

**UNIP**  
UNIVERSIDADE PAULISTA

## Aplicação Linguagem de Programação Orientada a Objeto

### ALPOO – 06.1 - Arquitetura web - HTML, CSS e Javascript

Prof. Msc Wanderley Gonçalves Freitas

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 1 de 15

**UNIP**  
UNIVERSIDADE PAULISTA

## Arquitetura

CLIENTE CLIENTE CLIENTE

SERVIDOR

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 2 de 15

**UNIP**  
UNIVERSIDADE PAULISTA

## Execução no Cliente (Browser)

- HTML
- CSS
- JavaScript
- XML
- XHTML
- HTML 5

CLIENTE CLIENTE CLIENTE

SERVIDOR

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 3 de 15

**UNIP**  
UNIVERSIDADE PAULISTA

## Execução no Cliente (Browser)

**Importante (1):**  
O **JavaScript (Comportamentos)** é uma linguagem de programação interpretada, onde sua finalidade é para que scripts possam ser executados do lado do cliente e interaja com o usuário sem a necessidade deste script passar pelo servidor.

- **CSS** (Cascading Style Sheets, ou Folhas de Estilo em Cascata) (Apresentação; Formatação, layout, cores, fontes, posicionamento) é uma linguagem mais voltada para a estética, utilizada para formatar os conteúdos já estruturados. O CSS é responsável pela aparência da página, como o posicionamento dos elementos, cores e tudo que torne o site mais visualmente agradável para a navegação.

**HTML** (HyperText Markup Language, ou Linguagem de Marcação de HiperTexto) (Conteúdo; Dados e estrutura;) é uma linguagem de publicação, utilizada para estruturar conteúdos, sejam textos, imagens, vídeos ou áudios

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 4 de 15

**UNIP**  
UNIVERSIDADE PAULISTA

## Execução no Cliente (Browser)

**XHTML** é a sigla inglesa para **EXtensible HyperText Markup Language** que em Português significa Linguagem Extensível para Marcação de Hipertexto. É uma linguagem de construção de páginas na internet criada a partir da linguagem HTML (versão anterior) juntamente com a linguagem XML, transformando-se em uma linguagem padronizada para web.

- **XML** (*eXtensible Markup Language*) é caracterizada como uma linguagem flexível e dinâmica que pode ser escrita em diversos formatos e interpretada com facilidade.
- A linguagem **HTML5** é a quinta versão do HTML e será o novo padrão para produção de páginas web englobando diferentes linguagens.

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 5 de 15

**UNIP**  
UNIVERSIDADE PAULISTA

## HTML

```
<html>
<head>
  <title>Título da Página</title>
</head>
<body>
  Esta é minha primeira página.
</body>
</html>
```

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 6 de 15

## Formulários HTML

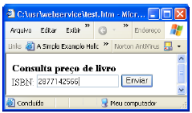
- Todo formulário em HTML é construído usando elementos dentro de um bloco `<FORM>`
- O bloco `<FORM>` define a URL que receberá o formulário e pode definir também o método usado

```
<FORM ACTION="URL para onde serão enviados os dados"
METHOD="método HTTP (pode ser GET ou POST)"
ENCTYPE="formato de codificação"
TARGET="nome da janela que mostrará a resposta" >
... corpo do formulário
(permite qualquer coisa permitida em <BODY>)
</FORM>
```

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 7 de 15

## Envio de dados com Formulários

- Vários elementos HTML servem para entrada de dados e são usados dentro de formulários. Todos os elementos de entrada de dados têm um *nome* e enviam um *valor*
- Exemplo de formulário para entrada de dados



```
<FORM ACTION="/cgi-bin/catalogo.pl"
METHOD="POST">
<H3>Consulta preço de livro</H3>
<P>ISBN: <INPUT TYPE="text" NAME="isbn">
<INPUT TYPE="Submit" VALUE="Enviar">
</FORM>
```

```
POST /cgi-bin/catalogo.pl HTTP/1.0
Content-type: text/x-www-form-urlencoded
Content-length: 15
isbn=2877142566
```

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 5 de 15

## Elementos para disparo de eventos

- Os elementos `<INPUT>` com atributo `TYPE` *Submit*, *Reset* e *Button* servem para disparar eventos
  - Envio do formulário (Submit)
  - Reinicialização do formulário (Reset)
  - Evento programado por JavaScript (Button)
- O *value* do botão define o texto que mostrará
- Apenas se o botão *contiver um atributo name*, o conteúdo de *value* será enviado ao servidor

Button	
<input type="button"/>	<code>&lt;input type="button"&gt;</code>
Apertar	<code>&lt;input type="button" value="Apertar"&gt;</code>
Submit	
Submit Query	<code>&lt;input type="submit"&gt;</code>
Enviar	<code>&lt;input type="submit" value="Enviar"&gt;</code>
Reset	
Reset	<code>&lt;input type="reset"&gt;</code>
Limpar	<code>&lt;input type="reset" value="Limpar"&gt;</code>

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 9 de 15

## Entrada de texto

- Elementos `<INPUT>` com `TYPE="text"` podem ser usados para entrada de texto
  - `<input type="text">`
  - `<input type="text" size=10>`
  - `<input type="text" value="texto inicial">`
- Com `TYPE="password"` o texto digitado é ocultado na tela do browser
  - `<input type="password" maxlength=8>`
  - `<input type="password" size=10>`

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 10 de 15

## Campos ocultos

- Campos ocultos consistem de um par nome/valor embutido no código HTML
- São úteis para que o autor da página possa enviar informações ao servidor
  - Informações sobre configuração da aplicação
  - Comandos, para selecionar comportamentos diferentes da aplicação
  - Parâmetros especiais para controle da aplicação, sessão ou dados que pertencem ao contexto da aplicação
- Sintaxe
  - `<INPUT TYPE="hidden" NAME="nome" VALUE="valor">`

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 11 de 15

## Chaves booleanas

- Há dois tipos: checkboxes e radio buttons
- Checkboxes permitem mais de uma seleção
  - Café `<input type="checkbox" name="refeicoes" value="Café" checked>` Café
  - Almoço `<input type="checkbox" name="refeicoes" value="Almoço">` Almoço
  - Jantar `<input type="checkbox" name="refeicoes" value="Jantar">` Jantar
- O código acima enviará nomes repetidos contendo valores diferentes na requisição
- Radio Buttons, se tiverem o mesmo nome, formam um grupo. No grupo, apenas uma seleção é aceita
  - GRUPO I
    - pela manhã `<input type="radio" name="turno" value="Manhã">` pela manhã
    - à tarde `<input type="radio" name="turno" value="Tarde">` à tarde
    - à noite `<input type="radio" name="turno" value="Noite" checked>` à noite
  - GRUPO II
    - Masculino `<input type="radio" name="sexo" value="M">` Masculino
    - Feminino `<input type="radio" name="sexo" value="F">` Feminino

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 12 de 15

## Área para entrada de texto

- Possibilitam a entrada de texto de múltiplas linhas
- Elemento: **<TEXTAREA>**

ssa a página HTML.  
 digitado pelo usuáric  
 ssa a página HTML.

<textarea rows=3 cols=20></textarea>

Texto inicial

<textarea rows=3 cols=20>Texto inicial</textarea>

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 13 de 15

## Menus de seleção

- Geram requisições similares a checkboxes e radio buttons
- Consistem de um par de elementos
  - **<SELECT>** define o nome da coleção
  - **<OPTION>** define o valor que será enviado

Domingo  
 Domingo  
 Segunda  
 Terça  
 Quarta  
 Quinta  
 Sexta  
 Sábado

<select name="umdia">  
 <option value="D">Domingo</option>  
 <option value="S">Segunda</option>  
 <option value="T">Terça</option>  
 <option value="Q">Quarta</option>  
 <option value="I">Quinta</option>  
 <option value="X">Sexta</option>  
 <option value="B">Sábado</option>  
 </select>  
  
 <select name="variosdias" size=4 multiple>  
 <option value="D">Domingo</option>  
 <option value="S">Segunda</option>  
 <option value="T">Terça</option>  
 <option value="Q">Quarta</option>  
 <option value="I">Quinta</option>  
 <option value="X">Sexta</option>  
 <option value="B">Sábado</option>  
 </select>

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 14 de 15

## HTML Tabelas

```

<table border="1">
  <tr>
    <th>Coluna 1</th>
    <th>Coluna 2</th>
  </tr>
  <tr>
    <td>linha 1, valor 1</td>
    <td>linha 1, valor 2</td>
  </tr>
  <tr>
    <td>linha 2, valor 1</td>
    <td>linha 2, valor 2</td>
  </tr>
</table>
  
```

Coluna 1	Coluna 2
linha 1, valor 1	linha 1, valor 2
linha 2, valor 1	linha 2, valor 2

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 15 de 15

## CSS Cascading Style Sheets

- Estilos definem como elementos html devem ser apresentados
- Permite a separação entre definição de conteúdo e formatação em HTML
- Style sheets externos são definidos através de arquivos CSS

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 16 de 15

## CSS – Maneiras de utilizar CSS

Existem três maneiras de se utilizar CSS

**CSS interno** – Definida no mesmo arquivo que o documento html. O CSS interno pode ser declarado:

- Dentro da tag **<body>**, definido como atributos das tags html normais. Escopo: estilo vale somente para um elemento da pagina
- Dentro da tag **<head>** através de uma tag **<style>**. Escopo: Estilo vale para todas as tags de um dado tipo, dentro de um único documento html

**CSS externo** – Definido em um arquivo .css separado do documento html. Escopo: Pode ser compartilhada por múltiplos documentos html.

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 17 de 15

## CSS - Arquivo(CSS externo)

- Existem 3 maneiras de definir um CSS para um documento HTML
  - Criando um link externo

```

<html>
<head>
  <link rel="stylesheet" type="text/css" href="estilo.css" />
</head>
<body>
  Esta é minha primeira página.
</body>
</html>
  
```

```

/* Conteúdo do arquivo estilo.css */
hr {color: sienna}
p {margin-left: 20px}
body {background-image:
  url("images/back40.gif")}
  
```

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 18 de 15

## CSS - Dentro da tag <head>

– Internamente (dentro da própria página HTML a ser exibida)

```
<html>
<head>
  <style type="text/css">
    hr {color: sienna}
    p {margin-left: 20px}
    body {background-image:url("images/back40.gif")}
  </style>
</head>
<body>
  Esta é minha primeira página.
</body>
</html>
```

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 19 de 15

## CSS - Dentro da tag <body>

– Definindo de forma

```
<html>
<head/>
<body style="color: sienna; text-align: center">
  Esta é minha primeira página.
  <P style="color: green; font-weight: bold; text-align: center"> O
  sapo nao lava o pe </P>

  <A style="font-size: 18px; color:red" HREF="http://www.sapo2.pt/">
  Veja por que</A>

</body>
</html>
```

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 20 de 15

## CSS - Estrutura de uma regra CSS

Formato geral:

```
seletor { propriedade: valor }
```

Uma definição de estilo em CSS será composta por uma sequência de definições como esta acima

- Exemplos:
  - body { color: black }
  - p { font-family: "Verdana"; text-align: center; color: red }
  - h1,h2,h3,h4,h5,h6 { color: green }
  - p { margin-left: 20px }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 21 de 15

## CSS - Estrutura de uma regra CSS

- Classes em CSS permitem que um mesmo elemento seja exibido de diferentes formas
- Formato usando classes:
 

```
seletor.classe { propriedade: valor }
```
- Exemplos:
  - p.direita { text-align: right } // Alinha à direita
  - p.centro { text-align: center } // Centraliza
  - .esquerda { text-align: left } // Aplicado a qualquer elemento html que contenha classe *esquerda*

```
<p class="direita">Este parágrafo será alinhado à direita</p>
<p class="centro">Este parágrafo será centralizado</p>
<p class="esquerda">Este será à esquerda</p>
```

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 22 de 15

## CSS - Estrutura de uma regra CSS

### CSS Externa - teste.css

```
@charset "utf-8";

body {
font-family: "Arial", sans-serif;
margin-left: 15%;
margin-right: 15%; }

.titulo1 {
color: red;
font-family: "Arial";
font-size: 25px; }

.texto{
font-family: "Times";
font-size: 16px; }

.codigo {
font-family: monospace;
font-size: 12px; }
```

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 23 de 15

## CSS - Estrutura de uma regra CSS

### CSS – Externa Exemplo

```
<html>
<head>
<link type="text/css" rel="stylesheet"href="teste.css">
</head>
<body class="body">
<P class="titulo1">Trabalho de Java</P>
<P class="texto">Desde seu lançamento, em maio de 1995, a plataforma Java foi adotada mais rapidamente do que qualquer outra linguagem de programação na historia da computacao. Em 2003 Java atingiu a marca de 4 milhoes de desenvolvedores em todo mundo. </P>
<P class="codigo">
public class OlaMundo { <BR>
  public static void main (String args[]); <BR>
  System.out.println("Ola Mundo"); }<BR>
}; <BR>
</P>
</body>
</html>
```

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 24 de 15

## CSS - Aplicações

Seletores mais simples

- Seletor **elemento**:
  - Aplicam a regra em **todos os elementos** da página;
- Seletor **id (#)** :
  - Aplicam a regra em um **elemento com id específico**;
  - **Não pode repetir** na página!
- Seletor **classe (.)**:
  - Aplicam a regra nos **elementos pertencentes a uma classe**;
  - **Atributo class** do HTML.

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 25 de 15

## CSS - Aplicações

- Comentários no CSS
  - H1 {color:red} /\* cor vermelha para a tag H1 \*/

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 26 de 15

## CSS - Aplicações

Elementos do CSS

- Uma regra CSS é formada pelo par seletor – declaração
- A declaração é composta pelo par propriedade – valor

**seletor** {prop1: valor; propr2: valor .. }

Ex:

```
P {
  Color: black;
  Font-Family: Verdana }
```

Obs: o seletor é a ligação entre o documento HTML e o CSS, e todos elementos HTML são possíveis seletores ( o inverso não é verdade ).

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 27 de 15

## CSS - Aplicações

Elementos do CSS

- Agrupamento de seletores  
H1, H2, H3 { font-family: helvetica }
- Agrupamento de declarações  
H1 { font-weight: bold;  
font-size: 12pt;  
font-family: helvetica; }
- Agrupamento de propriedades  
H1 { font: bold 12pt/14pt helvetica }

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 28 de 15

## CSS - Aplicações

Elementos do CSS

- Seletores de classes  
TD.class1 { color: purple }  
ou  
.class1 { color: purple }

no HTML:  
<TD class=class1> CSS é uma boa idéia </TD>

- Seletores ID  
#id1 { font-weight: bold; }

no HTML:  
<P ID=id1> ID deve ser único no documento </P>

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 29 de 15

## CSS - Aplicações

Elementos do CSS

- Seletores contextuais  
H1 B { color: #ff55ff }
- Seletores de pseudo-classes  
a:link {color: #FFF text-decoration: none;}  
a:visited {color: #FFF text-decoration: none;}  
a:active {color: #F00 text-decoration: none;}  
a:hover {color: #080 text-decoration:underline;}

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 30 de 15

**CSS - Aplicações**

- Com CSS podemos formatar:
  - Background
  - Textos
  - Fontes
  - Margens
  - Bordas
  - Listas
  - Tabelas

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 31 de 15

**CSS - HTML**

- **Plano de fundo**
  - **background-color** – especifica a cor de fundo do elemento
  - Valores possíveis - <cor> | transparent
  - Ex.: table { background-color: #cccc99 }
  - **background-image** – define a imagem de fundo
  - Valores possíveis – <url> | none
  - Ex.: body { background-image: url(foto.jpg) }
  - **background-attachment** – define se a imagem de fundo rola com o conteúdo ou não
  - Valores possíveis – scroll | fixed
  - Ex.: body { background-image: url(foto.jpg); background-attachment: fixed }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 32 de 15

**CSS - HTML**

**Plano de fundo**

- **background-repeat** – especifica se a imagem de fundo será repetida e como será a repetição
- Valores possíveis – repeat-x | repeat-y | repeat | no-repeat
- Ex.: body { background-image: url(foto.jpg); background-repeat: no-repeat }
- **background-position** – define a posição da imagem de fundo
- Valores possíveis – distância para o topo (percentual ou pixels) e/ou distância para a margem esquerda (percentual ou pixels) | [top|center|bottom] e/ou [left|center|right]
- Ex.: body { background-image: url(foto.jpg) background-position: center center }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 33 de 15

**CSS - HTML**

- **Plano de fundo**
  - **background** – especifica vários atributos de uma só vez.
  - Valores possíveis – os mesmos para os atributos background-color, background-image, background-repeat, background-attachment, background-position
  - Ex.: body { background: fixed url(foto.jpg) no-repeat center center }
- **Cores**
  - **color** – especifica a cor do texto de um elemento
  - Valores possíveis - <cor>
  - Ex.: p { color: #ffaabb }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 34 de 15

**CSS - HTML**

- **Propriedades de fontes**
  - **font-family** – especifica lista de fontes
  - Valores – <nomedafamilia>|<familiagenérica>
  - Ex.: a { font-family: "Times Roman", Arial, serif }

Obs.: as famílias genéricas podem ser: **serif, sans-serif, cursive, fantasy, monospace**

  - **font-size** – especifica o tamanho da fonte
  - Valores – <tam-abs>|<tam-relat>|<tamanho>|<percent>
    - tam-abs: xx-small, x-small, small, medium, large, x-large, xx-large
    - tam-relat: larger, smaller
    - tamanho: valor em pontos
    - percent: valor percentual em relação ao padrão
  - Ex.: body { font-size: 12pt }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 35 de 15

**CSS - HTML**

- **Propriedades de fontes**
  - **font-stretch** – especifica largura (expansão ou condensação) da fonte
  - Valores – normal | wider | narrower | ultracondensed | extra-condensed | condensed | semi-condensed | semi-expanded | expanded | extra-expanded | ultra-expanded
  - Ex.: a { font-stretch: condensed }
  - **font-style** – especifica o estilo da fonte
  - Valores – normal | italic | oblique
  - Ex.: p { font-style: italic }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 36 de 15

**CSS - HTML**

**Propriedades de fontes**

- **font-variant** – especifica se a fonte será exibida em caixa alta ou normal
- Valores – normal | small-caps
- Ex.: h1 { font-variant: small-caps }

- **font-weight** – especifica a espessura da fonte
- Valores – normal | bold | bolder | lighter | 100-900
- Ex.: p { font-weight: bold }
- span { font-weight: 800 }

- **font** – especifica vários atributos de uma só vez
- Ex.: td { font: italic bold 12pt Arial, Courier }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 37 de 15

**CSS - HTML**

- **Propriedades de textos**
  - **letter-spacing** – define a distância entre as letras
  - Valores – normal | <valor em px ou pt>
  - Ex.: th { letter-spacing: 12pt }
- **line-height** – espaçamento entre linhas de texto
- Valores – normal|<valor em px ou pt>|<percentual>
- span { line-height: 14pt }
- **text-align** – define o alinhamento do texto
- Valores – left | right | center | justify
- Ex.: td { text-align: justify }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 38 de 15

**CSS - HTML**

**Propriedades de textos**

- **text-decoration** – define efeitos adicionais ao texto
- Valores – none|underline|overline|line-through|blink
- Ex.: h3 { text-decoration: underline }

- **text-indent** – recuo da primeira linha de texto em um bloco
- Valores – <valor> | <percentual>
- span { text-indent: 30px }

- **text-transform** – controla a capitalização do texto
- Valores – none | capitalize | uppercase | lowercase
- Ex.: td { text-transform: uppercase }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 39 de 15

**CSS - HTML**

- **Propriedades de textos**
  - **vertical-align** – alinhamento em uma célula
  - Valores – top | middle | bottom
  - Ex.: th { vertical-align: middle }
- **white-space** – manipulação dos espaços em branco
- Valores – normal | pre | nowrap
  - pre – considera espaços adicionais (problema no IE)
  - nowrap – não quebra a linha nos espaços em branco
- p { white-space: nowrap }
- **word-spacing** – espaçamento entre palavras
- Valores – normal | <valor em px ou pt>
- Ex.: h1 { word-spacing: 12pt }

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 40 de 15

**CSS - HTML**

- **Caixas**
  - **border** – define largura, estilo e cor para as quatro bordas
  - Valores – <valor em px|thin|medium|thick> | <estilo> | <cor>
  - Ex.: table { border: 1px solid red }
- Opções para estilo:
  - none – sem borda
  - hidden – oculta
  - dotted – pontilhada
  - dashed – tracejada
  - solid – linha sólida
  - double – linha dupla
  - groove – entalhada

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 41 de 15

**CSS - HTML**

**Caixas**

- **border-bottom, border-left, border-right, border-top** – define largura, estilo e cor para as bordas individualmente
- mesmas opções de border.
- **border-color** – define as cores das quatro bordas.
- **border-bottom-color, border-left-color, border-right-color, border-top-color** – define a cor das bordas individualmente
- **border-style** – define o estilo para as quatro bordas
- **border-bottom-style, border-left-style, border-right-style, border-top-style** – define o estilo das bordas individualmente

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 42 de 15

**CSS - HTML**

**Caixas**

- `border-width` – define as larguras das quatro bordas.
- `border-bottom-width, border-left-width, border-right-width, border-top-width` – define as larguras das quatro individualmente
- `padding` – define as margens ao redor (externo) do elemento
- padding: acima direita abaixo esquerda
- Ex.: `img { padding: 10px 20px 15px 12px }`
- `padding-bottom, padding-left, padding-right, padding-top` – define as margens individualmente

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 43 de 15

**CSS - HTML**

**Caixas**

- `padding` – define as margens internas do elemento
- padding: acima direita abaixo esquerda
- Ex.: `th { padding: 10px 20px 15px 12px }`
- `padding-bottom, padding-left, padding-right, padding-top` – define as margens internas individualmente

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 44 de 15

**CSS - HTML**

**margin-top** (espaço entre borda superior e outros objetos)

**padding-top** - espaço entre a parte superior do objeto e a borda

**padding-bottom** - espaço entre a parte inferior do objeto e a borda

**padding-left**

**padding-right**

**border-top**

**border-bottom**

**border-left**

**border-right**

**margin-bottom**

**margin-left**

**margin-right**

**height (altura)**

**Width (largura)**

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 45 de 15

**CSS - HTML**

**Dimensões**

- `height` – define a altura do elemento
- height: auto|<valor em px>|<valor percentual>
- Ex.: `table { height: 100px }`
- `width` – define a largura do elemento
- width: auto|<valor em px>|<valor percentual>
- Ex.: `img { width: 100px }`
- Obs.: A imagem no IE não distorce ao se definir somente uma das dimensões.

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 46 de 15

**CSS - HTML**

**Posicionamento**

- `position` – define o modo de posicionamento do elemento
- position: absolute|relative|static

Ex.: `table { position: relative }`

- `left` – define a posição do elemento a partir da margem esquerda
- left: auto|<valor em px>|<valor percentual>

Ex.: `div { position: absolute; left: 10px }`

- `right` – define a posição do elemento a partir da margem direita
- right: auto|<valor em px>|<valor percentual>

Ex.: `h1 { position: absolute; right: 10px }`

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 47 de 15

**CSS - HTML**

- **Posicionamento**
  - `top` – define a posição do elemento a partir do topo
  - top: auto|<valor em px>|<valor percentual>
  - Ex.: `h2 { position: absolute; top: 20px }`
  - `bottom` – define a posição do elemento a partir da base
  - bottom: auto|<valor em px>|<valor percentual>

Ex.: `p { position: absolute; bottom: 5px }`

- `visibility` – define se o elemento está visível ou não
- visibility: inherit | visible | hidden

Ex.: `div { visibility: hidden }`

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 48 de 15



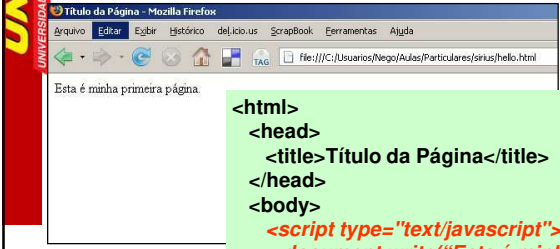
## JavaScript

- É a linguagem de script utilizada por milhões de páginas web
- Foi projetada para aumentar interatividade das páginas web:
  - Validação de formulários, interação com o usuário (p.ex., tratamento de cliques de botões), detecção de navegadores, etc
- É reconhecida pela maioria dos navegadores
- Seu processamento é feito na máquina cliente (browser)
- Não há relação com Java

UNIP UNIVERSIDADE PAULISTA

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 49 de 15

## JavaScript - Exemplo



```

<html>
<head>
<title>Título da Página</title>
</head>
<body>
<script type="text/javascript">
document.write("Esta é minha
primeira página.");
</script>
</body>
</html>

```

UNIP UNIVERSIDADE PAULISTA

Prof. Msc Wanderley Gonçalves Freitas 5

## JavaScript – Onde ocorrem

- Uma tag <script/> pode ser definida numa seção **head**, numa seção **body** e também pode ser definida **externamente**:
  - Na seção **head**, os scripts são executados quando são chamados ou quando algum evento ocorre;
  - Na seção **body**, os scripts são executados na carga da página web
  - Para definição externa, um arquivo **“.js”** precisa ser fornecido com as funções necessárias

UNIP UNIVERSIDADE PAULISTA

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 51 de 15

## JavaScript – Exemplo

```

<html>
<head>
<title>Título da Página</title>
<script src="hello.js"></script>
</head>
<body onload="message()">
<script type="text/javascript">
document.write("<h1>Esta é minha primeira
página.</h1>");
</script>
</body>
</html>

```

UNIP UNIVERSIDADE PAULISTA

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 52 de 15

## JavaScript - Sintaxe

Possui construções existentes na maioria das linguagens de programação (sintaxe similar a C):

- Declaração de variáveis (*var x;*)
- Comandos condicionais, repetições, definição de funções de usuário
- Operadores de atribuição, comparação, ...

UNIP UNIVERSIDADE PAULISTA

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 53 de 15

## JavaScript – Eventos

Eventos são ações que podem ser detectadas por um script

- Exemplos de eventos:
  - Clique do mouse, abertura de uma página web ou imagem, envio de um formulário html, uma tecla pressionada, etc

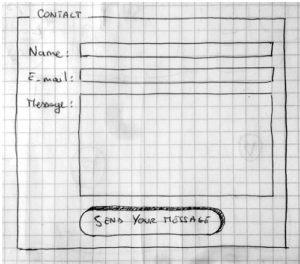
O tratamento destes eventos pode ser a chamada de funções do script

UNIP UNIVERSIDADE PAULISTA

Prof. Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 54 de 15

## Exercício prático

### Desenhando seu formulário



Agora estamos prontos para ir para o código HTML do nosso formulário. Para construir o nosso formulário de contato, vamos utilizar os seguintes elementos

**<form>**, **<label>**, **<input>**, **<textarea>**, e **<button>**

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 55 de 15

## Exercício prático - Linguagens

- O **JavaScript** é uma linguagem de programação interpretada, onde sua finalidade é para que scripts possam ser executados do lado do cliente e interaja com o usuário sem a necessidade deste script passar pelo servidor.
- CSS** (Cascading Style Sheets, ou Folhas de Estilo em Cascata) é uma linguagem mais voltada para a estética, utilizada para formatar os conteúdos já estruturados. O CSS é responsável pela aparência da página, como o posicionamento dos elementos, cores e tudo que torne o site mais visualmente agradável para a navegação.
- HTML** (HyperText Markup Language, ou Linguagem de Marcação de HiperTexto) é uma linguagem de publicação, utilizada para estruturar conteúdos, sejam textos, imagens, vídeos ou áudios.

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 56 de 15

## Exercício prático

### O Elemento **<form>**

Todos formulários HTML começam com um elemento **<form>** como este:

```

1 <form action="/pagina-processa-dados-do-form" method="post">
2
3 </form>
    
```

Este elemento define um formulário. Todos os seus atributos são opcionais, mas é considerada a melhor prática sempre definir pelo menos o atributo **action** e o atributo **method**.

- O atributo **action** define o local (uma URL) em que os dados recolhidos do formulário devem ser enviados.
- O atributo **method** define qual o método HTTP para enviar os dados (ele pode ser "GET" ou "POST" (veja as diferenças aqui)).

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 57 de 15

## Exercício prático

### Adicionar campos com os elementos **<label>**, **<input>**, e **<textarea>**

```

1 <form action="/pagina-processa-dados-do-form" method="post">
2   <div>
3     <label for="nome">Nome:</label>
4     <input type="text" id="nome" />
5   </div>
6   <div>
7     <label for="email">E-mail:</label>
8     <input type="email" id="email" />
9   </div>
10  <div>
11    <label for="msg">Mensagem:</label>
12    <textarea id="msg"></textarea>
13  </div>
14 </form>
    
```

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 58 de 15

## Exercício prático

- Os elementos **<div>** estão lá para estruturar nosso código e deixar a estilização mais fácil
- Observe o uso do atributo **for** em todos os elementos **<label>**; é uma maneira para vincular uma label à um campo do formulário. Este atributo faz referência ao id do campo correspondente.
- No elemento **<input>**, o atributo mais importante é o atributo **type** (**type="text"** ou **type="email"**)

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 59 de 15

## Exercício prático

### E um elemento **<button>**

```

1 <form action="/pagina-processa-dados-do-form" method="post">
2   <div>
3     <label for="nome">Nome:</label>
4     <input type="text" id="nome" />
5   </div>
6   <div>
7     <label for="mail">E-mail:</label>
8     <input type="email" id="mail" />
9   </div>
10  <div>
11    <label for="msg">Mensagem:</label>
12    <textarea id="msg"></textarea>
13  </div>
14  <div class="button">
15    <button type="submit">Enviar sua mensagem</button>
16  </div>
17 </form>
    
```

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 60 de 15

## Exercício prático

Um botão pode ser de três tipos: **submit**, **reset**, ou **button**.

- botão de **submit** envia os dados do formulário para a página de web definida pelo atributo action do elemento <form> .
- botão de **reset** redefine imediatamente todos os campos do formulário para o seu valor padrão.
- botão do tipo **button** útil para construir botões personalizados com JavaScript

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 61 de 15

## Exercício prático - HTML

Prof: Msc Wanderley Gonçalves Freitas

## Exercício prático – HTML+CSS

```

42 .textoarea {
43   /* Para alinhar corretamente os campos de
44   vertical-align: top;
45   /* Para dar espaço suficiente para digitar
46   height: 60px;
47   /* Para permitir aos usuários redimensionar
48   resize: vertical;
49   }
50 }
51
52 #botao {
53   /* Para posicionar os botões para a mesma
54   padding-left: 90px; /* mesmo tamanho que o
55   }
56
57 #superior {
58   margin: 0;
59   height: 70px;
60   background: #002972 /* #2494FD*/;
61   background-size: 90px 70px;
62   color: red;
63   background-repeat: no-repeat;
64   background-image: url("../imagem/logo.jpg");
65   font-size: 30px;
66   text-align: center;
67 }
    
```

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 63 de 15

## Exercício prático – HTML+CSS+Javascrpts

```

1 function mensagem() {
2   alert("Blink do botão!");
3 }
4
5 function enviadoDados() {
6
7   if (document.dados.txtNome.value == "")
8     alert("Preencha campo NOME corretamente!");
9   if (document.dados.txtEmail.value.length < 3) {
10    alert("Preencha campo E-MAIL corretamente!");
11    return false;
12   }
13   if (document.dados.txtEmail.value == "")
14     alert("Preencha campo E-MAIL corretamente!");
15   if (document.dados.txtEmail.value.indexOf('@') == -1) {
16    alert("Preencha o campo MENSAGEM!");
17    document.dados.txtEmail.focus();
18    return false;
19   }
20 }
21
22 if (document.dados.txtMensagem.value == "") {
23   alert("Preencha o campo MENSAGEM!");
24   document.dados.txtMensagem.focus();
25   return false;
26 }
    
```

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 64 de 15

## Exercício prático - Resumo

O **JavaScript** é uma linguagem de programação interpretada, onde sua finalidade é para que scripts possam ser executados do lado do cliente e interaja com o usuário sem a necessidade deste script passar pelo servidor.

- CSS** (Cascading Style Sheets, ou Folhas de Estilo em Cascata) é uma linguagem mais voltada para a estética, utilizada para formatar os conteúdos já estruturados. O CSS é responsável pela aparência da página, como o posicionamento dos elementos, cores e tudo que torne o site mais visualmente agradável para a navegação.
- HTML** (HyperText Markup Language, ou Linguagem de Marcação de HiperTexto) é uma linguagem de publicação, utilizada para estruturar conteúdos, sejam textos, imagens, vídeos ou áudios

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 65 de 15

## Referências Bibliográficas

BRUCE, Eckel. **Thinking in Java**. 4 ed. Massachusetts: Editora Prentice hall: 2006.

FOWLER, Martin. **UML essencial : um breve guia para a linguagem** – padrão de modelagem de objetos. Tradução Vera Pizerico e Christian Thomas Prices. 3. ed. Porto Alegre : Bookman, 2005.

FURLAN, Jose Davi **Modelagem de Objetos Através da UML** São Paulo: Editora Makron Books, 2000, ISBN 8534609241,

GRADY, Booch; RUMBAUGH, James; JACOBSON, Ivar. **UML, guia do usuário** . tradução de Fábio Freitas da Silva. Rio de Janeiro: Editora Campus, 2000.

SIERRA, kathy; BATES, Bert. **Certificação Sun para programadores e desenvolvedor Java 2**. Rio de Janeiro: Editora Altas Books. 2003.

SIERRA, kathy; BATES, Bert. **Head first java 5**. 2 ed. Sebastopol: Editora O' Reilly Media. Books. 2005.

TOM, Pender. **UML, a bíblia**. Rio de Janeiro: Elsevier, 2004.

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 66 de 15

**UNIP**  
UNIVERSIDADE PAULISTA

## Perguntas



Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 67 de 15

**UNIP**  
UNIVERSIDADE PAULISTA

## Obrigado

Wanderley

[Wanderley.unip@gmail.com](mailto:Wanderley.unip@gmail.com)

[www.wg.pro.br](http://www.wg.pro.br)

Prof: Msc Wanderley Gonçalves Freitas ALPOO 06 – Arquitetura Web 68 de 15