

Programação Web: Introdução ao CSS: Classes, Atributos, Pseudo-classes e Herança

Prof. Eduardo Lino
eduardo.lino@pucpr.br



O que é CSS?

- CSS (Cascading Style Sheets) é uma linguagem de formatação para adicionar estilos (cores, fontes, espaçamentos, margem, etc.) à um documento HTML.
- A maioria dos menus em cascata, estilos de cabeçalho e rodapé de páginas da internet, por exemplo, atualmente são desenvolvidos em CSS.

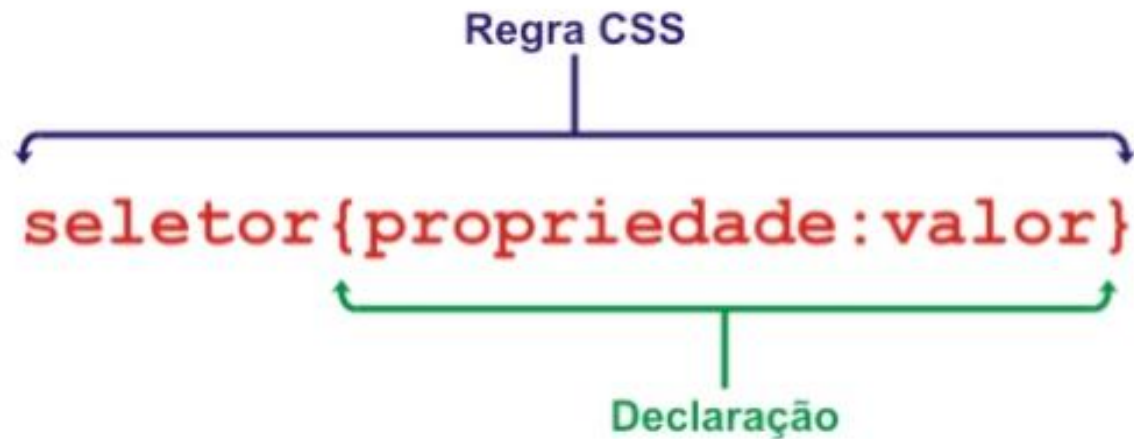


O que é CSS?

- CSS1: Foi a primeira versão, introduzida em 1996.
- CSS2: Foi introduzida em 1998.
- CSS2.1: Foi a segunda geração do CSS2.
- CSS3: É a última versão atualmente (2018), com muitas mudanças. Funções como: arredondamento dos cantos dos elementos, sombreamento, mais atributos para utilização do background dos elementos, etc.




Sintaxe



Sintaxe



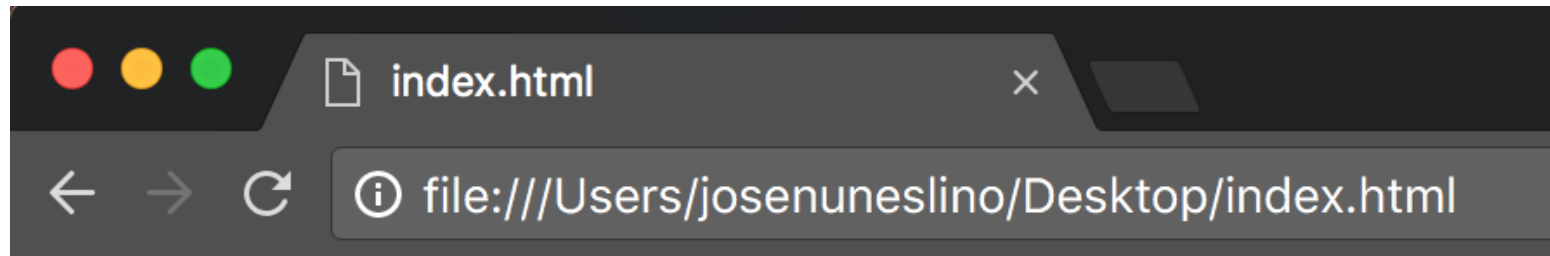
Desenhando um Botão com CSS



A screenshot of a code editor window titled 'index.html'. The editor shows the following HTML code:

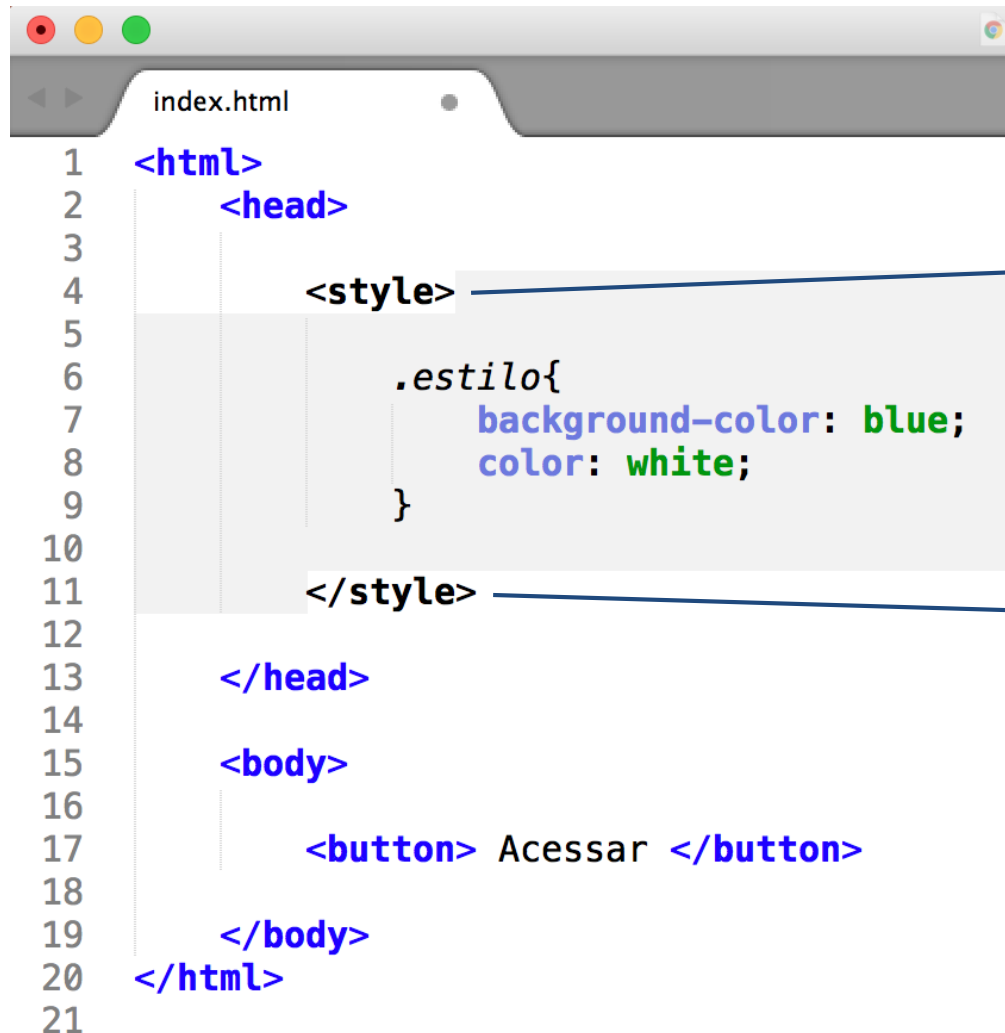
```
1 <html>
2   <head>
3
4   </head>
5
6   <body>
7
8       <button> Acessar </button>
9
10  </body>
11 </html>
12
```

Desenhando um Botão com CSS



Acessar

Desenhando um Botão com CSS



A screenshot of a web browser window showing a code editor for a file named 'index.html'. The code is as follows:

```
1 <html>
2   <head>
3
4     <style>
5
6       .estilo{
7         background-color: blue;
8         color: white;
9       }
10
11    </style>
12
13  </head>
14
15  <body>
16
17    <button> Acessar </button>
18
19  </body>
20 </html>
21
```

The CSS code is highlighted with a light gray background. A blue bracket on the right side of the code editor groups the CSS code from line 4 to line 11, pointing to the text 'Área do CSS'.

Área do CSS

Desenhando um Botão com CSS



```
1 <html>
2   <head>
3
4     <style>
5
6       .estilo {
7         background-color: blue;
8         color: white;
9       }
10
11    </style>
12
13  </head>
14
15  <body>
16
17    <button> Acessar </button>
18
19  </body>
20 </html>
21
```

Classe

Desenhando um Botão com CSS



```
1  <html>
2    <head>
3
4      <style>
5
6        .estilo{
7          background-color: blue;
8          color: white;
9        }
10     </style>
11
12   </head>
13
14   <body>
15
16     <button> Acessar </button>
17
18   </body>
19 </html>
20
21
```

Classe

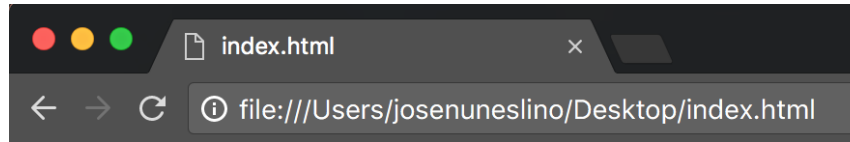
Desenhando um Botão com CSS



```
1 <html>
2   <head>
3
4     <style>
5
6       .estilo {
7         background-color: blue;
8         color: white;
9       }
10
11    </style>
12
13  </head>
14
15  <body>
16
17    <button class="estilo"> Acessar </button>
18
19  </body>
20 </html>
21
```

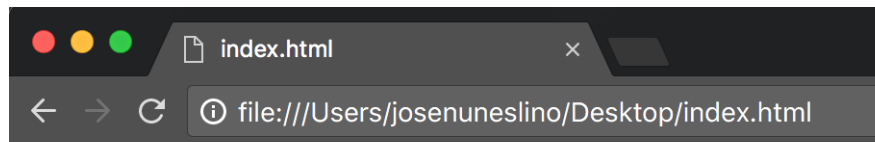
Classe

Desenhando um Botão com CSS



Sem CSS

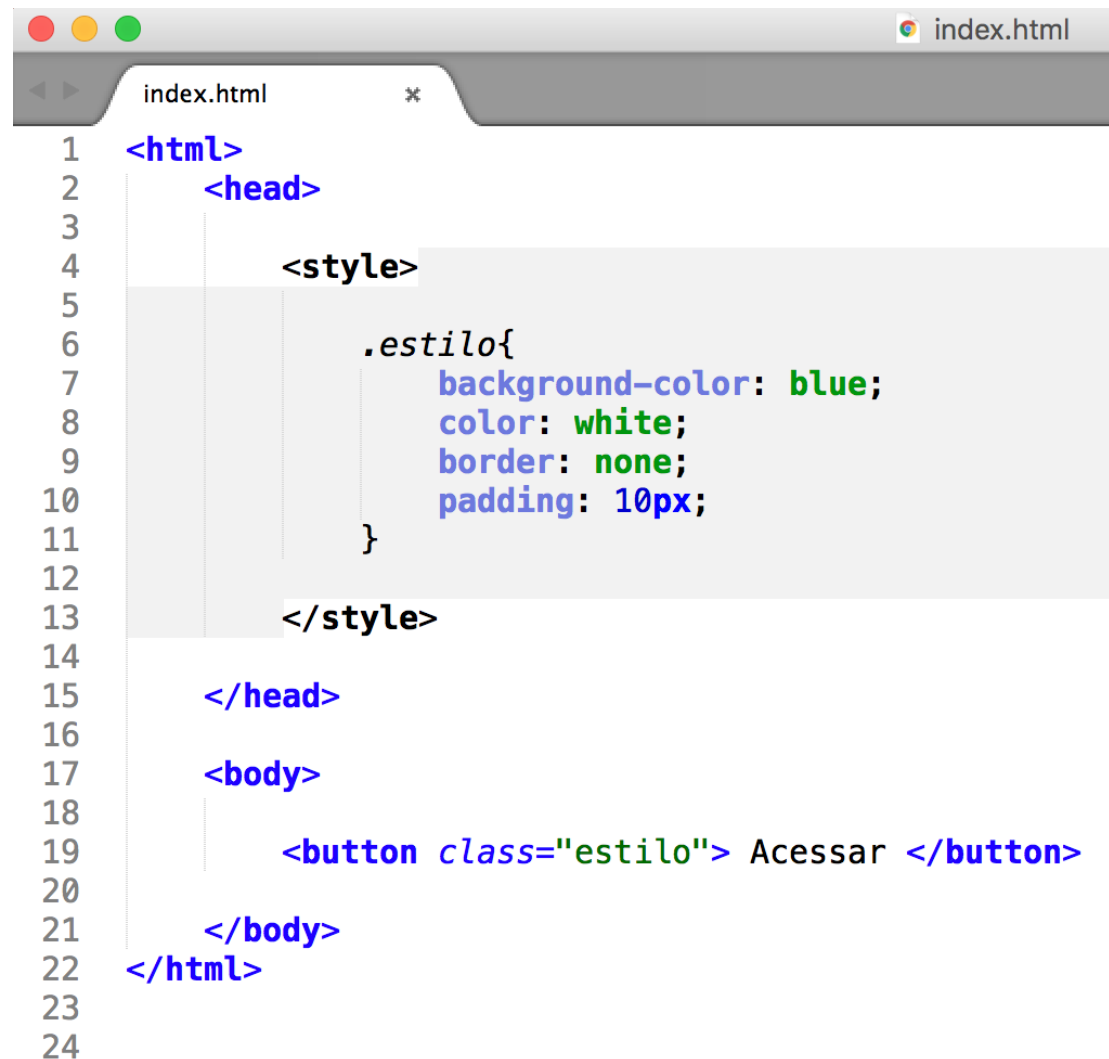
Acessar



Com CSS

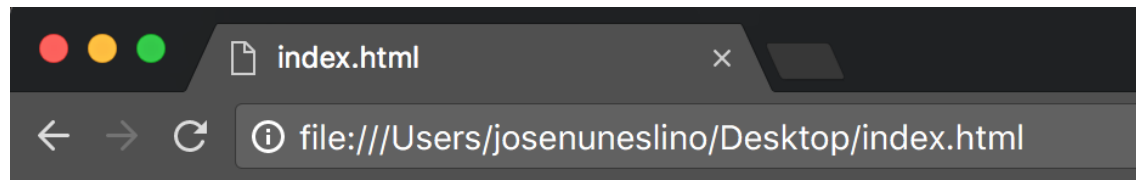
Acessar

Desenhando um Botão com CSS



```
1 <html>
2   <head>
3
4     <style>
5
6       .estilo{
7         background-color: blue;
8         color: white;
9         border: none;
10        padding: 10px;
11      }
12
13    </style>
14
15  </head>
16
17  <body>
18
19    <button class="estilo"> Acessar </button>
20
21  </body>
22 </html>
23
24
```

Desenhando um Botão com CSS



Acessar

CSS: Comentários

```
/* comentário */  
  
.estilo{  
    background-color: blue;  
    color: white;  
}
```

CSS: Atributos [W3C, 2018]

- Cores e Fundo.
- Fontes e Tipografia.
- Texto.
- Interface de Usuário.
- Bordas.
- Margem.
- Espaçamento.
- Formatação Visual.
- Detalhes da Formatação Visual.
- Efeito Visual.
- Modelo de caixa.

[W3C, 2018]: <http://www.w3c.br/divulgacao/guiasreferencia/css2/>

CSS: Cores e Fundo

Propriedade	Descrição	Valores	Descrição / Exemplos
color	Cor do texto presente no elemento	<color>	Exemplo: e {color: red; } e {color: rgb(255,0,0); }
background-color	Cor de fundo do elemento	[<color> transparent]	Para 'transparent' será adotada a cor de fundo do elemento superior e na ausência deste, a cor de fundo padrão do browser Exemplo: e {background-color: #f00; } e {background-color: transparent; }
background-image	Imagem de fundo do elemento	[url(...) none]	Exemplo: e {background-image: url(http://caminho/imagem.jpg); }
background-repeat	Define se e como a imagem de fundo utilizada se repetirá	[repeat repeat-x repeat-y no-repeat]	repeat: a imagem é repetida em ambas as direções: horizontal e vertical repeat-x: a imagem é repetida somente horizontalmente repeat-y: a imagem é repetida somente verticalmente. no-repeat: a imagem não será repetida e somente uma cópia da mesma será renderizada Exemplo: e {background-repeat: no-repeat;}

CSS: Fontes e Tipografia

Propriedade	Descrição	Valores	Descrição / Exemplos
font-family	Família(s) de fontes	[[<nome-fonte> <família-genérica>] [, <nome-fonte> <família-genérica>]*]	As seguintes famílias-genéricas de fontes são definidas em CSS2.1: 'serif', 'sans-serif', 'cursive', 'fantasy', 'monospace' Exemplo: e {font-family: "New Century Schoolbook", serif; }
font-style	Estilo utilizado na fonte	[normal italic oblique]	normal: fonte do elemento sem efeito itálico ou oblíquo italic: fonte do elemento com efeito itálico oblique: fonte do elemento com efeito oblíquo Exemplo: e {font-style: italic; }
font-weight	Peso da fonte	[normal bold bolder lighter 100 200 300 400 500 600 700 800 900]	normal: fonte do elemento com efeito padrão de peso (corresponde ao valor 400) bold: fonte do elemento com efeito de negrito (corresponde ao valor 700) bolder: fonte do elemento com o valor de peso imediatamente superior ao do valor herdado (p.ex: de 400 para 500) lighter: fonte do elemento com o valor de peso imediatamente inferior ao do valor herdado (p.ex: de 400 para 300) Exemplo: e {font-weight: bold; }
font-size	Tamanho da fonte	[[xx-small x-small small medium large x-large xx-large] [larger smaller] <tamanho> <porcentagem>]	medium: fonte do elemento com valor padrão de tamanho e equivalente ao padrão do browser. Os demais valores (xx-small x-small small large x-large xx-large) são calculados pelo browser considerando esse valor 'medium'. larger smaller: aumenta ou diminui o tamanho em relação ao valor herdado. tamanho: valor absoluto. Não são permitidos valores negativos. porcentagem: porcentagem do tamanho do valor herdado. Exemplo: e {font-size: 90%; }

CSS: Texto

Propriedade	Descrição	Valores	Descrição / Exemplos
text-indent	Recuo à esquerda da primeira linha em um bloco de texto	[<tamanho> <porcentagem>]	Exemplo: e {text-indent: 3em; }
text-align	Alinhamento do texto	[left right center justify]	left: alinhamento do texto à esquerda right: alinhamento do texto à direita center: alinhamento do texto centralizado justify: alinhamento do texto justificado Exemplo: e {text-align: center; }
letter-spacing	Espaçamento entre caracteres	[normal <tamanho>]	Exemplo: e {letter-spacing: 0.1em; }
word-spacing	Espaçamento entre palavras	[normal <tamanho>]	Exemplo: e {word-spacing: 1em; }
text-transform	Transforma o texto em maiúsculas/minúsculas	[capitalize uppercase lowercase none]	capitalize: coloca o primeiro caracter de cada palavra do elemento em maiúsculas (caixa alta) uppercase: coloca todos os caracteres de cada palavra do elemento em maiúsculas (caixa alta) lowercase: coloca todos os caracteres de cada palavra do elemento em minúsculas (caixa baixa) none: não aplica efeitos de capitalização Exemplo: e {text-transform: capitalize; }
white-space	Definições sobre espaços em branco num determinado elemento	[normal pre nowrap pre-wrap pre-line]	normal: utiliza a configuração padrão do browser para espaçamentos em branco pre: todos os espaços em branco presentes no código fonte serão exibidos no elemento nowrap: os espaços em branco serão tratados como 'normal', mas as quebras de linha serão suprimidas pre-wrap: todos os espaços em branco contantes no código-fonte serão exibidos no elemento pre-line: os espaços em branco serão tratados como normal. Exemplo: e {white-space: pre; }

CSS: Interface de Usuário

Propriedade	Descrição	Valores	Descrição / Exemplos
cursor	Especifica o cursor	[[http://.../cursor.ico,* [auto crosshair default pointer move e-resize ne-resize nw-resize n-resize se-resize sw-resize s-resize w-resize text wait help progress]	Exemplo: e {cursor:help; }
outline-style	Estilo da linha de contorno	<border-style>	Aceita os mesmos valores de 'border-style' Exemplo: e {outline-style: dotted; }
outline-color	Cor da linha de contorno	[<color> invert]	color: valor correspondente a uma cor invert: cor inversa à área do elemento (prática comum para tornar o 'focus' visível) Exemplo: e {outline-color: #foo; }
outline	Atalho para definir de uma só vez todas as propriedades de outline	[<outline-color> <outline-style> <outline-width>]	Exemplo: active {outline: black dotted thick; }

CSS: Bordas

Propriedade	Descrição	Valores	Descrição
border-top-width border-right-width border-bottom-width border-left-width	Largura das bordas em cada um dos quatro lados: superior, direito, inferior e esquerdo	[thin medium thick <tamanho>]	thin: espessura fina medium: espessura média thick: espessura grossa Exemplo: e {border-bottom-width: medium; }
border-width	Atalho para definir de uma só vez larguras de bordas para todos os lados	[thin medium thick <tamanho>] {1,4}	Exemplos: Todas as bordas com espessura 'thin': e {border-width: thin;} Bordas superior e inferior com espessura 'thin', bordas direita e esquerda com espessura 'thick': e {border-width: thin thick;} Borda superior com espessura 'thin', bordas direita e esquerda com espessura 'thick', borda inferior com espessura 'medium': e {border-width: thin thick medium;} Borda superior com espessura 'thin', borda direita com espessura 'thick', borda inferior com espessura 'medium', borda esquerda com espessura 'thin': e {border-width: thin thick medium thin;}
border-top-color border-right-color border-bottom-color border-left-color	Cor das bordas em cada um dos quatro lados: superior, direito, inferior e esquerdo	[<color> transparent]	Exemplo: e {border-top-color: black; }
border-color	Atalho para definir de uma só vez cores de bordas para todos os lados	[<color> transparent] {1,4}	Exemplos: Todas as bordas com a cor 'black': e {border-color: black;} Bordas superior e inferior com a cor 'black', bordas direita e esquerda com a cor 'silver': e {border-color: black silver;} Borda superior com a cor 'black', bordas direita e esquerda com a cor 'silver', borda inferior com a cor 'gray': e {border-color: black silver gray;} Borda superior com a cor 'black', borda direita com a cor 'silver', borda inferior com a cor 'gray', borda esquerda com a cor 'white': e {border-color: black silver gray white; }

CSS: Bordas (Continuação)

Propriedade	Descrição	Valores	Descrição
border-top-style border-right-style border-bottom-style border-left-style	Estilo de bordas em cada um dos quatro lados: superior, direito, inferior e esquerdo	[none hidden dotted dashed solid double groove ridge inset outset]	<p>none: nenhuma borda (largura computada como zero)</p> <p>hidden: idêntico à 'none', exceto na resolução de conflito de bordas nos elementos de uma tabela</p> <p>dotted: série de pontos</p> <p>dashed: série de segmentos de pequenas linhas</p> <p>solid: segmento de linha simples e contínua</p> <p>double: Duas linhas 'solid'. A soma das duas linhas e o espaço entre elas é igual ao valor definido em 'border-width'</p> <p>groove: Efeito 3D, entalhada</p> <p>ridge: Efeito 3D, oposto de 'groove': ressaltada</p> <p>inset: Efeito 3D, baixo relevo</p> <p>outset: Efeito 3D, alto relevo</p> <p>Exemplo: e {border-left-style: dotted; }</p>
border-style	Atalho para definir de uma só vez estilos de bordas para todos os lados	[none hidden dotted dashed solid double groove ridge inset outset] {1,4}	<p>Exemplos: Todas as bordas com estilo 'solid': e {border-style: solid;}</p> <p>Bordas superior e inferior com estilo 'solid', bordas direita e esquerda com estilo 'dotted': e {border-style: solid dotted;}</p> <p>Borda superior com estilo 'solid', bordas direita e esquerda com estilo 'dotted', borda inferior com o estilo 'dashed': e {border-color: solid dotted dashed; }</p> <p>Borda superior com estilo 'solid', borda direita com estilo 'dotted', borda inferior com estilo 'dashed', borda esquerda com o estilo 'double': e {border-color: solid dotted dashed double; }</p>
border-top border-right border-bottom border-left	Atalho para definir largura, estilo e cor das bordas superior, direita, inferior ou esquerda	[<border-top-width> <border-top-style> <border-top-color>]	Exemplo: e {border-bottom: thick solid red;}
border	Atalho para largura, estilo e cor das quatro bordas	[<border-top-width> <border-top-style> <border-top-color>]	Exemplo: e {border: thick solid red;}

CSS: Bordas (Continuação)

Propriedade	Descrição	Valores	Descrição
border-radius	Estilo de arredondamento de bordas	Valores em Pixel (px)	Exemplo: border-radius: 10px 10px 10px 10px; {canto superior esquerdo, canto superior direito, canto inferior esquerdo, canto inferior direito}

CSS: Margem

Propriedade	Descrição	Valores	Descrição
margin-top margin-right margin-bottom margin-left	Tamanho da margem para cada um dos lados: superior, direito, inferior e esquerdo	[<tamanho> <porcentagem> auto]	Exemplo: body {margin-top: 2em }
margin	Atalho para definir de uma só vez o tamanho da margem para todos os lados	[<tamanho> <porcentagem> auto]{1,4}	Exemplos: Todas as margens com 2em: body {margin: 2em } Margem superior e inferior com 1em, esquerda e direita com 2em: body {margin: 1em 2em } Margem superior com 1em; direita com 2em; inferior com 3em; e esquerda com 2em: body {margin: 1em 2em 3em } Margem superior com 1em; direita com 2em; inferior com 3em; e esquerda com 4em: body {margin: 1em 2em 3em 4em }

CSS: Espaçamento

Propriedade	Descrição	Valores	Descrição
padding-top padding-right padding-bottom padding-left	Distância utilizada para espaçamento em cada um dos quatro lados: superior, direito, inferior e esquerdo	[<tamanho> <porcentagem>]	Exemplo: body {padding-top: 2em }
padding	Atalho para definir de uma só vez a distância de espaçamento para todos os lados	[<tamanho> <porcentagem>]{1,4}	Exemplos: Todas as margens com 2em: body {padding: 2em } Margem superior e inferior com 1em, esquerda e direita com 2em: body {padding: 1em 2em } Margem superior com 1em; direita com 2em; inferior com 3em; e esquerda com 2em: body {padding: 1em 2em 3em } Margem superior com 1em; direita com 2em; inferior com 3em; e esquerda com 4em: body {padding: 1em 2em 3em 4em }

CSS: Formatação Visual

Propriedade	Descrição	Valores	Descrição / Exemplos
display	Modos de exibição do conteúdo	[inline block list-item run-in inline-block table inline-table table-row-group table-header-group table-footer-group table-row table-column-group table-column table-cell table-caption none]	<p>inline: O elemento renderizado em uma ou mais linhas block: renderizado como um bloco list-item: renderizado como um item de lista e em linha(s) run-in: renderizado como bloco ou linha(s), dependendo do contexto inline-block: renderizado em uma linha simples, mas comporta-se como um bloco</p> <p>table, inline-table, table-row-group, table-column, table-column-group, table-header-group, table-footer-group, table-row, table-cell, table-caption: esses valores farão com que o elemento se comporte como uma tabela (sujeito a restrições descritas no capítulo sobre tabelas)</p> <p>none: o elemento não será mostrado no modo visual</p> <p>Exemplo: <code>e {display: none;}</code></p>
position	Esquema de posicionamento	[static relative absolute fixed]	<p>static: Posicionamento normal de acordo com o fluxo normal dos elementos na página. As propriedades 'top', 'right', 'bottom', e 'left' não se aplicam</p> <p>relative: posicionamento é calculado de acordo com o fluxo normal. O efeito de 'position: relative' é indefinido nos elementos 'table-row-group', 'table-header-group', 'table-footer-group', 'table-row', 'table-column-group', 'table-column', 'table-cell', e 'table-caption'</p> <p>absolute: Posicionamento (e possivelmente o tamanho) é especificado com as propriedades 'top', 'right', 'bottom', e 'left'. Essas propriedades são especificadas em relação ao conteúdo do bloco</p> <p>fixed: Posicionamento é calculado tal como o modelo 'absolute', mas adicionalmente se manterá fixo, mesmo que se utilize a barra de rolagem</p> <p>Exemplo:</p> <pre>@media screen { e#exemplo {position: fixed; } } @media print { e#exemplo {position: static; } }</pre>
top right bottom left	Distância do elemento com relação à extremidade de seu elemento-pai	[<tamanho> <porcentagem> auto]	<p>top: Distância vertical em relação à margem superior right: Distância horizontal em relação à margem direita bottom: Distância vertical em relação à margem inferior left: Distância horizontal em relação à margem esquerda</p> <p>Exemplo:</p> <pre>e { top: 20px; left: 40px; }</pre>
z-index	Níveis de profundidade de sobreposição de elementos	[auto <número inteiro>]	<p>Os números serão considerados para ordenar os elementos que ficarão sobrepostos, do mais baixo para o mais alto: 'z-index: 1'; 'z-index: 2'; 'z-index: 3'; ...</p> <p>Exemplo: <code>e {z-index: 1;}</code></p>
direction	Define a direção de um texto: esquerda para direita; direita para esquerda	[ltr rtl]	<p>Utilizado em clientes que têm suporte a idiomas e escrita bidirecional.</p> <p>ltr - direção da esquerda para a direita rtl - direção da direita para a esquerda</p> <p>Exemplo:</p> <pre>e {direction: rtl;}</pre> <p>Para que a direção tenha o efeito atribuído, a propriedade 'unicode-bidi' deve conter os valores 'embed' ou 'bidi-override'</p>

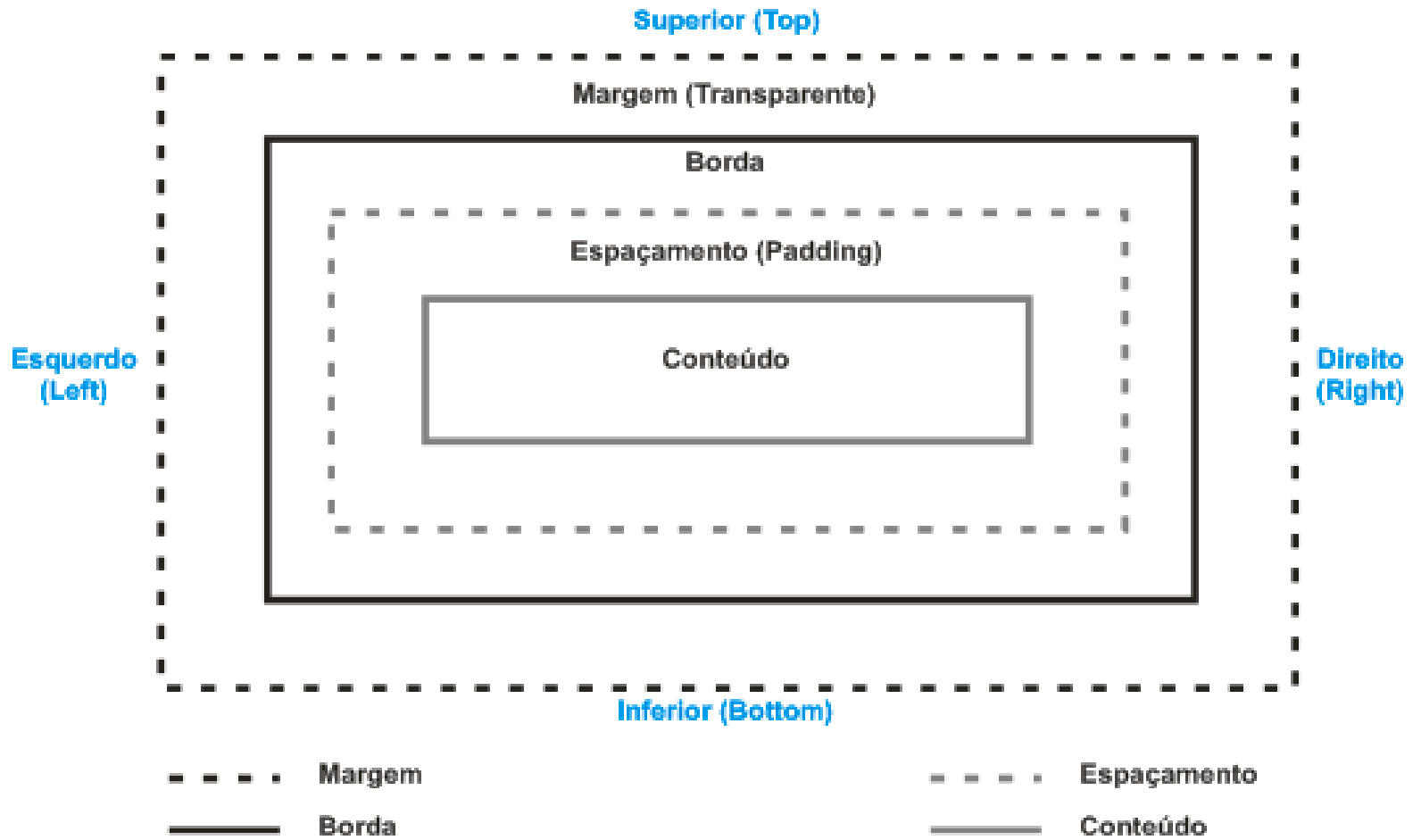
CSS: Detalhes da Formatação Visual

Propriedade	Descrição	Valores	Descrição / Exemplos
width	Largura	[<tamanho> <porcentagem> auto]	Exemplo: e {width: 25%;}
min-width	Largura mínima	[<tamanho> <porcentagem>]	Exemplo: e {min-width: 200px;}
max-width	Largura máxima	[<tamanho> <porcentagem> none]	Exemplo: e {max-width: 600px;}
height	Altura	[<tamanho> <porcentagem> auto]	Exemplo: e {height: 250px;}
min-height	Altura mínima	[<tamanho> <porcentagem>]	Exemplo: e {min-height: 30%;}
max-height	Altura máxima	[<tamanho> <porcentagem> none]	Exemplo: e {max-height: 70%;}
line-height	Distância entre as linhas de um texto	[normal <número> <tamanho> <porcentagem>]	Exemplo: e {line-height: 1.2em;}
vertical-align	Alinhamento vertical do texto	[baseline sub super top text-top middle bottom text-bottom <porcentagem> <tamanho>]	<p>baseline - A linha de base do elemento e do elemento superior (elemento-pai) serão consideradas para alinhamento. Se não houver linha de base no elemento, será considerado sua margem inferior</p> <p>sub - A linha de base do elemento será alinhada com a linha de base de conteúdo subscrito do elemento superior</p> <p>super - A linha de base do elemento será alinhada com a linha de base de conteúdo superescrito do elemento superior</p> <p>top - Alinha o topo do elemento com a linha de topo do elemento superior</p> <p>text-top - A base superior do elemento será alinhada com a linha superior do conteúdo de textos do elemento-pai</p> <p>middle - A linha central do elemento e de seu elemento superior serão consideradas para alinhamento. (linha central do elemento superior = ponto central da altura da letra 'x')</p> <p>bottom - Alinha o rodapé do elemento com o rodapé do elemento superior</p> <p>text-bottom - A base inferior do elemento será alinhada com a linha inferior do conteúdo de textos do elemento-pai</p> <p>Exemplo: e {vertical-align: sub;}</p>

CSS: Efeito Visual

Propriedade	Descrição	Valores	Descrição / Exemplos
overflow	Comportamento do browser quando o conteúdo do elemento for maior que sua área	[visible hidden scroll auto]	<p>visible - Todo o conteúdo do elemento estará visível (não cortado) e talvez conforme a área seja renderizado para fora do elemento</p> <p>hidden - O conteúdo será cortado (se maior que a área disponível) e não haverá barra de rolagens no elemento</p> <p>scroll - O conteúdo será cortado (se maior que a área disponível) mas haverá barra de rolagens no elemento. Quando a mídia for especificada como 'print', o conteúdo do elemento poderá ser impresso.</p> <p>auto - O método será atribuído pelo browser cliente</p> <p>Exemplo: e {overflow: scroll;}</p>
clip	Especifica região de um elemento a ser exibida	[rect (<tamanho topo>, <tamanho direito>, <tamanho rodapé>, <tamanho esquerdo>) auto]	<p>rect – Delimita área retangular do elemento (topo, direita, rodapé, esquerda) na qual o conteúdo ficará visível</p> <p>auto - O elemento não terá clip e sua própria área será idêntica à de visualização</p> <p>Exemplo: e {clip: rect(5px, 40px, 45px, 5px); }</p>
visibility	Define se um elemento estará ou não visível	[visible hidden collapse]	<p>visible - O elemento será exibido</p> <p>hidden - O elemento não será exibido, ficando totalmente transparente. Mas o layout da página continuará sendo afetado</p> <p>collapse - Mesmo efeito que hidden, mas aplicável a elementos de tabela</p> <p>Exemplo: e {visibility: hidden; }</p>

CSS: Modelo de caixa



CSS

- Boas práticas de programação web: **separarão das linguagens em diferentes arquivos.**

index.html

```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="css/estilos.css">
  </head>
  <body>
    <button class="botao"> Acessar </button>
  </body>
</html>
```

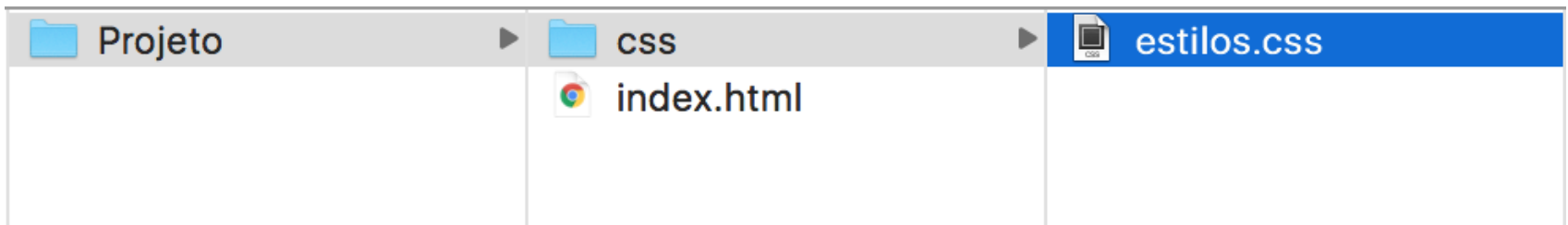
estilos.css

```
.botao{
  background-color: blue;
  color: white;
}
```

CSS

- Boas práticas de programação web: **separarão das linguagens em diferentes arquivos.**

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



CSS

- Boas práticas de programação web: **separarão das linguagens em diferentes arquivos.**

Atributos obrigatórios

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

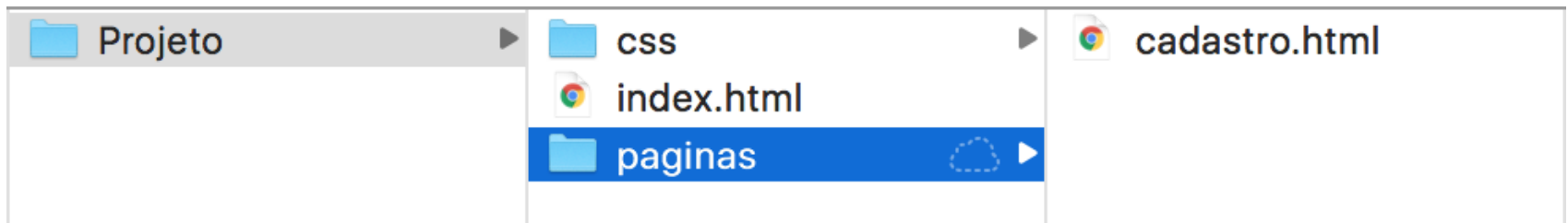


CSS Dinâmico

- Boas práticas de programação web: **separarão das linguagens em diferentes arquivos.**

cadastro.html

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



CSS: identificação com id

- Em uma representação comum apenas entre HTML e CSS, temos:

```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="../css/estilos.css" />
  </head>
  <body>
    <button id="botao"> Acessar </button>
    <button> Acessar </button>
  </body>
</html>
```

```
#botao{
  background-color: blue;
  color: white;
}
```

CSS: identificação com id

- Em uma representação comum apenas entre HTML e CSS, temos:

```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="../css/estilos.css" />
  </head>
  <body>
    <button id="botao"> Acessar </button>
    <button class="botao"> Acessar </button>
  </body>
</html>
```

```
#botao{
  background-color: blue;
  color: white;
}
```

CSS: identificação com id

- Em uma representação comum apenas entre HTML e CSS, temos:

```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="../css/estilos.css" />
  </head>
  <body>
    <button id="botao"> Acessar </button>
    <button class="botao"> Acessar </button>
  </body>
</html>
```

```
#botao{
  background-color: blue;
  color: white;
}
```

Não irá carregar os valores do **#botao**

CSS: Pseudo-classe

- Uma ***pseudo-classe*** CSS é uma palavra-chave adicionada a seletores que especifica um estado especial do elemento selecionado. Por exemplo, **:hover** pode ser usado para alterar a cor de um botão quando o usuário passar o cursor sobre ele.

```
.estilo{  
    background-color: blue;  
    color: white;  
}  
  
.estilo:hover{  
    background-color: red;  
}
```

CSS: Pseudo-classe

`:active`

`:checked`

`:default`

`:dir()`

`:disabled`

`:empty`

`:enabled`

`:first`

`:first-child`

`:first-of-type`

`:fullscreen`

`:focus`

`:hover`

`:indeterminate`

`:in-range`

`:invalid`

`:lang()`

`:last-child`

`:last-of-type`

`:left`

`:link`

`:not()`

`:nth-child()`

`:nth-last-child()`

`:nth-last-of-type()`

`:nth-of-type()`

`:only-child`

`:only-of-type`

`:optional`

`:out-of-range`

`:read-only`

`:read-write`

`:required`

`:right`

`:root`

`:scope`

`:target`

`:valid`

`:visited`

CSS: Pseudo-classe

`:active`

`:checked`

`:default`

`:dir()`

`:disabled`

`:empty`

`:enabled`

`:first`

`:first-child`

`:first-of-type`

`:fullscreen`

`:focus`

`:hover`

`:indeterminate`

`:in-range`

`:invalid`

`:lang()`

`:last-child`

`:last-of-type`

`:left`

`:link`

`:not()`

`:nth-child()`

`:nth-last-child()`

`:nth-last-of-type()`

`:nth-of-type()`

`:only-child`

`:only-of-type`

`:optional`

`:out-of-range`

`:read-only`

`:read-write`

`:required`

`:right`

`:root`

`:scope`

`:target`

`:valid`

`:visited`

CSS: Herança

- Podemos ainda utilizar um seletor hierárquico que permite aplicar estilos aos elementos filhos de um elemento pai:

```
.rodape img {  
    margin-right: 35px;  
    vertical-align: middle;  
    width: 94px;  
}
```


CSS: Herança

- Podemos ainda utilizar um seletor hierárquico que permite aplicar estilos aos elementos filhos de um elemento pai:

```
.rodape img {  
    margin-right: 35px;  
    vertical-align: middle;  
    width: 94px;  
}
```

- No exemplo acima, o elemento pai rodape é selecionado pela classe. O estilo será aplicado apenas nos elementos img filhos do elemento com class= "rodape".

CSS: Herança

```
.classDivMarcas #tabelaMarcas tr td{  
    padding: 50px;  
}
```

CSS: Pseudo-classe

- O CSS tem um recurso chamado de pseudo-classes que são como classes CSS já pré-definidas para nós. É como se o navegador já colocasse certas classes por padrão em certos elementos, cobrindo situações comuns como a de selecionar o primeiro ou o último elemento.

```
<ul>
  <li>Primeiro item</li>
  <li>Segundo item</li>
  <li>Terceiro item</li>
  <li>Quarto item</li>
</ul>
```

```
li:first-child {
  color: red;
}
li:last-child {
  color: blue;
}
```

CSS: Pseudo-classe

- **selector:first-child**
- **selector:last-child**

CSS: Pseudo-classe

- **selector:first-child**
- **selector:last-child**
- **selector:nth-child** (mais genérico)

CSS: nth-child

- Um seletor ainda mais genérico do CSS3 é o :nth-child() que nos permite passar o índice do elemento. Por exemplo, podemos pegar o terceiro item com:

```
li:nth-child(3) { color: yellow; }
```

CSS: nth-child

- Porém, o mais interessante é que o nth-child pode receber uma expressão aritmética para indicar quais índices selecionar. É fácil fazer uma lista zebra, com itens ímpares de uma cor e pares de outra:

```
li:nth-child(2n) { color: green; } /* elementos pares */  
li:nth-child(2n+1) { color: blue; } /* elementos ímpares */
```

CSS: Pseudo-classe

- :hover

- Quando se deseja mudar a cor de um elemento quando o usuário passa o mouse por cima. Ou seja, queremos mudar seu visual a partir de um evento do usuário (no caso, passar o mouse em cima).

- :link

- Seleciona todos os elementos que possuem o atributo "href".

- :visited

- Seleciona todos os elementos que possuem o atributo "href" e que já foram visitados.

Exercício em Sala

- Elaborar um programa web que possua uma tabela de no mínimo 5 linhas e 3 colunas. Esta tabela deverá possuir um cabeçalho e de cor única. O restante das linhas da tabela deverão possuir cores diferentes, as linhas pares de uma cor, e as ímpares de outra cor. Ao passar o mouse por cima da tabela, as linhas deverão ficar com outra cor, diferente de qualquer outra já utilizada no programa. Lembre-se que o cabeçalho não faz parte do corpo da tabela, sendo assim, ao passar o mouse por cima dele, não deverá ser mudado a cor.

Exercício em Sala

Marca	Modelo	Ano	Cor	Portas
Chevrolet	Corsa	2011	Verde	2
Chevrolet	Corsa	2009	Vermelho	2
Chevrolet	Corsa	2010	Preto	2
Chevrolet	Corsa	2010	Preto	2
Chevrolet	Corsa	2010	Preto	2



Exercício em Sala

Marca	Modelo	Ano	Cor	Portas
Chevrolet	Corsa	2011	Verde	2
Chevrolet	Corsa	2009	Vermelho	2
Chevrolet	Corsa	2010	Preto	2
Chevrolet	Corsa	2010	Preto	2
Chevrolet	Corsa	2010	Preto	2



Elemento <div>

- A tag **<div>** define uma divisão ou seção em um documento HTML, é frequentemente utilizado para bloquear o grupo de elementos para formatá-los com estilos.
- A **<div>** serve para formatar as caixas dando tamanho, bordas, cores e outras funcionalidades.
- O elemento **<div>** deve ser inserido na **<body>**
- Entre **<div>** e **</div>** coloca-se o conteúdo.

Elemento <div>

A propriedade **float** do CSS determina que um elemento deve ser retirado do seu fluxo normal e colocado ao longo do lado direito ou esquerdo do seu container, onde textos e elementos em linha irão se posicionar ao seu redor.

Um elemento flutuante é um tipo de elemento cujo valor de **float** é diferente de **none**.

Elemento <div>

O valor de left para o atributo float, indica que o elemento deve flutuar à esquerda do bloco.

- **Left**

- É uma palavra-chave que indica que o elemento deve flutuar à esquerda do bloco.

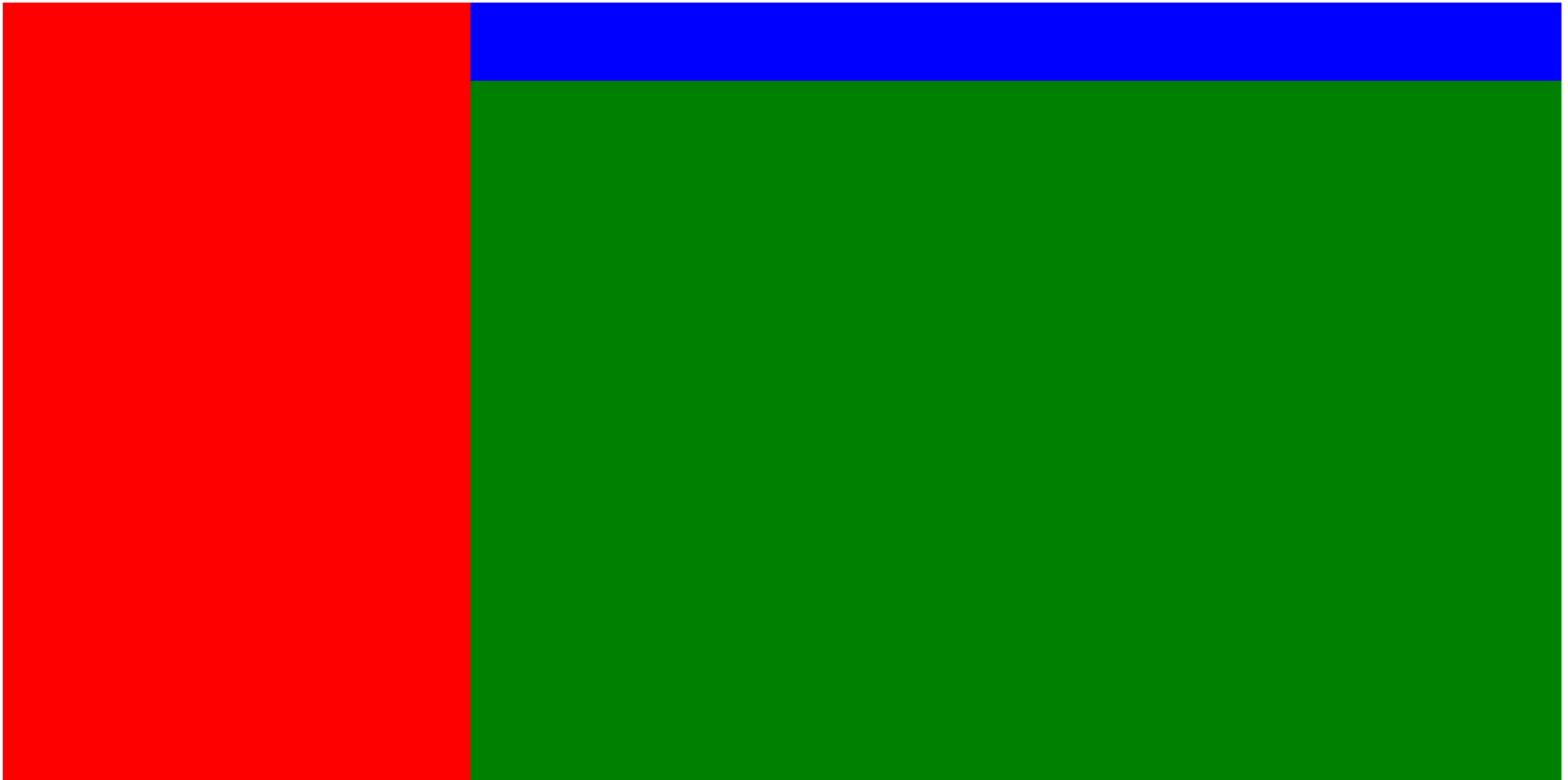
- **Right**

- É uma palavra-chave que indica que o elemento deve flutuar à direita do bloco.

- **None**

- É uma palavra-chave que indica que o elemento não deve flutuar.

Elemento <div>



Exercício em Sala

- Criar um formulário de cadastro EXATAMENTE igual ao formulário da próxima página.
- Pelo menos um campo deverá ter a cor da borda em vermelho.
- O Cadastro deverá ficar dentro da **<div>** de cor **verde** apresentada no slide anterior.
- Na **<div>** de cor **vermelha**, deverá ficar um menu com pelo menos 5 botões.
- No **<div>** de cor **azul**, deverá ficar o cabeçalho da página, com o título da página.

Exercício em Sala

← → ↻ ⓘ file:///Users/josenuneslino/Desktop

Nome
Sobrenome
E-mail
Matricula
Usuário
Senha
Confirmar Senha

Salvar

← → ↻ ⓘ file:///Users/josenuneslino/Desktop

Eduardo
Lino
E-mail
1018912
eduardo.lino
.....
.....

Salvar



Exercício em Sala (Chrome)



← → ↻ ⓘ file:///Users/josenuneslino/Desktop

10px
20px

Nome

Sobrenome

E-mail

Matricula

Usuário

Senha

Confirmar Senha

3px

20px

40px

100px

2px

#CCC 1px

#088A68

← → ↻ ⓘ file:///Users/josenuneslino/Desktop

Eduardo → Arial

Lino

E-mail

1018912

eduardo.lino

.....

.....

#FA5858

Salvar

#04B486

Contato

eduardo.lino@pucpr.br