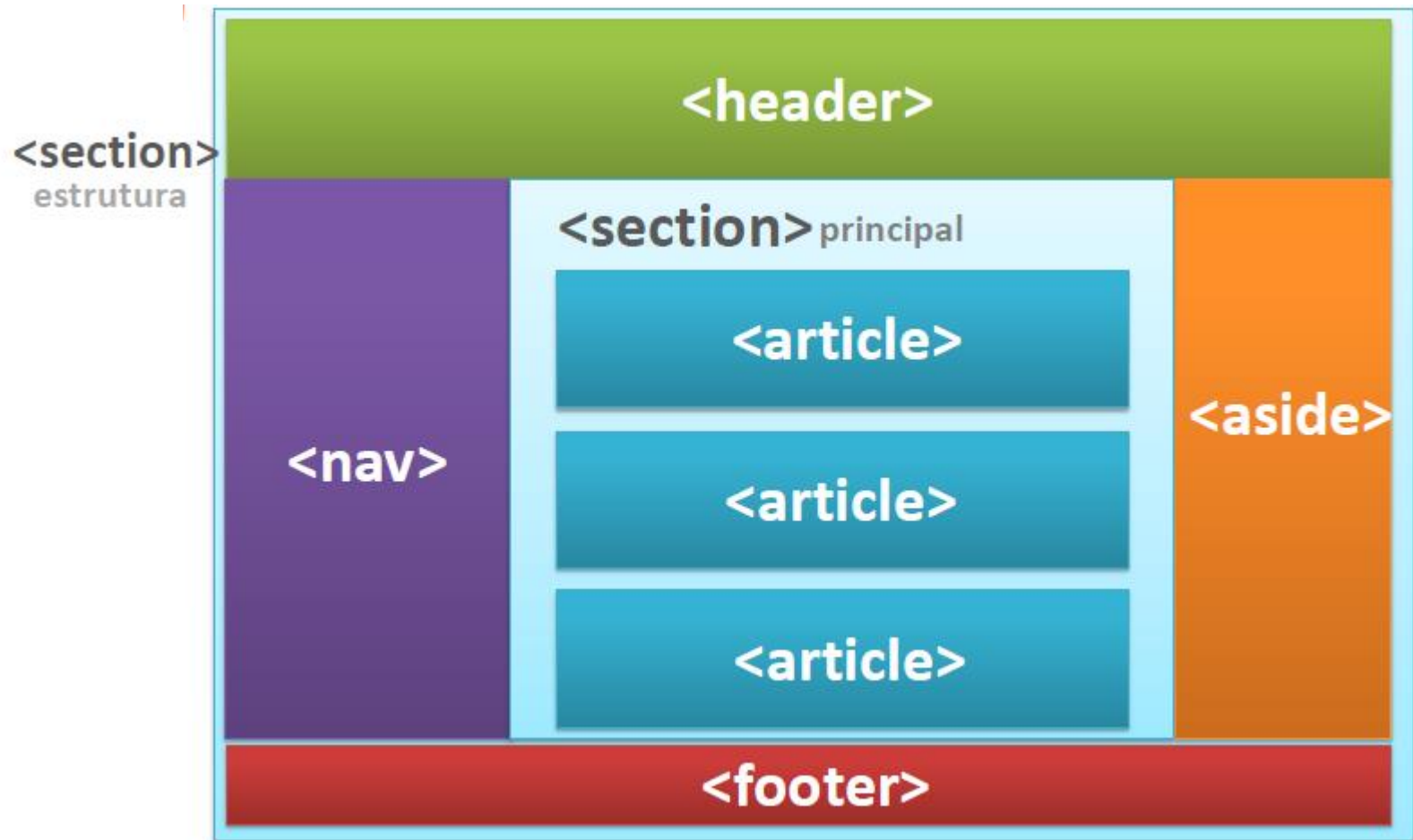
Novos Elementos e Atributos

- O HTML5 trouxe elementos que definem setores principais no documento HTML, como sidebar, rodapé e cabeçalho, área de conteúdo.
- Com isso, os buscadores conseguem vasculhar o código de maneira mais eficaz.

Elementos de Estruturação

- <section>** : permite definir o layout da página em seções que podem ser identificadas por um atributo id.
- <header>** : permite estabelecer a área de cabeçalho da página.
- <nav>** : agrupa elementos de navegação da página (menu).
- <article>** : permite definir blocos para conteúdos de texto na página.
- <aside>** : Representa a parte do conteúdo que pouco se refere com a página.
- <footer>** : indica a área de rodapé da página.

Exemplificando



Futuro da Web - HTML5

HTML 5 [HyperText Markup Language]



section.container

header

nav

article

hgroup

p

aside

section

aside

footer

**Novos
elementos
(semânticos)**

HTML5 + CSS

CSS

Cascading Style Sheets



CSS: Cascading Style Sheets

- Cascading Style Sheets (ou simplesmente CSS) é uma linguagem de estilo utilizada para definir a apresentação de documentos escritos em uma linguagem de marcação, como HTML ou XML.
- Seu principal benefício é prover a separação entre o formato e o conteúdo de um documento.
- Pode-se definir apresentações específicas para diferentes dispositivos (Tvs, Celulares, Impressoras, etc) apenas criando folhas de estilo alternativas.



CSS: Cascading Style Sheets

- CSS 1.0 ... CSS level 1
- Em setembro de 1994 surgiu a primeira proposta
- Em dezembro de 1996 lançada como Recomendação oficial do W3C
- CSS 2.0 ... CSS level 2
 - Publicada em maio de 1998
 - Última revisão: 12 de maio de 2008
- CSS 2.1 ... CSS level 2 revision 1
- Candidata a Recomendação oficial do W3C em 23 de abril de 2009
- CSS 3.0 ... CSS level 3
 - Proposta em 2001

Associar CSSs a documentos HTML

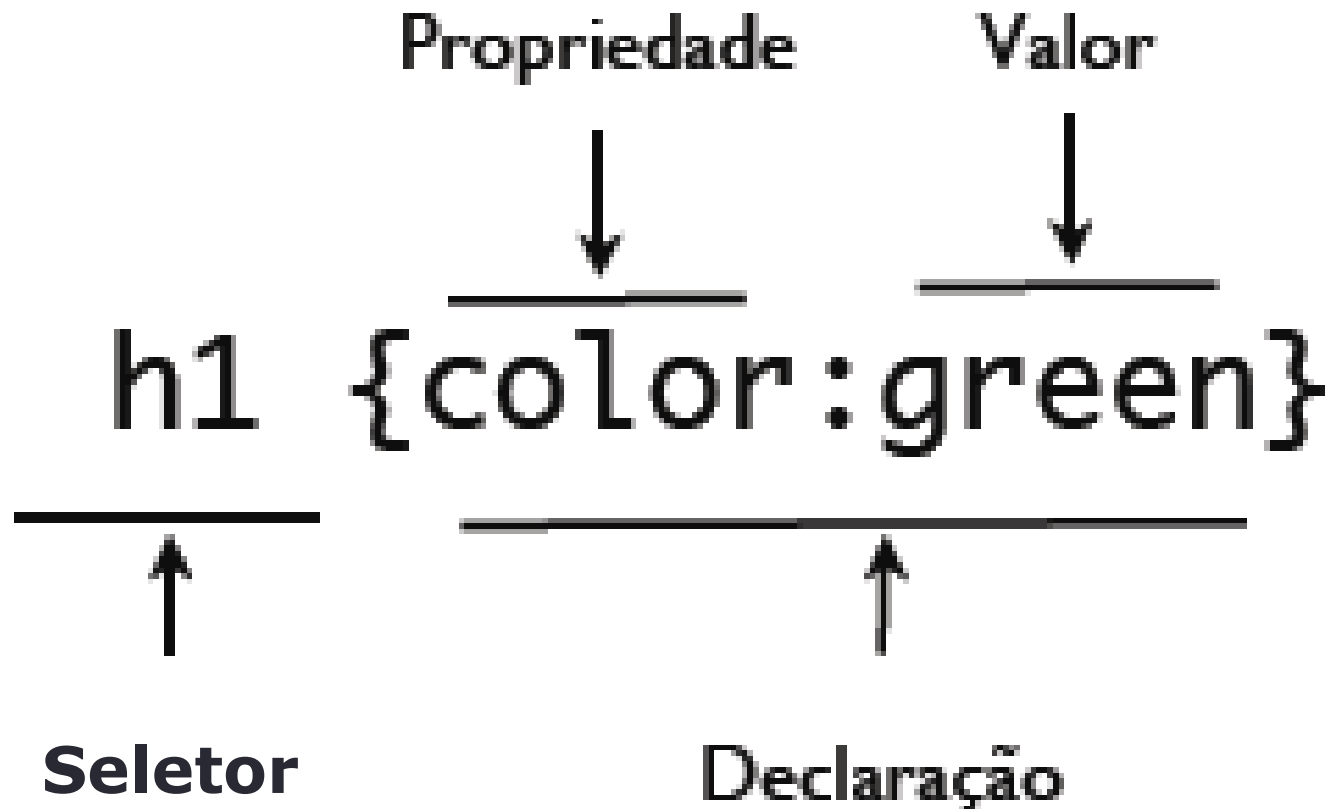
- **Inline**
 - Através do atributo style diretamente dentro de uma marcação
- **Interno/Incorporado**
 - Através da tag style entre as marcações <head></head> do documento HTML
- **Externo**
 - Criar um link (ligação) para uma página que contém os estilos.

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

CSS: Comentários

- Assim como em qualquer linguagem de programação, em CSS é possível adicionar comentários para melhor documentar o código.
- Exemplo:
`/ Comentário */`*

CSS: Cascading Style Sheets



Exemplo - CSS

- Parágrafo com cor preta, fundo amarelo, textos em itálico e alinhados à direita.

-
- **Comandos:** *salvar na pasta css como* **modelo1.css**

p {

color:#000;

background-color:#ff0;

font-style:italic;

text-align: right;

}

Exemplo

salvar como [exemplo1.html](#)

```
<!DOCTYPE HTML>
```

```
<html lang="pt-br">
```

```
  <head>
```

```
    <meta charset="UTF-8">
```

```
    <link rel="stylesheet" type="text/css" href="css/modelo1.css"/>
```

```
    <title> Título da Página </title>
```

```
  </head>
```

```
  <body>
```

```
    <p> Conteúdo da página </p>
```

```
  </body>
```

```
</html>
```

Visualização

Conteúdo da página

Atributo **media** do elemento **link**

- O atributo **media** informa qual o tipo de mídia deve ser aplicados aos estilos.
- Na ausência do tipo de mídia, os estilos serão aplicados para todas as mídias.

```
<link rel="stylesheet"  
type="text/css"  
href="estilo.css"  
media="screen,projection" />
```

Atributo **media** do elemento **link**

Valor	Mídia
-------	-------

screen	→ Telas de monitores.
---------------	-----------------------

tv	→ Dispositivos tipo televisão.
-----------	--------------------------------

projection	→ Projetores.
-------------------	---------------

handheld	→ Dispositivos portáteis.
-----------------	---------------------------

print	→ Impressoras e visualização no modo de impressão.
--------------	--

braille	→ Dispositivos táteis.
----------------	------------------------

aural	→ Sintetizadores de voz.
--------------	--------------------------

all	→ Todos os tipos de mídia.
------------	----------------------------

Regra CSS

- Quando o valor de uma propriedade for uma palavra composta, separada por espaços, deve-se usar sinais de (“”) ou, alternativamente, de (“”).

```
p {  
    font-family: “times new roman”;  
}
```

ou

```
p {  
    font-family: ‘times new roman’;  
}
```

Regra CSS

- Não se usam aspas em palavras compostas separadas por hífen:

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

Agrupamento de Seletores

Exemplo

```
h1 { color: red; }  
h2 { color: red; }  
p  { color: red; }
```

Agrupando os seletores

```
h1, h2, p { color: red; }
```

Unidades de medida absolutas

abreviatura**Unidade****in**

→ Polegada

cm

→ Centímetro

mm

→ Milímetro

pt→ Ponto. $1\text{pt} = 1/72 \text{ inch.}$ **pc**→ Pica. $1\text{pc} = 12 \text{ pt}$

Unidades de medida relativas

- As unidades de medida relativas são aquelas que tomam como base um valor de referência anteriormente definido.

abrev.	Unidade
--------	---------

em	→ é calculada em relação a um tamanho de fonte predefinido.
-----------	---

ex	→ é igual à altura da letra x minúscula da fonte adotada.
-----------	---

px	→ pixel
-----------	---------

Unidade *em*

salvar como **unidade_em.css**

```
h1 { font-size: 50px; }  
h2 { font-size: 40px; }  
h3 { font-size: 30px; }  
h4 { font-size: 20px; }  
h1, h2, h3, h4 { margin-left: 1em; }
```

salvar como **unidade_em.html**

```
<h1> Cabeçalho nível 1 </h1>  
<h2> Cabeçalho nível 2 </h2>  
<h3> Cabeçalho nível 3 </h3>  
<h4> Cabeçalho nível 4 </h4>
```

Visualização

Cabeçalho nível 1

Cabeçalho nível 2

Cabeçalho nível 3

Cabeçalho nível 4

CSS: Porcentagens

- São calculadas em relação a um valor preexistente, em geral uma unidade de medida.
- Consiste de um valor de 0 a 100 seguido do sinal de porcentagem (%).

Porcentagem– Exemplo

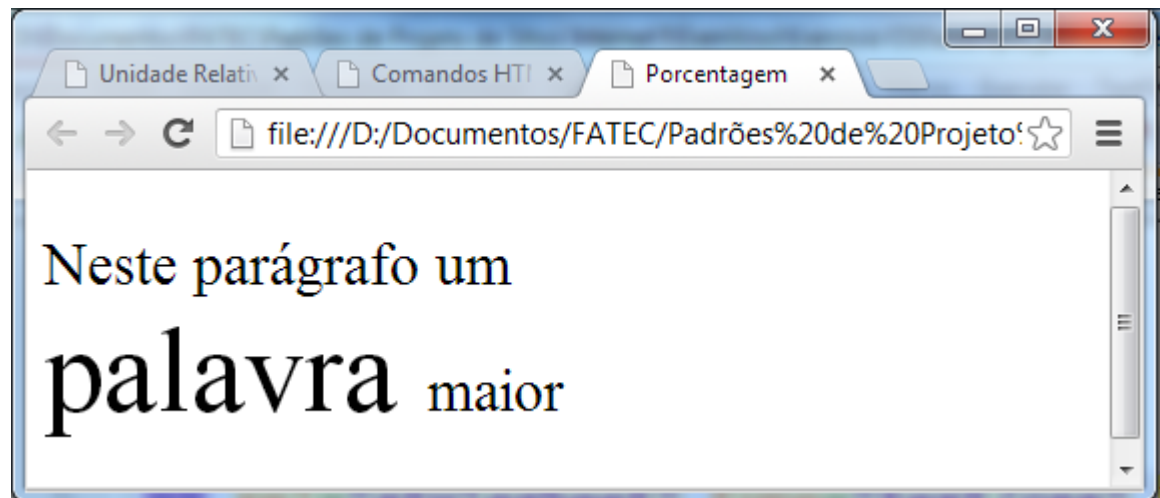
salvar como **porcentagem.css**

```
p    { width: 50%;  
      font-size: 30px;}  
span {font-size: 200%;}
```

salvar como **porcentagem.html**

<p>Neste parágrafo tem uma **** palavra **** maior **</p>**

visualização



CSS: Seletor classe

- Uma classe define a variação de um estilo.
- É referenciado através de uma ocorrência específica de um elemento utilizando o atributo **class**.
- Serve para definir vários estilos diferentes para o mesmo elemento.
- Definido pelo ■ (ponto)

Seletor classe – Exemplo 1

seletor.css

```
h4 { color: blue; }
```

```
.diferente { color: red; }
```

seletor_classe.html

```
<h4> Cabeçalho Azul</h4>
```

```
<h4 class="diferente">Cabeçalho Diferente Vermelho</h4>
```

visualização

Cabeçalho Azul

Cabeçalho Diferente - Vermelho

Seletor classe – Exemplo 2

seletor2.css

```
p { color: blue; }
```

```
p.cor-um { color: green; }
```

```
p.cor-dois { color: red; }
```

seletor_classe2.html

```
<p> Texto do parágrafo na cor azul </p>
```

```
<p class="cor-um"> Texto do parágrafo na cor verde </p>
```

```
<p class="cor-dois"> Texto do parágrafo na cor vermelha</p>
```

visualização

Texto do parágrafo na cor azul

Texto do parágrafo na cor verde

Texto do parágrafo na cor vermelha

CSS: Seletor id

- Semelhante ao Seletor classe.
- A diferença é que utiliza o atributo id, que serve para **identificar exclusivamente um determinado elemento** no documento.
- Definido pelo #

Seletor id – Exemplo

salvar como **seletor_id.css**

```
div#diferente { color: red;  
                border: 1px solid red }
```

salvar como **seletor_id.html**

```
<div id= “diferente”> seletor id </div>
```

visualização



seletor id

CSS – PSEUDO CLASSES

Descrição

- As Pseudo-classes são usadas na CSS para adicionar efeitos diferentes em alguns seletores, ou a uma parte de alguns seletores.
- Exemplo:
 - Personalizar cores de links em uma página, fazendo assim com que o padrão de cores seja seguido.

Sintaxe básica

- A Sintaxe das pseudo-classes:

***selector**:pseudo-class {property: value}*

- Classes CSS também podem ser usadas com pseudo-classes:

***selector.class**:pseudo-class {property: value}*

Pseudo-classes Âncora

- Um link ativo, visitado, não visitado, ou quando o mouse está sobre um link, podem todos ser exibidos de diferentes maneiras em um navegador que suporta CSS:

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

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

```
a:hover {color: #FF00FF} /* mouse sobre link */
```

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

Pseudo-Classes – Exemplo

pseudo_classe.css

```
a:link {color: blue} /* link */
```

```
a:visited {color: gray} /* link visitado */
```

```
a:hover {color: lightblue} /* mouse sobre link */
```

```
a:active {color: red} /* link selecionado */
```

pseudo_classe.html

```
<a href="http://www.fatecjales.edu.br/"> Fatec Jales </a>
```

```
<br/>
```

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

visualização

Fatec Jales

Google

Pseudo-classes Âncora

- **Observações:**

- **a:hover** DEVE vir após a:link e a:visited na definição do CSS de modo a ser efetivo!!
- **a:active** DEVE vir após a:hover na definição do CSS de modo a ser efetivo!!

Pseudo-classes e Classes CSS

- Pseudo-classes podem ser combinadas com classes CSS:

```
a.amarelo: hover {color: yellow}
```

```
<a class="amarelo" href="http://www.gmail.com/"> E-mail </a>
```

Se o link no exemplo acima for visitado, ele será exibido em amarelo.

Exercício

Web Semântica



Evoluindo a já clássica e conhecida “Web de documentos”, o W3C ajuda no desenvolvimento de tecnologias que darão suporte à “Web dos dados”, viabilizando pesquisas como num banco de dados.

O objetivo final da Web de dados é possibilitar com que computadores façam coisas mais úteis e com que o desenvolvimento de sistemas possa oferecer suporte a interações na rede. O termo “Web Semântica” refere-se à visão do W3C da Web dos Dados Conectados.

A Web Semântica dá às pessoas a capacidade de criarem repositórios de dados na Web, construírem vocabulários e escreverem regras para interoperarem com esses dados.

Fonte: www.w3c.br/Padroes/WebSemantica

PROPRIEDADES PARÁGRAFOS

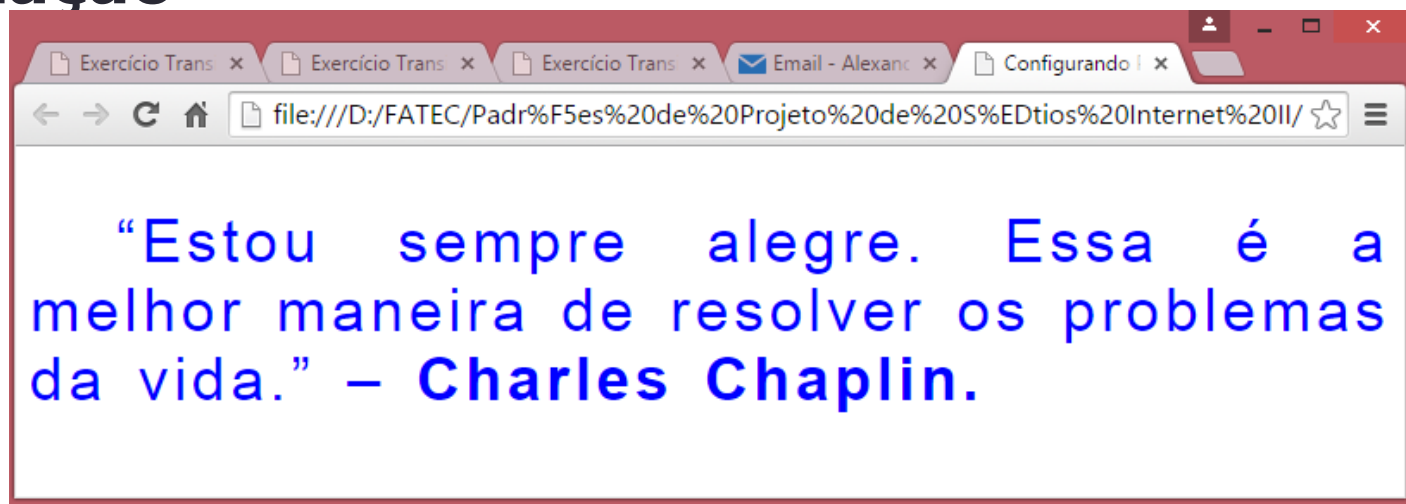
salvar como: **paragrafo.css**

```
p {  
  line-height: 40px; /* 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: 50px; /* tamanho da margem da primeira linha*/  
  text-align: justify; /* alinhamento do texto*/  
  font-size: 35 px; /* tamanho do texto*/  
  font-family: arial, calibri; /* tipo de letra do texto*/  
  color: blue; /* cor do texto*/  
}
```

salvar como **paragrafo.html**

`<p>` “Estou sempre alegre. Essa é a melhor maneira de resolver os problemas da vida.”
– `` Charles Chaplin. ``
`</p>`

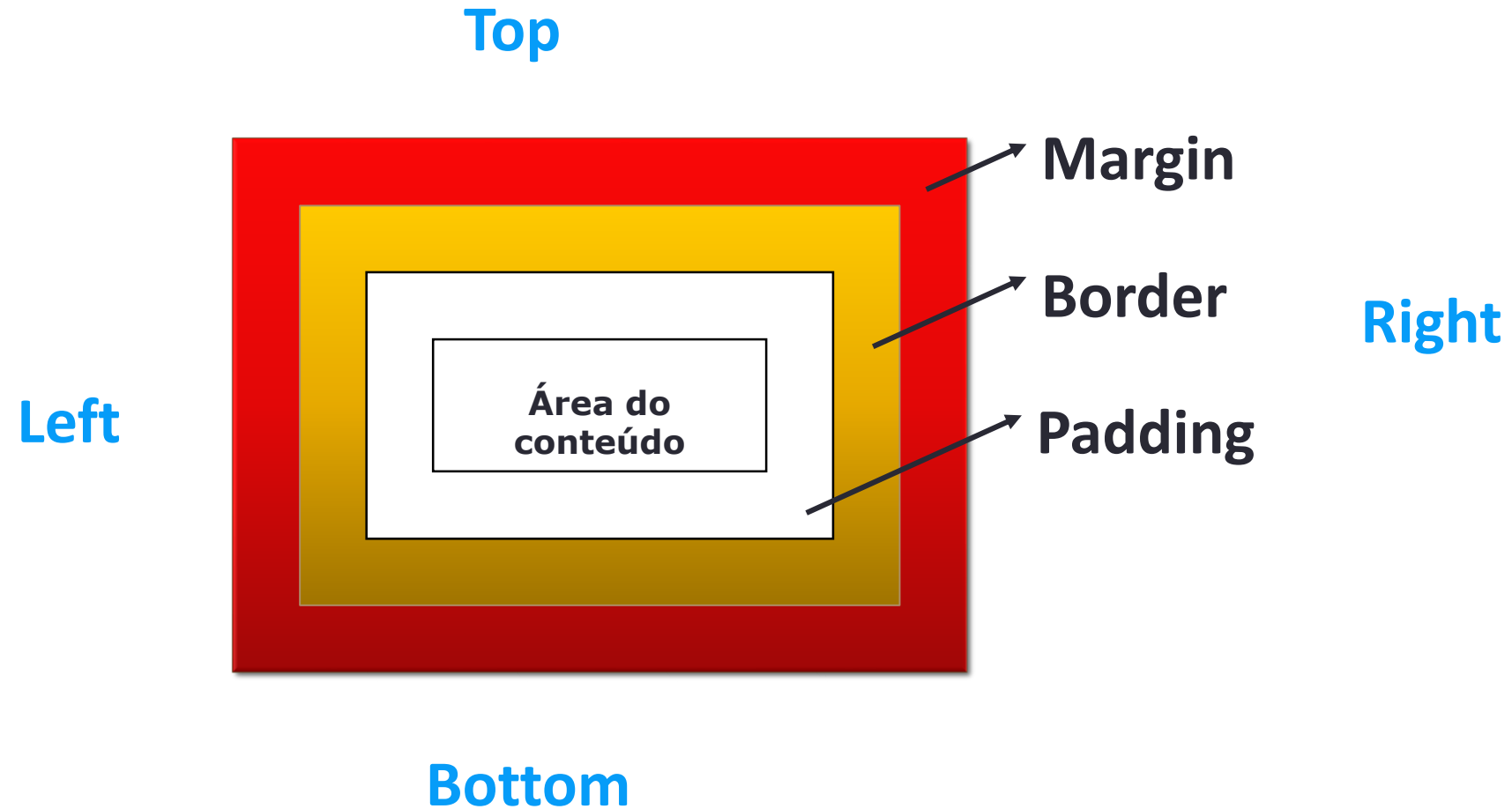
visualização



Box Model (Modelo de Caixa)

- O modelo de uma caixa determina a maneira como os elementos são exibidos e como eles se interagem.
- Cada elemento na página é considerado uma caixa retangular tendo o conteúdo do elemento composto do
 - **Padding (preenchimento),**
 - **Border (borda) e**
 - **Margin (margem)**

Ilustração do modelo de caixa



Modelo de caixa

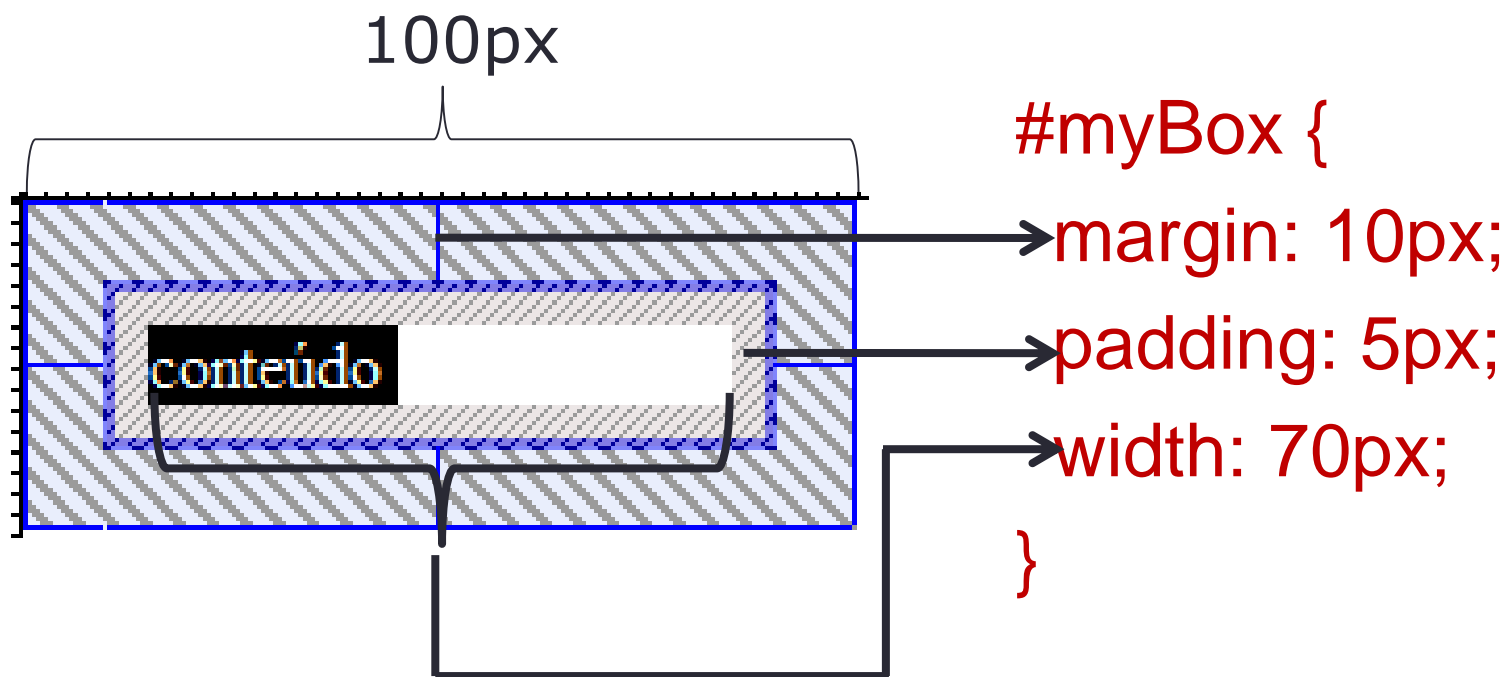
- **Padding** (Preenchimento)
 - É aplicado em torno da área de conteúdo.
 - Utilizado principalmente para criar um espaço em torno do conteúdo para que não encoste nas margens
- **Border** (Bordas)
 - Aplica uma linha em torno da área de preenchimento.
- **Margin** (Margens)
 - São transparentes
 - Utilizadas para controlar o espaçamento entre elementos

Modelo caixa

- Você pode sobrescrever esses estilos do navegador utilizando:

```
* {  
    margin:0;  
    padding:0;  
}
```


Modelo Caixa



Box Model (Caixa Modelo)

Margin (margem)

```
#caixa{ margin-top: 20px;  
        margin-right: 30px;  
        margin-bottom: 5px;  
        margin-left: 10px;}
```

OU

```
#caixa{ margin: 20px 30px 5px 10px;}
```

Estilos de bordas (border-style)

Estilo

Descrição

none

→ define espessura 0 para a borda.

hidden

→ o mesmo efeito de none, mas com precedência na resolução de bordas conflitantes

Borda dotted

Borda dashed

Borda solid

Borda double

Borda groove

Borda ridge

Borda inset

Borda outset

Box Model (Caixa Modelo)

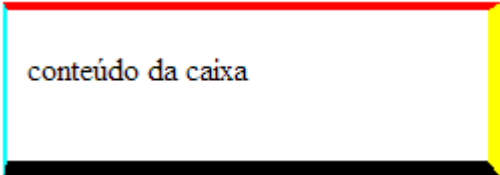
modelo_caixa.css

```
#caixa1{ margin: 20px 30px 5px 10px;      → margem cx.  
        padding: 20px 30px 5px 10px;     → preench. cx.  
        border-width: 4px 7px 9px 2px;   → comp. borda  
        border-color: red yellow black cyan; → cor borda  
        border-style: solid;             → estilo. borda  
        width: 200px;                    → largura caixa  
        height: 50px;                     → altura caixa  
}
```

modelo_caixa.html

```
<div id= "caixa1"> conteúdo da caixa </div>
```

visualização



conteúdo da caixa

Box Model (Caixa Modelo)

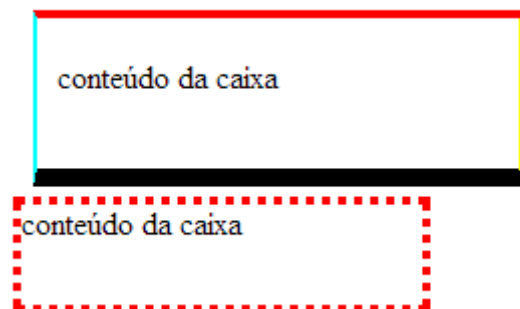
modelo_caixa.css

```
#caixa2{ border: 4px dotted red;  
        width: 200px;  
        height: 50px;  
}
```

modelo_caixa.html

```
<div id= "caixa1"> conteúdo da caixa </div>  
<div id= "caixa2"> conteúdo da caixa </div>
```

visualização



Modelo de Formatação visual

- Há três tipos de posicionamento na CSS:
 - **Fluxo Normal;**
 - **Elementos Flutuantes, e**
 - **Posicionamento Absoluto.**
- A menos que seja especificado, todas as caixas são inicialmente posicionadas no fluxo normal, ou seja, caixa será determinada pela posição no HTML5.

Exemplo: Posicionamento Normal

modelo_caixa_tipo.css

.caixa_normal{

margin:20px;

padding:10px;

width:140px;

float:left;

border:2px solid red;

text-align:center;

background-color:white;

}



Caixa Modelo – Posicionamento Normal

modelo_caixa_normal.html

`<div class="caixa_normal"> conteúdo da Caixa Normal 1 </div>`

`<div class="caixa_normal"> conteúdo da Caixa Normal 2 </div>`

`<div class="caixa_normal"> conteúdo da Caixa Normal 3 </div>`

visualização

conteúdo da Caixa
Normal 1

conteúdo da Caixa
Normal 2

conteúdo da Caixa
Normal 3

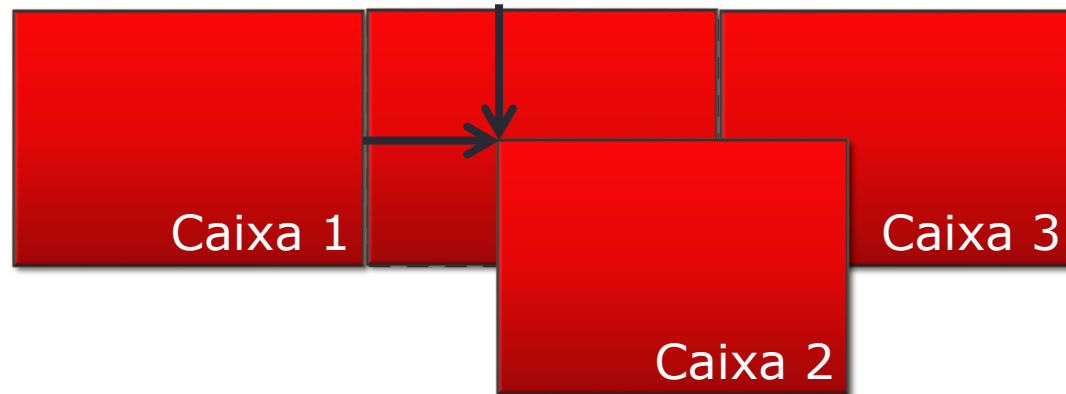
Posicionamento Relativo

- O elemento permanecerá no mesmo lugar onde está.
- Mas você poderá deslocar o elemento “em relação” ao seu ponto inicial (posição original) configurando uma posição vertical ou horizontal.

Exemplo: Posicionamento Relativo

Acrescentar no arquivo **modelo_caixa_tipo.css**

```
.caixa_relativa{  
  position:relative;  
  left:20px;  
  top:20px;  
  margin:20px;  
  padding:10px;  
  width:140px;  
  float:left;  
  border:2px solid red;  
  text-align:center;  
  background-color:white;  
}
```



Caixa Modelo – Posicionamento Relativo

modelo_caixa_relativa.html

`<div class="caixa_normal">` conteúdo da Caixa Normal 1 `</div>`

`<div class="caixa_relativa">` conteúdo da Caixa Relativa 1 `</div>`

`<div class="caixa_normal">` conteúdo da Caixa Normal 2 `</div>`

visualização

conteúdo da Caixa
Normal 1

conteúdo da Caixa
Relativa 1

conteúdo da Caixa
Normal 2

Posicionamento Absoluto

- O posicionamento absoluto remove o elemento do fluxo do documento, não ocupando, portanto, nenhum espaço.
- Outros elementos no fluxo normal do documento agirão como se o elemento de posicionamento absoluto nunca estivesse lá.
- É posicionado em relação a seu ancestral posicionado mais proximamente.

Exemplo: Posicionamento Absoluto

Acrescentar no arquivo **modelo_caixa_tipo.css**

```
.caixa_absoluta{  
  position: absolute;  
  left: 20px;  
  top: 20px;  
  margin: 20px;  
  padding: 10px;  
  width: 140px;  
  float: left;  
  border: 2px solid red;  
  text-align: center;  
  background-color: white;  
}
```



Caixa Modelo – Posicionamento Normal

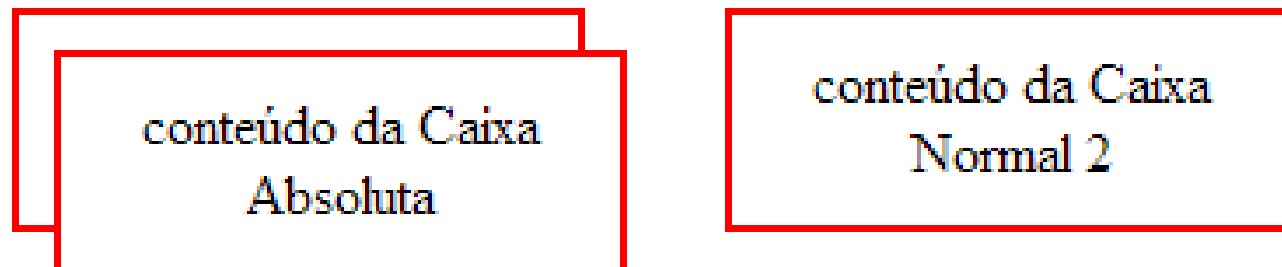
modelo_caixa_absoluta.html

`<div class="caixa_normal">` conteúdo da Caixa Normal 1 `</div>`

`<div class="caixa_normal">` conteúdo da Caixa Normal 2 `</div>`

`<div class="caixa_absoluta">` conteúdo da Caixa Absoluta `</div>`

visualização



Posicionamento Fixo

- O posicionamento fixo é uma subcategoria do posicionamento absoluto.
- A diferença é que o bloco de um elemento fixo é o ponto fixo de visualização.
- Isso permite criar elementos flutuantes que permanecem sempre na mesma posição na janela.

Exemplo: Posicionamento Fixo

Acrescentar no arquivo **modelo_caixa_tipo.css**

```
.caixa_fixa{
```

```
  position:fixed;
```

```
  left:70px;
```

```
  top:70px;
```

```
  margin:20px;
```

```
  padding:10px;
```

```
  width:140px;
```

```
  float:left;
```

```
  border:2px solid red;
```

```
  text-align:center;
```

```
  background-color:white;
```

```
}
```



Caixa Modelo – Posicionamento Normal

modelo_caixa_absoluta_fixa.html

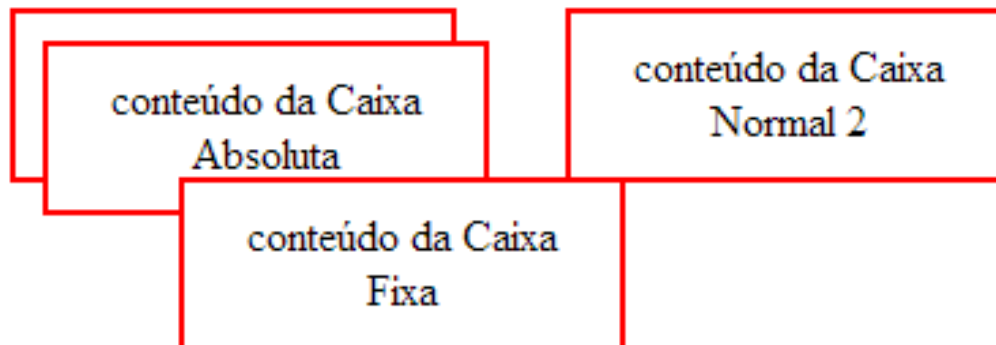
`<div class="caixa_normal">` conteúdo da Caixa Normal 1 `</div>`

`<div class="caixa_normal">` conteúdo da Caixa Normal 2 `</div>`

`<div class="caixa_absoluta">` conteúdo da Caixa Absoluta `</div>`

`<div class="caixa_fixa">` conteúdo da Caixa Fixa `</div>`

visualização



ELEMENTO FIGURE

salvar como: figure.css

```
figure {  
    background-color: #F2EDED;  
    border: 1px solid #ccc;  
    text-align: center;  
    width: 50%;  
    margin: 0px auto;  
}
```

```
figcaption {  
    font-style: italic;  
}
```

salvar como **figure.html**

```
<figure>
```

```
  
```

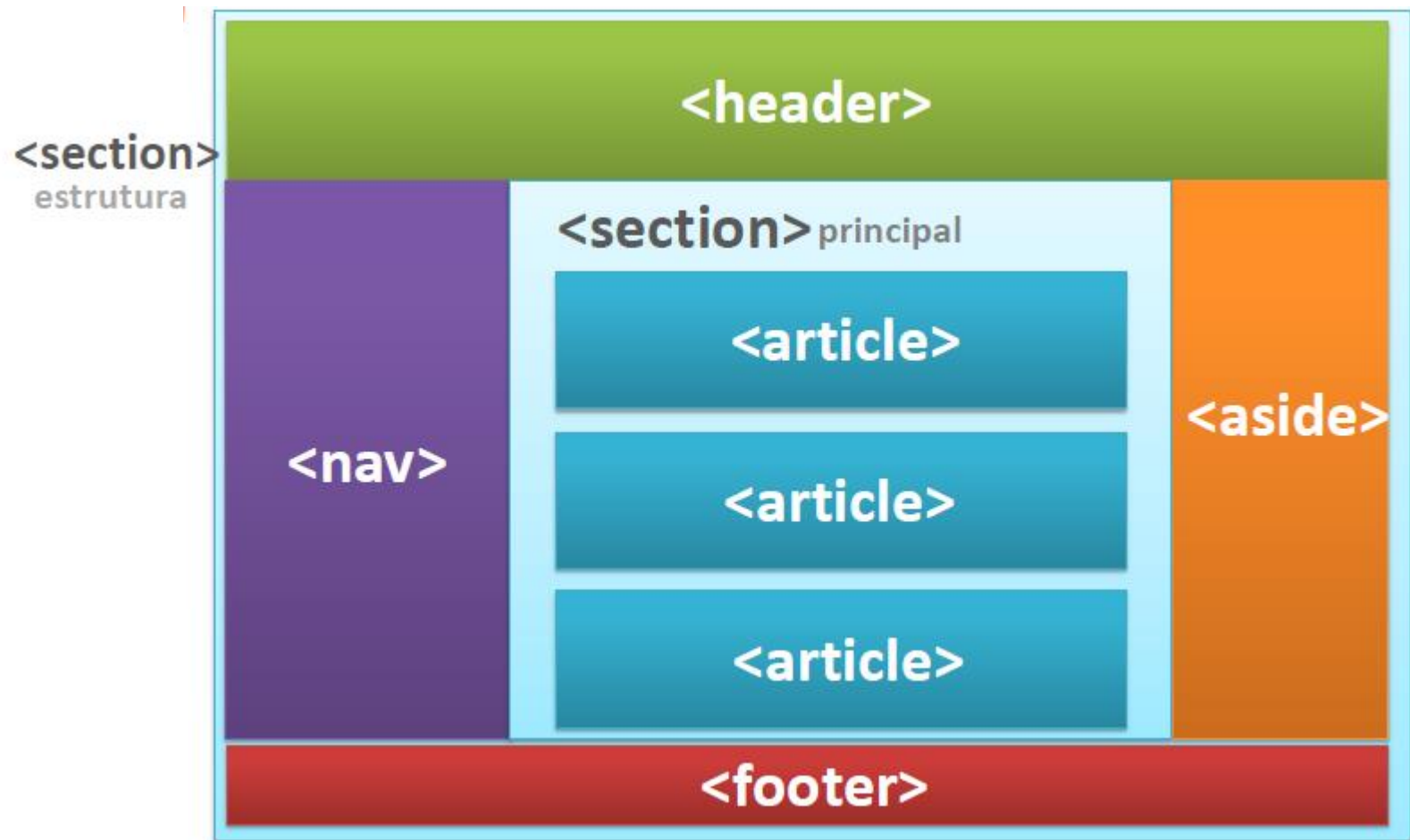
```
  <figcaption>Logo do HTML 5</figcaption>
```

```
</figure>
```

visualização



Exemplo da Estrutura Básica do HTML5



Exemplo

estrutura.css

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

```
section, article, header, footer, aside {
```

```
    border:3px solid black;
```

```
    margin: 10px;}
```

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

```
article{ background-color: green;}
```

```
aside{ height:80px;background-color: yellow;}
```

```
header, footer{
```

```
    height: 70px;
```

```
    background-color:blue;}
```

Exemplo_estrutura.html

<section>

<header>

<h1> Header (Cabeçalho) da Página</h1>

</header>

<article>

<h2> Título do Artigo </h2>

<section>

<h3> Subtítulo da Página - Nova Section</h3>

Texto da Section

<article>

Artigo dentro do section dentro do article

</article>

<aside>

Aside Subtítulo do Conteúdo Complementar

</aside>

</section>

</article>

<footer> footer </footer>

</section>

Visualização.



Validadores

- Validador de HTML, XHTML, XML e HTML5

<http://validator.w3.org>

- Validador de Folhas de Estilo CSS

<http://jigsaw.w3.org/css-validator>



Exemplo

estilo_validacao.css

```
p{ text-align:justify;  
    font-family:Arial, Helvetica, sans-serif;  
    font-size:20px;  
}  
  
p.direito{  
    text-align:right;  
}  
  
h1{color:red;  
}  
  
img{  
    float:left;  
}
```

Exemplo_validacao_erro.html

`<h1 align=center> HTML e CSS </h1>`

``

`<p>`

``HTML e CSS`` são as tecnologias fundamentais para o desenvolvimento de páginas Web: HTML (html e xhtml) para a estrutura, ``CSS para o estilo e leiaute, incluindo WebFonts.``

Pesquise recursos para melhores projetos de páginas web e ferramentas de ajuda.

`</p>`

`<p align="right"> fonte:`

`<a`

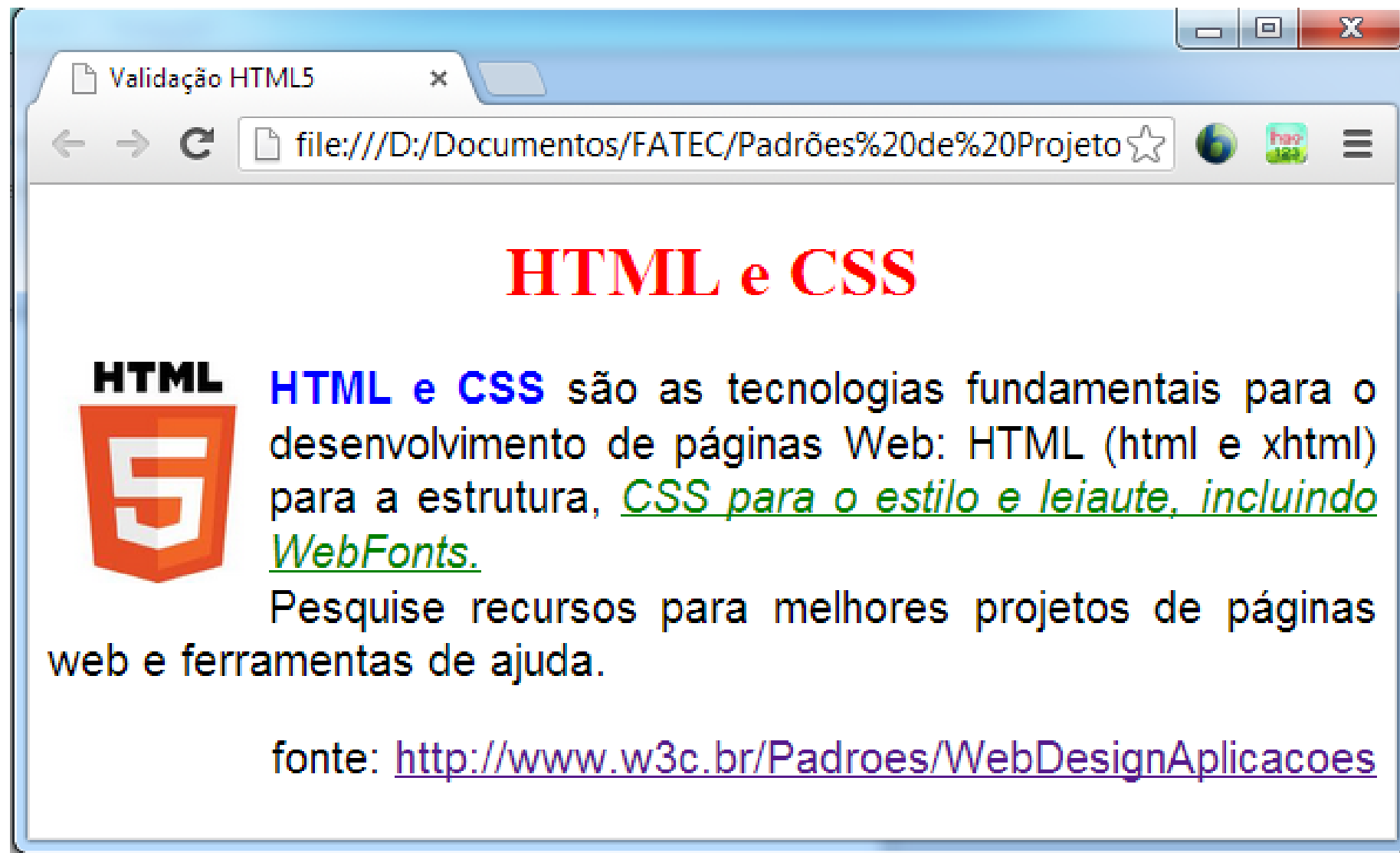
`href="http://www.w3c.br/Padroes/WebDesignAplicacoes">`
`http://www.w3c.br/Padroes/WebDesignAplicacoes`

<http://validator.w3.org>

Visualização



Visualização Correta



Exemplo

estilo_validacao.css

```
@charset "UTF-8";
```

```
p{ text-align:justify;
```

```
    font-family:Arial, Helvetica, sans-serif;
```

```
    font-size:20px;}
```

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

```
h1{ color:red;
```

```
    text-align:center; }
```

```
strong{color:blue;}
```

```
em{color:green;
```

```
    text-decoration: underline; }
```

```
img{float:left;
```

```
    height: 100px;
```

```
    width: 100px; }
```

Exemplo_validacao_correto.html

<h1> HTML e CSS </h1>

<p>

HTML e CSS são as tecnologias fundamentais para o desenvolvimento de páginas Web: HTML (html e xhtml) para a estrutura, CSS para o estilo e leiaute, incluindo WebFonts.

Pesquise recursos para melhores projetos de páginas web e ferramentas de ajuda.

</p>

<p class="direito"> fonte:

<a href=

"http://www.w3c.br/Padroes/WebDesignAplicacoes">http://www.w3c.br/Padroes/WebDesignAplicacoes

</p>

ÁUDIO E VÍDEO

Audio

- Elemento que destina-se a incorporar um som ou um stream de áudio.

salvar como **audio.html**

<audio>

<source src="audio/audio.mp3" type="audio/mpeg">

<p> Seu navegador não suporta o elemento áudio da HTML5

Faça download do audio.mp3

</p>

</audio>

Atributos do elemento Audio

Propriedade

Definição

src : caminho do arquivo de som

autoplay : inicia o som automaticamente

loop : repetir o som indefinidamente

controls : renderiza uma barra de controle nativa do navegador

Atributos do elemento Audio

Propriedade

Definição

preload: define como será o pré-carregamento do som na página. Tipos de valores:

none : não faz o pré-carregamento do som na página

metadata : pré-carregamento somente dos metadados do som, como dimensões, duração, controles etc

auto : define que o som será pré-carregado.

Audio

alterar **audio.html**

`<audio autoplay loop controls>`

`<source src="audio/audio.mp3" type="audio/mpeg">`

`<p>` Seu navegador não suporta o elemento áudio da HTML5

`
`

Faça `` download da audio.mp3``

`</p>`

`</audio>`

Audio

- **Observação:**

- A sintaxe XHTML5 por ser compatível com XML requer que os atributos booleanos, deve-se ser declarado:

`autoplay="autoplay"`

Não confundir atributo boolean com valor booleano.

INCORRETO

~~`controls="true"`~~

Vídeo

*Salvar como: **video.html***

```
<video autoplay loop controls>
```

```
  <!-- <source src="video/video.ogv" type="video/ogg">
```

```
    <source src="video/video.mp4" type="video/mp4"> -->
```

```
<source src="video/video.webm" type="video/webm">
```

```
<p> Seu navegador não suporta o elemento vídeo da HTML5
```

```
<br/>
```

```
  Faça <a href="video/video.webm"> download do  
  video.webm</a>
```

```
</p>
```

```
</video>
```

FORMULÁRIOS

Um formulário é um modelo para a entrada de um conjunto de dados.

Sintaxe - Formulário

Exemplo

```
<form method="método" action="URL_de_script">
```

.....

```
</form>
```

Atributo Action:

action: Especifica o URL do *script* ao qual serão enviados os dados do formulário.

Atributo Method

- Seleciona um método para acessar o URL de ação. Os métodos usados atualmente são **GET** e **POST**. Ambos os métodos transferem dados do browser para o servidor, com a seguinte diferença básica:

post → - os dados entrados fazem parte do corpo da mensagem enviada para o servidor;
- transfere grande quantidade de dados;

get → os dados entrados fazem parte do URL (endereço) associado à consulta enviada para o servidor; - suporta até 128 caracteres.

Exemplo Formulário

estilo_formulario.css

```
input, textarea, select {  
    display: block;  
    margin-bottom: 10px;  
}  
input.dis_inline {  
    display: inline-block;  
}
```

exemplo_formulario.html

```
<form method="post" action="#">  
    <fieldset>  
        <legend>Dados pessoais</legend>  
    </fieldset>  
</form>
```

Exemplo Formulário

Fim... exemplo_formulario.html

```
<input class="dis_inline" type="reset" value="Limpar"/>
```

```
<input class="dis_inline" type="submit" value="Enviar"/>
```

```
</fieldset>
```

```
</form>
```

Visualização



A visual representation of the HTML form. It consists of a rectangular box with a thin gray border. Inside the box, there are two buttons side-by-side. The left button is labeled 'Limpar' and the right button is labeled 'Enviar'. Both buttons have a light gray background with a subtle gradient and a thin gray border.

Exemplo Formulário

Continuação... exemplo_formulario.html

```
<label for="nome">Nome</label>
```

```
<input type="text" id="nome" name="nome" maxlength="40" size="42"/>
```

```
<label for="senha">Senha</label>
```

```
<input type="password" id="senha" name="senha" maxlength="8" size="10"/>
```

Visualização

Dados pessoais

Nome

Senha

Exemplo Formulário

Continuação... exemplo_formulario.html

```
<label for="cep">CEP</label>
```

```
<input type="text" required pattern="\d{5}-?\d{3}"  
      id="cep" placeholder="00000-000"/>
```

```
<label for="tel">Telefone</label>
```

```
<input type="tel" id="tel" placeholder="digite aqui seu telefone"/>
```

Visualização

CEP



É preciso que o formato corresponda ao exigido.

Telefone

Exemplo Formulário

Continuação... exemplo_formulario.html

```
<label for="email">E-mail</label>
```

```
<input type="email" id="email" required/>
```

```
<label for="url">URL site</label>
```

```
<input type="url" id="url" autofocus/>
```

Visualização

E-mail



Insira um endereço de e-mail.

URL site

Exemplo Formulário

Continuação... exemplo_formulario.html

Esporte:

```
<label for="esporte1"> Futebol </label>
```

```
<input class="dis_inline" type="checkbox" name="esporte1 "
id="esporte1 " value="futebol" checked="checked"/>
```

```
<label for="esporte2"> Vôlei </label>
```

```
<input class="dis_inline" type="checkbox" id="esporte2"
value="volei"/>
```

```
<br/>
```

Visualização

Esporte: Futebol ☒ Vôlei ☐

Exemplo Formulário

Continuação... exemplo_formulario.html

Sexo:

```
<label for="fem"> Feminino </label>
```

```
<input class="dis_inline" type="radio" name="sexo" id="fem"  
      value="feminino" checked="checked"/>
```

```
<label for="masc"> Masculino </label>
```

```
<input class="dis_inline" type="radio" name="sexo" id="masc"  
      value="masculino"/>
```

```
<br/>
```

Visualização

Sexo: Feminino ☐ Masculino ☒

Exemplo Formulário

Continuação... exemplo_formulario.html

Visualização

`<label for="search">Busca</label>`

`<input type="search" id="search"/>`

`<label for="date">Date</label>`

`<input type="date" id="date"/>`

Busca

Date

« « maio de 2013 » »

dom	seg	ter	qua	qui	sex	sáb
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Hoje

Limpar

Exemplo Formulário

Continuação... exemplo_formulario.html

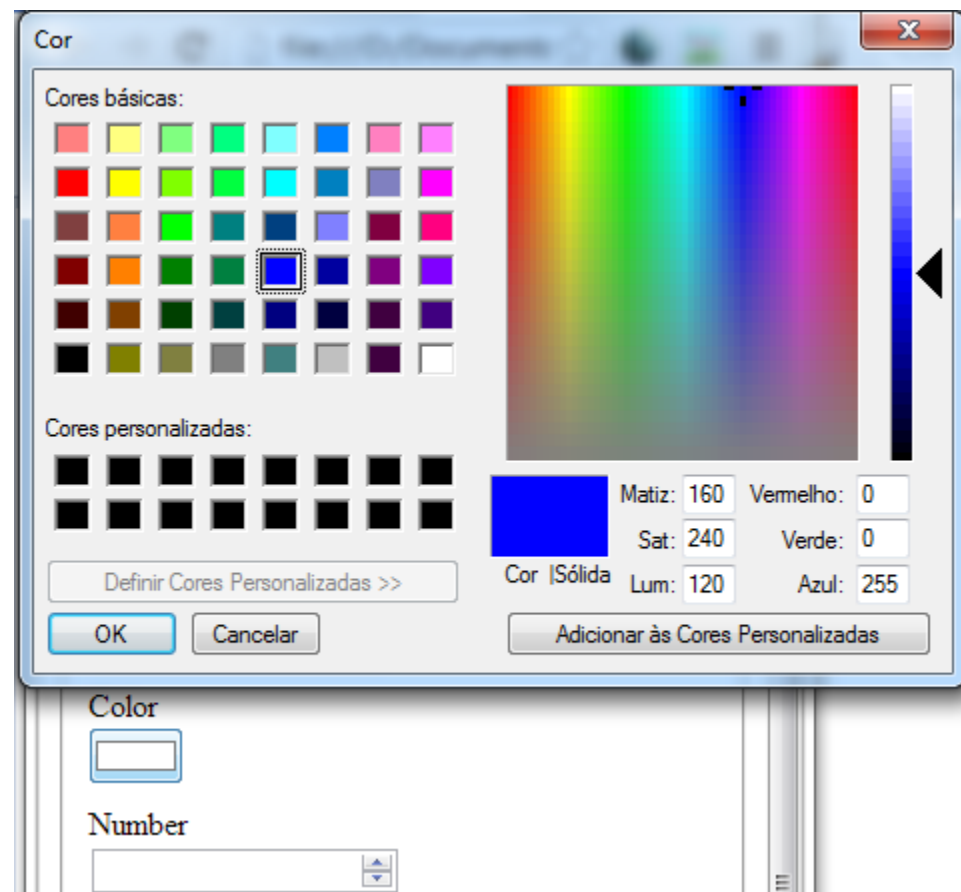
Visualização

```
<label for="color"> Color </label>
```

```
<input type="color" id="color"  
value="#FFFFFF"/>
```

```
<label for="number"> Number  
</label>
```

```
<input type="number" min="0"  
max="100" step="5"  
id="number"/>
```



Exemplo Formulário

Continuação... exemplo_formulario.html

```
<label for="range">Range</label>
```

```
<input type="range" id="range"/>
```

```
<label for="time">Time</label>
```

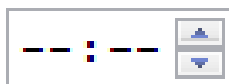
```
<input type="time" id="time" step="300"/>
```

Visualização

Range



Time

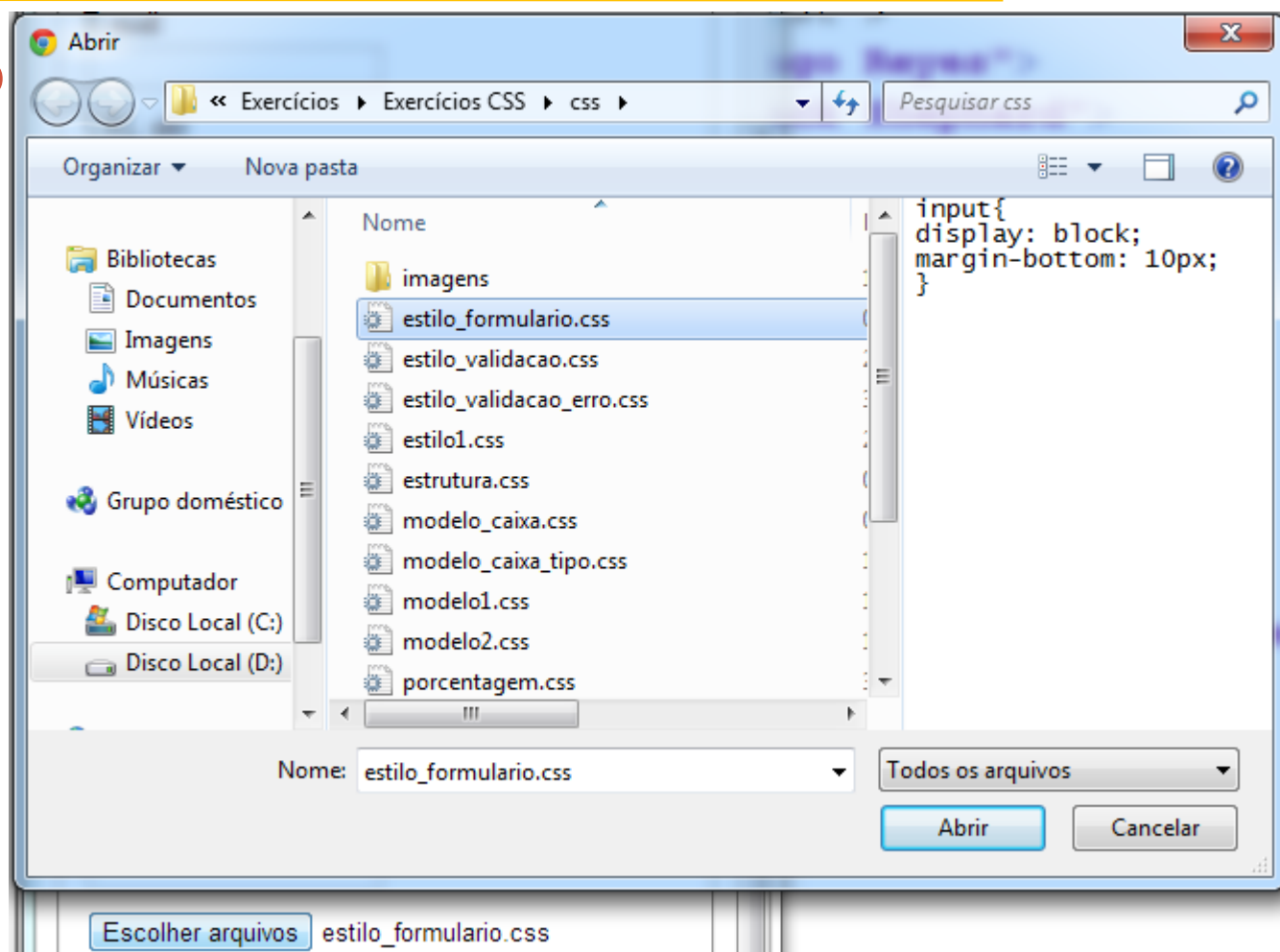


Exemplo Formulário

Continuação... exemplo_formulario.html

`<input type="file" multiple/>`

Visualização



Exemplo Formulário

Continuação... exemplo_formulario.html

```
<label for="cor"> Cores </label>  
<select id="cor" name="cor" >  
  <option value="sem_cor"> </option>  
  <option value="azul">Azul</option>  
  <option value="verde">Verde</option>  
  <option value="amarela">Amarela</option>  
</select>
```

Visualização

Cores

Azul
Verde
Amarela

Exemplo Formulário

Continuação... exemplo_formulario.html

```
<label for="lista">Selecione um nome</label>
```

```
<input type="text" id="lista" name="characters" list="data-list"/>
```

```
<datalist id="data-list">
```

```
<option value="Hugo Reyes"/>
```

```
<option value="Jack Shephard"/>
```

```
<option value="James 'Sawyer' Ford"/>
```

```
<option value="John Locke"/>
```

```
<option value="Sayid Jarrah"/>
```

```
</datalist>
```

Visualização

Selecione um nome



Hugo Reyes
Jack Shephard
James 'Sawyer' Ford
John Locke
Sayid Jarrah

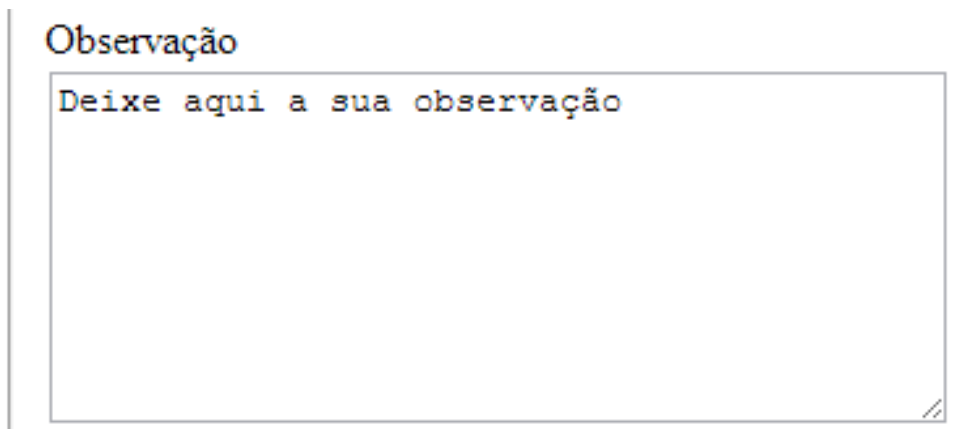
Exemplo Formulário

Continuação... exemplo_formulario.html

```
<label for="observacao">Observação</label>
```

```
<textarea cols="40" rows="8" id="observacao" size="100px"  
placeholder="Deixe aqui a sua observação "></textarea>
```

Visualização



Observação

Deixe aqui a sua observação

CSS: Comentários

- **display:** Define como um elemento é exibido.
 - Ex: `display: none;`
- **float:** Faz o elemento flutuar à esquerda ou à direita do restante do conteúdo.
 - Ex: `float: left;`

CSS: Comentários

- `margin-top`, `margin-right`, `margin-bottom` e `margin-left`: Define a margem superior, direita, inferior e esquerda de um elemento.
 - Ex: `margin-top: 10px;`
- `padding-top`, `padding-right`, `padding-bottom` e `padding-left`: Define a área superior, direita, inferior e esquerda de espaçamento de um elemento.
 - Ex: `padding-right: 5px;`

CSS: Comentários

- **font-family:** Relação de nomes específicos de famílias de fontes ou de seus nomes genéricos.
 - Ex: font-family: Arial, Helvetica, sans-serif;
- **font-size:** Define o tamanho de uma fonte.
 - Ex: font-size: 11px;
- **font-weight:** Define o peso de uma fonte.
 - Ex: font-weight: bold;

CSS: Comentários

- **color:** Define a cor do texto.
 - Ex: `color: red;`
- **text-align:** Alinha o texto em um elemento.
 - Ex: `text-align: center;`
- **text-decoration:** Acrescenta decoração de texto.
 - Ex: `text-decoration: underline;`

CSS: Comentários

- `background-color`: Define a cor de fundo de um elemento.
 - Ex: `background-color: blue;`
- `background-image`: Define uma imagem como plano de fundo.
 - Ex: `background-image: url(images/bg.jpg);`
- `background-repeat`: Define como uma imagem de fundo será repetida.
 - Ex: `background-repeat: no-repeat;`

CSS: Comentários

- **border-width:** Define a largura da borda de um elemento.
 - Ex: `border-width: 3px;`
- **border-style:** Define o estilo da borda.
 - Ex: `border-style: dotted;`
- **border-color:** Define a cor da borda.
 - Ex: `border-color: #000000;`

O Poder do CSS

- O CSS Zen Garden tem como alvo entusiasmar, inspirar e encorajar a separação do Conteúdo (HTML) da Apresentação (CSS)
- Mostrar as enormes possibilidades de se obter belos Layouts através do CSS
- Como exemplo, podemos Acessar os links e visualizar diversos Layouts diferentes. O código HTML permanece o mesmo, a única coisa que muda é o arquivo CSS externo.

Propriedade	Descrição	Valores	NN	IE	W3C
background	Uma propriedade estenográfica para configurar todas as propriedades do plano de fundo em uma declaração	<i>background-color</i> <i>background-image</i> <i>background-repeat</i> <i>background-attachment</i> <i>background-position</i>	6.0	4.0	CSS 1
background-attachment	Especifica se uma imagem de plano de fundo é fixa ou rola com o resto da página	scroll fixed	6.0	4.0	CSS 1
background-color	Configura a cor do plano de fundo de um elemento	<i>color-rgb</i> <i>color-hex</i> <i>color-name</i> transparent	4.0	4.0	CSS 1
background-image	Configura uma imagem como plano de fundo	<i>url</i> none	4.0	4.0	CSS 1
background-position	Configura a posição inicial de uma imagem de plano de fundo	top left top center top right center left center center center right bottom left bottom center bottom right <i>x-% y-%</i> <i>x-pos y-pos</i>	6.0	4.0	CSS 1
background-repeat	Estabelece se/como uma imagem de plano de fundo será repetida	repeat repeat-x repeat-y no-repeat	4.0	4.0	CSS 1

Propriedade	Descrição	Valores
Background	Uma propriedade estenográfica para configurar todas as propriedades do plano de fundo em uma declaração	<i>background-color</i> <i>background-image</i> <i>background-repeat</i> <i>background-attachment</i> <i>background-position</i>
background-attachment	Especifica se uma imagem de plano de fundo é fixa ou rola com o resto da página	scroll fixed

Propriedade	Descrição	Valores
background-color	Configura a cor do plano de fundo de um elemento	<i>color-rgb</i> <i>color-hex</i> <i>color-name</i> transparent
background-image	Configura uma imagem como plano de fundo	<i>url</i> none

Propriedade	Descrição	Valores
background-position	Configura a posição inicial de uma imagem de plano de fundo	top left top center top right center left center center center right bottom left bottom center bottom right <i>x-% y-%</i> <i>x-pos y-pos</i>
background-repeat	Estabelece se/como uma imagem de plano de fundo será repetida	repeat repeat-x repeat-y no-repeat

Referências

- Silva, Maurício Samy – Construindo sites com CSS e (X)HTML. - São Paulo: Novatec, 2008.
- www.w3.org
- www.w3schools.com
- www.maujor.com
- www.tableless.com.br
- www.cssnolanche.com.br

Bibliografia

- [1] w3c.org.
- [2] SILVA, Maurício Samy. HTML 5. A Linguagem de Marcação que revolucionou a WEB.
- [3] SILVA, Maurício Samy. **Construindo sites com CSS e (X)HTML: sites controlados por folhas de estilo em cascata**, São Paulo: Novatec, 2008.
- [4] apostila W3C.

DÚVIDAS???

E-mail:

alexandre.bernardes2@fatec.sp.gov.br