



1. Apresentação das linguagens e dos conceitos básicos

OBJETIVOS



OBJETIVOS

- Delinear um breve histórico sobre o HTML e CSS
- Apresentar as boas práticas e o W3C
- Fazer considerações importantes
- Produzir um briefing
- Montar a arquitetura da informação do projeto

HTML

- HTML: linguagem criada para facilitar a comunicação
- HTML: *hypertext markup language* (linguagem de marcação de hipertexto)
- HTML não é uma linguagem de programação
- A função do HTML é definir a estrutura do site, ou seja, a parte editorial e criar links

HTML5

- A principal função do HTML5 é explorar a comunicação homem-máquina
- Código mais leve para que o conteúdo chegue de forma rápida aos olhos dos leitores
- Surgimento de novos elementos:
 - Sessão
 - Cabeçalho
 - Rodapé
 - Formulário
 - Áudio
 - Vídeo

HTML5

- Quando o HTML5 é associado com essas novas possibilidades ao JavaScript e ao CSS, pode-se criar jogos e animações para todos os dispositivos

CSS

- É o responsável pelo aspecto da página, pelo layout
- Além de garantir acessibilidade, tanto para máquinas quanto para pessoas
- Veio para agilizar a leitura das páginas web pelos navegadores
- E também para facilitar a vida de usuários que tem necessidades especiais ou não

CSS

- É uma sequência de declarações de propriedades e seus valores para manipulação da forma como o conteúdo de uma página web será exibido
- Deve-se saber qual elemento está selecionado para aplicar um determinado estilo
- Utilizando corretamente o HTML e CSS, a página web pode ficar até 56% mais leve

CSS

- O uso correto dos estilos permite ao desenvolvedor uma variação incrível de design para suas páginas
- Exemplo:
 - <http://www.csszengarden.com>
- É sempre o mesmo conteúdo, variando apenas as folhas de estilos

Webstandards (boas práticas)

- São regras para que a WWW não vire um caos absoluto
- São criados por uma porção de pessoas e empresas que se unem ao W3C (página internacional: <http://www.w3c.org>; página do escritório brasileiro: <http://www.w3c.br>)
- Tem como função regulamentar as normas técnicas para a web

Webstandards (boas práticas)

- Os quatro objetivos primários são:
 - Web para todos
 - Web em qualquer aparelho
 - Base de conhecimento e credibilidade
 - Confiança
- Humanização da interação homem/máquina

Um programa para escrever, outro para ler

- Existem várias opções de programas para escrever o código
 - Bloco de notas: editor padrão em ambiente Windows
 - Notepad++: editor HTML gratuito (<https://notepad-plus-plus.org>)
 - Sublime
 - Visual Studio Code
- Para leitura utilizamos os navegadores: Microsoft Edge, Firefox, Chrome, IE, etc.

Organização

- É a palavra chave para um desenvolvedor web
- HTML é uma linguagem de marcação e trabalha o tempo todo com referências, para buscar hiperlinks, exibir imagens, vincular folhas de estilo, etc.
- Para isso seguiremos o padrão a seguir para estruturar as atividades desenvolvidas

Organização

- Html5-css3
 - _finalizados
 - estilos
 - interação
 - multimídia
 - recursos
 - cap02
 - _propostos
 - projeto_inicio
 - cap03
 - _propostos
 - projeto_inicio
- Html5-css3
 - cap04
 - _propostos
 - projeto_inicio
 - cap05
 - _propostos
 - projeto_inicio
 - teoria

Nomeando seus arquivos

- Os nomes de arquivos e das pastas de objetos que serão publicados na web NÃO podem ter espaços e nem caracteres especiais, como aspas, acentos, cedilhas, etc.
- De preferência use nome com letras minúsculas e sem começar por número
- No máximo use o sinal de hífen ou underscore para facilitar a leitura do nome

Layout versus conteúdo

- Primeiro será feito a parte estrutural do documento usando o HTML
- Depois deixará o conteúdo mais atrativo com CSS
- Sabemos que no começo queremos muito alinhar, encher de cores, imagens de fundo e outras belezas, mas vamos com calma
- O conteúdo vem primeiro





2. HTML 5 – Elementos iniciais

OBJETIVOS



OBJETIVOS

- Conhecer o corpo básico
- Criar a sua primeira página
- Utilizar as tags básicas do HTML
- Aprender a utilizar tabelas
- Inserir imagens
- Utilizar links
- Trabalhar com formulários

Preparando-se para utilizar o HTML5

- É uma linguagem utilizada para descrever páginas web
- Também conhecida como uma linguagem de marcação, pois sua estrutura é descrita por uma série de tags (marcas)

Preparando-se para utilizar o HTML5

- Versões do HTML:

Versão	Ano
HTML	1991
HTML+	1993
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2012

Tags

- Todo documento terá esse conjunto de instruções que formam o corpo básico da maioria das páginas
- Tags: são palavras-chave cercadas por colchetes angulares, por exemplo, <html>

Tags

- Tags geralmente se apresentam em pares, como <body> e </body>, sendo que:
 - a primeira tag em um par é a tag inicial ou de abertura, e a segunda é a tag final, ou de fechamento
 - Existem tags especiais, chamadas abertas ou vazias. Elas não possuem fechamento. Por exemplo:
, <a>
 - A tag final é escrita como a marca inicial, com a adição de uma barra antes do nome da tag, como em </body>
- Sintaxe: <TagName> conteúdo </TagName>

Boas práticas para utilizar as tags

- Abrir e fechar as tags
- Letras minúsculas nas tags
- Atributos entre aspas (veremos mais adiante)

Elementos HTML

- Corresponde ao conjunto que vai desde a tag inicial até a tag final

Tag inicial	Conteúdo	Tag final
<p>	Este é um parágrafo	</p>

Escolhendo um editor HTML

- Notepad++ (adotaremos este a princípio)
 - <https://notepad-plus-plus.org/>
- Sublime Text
 - <https://www.sublimetext.com/>
- Visual Studio Code
 - <https://code.visualstudio.com/>
- Atom
 - <https://atom.io/>

Navegadores

- É importante que tenha o maior número de navegadores para testar sua página
 - Chrome
 - Firefox
 - Internet Explorer
 - Opera
 - Safari
 - Microsoft Edge

Atividade 1 – Corpo básico e primeiras instruções

- Objetivo:
 - Criar a primeira página na web e aprender a utilizar as instruções básicas
- Tarefas:
 - Digitar as instruções do corpo básico
 - Salvar sua página
 - Abrir sua página em um navegador

Utilizando um editor de texto e criando o corpo básico

1. Abra o editor
2. Digite as instruções a seguir:

```
<!DOCTYPE html>  
  
<html>  
  
<head> </head>  
  
<body> </body>  
  
</html>
```

Conhecendo a estrutura de uma página HTML

- <!DOCTYPE>
- <html> conteúdo </html>
- <head> conteúdo </head>
- <body> conteúdo </body>

<!DOCTYPE>

- Primeira coisa que deve ser inserida
- Desta forma você está dizendo a qualquer navegador o tipo de documento e que versão está sendo utilizada
- Essa instrução deve ser colocada em todos os arquivos HTML como primeiro item

```
<html> . . . </html>
```

- Informa ao navegador que o documento em questão é um documento HTML
- Esse elemento representa a raiz de um documento HTML com suas respectivas instruções e se comportará como um recipiente para outras tags
- Seu conteúdo é definido como corpo

```
<head> . . . </head>
```

- É utilizado para indicar os elementos do cabeçalho da página do seu navegador
- Dentro da <head> é possível incluir scripts, adicionar um título ao documento, instruir o navegador onde encontrar folhas de estilo e informações meta, entre outros

<head> . . . </head>

- As seguintes tags podem ser adicionadas à seção de cabeça:
 - <title>: define o título do documento
 - <style>: define as informações de estilo para um documento HTML
 - <meta>: contém elementos para especificar o comportamento e a descrição da página, as palavras-chave, o autor do documento, a última modificação, entre outros metadados
 - <link>: é mais usado para buscar as orientações de estilo de páginas em outro arquivo

<head> . . . </head>

- As seguintes tags podem ser adicionadas à seção de cabeça:
 - <script>: é usado para definir um script do lado do cliente, como um JavaScript
 - <noscript>: define um conteúdo alternativo para os usuários que têm os scripts com deficiência em seu navegador ou tem um navegador que não suporta script
 - <base>: especifica uma URL padrão e um tipo de abertura de janela para os links que estiverem contidos naquela página

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

- Define o corpo do elemento
- Dentro desse elemento estará todo o conteúdo de um documento HTML, como textos, hiperlinks, imagens, tabelas, listas, etc.

Salvando sua página

- Ao salvar um arquivo HTML, use a extensão html
- Na caixa *Nome*, digite *meuprimeirosite.html*
- Na caixa *Tipo*, selecione *Todos os arquivos*
- Na caixa *Codificação*, escolhe a opção *UTF-8*
 - Utilizado para representar qualquer caractere universal padrão
 - Garante que os caracteres especiais serão utilizados de acordo com o S.O. utilizado pelo internauta
- Não utilize acentos ou letras maiúsculas para os nomes dos arquivos

Inserindo os primeiros textos

1. Feche o seu navegador e volte ao bloco de notas.
2. Acrescente aos comandos anteriormente digitados, dentro da tag <body>, a frase a seguir (indicada em negrito):

```
<!DOCTYPE html>
<html>
<head> </head>
<body>
Saudações. Bem-vindo à minha primeira página.
</body>
</html>
```

3. Salve sua página e abra em seu navegador.

Conhecendo atributos HTML

- Elementos HTML podem ter atributos que fornecem informações adicionais sobre um elemento
- Eles são sempre especificados em uma tag inicial e devem ser informados em pares com nome e valor
- Exemplo:
 - ` Acesse o link `

Conhecendo atributos HTML

- Elementos HTML podem ter atributos que fornecem informações adicionais sobre um elemento
- Eles são sempre especificados em uma tag inicial e devem ser informados em pares com nome e valor
- Exemplo:
 - ` Acesse o link `

Trabalhando novos atributos

- Para evitar problemas de acentuação, vamos adicionar outros elementos:

1. Insira a sequência a seguir dentro o elemento

```
<head>
```

```
...
```

```
<head>
```

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

```
</head>
```

```
...
```

2. O atributo *charset* especifica o conjunto de caracteres para o documento HTML

Trabalhando novos atributos

- Outra coisa que deve-se informar é o idioma em que a página vai ser escrita
- Isso vai auxiliar os mecanismos de busca, como Google, Bing, etc.

3. Digite a sequência de comandos dentro do elemento <html>

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

...

Trabalhando novos atributos

4. Digite o seguinte código dentro da tag <head>

```
...
<head>
    <meta charset="utf-8"/>
    <title>Primeira página</title>
</head>
...
```

5. Salve a página e abra o navegador

- Observe que o título *Primeira página* foi inserido na aba do seu navegador

Trabalhando novos atributos

- Com isso, já conhecemos o corpo básico para cada página que criará para seu site

```
<!DOCTYPE html>  
  
<html lang="pt-br">  
  
    <head>  
        <meta charset="utf-8">  
        <title>XXXXXXXXXXXX</title>  
    </head>  
  
    <body>  
        XXXXXXXXXXXXXXXXXXXXXXX  
    </body>  
  
</html>
```

Comentários no HTML

- São usadas para inserir um orientação ou uma documentação sobre o que está sendo feito com a linguagem HTML
- Sintaxe: < ! Escreva seus comentários>
- Comentários não são exibidos pelo navegador
- Exemplos:

```
<!--Este é um comentário-->  
<p>Este é um parágrafo.</p>  
<!--Lembre-se de adicionar outras informações aqui-->
```

Atividade 2 – Quebra de linhas e parágrafos

- Objetivo:
 - Inserir novas tags na sua primeira página
- Tarefas:
 - Abrir o arquivo existente
 - Trabalhar com as tags p, br e h1 a h6
 - Salvar a sua página
 - Abrir a sua página no navegador

Atividade 2 – Quebra de linhas e parágrafos

- Suponha que você queira que sua página apresente o seguinte texto, com as quebras de linha e parágrafo conforme a seguir:

Saudações. Bem-vindo à minha primeira página.

Aqui será apresentada a nova versão do HTML.

Essa versão traz consigo importantes mudanças quanto ao papel do HTML no mundo da web, por meio de novas funcionalidades, como semântica e acessibilidade.

Novos recursos que antes só eram possíveis por meio de outras tecnologias.

- Salve o seu arquivo e abra a página no seu navegador

Inserindo uma quebra de linha

- Sintaxe:

- Exemplo:
 - Este é
 um texto
 com quebra de linhas
- Permite a quebra de linhas em um texto

Inserindo uma quebra de linha

1. No seu editor HTML insira as tags conforme a seguir:

```
...
<body>
    Saudações. Bem-vindo à minha primeira página. <br>
    Aqui será apresentada a nova versão do HTML. <br>
    Essa versão traz consigo importantes mudanças quanto ao papel do
    HTML no mundo da web, por meio de novas funcionalidades, como
    semântica e acessibilidade.
    Novos recursos que antes só eram possíveis por meio de outras
    tecnologias.
</body>
...
```

Trabalhando com parágrafos <p>

- Sintaxe: <p> conteúdo </p>
- Exemplo:
 - <p>Este é um parágrafo.</p>
 - <p>Este é outro parágrafo.</p>
- A necessidade de exibição do próximo parágrafo para alguns programadores poderia ser solucionada com a inserção de duas tags
, porém essa não é a melhor alternativa

Trabalhando com parágrafos <p>

1. No seu edito HTML, insira a tag conforme a seguir:

...

Essa versão traz consigo importantes mudanças quanto ao papel do HTML no mundo da web, por meio de novas funcionalidades, como semântica e acessibilidade.

```
<p>Novos recursos que antes só eram possíveis por meio de outras  
tecnologias. </p>  
</body>
```

...

2. Salve o seu arquivo e o verifique no navegador

Atividade 3 – Estrutura e formatação

- Objetivo:
 - Criar a página “Quem somos” para o seu projeto
- Tarefas:
 - Abrir o arquivo existente
 - Salvar o arquivo em outra subpasta
 - Trabalhar com as tags *Strong*, *br*, *p* e *h1* a *h6*
 - Conhecer as tags de formatação *b*, *i*, *em*, *sub*, *sup*, *del*, *ins*, *mark*, *pre* e *blockquote*
 - Salvar a sua página

Entendendo o projeto

- Nas próximas atividade, o seu foco será o desenvolvimento de um site de um pet shop
- Algumas páginas serão desenvolvidas e serão utilizadas outros comandos do HTML para isso
- Atenção: é importante definir em que locais estarão arquivados os seus arquivos

Entendendo o projeto

The image shows a Windows desktop environment. On the left, there is a 'Bloco de notas' (Notepad) window titled 'quem-somos'. It contains the following HTML code:

```
<!doctype html>
<html lang="pt-br">
<head>
    <meta charset="utf-8">
    <title>Quem somos - Meu amigo cão</title>
</head>
<body>

    Quem somos

    Em que nós acreditamos
    Na Meu amigo cão, o nosso sucesso como uma
    organização é conduzido pelo cuidado
    que temos com nossos animais.

    Cras semper magna in velit tempus tristique quis
    non lorem. Mauris vitae fringilla sapien, sed
    volutpat elit. Fusce molestie euismod odio, ut
    blandit nulla bibendum ac. Fusce posuere turpis sed
    sem tincidunt, sed auctor tellus faucibus. Proin
    ultrices enim vitae ipsum luctus, sed malesuada
    orci ornare. Vivamus eget molestie magna, sed
    aliquam risus. Aliquam adipiscing laoreet felis,
    eget tincidunt purus sollicitudin sit amet. Nulla
    facilisi.

    Donec dapibus nibh eu ipsum elementum, non


```

On the right, there is a browser window titled 'Quem somos - Meu amigo cão'. The page content is as follows:

Quem somos Em que nós acreditamos Na Meu amigo cão, o
nosso sucesso como uma organização é conduzido pelo
cuidado que temos com nossos animais. Cras semper magna
in velit tempus tristique quis non lorem. Mauris vitae
fringilla sapien, sed volutpat elit. Fusce molestie euismod
odio, ut blandit nulla bibendum ac. Fusce posuere turpis sed
sem tincidunt, sed auctor tellus faucibus. Proin ultrices enim
vitae ipsum luctus, sed malesuada orci ornare. Vivamus eget
molestie magna, sed aliquam risus. Aliquam adipiscing
laoreet felis, eget tincidunt purus sollicitudin sit amet. Nulla
facilisi. Donec dapibus nibh eu ipsum elementum, non
consectetur massa facilisis. Nam commodo at leo non
commodo. In blandit arcu sit amet sem convallis, rutrum
gravida quam elementum. Cras accumsan orci vel lectus
blandit, at dictum tellus rhoncus. Etiam vitae urna nec leo
facilisis auctor a quis lorem. Integer dolor urna, luctus quis
consequat vitae, vestibulum non lectus. In turpis mi,
tincidunt non iaculis ac, feugiat eget purus. Visão Animais
bem cuidados. Donos mais felizes. Lorem ipsum dolor sit
amet, consectetur adipiscing elit. Vestibulum nec tempus
justo, nec sagittis enim. Vestibulum eget augue dui.
Vestibulum consequat pretium lectus vitae varius. Etiam
facilisis quam et nulla posuere feugiat. Quisque eu augue
mi. Vivamus sagittis metus tortor, vitae vehicula mauris
interdum id. Maecenas nulla velit, posuere in pharetra eu,

Formatando textos

- É comum que nas suas páginas você necessite aplicar formatações e destacar alguma palavra ou sentenças inteiras
- Existem várias possibilidades, monte o exemplo a seguir:

Formatando textos

```
<!DOCTYPE html>
<html>
<body>
    <p><b>O texto está em negrito</b></p>
    <p><strong>Este é um texto utilizando Strong</strong></p>
    <p><em>Este é um texto utilizando Emphasized</em></p>
    <p><i>O texto está em itálico</i></p>
    <p><small>Este é um texto utilizando o Small</small></p>
    <p>Aqui foi utilizado o <sub> subscript</sub> e aqui o
        <sup>superscript</sup></p>
</body>
</html>
```

Formatando textos

- Os resultados desses elementos em um navegador seria:

O texto está em negrito

Este é um texto utilizando Strong

Este é um texto utilizando Emphasized

O texto está em itálico

Este é um texto utilizando o Small

Aqui foi utilizado o _{subscript} e aqui o ^{superscript}

Formatando textos

- Observe que as tags como `` e `<i>` são utilizadas como negrito ou itálico no texto
- A tag `` exibe o mesmo que a tag ``, e a tag `` o mesmo que a tag `<i>`
- Há uma diferença no significado dessas tags:
 - `` e `<i>` definem apenas o texto em negrito ou itálico
 - Já as tags `` e `` significa que você deseja que o texto seja exibido de forma que o usuário entenda como “importante”

- Sintaxe: conteúdo
 - Utilizada em parte de uma frase
 - Faz uma palavra ou um texto ser definido como mais importante em uma parte do site
1. No arquivo *quem-somos.html*, digite os elementos em seu programa conforme a seguir:


```
...  
<body>  
Quem somos  
<p>Em que nós acreditamos <br>  
Na <strong>Meu amigo cão</strong>, o nosso sucesso como uma  
organização é conduzido pelo cuidado que temos com nossos animais.</p>  
. . .  
</body>  
. . .
```

Outras formatações

- **Veja outras formatações:**

- <ins>: define o texto a ser inserido;
- : define o texto a ser excluído;
- <mark>: define uma marca de realce em um texto;
- <pre>: define um texto pré-formatado;
- <abbr>: define uma abreviatura ou um acrônimo;
- <address>: define as informações de contato do autor de um documento;
- <bdo>: define a direção de um texto;
- <blockquote>: define um recuo de uma seção.

Outras formatações

- Vamos montar a estrutura a seguir:

```
<!DOCTYPE html>
<html>
<body>
<p>Comer uma <del>pera</del> <ins>maçã</ins> diariamente é ótimo para
a saúde!</p>
<p>Não se esqueça de usar <mark>gravata</mark> na festa.</p>
<pre>
A exibição de texto utilizando o elemento PRE
mantém os      espaços
e as quebras de
linhas
</pre>
<p>Veja os requisitos de software para o treinamento:</p>
<blockquote>
Processador x86 ou x64 bits de 1 gigahertz (Ghz), ou mais rápido,
com conjunto de instruções SSE2, 1 gigabyte (GB) de RAM (32 bits);
2 gigabytes (GB) de RAM (64 bits). 3 gigabytes (GB) disponíveis
</blockquote>
</body>
```

Outras formatações

- O texto será exibido da seguinte maneira:

Comer uma pera maçã diariamente é ótimo para a saúde!

Não se esqueça de usar gravata na festa.

A exibição de texto utilizando o elemento PRE
mantém os espaços
e as quebras de
linhas

Veja os requisitos de software para o treinamento:

Processador x86 ou x64 bits de 1 gigahertz (Ghz), ou mais rápido, com conjunto de instruções SSE2, 1 gigabyte (GB) de RAM (32 bits); 2 gigabytes (GB) de RAM (64 bits). 3 gigabytes (GB) disponíveis

Definindo cabeçalhos

- <h1> até <h6>
- Sintaxe: <h1>conteúdo</h1>
- Sintaxe: <h2>conteúdo</h2>
- Sintaxe: <h3>conteúdo</h3>
- Sintaxe: <h4>conteúdo</h4>
- Sintaxe: <h5>conteúdo</h5>
- Sintaxe: <h6>conteúdo</h6>

Definindo cabeçalhos

- O HTML permite até seis tipos, sendo o primeiro quando deseja definir o maior nível de importância `<h1>`, e o último, o menor nível `<h6>`
- Isso faz com que os mecanismos de busca consigam identificar quais são os títulos mais importantes da publicação

Definindo cabeçalhos

- Vamos montar a seguinte estrutura:

```
<h1>Cabeçalho 1</h1>
```

```
<h2>Cabeçalho 2</h2>
```

```
<h3>Cabeçalho 3</h3>
```

```
<h4>Cabeçalho 4</h4>
```

```
<h5>Cabeçalho 5</h5>
```

```
<h6>Cabeçalho 6</h6>
```

Definindo cabeçalhos

- O que resultaria em:

Cabeçalho 1

Cabeçalho 2

Cabeçalho 3

Cabeçalho 4

Cabeçalho 5

Cabeçalho 6

Definindo cabeçalhos

1. No arquivo *quem-somos.html*, digite os elementos em seu programa conforme a seguir:
2. Salve o arquivo e verifique-o no seu navegador

```
...
<h1>Quem somos</h1>
.
.
.
<h2>Visão</h2>
.
.
.
<h2>Valores</h2>
...
.
```

Trabalhando com divisões de temas

<hr>

- Sintaxe: <hr>
- A tag cria uma linha horizontal em uma página HTML
- Pode ser usado para separar o conteúdo, uma vez que é comum existirem vários temas sendo abordados
- Essa tag não possui fechamento

Trabalhando com divisões de temas

<hr>

- No arquivo *quem-somos.html*, digite os elementos em seu programa conforme a seguir:
...
<hr>
<h1>Visão</h1>
•
•
•
<hr>
<h1>Valores</h1>
...
- Salve seu arquivo e verifique-o no seu navegador

Utilizando caracteres e símbolos especiais

- Também é possível utilizar letras gregas e símbolos, ou até mesmo os sinais de maior e menor (“>” e “<”)
- Eles são representados pelo símbolo “&” seguido de caracteres comuns, finalizados por “;”
- Vamos conhecer alguns exemplos:
 - (non-break space): permite a inserção de um espaço entre palavras

Utilizando caracteres e símbolos especiais

<	Menor que	<
>	Maior que	>
&	E comercial	&
¢	Centavos	¢
£	Libra	£
¥	Yen	¥
€	Euro	€
©	Copyright	©
®	Marca Registrada	®

Atividade 4 – Listas

- Objetivo:
 - Criar a sua página curiosidades
- Tarefas:
 - Abrir o arquivo existente
 - Trabalhar com listas ordenadas, não ordenadas e de definição
 - Salvar a sua página
 - Abrir a sua página do navegador

Trabalhando com listas

- Uma lista é uma sequência encadeada a qual contém mais de um elemento que você precisa exibir em sua página
- Existem 3 tipos:
 - Lista não ordenada
 - Lista ordenada
 - Lista de definição

*Lista não ordenada *

- Sintaxe: item
- Apresenta seus itens com marcadores

```
<ul>  
  <li>Item 1</li>  
  <li>Item 2</li>  
  <li>Item 3</li>  
</ul>
```



- Item 1
- Item 2
- Item 3

*Lista não ordenada *

- Também é possível usar uma lista não ordenada dentro de outra lista não ordenada

```
<ul>
  <li>Item 1</li>
  <li>Item 2</li>
    <ul>
      <li>Item 2.1</li>
      <li>Item 2.2</li>
        <ul>
          <li>Item 2.2.1</li>
          <li>Item 2.2.2</li>
        </ul>
    </ul>
  <li>Item 3</li>
</ul>
```



- Item 1
- Item 2
 - Item 2.1
 - Item 2.2
 - Item 2.2.1
 - Item 2.2.2
- Item 3

*Lista ordenada *

- **Sintaxe:** item
- A lista ordenada apresenta seus itens numerados:

```
<ol>  
<li>Item 1</li>                                1. Item 1  
<li>Item 2</li>                                2. Item 2  
<li>Item 3</li>                                3. Item 3  
</ol>
```



*Lista ordenada *

- Também é possível usar uma lista ordenada dentro de outra lista ordenada

```
<ol>
  <li>Item 1</li>
  <li>Item 2</li>
    <ol>
      <li>Item 2.1</li>
      <li>Item 2.2</li>
    </ol>
  <li>Item 3</li>
</ol>
```



1. Item 1
2. Item 2
1. Item 2.1
2. Item 2.2
3. Item 3

*Lista ordenada *

- Também é possível alterar a numeração de início de sua lista ordenada, para isso utilizamos o atributo **start**

```
<ol start="6">  
  <li>Item 1</li>          6. Item 1  
  <li>Item 2</li>          7. Item 2  
  <li>Item 3</li>          8. Item 3  
</ol>
```



*Lista ordenada *

- Também é possível reverter a numeração de sua lista, para isso utilizamos o atributo *reversed* (*é suportado nos principais navegadores, exceto no IE*)

```
<ol reversed>  
  
<li>Item 1</li>                                3. Item 1  
<li>Item 2</li>                                2. Item 2  
<li>Item 3</li>                                1. Item 3  
  
</ol>
```



*Lista ordenada *

- Também podem ter a descrição da numeração alterada. Para isso deve-se utilizar o atributo *type*
- *type = “val”*

Atributo	Tipo de numeração	
1	Números arábicos	1, 2, 3, ...
a	Letras minúsculas	a, b, c, ...
A	Letras maiúsculas	A, B, C, ...
i	Números romanos em minúsculas	i, ii, iii, ...
I	Números romanos em maiúsculas	I, II, III, ...

*Lista ordenada *

- Também podem ter a descrição da numeração alterada. Para isso deve-se utilizar o atributo *type*
- *type = “val”*

```
<ol type="I">  
  <li>Item 1</li>  
  <li>Item 2</li>  
  <li>Item 3</li>  
</ol>
```



I. Item 1

II. Item 2

III. Item 3

Lista de definição <dl>

- Sintaxe:
 - <dl> <dt> termo </dt> <dd> Descrição </dd> </dl>
- Esse tipo de lista não utiliza marcadores ou números
- Deve ser usado quando se deseja descrever um termo <dt> e suas descrições <dd>

Lista de definição <dl>

```
<dl>
  <dt>Termo 1</dt>                                Termo 1
    <dd>Descrição A</dd>                          Descrição A
    <dd>Descrição B</dd>                          Descrição B
  <dt>Termo 2 </dt>                                Termo 2
    <dd>Descrição C</dd>                          Descrição C
  <dt>Termo 3 </dt>                                Termo 3
    <dd>Descrição D</dd>                          Descrição D
    <dd>Descrição E</dd>                          Descrição E
</dl>
```



Combinando listas

- As listas poderão ainda ser integradas na sua página, em conjunto com diferentes tipos de listas
- Faremos o exemplo a seguir que é uma receita de omelete, que utiliza a lista de definição, contendo uma lista não ordenada (os ingredientes) e uma lista ordenada (o preparo)

Combinando listas

INGREDIENTES

- 2 ovos
- 1 fatia de presunto picado
- 2 fatias de queijo picado
- 1 pitada de sal
- Tomate cortado
- Caldo de galinha

PREPARO

1. Bata os dois ovos.
2. Coloque-os na frigideira já untada com óleo.
3. Coloque o sal, o presunto picado em quadradinhos e as duas fatias de queijo.
4. Coloque os temperos a gosto, espere ficar firme e vire o omelete.
5. Está pronto. Bom apetite!

Combinando listas

1. Abra com o seu editor de códigos o arquivo *curiosidades.html*
2. Digite os novos elementos conforme a seguir:
 - Criar uma lista ordenada para as 7 perguntas
 - Criar uma lista não ordenada para as raças
 - E na ultima parte criar uma lista de definição que vai desde *Bulldog* até *Lhasa Apso*

Combinando listas

Antes de mais nada, é importante que você tenha em mente a resposta para algumas questões:

1. Mora em casa ou apartamento?
2. Quer um cachorro para ser parceiro em atividades? Brincar? Terapia?
3. Você mora no campo ou na cidade?
4. Alguém da família é alérgico a cães?
5. Está disposto a cuidar do cão com frequência?
6. É rígido com limpeza?
7. Fica preocupado com bagunça?

Veja algumas raças:

- Bulldog
- Beagle
- Bull Terrier
- Collie
- Labrador
- Lhasa Apso



Combinando listas

Bulldog

Embora a aparência do bulldog inglês possa ser um pouco intimidante, ele está entre os mais gentis dos cães. Poucos arriscariam um encontro próximo com um cão corajoso o suficiente para enfrentar um touro. É descrito como um animal muito carinhoso e confiável, gentil com as crianças, mais conhecido por sua coragem e suas excelentes habilidades de guarda.

Beagle

O Beagle é amoroso, doce e gentil. Fica feliz de ver todos e cumprimenta com a cauda abanando. É sociável, corajoso e inteligente. O beagle é excelente com crianças e geralmente se dá bem com outros cães, mas, por causa de seus instintos de caça, não deve ser confiado com animais de estimação não caninos, a não ser que seja socializado com gatos e outros animais domésticos quando jovem.

Bull Terrier

Antigamente, essa raça era considerada um gladiador feroz, agora é vista como um cão dócil. O bull terrier pode defender seu dono em uma situação verdadeiramente crítica, mas não é criado para ser um cão de guarda. Corajoso, divertido, carinhoso, ativo, palhaço e destemido, o bull terrier é um cão educado, obediente e leal. Ele se torna muito ligado aos seus donos.

Collie

O collie é um cão muito inteligente. Sensível, meigo, dócil, fácil de treinar e leal, é geralmente bom com outros animais de estimação e amigável com outros cães. Eles são pastores naturais; filhotes podem ter o comportamento de pastoreio com os humanos e precisam ser educados a não fazer isso. Fiel, brincalhão, dócil, bom com crianças e protetor de seus familiares.

Labrador

Uma das raças mais populares nos EUA, o labrador é leal, amoroso, carinhoso e paciente, o que o torna um grande cão de família. Muito inteligente, bem-humorado e muito disposto a agradar, está entre as melhores opções para ser um cão de serviço a deficientes. Labradores gostam de brincar, especialmente com água, nunca perdem a oportunidade de dar um bom mergulho. Esses cães têm um excelente temperamento e são confiáveis, amigáveis e excelentes com crianças e outros cães.

Lhasa Apso

Esse é um cão resistente, amigável e assertivo. Inteligente e animado, é um bom animal de estimação. Lhasa apsos são pequenos, espírituosos, afetuosa e obedientes a seus donos. Essa raça responde ao treinamento motivacional. Eles têm uma ótima audição, são bons cães de guarda e acostumam-se a viajar. Infelizmente, a raça tende a apresentar a Síndrome do Cachorro Pequeno, um comportamento negativo em que o cão pensa que é o líder da casa. Tomam-se desconfiados com estranhos e podem não tolerar crianças.

Atividade 5 – Imagens

- Objetivo:
 - Modificar a página “Quem somos” e “Curiosidades”
- Tarefas:
 - Abrir o arquivo existente
 - Inserir a imagem
 - Salvar a sua página
 - Abrir a sua página do navegador

Tipos de imagens

- **JPG:** é a extensão mais utilizada, obtido nas fotografias e celulares e câmeras digitais. A maior vantagem é a possibilidade de compressão, permitindo um rápido carregamento de uma página web
- **GIF:** extensão utilizada para criar imagens animadas
- **PNG:** Também traz alta compressão de arquivos sem perder a qualidade. O W3C recomenda esse tipo de arquivo, indicado como formato ideal para internet
- **SVG:** formato de arquivo baseado em texto e descreve as imagens com formatos vetoriais. É ideal para dispositivos móveis

Inserindo imagens

- São definidas utilizando a tag
- Sintaxe:
- *src* é o atributo onde é colocado o endereço exato onde a imagem encontra-se armazenada
- Ex.:

Inserindo legenda nas imagens

- <figure> e <figcaption>
- O elemento <figure> tem a finalidade de marcar fotos e ilustrações
- O elemento <figcaption> cria uma descrição (legenda) dentro de uma ou mais imagens
- Esse elementos são usado em conjunto

Inserindo legenda nas imagens

- Sintaxe:

```
<figure>  
    
    
    
  <figcaption> descrição</figcaption>  
</figure>
```

Inserindo legenda nas imagens

1. Abra o arquivo *quem-somos.html* na pasta do seu projeto
2. Insira a imagem *meuamigocao.jpg* com a descrição “Meu amigo cão – Pet shop” dentro do elemento `<h1>Quem somos</h1>`
3. Salve e abra-o no navegador

Inserindo legenda nas imagens

1. Abra o arquivo *curiosidades.html* na pasta do seu projeto
2. Insira as figuras *bolinha.jpg* e *esquentar-pes.jpg* com as descrições “brincando com a bolinha” e “esquentando os pés” respectivamente e adicione a legenda *Para brincar ou esquentar os seus pés*
3. Salve e abra-o no navegador

Atividade 6 – Tabelas e novos elementos

- Objetivo:
 - Criar a página “Produtos”
- Tarefas:
 - Abrir o arquivo existente
 - Trabalhar com tabela
 - Salvar a sua página
 - Abrir a sua página do navegador

Utilizando tabelas

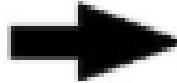
- Dentro das tabelas pode-se utilizar número, textos, figuras, etc.
- A elaboração de uma tabela consiste em uma combinação de linhas e colunas, resultando em uma forma objetiva e ordenada de expor informações

Utilizando tabelas

- Sintaxe: <table>...</table>
- As tabelas são definidas com a tag <table> que necessita de outros elementos para a construção de uma tabela
- <th>, <tr> e <td>
- O elemento <th> significa o início de um cabeçalho
- O elemento <tr> define o início de uma linha da tabela
- A cada linha <tr>, uma ou mais células são acrescentadas <td>. Você deve acrescentar a mesma quantidade de células com base no cabeçalho criado

Utilizando tabelas

```
<table>  
  <tr>  
    <th>Livro</th>  
    <th>Quantidade</th>  
  </tr>  
  
  <tr>  
    <td>Excel 2013</td>  
    <td>200</td>  
  </tr>  
  
  <tr>  
    <td>HTML5 / CSS3</td>  
    <td>180</td>  
  </tr>  
  
</table>
```



Livro	Quantidade
Excel 2013	200
HTML5 / CSS3	180

Definindo bordas

- A tag <table> possui vários atributos que não são mais suportados pelo HTML5, mas que ainda funcionam nos navegadores
- Sintaxe: <table border="1">...</table>
- Adiante aprenderemos a utilizar o CSS em conjunto com a tag <table>

Definindo bordas

```
<table border="1">  
  
<tr>  
  <th>Livro</th>  
  <th>Quantidade</th>  
</tr>  
  
<tr>  
  <td>Excel 2013</td>  
  <td>200</td>  
</tr>  
  
<tr>  
  <td>HTML5 / CSS3</td>  
  <td>180</td>  
</tr>  
</table>
```



Livro	Quantidade
Excel 2013	200
HTML5 / CSS3	180

Definindo bordas

Livros	Quantidade
Excel 2013	200
HTML5 / CSS3	180
PowerPoint 2013	100
AutoCAD 2015	120

Definindo bordas

- Você também pode trabalhar com os cabeçalhos na vertical
- Para isso terá que combinar os cabeçalhos <th> com as células <td> na mesma linha <tr>
- Desafio:

Livro	Excel 2013	HTML5 / CSS3
Quantidade	200	180

Criando legendas para tableas

<caption>

- **Sintaxe:** <caption> Legenda </caption>
- Esta tag define uma legenda da tabela
- Deve ser inserida imediatamente após a tag <table>
- Por padrão, a legenda estará alinhada ao centro da tabela, porém as propriedades CSS podem ser usadas para modificar esse alinhamento

Editora Senac		
Livro	Excel 2013	HTML5 / CSS3
Quantidade	200	180

Mesclado células na tabela

- Ao construir uma tabela, é necessário que faça uma mescla ou expanda o tamanho da célula para que o conteúdo faça referência a uma quantidade maior de linhas ou colunas
- <colspan> e <rowspan>
- Seu uso é recomendado para títulos que englobem várias colunas ou linhas
- Devem ser utilizada nas tags <th> e <td>

Mesclado células na tabela

- Desafios:

Célula expandida em várias colunas	
Exemplo de Colspan	50
Fevereiro	10

Mês	Valores	Exemplo de Rowspan
Janeiro	30	
Fevereiro	15	Célula expandida em várias linhas

Mesclado células na tabela

1. Abra o arquivo *produtos.html* na pasta do seu projeto
2. Crie a tabela conforme a seguir:

Mesclado células na tabela

Produtos			
Porte	Marca		Preço
Pequeno	Vestibulum		R\$ 5,00
	Nunc		R\$ 4,80
Médio	Mollis		R\$ 6,90
	Portitor		R\$ 3,90
Grande	Lacinia		R\$ 6,90
	Sapiens		R\$ 3,90

Consulte nossos preços para grandes quantidades.

Novos elementos da tabela

- Com o HTML5 novas tags foram criadas para facilitar o uso da semântica e da formatação nas tabelas
- <thead>: esse elemento é o cabeçalho da tabela
- <tbody>: esse elemento define o corpo da tabela, ou seja, todas as linhas <tr> e os seus dados <td>
- <tfoot>: é usado para mostrar o conteúdo do rodapé de uma tabela

Novos elementos da tabela

- Desafio:

TABELA THEAD/TFOOT/TBODY			
Coluna 1	Coluna 2	Coluna 3	Coluna 4
conteúdo a	conteúdo b	conteúdo c	conteúdo d
conteúdo e	conteúdo f	conteúdo g	conteúdo h
conteúdo i	conteúdo j	conteúdo k	conteúdo l
resultado 1	resultado 2	resultado 3	resultado 4

Atividade 7 – Página “Index” e links de navegação

- Objetivo:
 - Criar a página “Menu” e inserir os links nas demais páginas
- Tarefas:
 - Abrir o arquivo existente
 - Inserir links
 - Salvar a sua página
 - Abrir a sua página do navegador

Criando a página “Index”

- Em qualquer projeto de site existe um conjunto de páginas
- Deveremos criar os links que darão acesso às demais páginas

Inserindo links

- Um dos grandes recursos de um site é a criação de hiperlinks
- Um local na página que permite acessar outro ponto dentro da mesma página (link interno), ou outra página (link externo)
- Seria complicado digitar todo o endereço para ir de uma página a outra

Inserindo links

- Um link pode ser:
 - uma palavra
 - uma sentença
 - uma opção de lista
 - uma imagem
- O link em sua forma padrão aparece sublinhado e com uma coloração azul

Inserindo links

- **Sintaxe:** ` texto explicativo `
- O elemento `<a>` (âncora) é utilizado para criar um link, e deve utilizar o atributo `href` para fazer referência ao hipertexto

Inserindo links

Para abrir uma página do seu projeto	<code>Clique para ir à página de produtos</code>
Para abrir uma página em uma pasta do seu projeto. É necessário informar o caminho	<code> Clique para ir à página de produtos</code>
Para abrir uma página de outro site Nesse caso, é imprescindível usar a referência completa do protocolo para o endereço do site	<code> Portal do Senac São Paulo</code>

Inserindo links

1. Abra o arquivo *index.html* na pasta *projeto*
2. Crie os links para as páginas:
 - *index.html*
 - *quem-somos.html*
 - *produtos.html*
 - *curiosidade.html*
- Faça os teste navegando entre as páginas e voltando para *index.html*

Inserindo links para voltar à página “Index”

- Deverá inserir links nas páginas criadas para que tenham a opção de retornar à página principal do seu site
- <p> Home </p>

Links absolutos e relativos

- O endereço relativo é a informação para a localização da página desejada a partir da localização da página atual, ou seja, estão na mesma página
- Em alguns casos, optamos por trabalhar com algumas páginas em pastas diferentes no projeto, nesse caso, há a necessidade de especificar o endereço completo para o acesso

Links absolutos e relativos

- Esse tipo de endereço é chamado de absoluto, nada mais é que a informação para a localização de outra página, independentemente do local da página atual (protocolo + domínio + caminho)
- Exemplo:
 - <http://www.wyzw.com.br/pagina.html>
 - <http://www.wyzw.com.br/livros/pagina.html>

Criando um link para a navegação na mesma página

- Em alguns casos, existem muitas informações na página, o que torna necessária a utilização da barra de rolagem no navegador
- Para facilitar o HTML permite que você utilize o elemento `<a>` associado a um nome em um determinado local da sua página

Criando um link para a navegação na mesma página

1. Abra o arquivo *curiosidades.html* na pasta do projeto
2. Digite os novos elemento conforme a seguir:

...

...

```
<figcaption>Para brincar ou esquentar os seus pés</figcaption>
</figure>
```

```
<a id="top"> </a> <h3>Veja algumas raças:</h3>
```

```
<ul>
```

...

Criando um link para a navegação na mesma página

```
...  
<dt>Bulldog</dt>  
  
<dd>  
Embora a aparência do bulldog inglês possa ser um pouco intimidante,  
ele está entre os mais gentis dos cães. .....habilidades de guarda.  
<br>  
<p> <a href="#top">Voltar</a> </p>  
</dd>
```

- ```
...
• Assim, estaremos criando um link de referência à
âncora top, nesse caso deve-se utilizar o símbolo
“#” à frente do nome do ponto de identificação
```

# Criando um link para a navegação na mesma página

- Insira os demais links nas outras raças de cães
- Salve o arquivo e abra-o no navegador e teste os links com a palavra voltar abaixo das descrições de cada cachorro

# Criando um link para a navegação na mesma página

- Faça o mesmo agora criando identificadores para cada raça conforme ao lado:

```
...
<dt> Bulldog</dt>
.
.
.
<dt> Beagle</dt>
.
.
.
<dt> Bull Terrier</dt>
.
.
.
<dt> Collie</dt>
.
.
.
<dt> Labrador</dt>
```

# Criando um link para a navegação na mesma página

- Com isso, criamos identificação para cada cão
- Agora crie os novos elementos para fazer referência para cada identificação feita para os cães

```
 <h3>Veja algumas raças:</h3>

 Bulldog
 Beagle
 Bull Terrier
 Collie
 Labrador
 Lhasa Apso


```

# Criando um link externo

- Até o momento, utilizamos links relativos
- Seria interessante utilizar links absolutos, que permitissem visitar um outro site com uma maior descrição sobre o referido cachorro

# Criando um link externo

## 1. Digite os novos elementos conforme a seguir:

```
<dt> Bulldog</dt>
.
.
.....excelentes habilidades de guarda.

Tenha mais
informações

<p> Voltar </p>
</dd>
```

## 2. Faça o mesmo com os demais cães

# Abrindo o link em nova janela

- Também podemos usar o atributo *target* para especificar se, ao abrir um link, ele será aberto na mesma janela do navegador ou em uma nova
- Digite os novos elementos conforme a seguir:

```

Tenha mais informações
```

- Clique no link e observe que será aberto em uma nova janela

# Inserindo um link de e-mail

- O HTML permite, por meio do uso de um link, o acesso ao programa padrão do correio eletrônico, trazendo o campo destinatário já preenchido
- **Sintaxe:** <a href="mailto:endereco@email.com">texto</a>
- Faça a alteração no e-mail contido na página referenciando para o e-mail atendimento@meuamigocaopetshop.com

# Criando um link de download

- Também pode-se utilizar um link externo para que, em vez de seu arquivo ser visitado, ele possa ser baixado diretamente no seu computador
- Para isso, deve-se utilizar o atributo *download*
- Esse atributo indica o nome que será dado ao arquivo a ser baixado

# Criando um link de download

- Sintaxe:

```
mensagem
```

- Veja o exemplo:

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<p>
<a href="files/arquivo.pdf"
download="relatorio.pdf">
Clique para fazer o download do arquivo
</p>
</body>
</html>
```



Clique para fazer o download do arquivo

# Atividade 8 – Formulários

- Objetivo:
  - Criar formulário no seu site: Boletim e Agendamento para Banho e Tosa
- Tarefas:
  - Abrir o arquivo existente
  - Inserir links
  - Inserir formulários
  - Salvar a sua página
  - Abrir a sua página do navegador

# Atividade 8 – Formulários

- É muito importante para as empresas coletar dados para a execução de serviços e obter informações dos seus clientes
- Em uma página, isso é possível com o uso dos formulários
- É composto de rótulos (descritivos do que será informado) e campos (local onde serão digitadas ou selecionadas as informações descritas nas legendas)

# Conhecendo as primeiras tags e os atributos para formulários

- Formulários HTML são usados para transmitir dados a um servidor
- A tag <form> é usada para criar um formulário HTML
- Sintaxe: <form> Elementos de entrada </form>
- Um formulário HTML pode conter elementos de entrada, como campos de texto, caixas de seleção, botões de rádio, etc.

# Conhecendo as primeiras tags e os atributos para formulários

- <label>: é um elemento de entrada que define um rótulo para um elemento que será digitado
- Sintaxe:

```
<label for="nome do rótulo">Descrição
do rótulo</label>
```

# Conhecendo as primeiras tags e os atributos para formulários

- <input>: é um elemento de entrada que especifica um campo de entrada onde o usuário pode inserir dados
- Deve ser utilizado junto a uma tag <label>
- O elemento <input> é o mais importante de um formulário, pois todas as informações do usuário virão por meio dele
- Esse elemento pode variar e ter várias formas

# Conhecendo as primeiras tags e os atributos para formulários

- Dependendo do atributo, pode ser dos tipos:
  - Campo de texto
  - Caixa de verificação
  - Senha
  - Botão rádio
  - Botão enviar
  - Etc.
- Veremos os mais importantes no projeto

# Atributo *type*

- Esse atributo especifica o tipo de elemento
  - *checkbox*: permite ao usuário um número limitado de opções em caixas de seleção
  - *email*
  - *password*: permite a entrada de caracteres mascarados, como uma senha
  - *radio*: permite selecionar um item em uma lista com várias escolhas

# Atributo *type*

- Esse atributo especifica o tipo de elemento
  - *submit*: essa opção apresenta um botão de envio, que em conjunto com uma linguagem de programação é usado para enviar dados de um formulário para um servidor
  - *text*: define a entrada de dados no formato texto

# Atributo *name*

- É utilizado para referenciar o elemento no formulário via programação

# Criando um formulário básico

- O pet shop organiza um boletim semanal contendo informações, dicas e outras notícias a respeito dos animais
- Deveremos colher o nome e e-mail para o envio desse boletim digital

# Criando um formulário básico

1. Abra o arquivo *boletim.html* na pasta do projeto
2. Digite os novos elementos conforme a seguir:
  - Crie um formulário
  - Adicione dentro de um parágrafo uma *label* e um *input* para receber o nome do usuário

# Criando um formulário básico

- Para o tipo *text*, você pode utilizar outros atributos:
  - *id*: serve para identificar o elemento e possibilitar o uso da tag `<label>`
  - *placeholder*: especifica uma dica que descreve o valor esperado de um campo de entrada, uma orientação do que deve ser digitado
  - *size*: especifica a largura visível, em caracteres, do elemento. É útil quando utilizado em conjunto com o atributo *placeholder* para grandes descrições

# Criando um formulário básico

3. Digite os novos elementos conforme a seguir:
  - adicione um *id* para o nome e coloque uma dica no elemento *input*
4. Digite os novos elementos conforme a seguir:
  - Crie agora os mesmos itens só que para o e-mail, com uma largura de 30
  - Também crie um botão (*submit*) que simule o envio do formulário

# Criando um formulário básico

**Meu amigo cão – Pet shop**

**Boletim mensal**

**Assine já!**

Fazendo sua assinatura  
você receberá mensalmente informações sobre nossos produtos, serviços, eventos e descontos.

Nome:

Email:

# Enviando dados por meio de um formulário

- **Sintaxe:** <form method="tipo" id="nome" action="nome da página">
- Atributo *method*: especifica como será o envio dos dados do seu formulário
- Existem 2 métodos utilizados na linguagem HTML:
  - GET (por padrão é geralmente utilizado)
  - POST

# Enviando dados por meio de um formulário

- Quando se digita uma URL, o navegador faz uma requisição HTTP para o servidor da URL digitada e o método de requisição é o GET
- Os dados serão enviados por meio da URL, separados por um sinal de interrogação
- O método GET não altera o estado do servidor

# Enviando dados por meio de um formulário

- O Google utiliza método semelhante em suas pesquisas
- Ao digitar uma frase em seu navegador, por exemplo, “cursos no Senac”, o motor de pesquisa apresentará na URL a seguinte descrição

[https://www.google.com/search?hl=en&q=cursos+no+senac&gws\\_rd=ssl](https://www.google.com/search?hl=en&q=cursos+no+senac&gws_rd=ssl)

# Enviando dados por meio de um formulário

- O que se deve ter em mente ao utilizar o método GET:
  - Ele exibirá os dados do seu formulário na URL
  - Lembre-se de que a URL tem um comprimento limitado (cerca de 3.000 caracteres)
  - Em caso de senha ou dados que necessitem sigilo, não utilizar GET, pois eles serão visíveis na URL
  - O envio de dados é mais rápido
  - Só envia textos

# Enviando dados por meio de um formulário

- Já o método POST envia os dados encapsulando-os com a requisição HTTP, de forma que não podem ser vistos:
- Ao utilizar o método POST:
  - Não terá limitações de comprimento da mensagem, pois ela é enviada no corpo da requisição HTTP
  - O envio é mais demorado
  - Envia textos e dados binários
  - Por não ser padrão, deve ser indicado no atributo *method*

# Enviando dados por meio de um formulário

- Atributo *action*: especifica para onde os dados do formulário serão enviados
- Existem duas possibilidades de envio:
  - URL absoluta: indica, por exemplo, outro site:
    - `action="http://www.pr.senac.br/eventos.html"`
  - URL relativa: indica um arquivo dentro de um website
    - `action="eventos.html"`

# Enviando dados por meio de um formulário

1. Digite os novos elementos conforme a seguir:
  - Adicione o *method* post, um identificador com o nome de boletim e direcione para a página *obrigado.html*
  - Essa página simula o envio dos seus dados a um servidor
  - Crie um link no arquivo *index.html* com o texto “Assine o boletim mensal” que direcione para página *boletim.html*

# Enviando dados por meio de um formulário

## Meu amigo cão – Pet shop

Assine o boletim mensal

- [Home](#)
- [Quem somos](#)
- [Produtos](#)
- [Banho e Tosa](#)
- [Curiosidades](#)

Atendimento: [atendimento@meuamigocaopetshop.com](mailto:atendimento@meuamigocaopetshop.com) - (0XX) 99999-9999 / (0XX) 88888-8888

# Criando um formulário complexo

- Iremos criar agora a página *banho e tosa*
- Nessa página o cliente informará seus dados para que a empresa retire e entregue o animal, as informações básicas sobre o animal e o horário de preferência para o procedimento

# Criando um formulário complexo

1. Abra o editor de códigos o arquivo *banho-e-tosa.html*
2. Crie o formulário conforme imagem a seguir:

# Criando um formulário complexo

[Home](#)

## Agendamento para banho e tosa

**Faça o seu agendamento com pelo menos 48 horas de antecedência**

Nome:

Endereço:

CEP:

Email:

Tel/Cel:

Nome:

# Definindo o tamanho máximo de caracteres

- O atributo *maxlength* especifica o número máximo de caracteres a serem digitados no elemento <input>
- O atributo *size* determina o tamanho da caixa de texto, porém, não limita os caracteres digitados
- Vamos criar o arquivo *banho-e-tosa.html* conforme imagem a seguir:

# Definindo o tamanho máximo de caracteres

---

[Home](#)

## Agendamento para banho e tosa

**Faça o seu agendamento com pelo menos 48 horas de antecedência**

Nome:

Endereço:

CEP:

Email:

Tel/Cel:

Nome:

# Definindo o tamanho máximo de caracteres

1. No arquivo *banho-e-tosa.html*, digite os novos elementos conforme a seguir:
  - Nome do dono: máximo de 100 caracteres
  - Endereço do animal: máximo de 60 caracteres
  - CEP: máximo de 9 caracteres
  - Telefone do dono: máximo de 13 caracteres
  - Nome do animal: máximo de 40 caracteres
2. Salve o arquivo e faça os testes necessários

# Agrupando elementos em um formulário

- A tag <fieldset> é utilizada para agrupar elementos relacionados em um formulário
  - Essa tag desenha uma caixa em torno do agrupamento
1. Adicione o elemento de agrupamento com o identificador “ident\_dono” aos dados relacionados do dono do cão

# Agrupando elementos em um formulário

[Home](#)

## Agendamento para banho e tosa

Faça o seu agendamento com pelo menos 48 horas de antecedência

Nome:

Endereço:

CEP:

Email:

Tel/Cel:

Nome:

# Definindo uma legenda para os agrupamentos

- Uma alternativa interessante é que, em cada agrupamento, você pode indicar qual assunto representa esse conjunto por meio de uma descrição (legenda)
  - A tag <legend> define uma legenda para o elemento *fieldset*
1. Adicione a legenda “Identificação do dono do animal” ao elemento *fieldset* criado anteriormente no arquivo *banho-e-tosa.html*

# Definindo uma legenda para os agrupamentos

Home

## Agendamento para banho e tosa

**Faça o seu agendamento com pelo menos 48 horas de antecedência**

Identificação do dono do animal

Nome:

Endereço:

CEP:

Email:

Tel/Cel:

Nome:

# Trabalhando com botões de rádio

- São utilizadas para a seleção de uma opção já predefinida
- São apresentados como pequenos círculos à frente de legendas que você mesmo define
- Pode-se ter várias opções, mas só pode escolher uma
- Caso escolha outra opção, o comando automaticamente desmarcará a anterior

# Trabalhando com botões de rádio

- **Sintaxe:** <input type="radio" name="nome" id="identificação" value="valor">Descrição
1. No arquivo *banho-e-tosa.html*, digite os novos elementos conforme a seguir:
    - Adicione um agrupamento com identificador “ident\_animal” e legenda “Animal”
    - Vamos adicionar um input do tipo radio dentro do parágrafo Porte do cão conforme imagem a seguir:

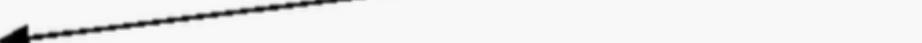
# Trabalhando com botões de rádio

- Atributo *nome*: `porte_animal`
- Atributos *identificador* e *valor*  
`animal_pequeno`,  
`animal_medio`,  
`animal_grande`

Animal

Nome:

Porte do cão:  pequeno  médio  grande



# Definindo uma opção rádio já marcada

- É possível iniciar uma página com a opção de rádio mais utilizada predefinida
- Isso facilita o cadastramento e permite mudança apenas em caso de outra opção ser escolhida
- Vamos supor que no pet shop a maior parte dos cachorros seja pequena
- O atributo *checked* faz com que o elemento já venha marcado

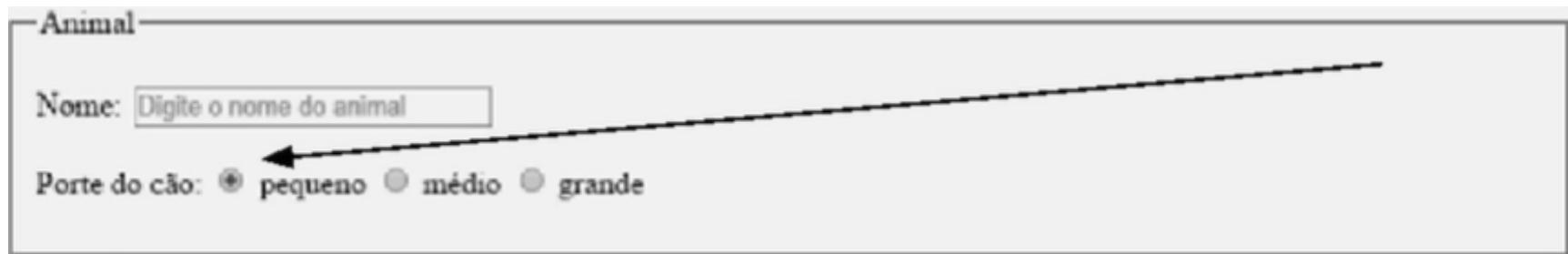
# Definindo uma opção rádio já marcada

- Adicione o atributo *checked* ao final do elemento input do tipo radio do cão de porte pequeno

Animal \_\_\_\_\_

Nome:

Porte do cão:  pequeno  médio  grande



# Criando uma lista drop-down

- Também conhecida como menu pop-up ou lista suspensa
- A lista drop-down é semelhante a uma caixa de listagem na qual você poderá escolher um valor

```
<select name="nome" id=" identificação" >
 <option>opção 1</option>
 <option>opção 2</option>
 <option>opção n</option>
</select>
```

# Criando uma lista drop-down

- A tag `<select>` é utilizada para criar a lista e a tag `<option>` serve para indicar as opções dentro da lista
1. No arquivo *banho-e-tosa.html*, digite os novos elementos abaixo:
    - Adicione dentro do elemento parágrafo com texto Tipo de tosa: uma lista dowp-down com os textos:
      - Nenhuma, Tosa Padrão, Tosa Higiênica, Tosa na Tesoura, Tosa da Raça, Tosa Bebê
      - Para os atributos valores das opções, mantenha para nenhuma em branco (""), padrao, higienica, tesoura, raca, bebe

# Criando uma lista drop-down

Animal

Nome:

Porte do cão:  pequeno  médio  grande

Tipo de tosa:  

- Nenhuma
- Tosa Padrão
- Tosa Higiênica
- Tosa na Tesoura
- Tosa da Raça
- Tosa Bebê

# Definindo um campo para a digitação de texto livre

- A tag <textarea> define uma entrada de texto em várias linhas
- Em alguns casos, uma entrada de dados necessita de um número ilimitado de caracteres
- Essa tag pode ser criada utilizando atributos como <cols> e <rows> que determinam o número de colunas (largura) e o de linhas (altura)
- Porém, com uso do CSS poderá ser definido melhor a altura e largura mais adiante

# Definindo um campo para a digitação de texto livre

- Sintaxe:  

```
<textarea name="nome"
id="identificação" placeholder="Descrição
dentro da caixa de texto"></textarea>
```
- Adicione dentro o elemento parágrafo com texto Observações:
  - Crie um textarea com nome `q_obs`, identificador `q_obs` e coloque uma mensagem dentro da caixa de texto “Digite alguma informação sobre o animal se necessário”

# Definindo um campo para a digitação de texto livre

Animal

Nome:

Porte do cão:  pequeno  médio  grande

Tipo de tosa:

Observações:

# Criando caixas de seleção

- O atributo `<type="checkbox">` define uma caixa de seleção
- Permite que o usuário selecione um ou mais itens de um número limitado de opções
- Sintaxe: `<input type="checkbox" name="nome" value="valor">` opção
- Criaremos um novo agrupamento que estarão escolhas relativas ao animal, onde o usuário poderá escolher mais de um dia da semana para agendar banho e tosa

# Criando caixas de seleção

1. No arquivo *banho-e-tosa.html*, digite os novos elementos abaixo:
  - Crie um novo agrupamento com legenda Horário e identificador `ident_marcação`
  - Dentro do parágrafo com texto Indique os dias... crie caixas de seleção para os dias da semana: Terça-feira, Quarta-feira, Quinta-feira, Sexta-feira e Sábado, coloque o atributo nome “semana” e o atributo valor conforme o dia da semana, exemplo: “idterca”, “idquarta” e etc.

# Criando caixas de seleção

**Animal**

Nome:

Porte do cão:  pequeno  médio  grande

Tipo de tosa:

Observações:

Horário - (Não atendemos aos domingos e às segundas-feiras)

Indique os dias da semana de sua preferência:

Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira  Sábado

# Utilizando opções de tipo data

- Como a tag <input> possui vários tipos de entrada, permite um melhor controle de entrada e validação de dados
- Em alguns casos, é necessário que os dados sejam do tipo data, ou seja, dia, mês e ano
- Com isso, evita que o usuário insira outra informação, atrapalhando o controle desejado

# Utilizando opções de tipo data

- A tag <input> possui o atributo *type="date"* que é utilizada para campos com entrada de uma data
- Sintaxe: <input type="date" name="nome" id="identificação">
- Criaremos duas entradas do tipo data para o agendamento de banho e tosa

# Utilizando opções de tipo data

1. No arquivo *banho-e-tosa.html*, digite os novos elementos conforme a seguir:
  - Dentro o parágrafo Primeira opção de horário crie um input do tipo data com nome e identificador “*Pdata*”
  - Dentro do parágrafo Segunda opção de horário crie um input do tipo data com nome e identificador “*Sdata*”

# Utilizando opções de tipo data

**Animal**

Nome:

Porte do cão:  pequeno  médio  grande

Tipo de tosa:

Observações:

Horário - (Não atendemos aos domingos e às segundas-feiras)

Indique os dias da semana de sua preferência:

Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira  Sábado

Primeira opção de horário:

Segunda opção de horário:

# Exercício 1

- A partir dos itens aprendidos, crie uma página seguindo as instruções:
  - Substitua os itens entre os colchetes por dados que achar convenientes
  - Utilize, se necessário, dados de *Lorem ipsum*
  - Trabalhe, quando possível, com cabeçalhos, divisão de temas e listas
  - Salve a página com nome de *curriculo.html*

# Exercício 1

- [Endereços]
- [Cidade, Estado e Cep]
- [Telefone]
- [Email]

## [Seu nome]

### OBJETIVO:

Trabalhar na sua empresa consultada. Libere algumas ações avançadas para que o seu negócio funcione.

### HABILIDADES & COMPETÊNCIAS

Flexibilidade - Alguém só pode trabalhar algumas horas, não precisa desistir deles.  
Capacidade para resolver problemas - Adaptação constante em diferentes situações, resolvendo elas de forma eficiente.  
Trabalho em equipe - Um lar é um exemplo de grandeza, junto autor de adaptação.  
Envolvimento - Não só se envolve com a família, mas também com amigos e vizinhos.  
Positivo - Trabalha para superar todos os seus objetivos, mesmo os mais difíceis.  
Aprendizado qualificado - Muito importante para o seu sucesso, sempre buscando novas maneiras.  
Comunicativo - Sabe falar com pessoas de diferentes culturas e idiomas.

### EXPERIÊNCIA

- [CARGO: NOME DA EMPRESA] [Data De - Até]
- [CARGO: NOME DA EMPRESA] [Data De - Até]
- [CARGO: NOME DA EMPRESA] [Data De - Até]
- [CARGO: NOME DA EMPRESA] [Data De - Até]

### EDUCAÇÃO

- [NOME DA INSTITUIÇÃO—LOCAL—GRADUAÇÃO]
- [NOME DA INSTITUIÇÃO—LOCAL—GRADUAÇÃO]
- [NOME DA INSTITUIÇÃO—LOCAL—GRADUAÇÃO]

### COMUNICAÇÃO

Autônomo liberto falar com alguém através de e-mail, telefone, fax ou correio eletrônico, através de outras formas.

### REFERÊNCIAS

1. [NOME DE REFERÊNCIA, Cargo, Empresa, contato]
2. [NOME DE REFERÊNCIA, Cargo, Empresa, contato]
3. [NOME DE REFERÊNCIA, Cargo, Empresa, contato]
4. [NOME DE REFERÊNCIA, Cargo, Empresa, contato]

# Exercício 2

- Crie uma página com um mural para cães desaparecidos ou encontrados
- Utilize cabeçalhos e imagens
- Adicione  *Lorem ipsum*, se necessário

# Exercício 2

Morai - Encante-se aqui com desaparecidos:

REAGOL: desaparecido de casa há dois dias. Peço favor a todos, o preceito da solidariedade continue para o cãozinho. Desapareceu por volta das 10h30 da noite de ontem (sábado), atende pelo nome de Toly. Encontre em: [www.desaparecidos.com.br](http://www.desaparecidos.com.br) - 0800-1000.



BILL TERRIER: resgatado na região da Lapa, em São Paulo, está em ótimo estado. Obrigado a todos que o resgataram Toly - 0800-1000.



COLAR: perdido no dia 20 na rua Júlio Ribeiro, bairro Ipiranga, São Paulo. Obrigado a todos que o resgataram Toly - 0800-1000.



Atendimento: [atendimento@desaparecidos.com.br](mailto:atendimento@desaparecidos.com.br) - (030) 99999-9999 / (030) 88888-8888

# Exercício 3

- Crie uma página com a grade de cursos oferecidos no Senac
- Utilize cabeçalhos e tabela com a primeira linha em negrito

# Exercício 3

## Cursos Senac - Informática para Escritório

Cursos *****	Carga Horária ****	Idade Mínima*****	Escolaridade *****	Inicio *****	Valor *****
Excel Básico	24 horas	14 anos	6º ano do ensino fundamental	Fevereiro, março e abril	R\$ 390,00
Excel Avançado	39 horas	16 anos	Cursando o ensino médio	Fevereiro, março e abril	R\$ 723,00
Informática para Maturidade	36 horas	Acima de 40 anos	6º ano do ensino fundamental	Fevereiro e maio	R\$ 622,00
Operador de Computador	160 horas	16 anos	Ensino fundamental 2 incompleto	Março e outubro	R\$ 1.757,00
PowerPoint	20 horas	14 anos	Ensino fundamental 2 incompleto	Março e junho	R\$ 216,00
Project	20 horas	16 anos	Ensino fundamental completo	Fevereiro e abril	R\$ 570,00
Word	24 horas	14 anos	Ensino fundamental 2 incompleto	Março e junho	R\$ 250,00

# Exercício 4

- Crie um formulário contendo os seguintes dados:
  - Título: informações pessoais
  - Campos: Nome, Idade, Endereço, Cidade, Estado, CEP, Telefone Res., Celular e E-mail
  - Salve com o nome de *Cadastro*

# Exercício 4

## Informações Pessoais

Cadastro do Cliente

Nome:

Idade:

Endereço:

Cidade:

Estado:

CEP:

Tel/Res:

Tel/Cel:

Email:

# Exercício 5

- Crie uma página “Index” que tenha link para acesso às páginas criadas:
  - Menu Exercícios:
    - Currículo
    - Mural de cães
    - Cursos
    - Cadastro

# Resumo do capítulo

- Crie uma tabela que contenha os tópicos estudados neste capítulo
- A tabela irá possuir 2 colunas:
  - *Item* e *Procedimento*
- Os itens serão:
  - Criar o corpo básico; Inserir uma quebra de linha; Inserir uma quebra de parágrafo; Definir cabeçalhos; Trabalhar com divisão de temas; Utilizando caracteres especiais; Listas não ordenadas; Listas ordenadas; Listas ordenadas com atributo *start*; Listas ordenadas com atributo *reversed*; Listas ordenadas com atributo *type*; Lista de definição;

# Resumo do capítulo

- Os itens serão:
  - Combinando listas; Inserindo imagens; Inserindo legenda nas imagens; Inserindo tabelas; Criando legenda para tabelas; Mesclando células na tabela; Inserindo links; Inserindo um link de e-mail; Criando um link de download; Formulário; Enviando dados por meio de um formulário; Definindo o tamanho máximo de caracteres; Agrupando elementos em um formulário; Definindo legenda para agrupamentos;

# Resumo do capítulo

- Os itens serão:
  - Trabalhando com botões de rádio; Criando uma lista drop-down; Definindo um campo de digitação de texto livre; Criando caixas de seleção; Utilizando opções de tipo data





## 3. CSS Estilos Iniciais

# OBJETIVOS



# OBJETIVOS

- Conhecendo as diferentes formas de declaração de estilos
- Manipular as propriedades:
  - *Text*
  - *Font*
  - *Background*
  - *Lista*
  - *Border*
  - *Margin*
  - *Padding*

# Introdução

- Neste capítulo veremos os primeiros passos para a criação de um layout com o CSS
- Conheceremos os locais e suas formas de declaração, juntamente das regras básicas de sintaxe
- Manipular as propriedades, reconhecendo os seus nomes e as possibilidades de valores

# Introdução

- As atividades a seguir dependem dos exercícios produzidos no Capítulo anterior
- Será dado um visual mais atrativo às páginas do projeto criado até o momento
- Copie os arquivos do capítulo anterior para a pasta deste capítulo

# Atividade 1 – Conhecendo as formas de declaração de estilos

- Objetivos:
  - Conhecer as formas de declaração
  - Conhecer a sintaxe das folhas de estilo

# Formatando com CSS

- CSS é uma linguagem de estilo, na qual um punhado de regras (declarações) diz como o conteúdo deve ser apresentado
- CSS significa *cascading style sheet* (folhas de estilo em cascata)
- A forma mais comum e prática para colocar estilos em sites com várias páginas é criar uma arquivo externo contendo todas as declarações de CSS

# Formatando com CSS

- A segunda opção, mais comum em sites de página única ou em sites com páginas bem diferentes entre si, é fazer a declaração na própria página HTML, entre os elementos <head> e </head>

# Formatando com CSS

- A terceira opção, que deve ser utilizada para as exceções às regras comuns de uma folha de estilo e deve ser usada somente em último caso, é a declaração junto ao elemento

# Formatando com CSS

- É possível combinar os três métodos de posicionamento de estilos em uma mesma página
- Tome cuidado para manter o código organizado, evitando criar regras conflitantes, a menos que seja necessário

# Formatando com CSS

- O efeito cascata que dá nome a linguagem faz referência ao conceito de peso que cada regra tem, no caso de haver regras conflitantes
- Isso porque é possível ter folhas de estilos de estilos de três origens diferentes
  - As que o autor da página escreveu
  - As que o usuário alterou
  - E as do navegador

# Formatando com CSS

- As de maior peso são definidas pelo autor, em segundo as do usuário e por último as do navegador
- Assim como efeito cascata, existem os conceitos de especificidade e herança de regras, que serão vistas mais adiante

# Sintaxe básica

- Em geral, uma declaração é feita primeiro invocando o “seletor”, que é o elemento a ser modificado, seguidos pelas propriedades e valores de mudança

**Regra CSS**

**seletor { propriedade: valor; }**

**Declaração**

# Sintaxe básica

- Alterando a cor de um parágrafo

```
p {
 color: #990099;
```

- p: seletor
- color: propriedade
- #990099: valor

# Sintaxe básica

- No exemplo, *p* é o seletor, nesse caso o seletor de elemento. É escrito assim, somente com a letra (ou as letras) do elemento, depois abre chave {} para iniciar a relação de todas as propriedades que serão alteradas para esse seletor (elemento *p*)
- Nesse exemplo estamos apenas mudando a propriedade de cor (color), e o sinal de dois pontos (:) seguido do código hexadecimal da cor (#990099)
- O sinal ponto e vírgula (;) encerra essa linha de propriedade/valor e o sinal de fechar chave () encerra a declaração do que deve ser alterado para o elemento *p*.

# Locais de declaração

- Folhas de estilo externa: as declarações são descritas em um arquivo externo de nome “estilo.css”, e depois o arquivo todo é chamado para dentro de sua página, entre os elementos <head> e </head>, assim:

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

# Locais de declaração

- Folhas de estilo incorporada: as declarações são descritas diretamente na sua página HTML, entre os elementos <head> e </head>, assim:

```
<style type="text/css">
p{
color: #990099;
}
</style>
```

# Locais de declaração

- Folhas de estilo em escopo: outra forma de trabalhar com estilos incorporados, em que a declaração é válida apenas dentro de um contexto específico:

# Locais de declaração

```
<div id="destaque">
 <style rel="stylesheet" type="text/css" scoped="scoped">
 /* seletores / propriedades e valores */
 </style>
 <!-- conteúdo HTML dentro do div -->
 <!-- As regras CSS serão aplicadas apenas aqui dentro -->
</div>
```

- Porém sua aceitação por parte dos navegadores é pequena. Até o momento, apenas o Firefox dá suporte a ela.

# Locais de declaração

- Folhas de estilo inline: as declarações são descritas junto ao seu texto, dentro do elemento que modificam, ficando assim:  
`<p style="color: #990099;"> Parágrafo sendo alterado por uma regra de estilo CSS inline </p>`
- A recomendação e boa prática utilizada por desenvolvedores é usar a opção de folha de estilo externa

# Propriedades CSS para estilização das fontes

- Definem os diferentes aspectos de apresentação dos glifos (letras e caracteres) que compõe conteúdos textuais dos diferentes elementos da marcação HTML
- Vejamos a seguir algumas propriedades básicas

# Propriedades CSS para estilização das fontes

- **color** - cor da fonte;
- **font-family** – família (tipo) de fontes;
- **font-size** - tamanho da fonte;
- **font-style** – estilo da fonte;
- **font-variant** – fontes maiúsculas de menor altura;
- **font-weight** – peso da fonte;
- **font-stretch** – grau de expansão/contração dos glifos;
- **font** – maneira abreviada para declarar todas as propriedades anteriores (exceto cor)

# Valores válidos para as propriedades das fonte

- Color (cor):
  1. código hexadecimal: #ffc6d9
  2. código rgb: rgb(255,235,0)
  3. código rgba: rgb(255,235,0, 0.7)
  4. código hsl: hsl(210,100%,40%)
  5. código hsla: hsla(155,80%,35%,0.4)
  6. palavra-chave: red, blue, green...etc
  7. transparente: transparent

# Valores válidos para as propriedades das fonte

- font-size (tamanho da fonte):
  1. xx-small
  2. x-small
  3. small
  4. medium
  5. large
  6. x-large
  7. xx-large
  8. smaller
  9. larger
  10. length: medida CSS de comprimento  
exemplos: px, em, rem, % (porcentagem) ...

# Valores válidos para as propriedades das fonte

- font-style (estilo da fonte):
  1. normal: fonte normal (em pé)
  2. italic: fonte inclinada
  3. oblique: fonte obliqua

# Valores válidos para as propriedades das fonte

- font-variant (fontes maiúsculas de menor altura):
  1. normal: fonte normal
  2. small-caps: transforma em maiúsculas de menor altura

# Valores válidos para as propriedades das fonte

- font-weight (peso da fonte):
  1. normal
  2. bold
  3. bolder
  4. lighter
  5. 100
  6. 200
  7. 300
  8. 400
  9. 500
  10. 600
  11. 700
  12. 800
  13. 900

# Valores válidos para as propriedades das fonte

- font-stretch (grau de expansão/contração dos glifos):
  1. normal
  2. ultra-condensed
  3. extra-condensed
  4. condensed
  5. semi-condensed
  6. semi-expanded
  7. expanded
  8. extra-expanded
  9. ultra-expanded

# Propriedades CSS para estilizar textos

- Textos são inseridos em uma página com o uso de elementos de marcação, tais como: h1, p, div, span, body etc.
- text-align: destina-se a definir o alinhamento horizontal do texto. Os valores possíveis são:
  1. left : Alinha o texto à esquerda do box
  2. right: Alinha o texto à direita do box
  3. center: Alinha o texto no centro do box
  4. justify: Alinha o texto de modo que cada linha estenda-se da esquerda à direita do box

# Propriedade CSS para estilizar bordas

- Definem as características das 4 bordas de um elemento HTML
- border-color: define a cor da borda
- color:
  - código hexadecimal: #ffc6d9
  - código rgb: rgb(255,235,0)
  - código rgba: rgb(255,235,0, 0.7)
  - código hsl: hsl(210,100%,40%)
  - código hsla: hsla(155,80%,35%,0.4)
  - palavra-chave: red, blue, green...etc
  - transparente: transparent

# Controlando as entrelinhas com *line-height*

- line-height (altura da linha): permite controlar o espaçamento entre as linhas de um texto
  1. normal: é o valor padrão nativo do navegador (entre 1 e 1.2);
  2. número: um valor expresso por um número inteiro ou decimal; line-height: 1.5;
  3. comprimento: um valor CSS para comprimento (px, em, pt, etc); line-height: 2em;
  4. porcentagem: um valor expresso em porcentagem; line-height: 200%;

# Outros elementos

- first-line (primeira linha): O pseudoelemento :first-line casa com a primeira linha do texto de um elemento. Ele só causa efeito textos dentro de elementos nível de bloco e não causa efeito em elementos inline
- Quando usado para estilizar um parágrafo, somente a primeira linha do parágrafo será estilizada, mesmo com o texto se distribuindo por várias linhas
- Ex.:

```
p:first-line {
 • /* propriedades a serem aplicadas */
}
```

# Atividade 2 – Estilos de fonte e texto

- Objetivos:
  - Criar layout da página “Quem somos”
  - Manipular propriedades de texto e fonte
- Tarefas:
  - Criar folha de estilo externa
  - Manipular seletores de elemento
  - Manipular propriedades de texto e fonte
  - Observar resultados
  - Salvar o trabalho

# Vamos praticar!

- Criar uma folha de estilo com o nome *principal.css*
- Utilizar referência externa do arquivo CSS criado dentro do HTML
- Atribuir a cor #C0262C ao elemento `<h1>`
- Colocar o tamanho da fonte do `<h1>` para 40 px utilizando a unidade de medida *em*

# Vamos praticar!

- Alterar o tamanho da fonte do elemento `<h2>` para 28 px de altura utilizando a unidade de medida *em*
- Alterar o tamanho das letras do `<p>` para *1.1em*
- Alterar a altura da linha do `<p>` para *1.3em*
- Atribuir alinhamento do texto do `<p>` para justificado

# Vamos praticar!

- Alterar cor da borda do elemento `<hr>` para `#C0262C`
- Atribuir ao elemento `<body>` as famílias de fontes Helvetica, Arial, sans-serif
- Atribuir o pseudoelemento `first-line` ao `<p>` e colocar tamanho da fonte `1.17em`, colocar fonte-variant `small-caps`, colocar peso da fonte `bolder`, e trocar o estilo da fonte para `itálico`

# Propriedade font-family

- Deve-se colocar, no mínimo, duas opções de fontes caso não exista a fonte escolhida na máquina do usuário
- Só funcionará com a fonte declarada se o usuário possuir a fonte instalada em sua máquina
- Caso queira ser universal, utilize a fonte mais simples possível, por exemplo: arial, arial black, trebuchete MS, verdaba; georgia, times, times new romam; courier, courier new; comic sans MS

# Família de fonte CSS3

- Outra forma de modificar a fonte é usando a propriedade *@font-face*, homologada na CSS3
- Porém é necessário verificar quais navegadores são capazes de lidar corretamente com elas (acesse o link <https://caniuse.com/#feat=fontface>)
- Deverá ser analisado os navegadores mais utilizados pelos usuários para utilizar a propriedade *@font-face*

# Google fonts

- Uma forma rápida de usar a propriedade `@font-face` é por meio das fontes oferecidas do Google Fonts (<https://fonts.googleapis.com/>) e localizar a fonte desejada
- Após localizado a fonte basta copiar a linha de código e adicionar na `<head></head>`
- `<link href="https://fonts.googleapis.com/css?family=Cute+Font" rel="stylesheet">`

# Font Squirrel

- Para quem gosta de ter controle total sobre os próprios arquivos, outra opção para se usar a propriedade *@font-face* é por meio do kit de fontes web oferecido para algumas fontes do site

Font Squirrel (<https://www.fontsquirrel.com/>)

# Atividade 3 – Estilos de fundo e lista

- Objetivos:
  - Criar o layout da página “Curiosidades”
  - Manipular propriedades de fundo, lista, espaçamento, margem e borda
- Tarefas:
  - Abrir múltiplos arquivos
  - Criar folha de estilo interna
  - Manipular seletores de elemento
  - Manipular propriedades de fundo, lista, espaçamento, margem e borda
  - Observar resultados
  - Salvar o trabalho

# Propriedade CSS background

- São 8 as propriedades CSS destinadas a estilizar o fundo de um box HTML:
  1. background-color (define a cor do fundo);
  2. background-image (define uma imagem de fundo);
  3. background-repeat (define a maneira como a imagem de fundo é posicionada);
  4. background-attachment (define se a imagem de fundo "rola" ou não com a tela);

# Propriedade CSS background

5. background-position (define como e onde a imagem de fundo é posicionada);
6. background-clip (define a área do box onde a imagem de fundo é aplicada);
7. background-origin (define a posição de origem da imagem no box);
8. background-size (define as dimensões da imagem no box);
9. background (maneira abreviada para declarar todas as propriedades anteriores).

# background-color

1. código hexadecimal: #ffc6d9
2. código rgb: rgb(255,235,0)
3. código rgba: rgb(255,235,0, 0.7)
4. código hsl: hsl(210,100%,40%)
5. código hsla: hsla(155,80%,35%,0.4)
6. palavra-chave: red, blue, green...etc
7. transparente: transparent

# background-image

1. URL: url(caminho/imagem.gif)
  2. URL: url('caminho/imagem.gif')
  3. URL: url("caminho/imagem.gif")
  4. Gradiente linear: linear-gradient(45deg, blue, red)
  5. Gradiente radial: radial-gradient(20px 50px, green, blue, red)
- O uso de aspas simples ou duplas no URL da imagem é facultativo

# background-repeat

1. imagem não repete: no-repeat
2. imagem repete na vertical e horizontal: repeat
3. imagem repete na vertical: repeat-y
4. imagem repete na horizontal: repeat-x
5. imagem toca as quatro bordas internas do box e são espaçadas de modo a se distribuírem igualmente: space
6. imagem toca as quatro bordas internas do box e são redimensionadas de modo a preencherem o fundo tocando umas nas outras: round

# background-position

- 1. Xpx Ypx
- 2. X% Y%
- 3. top left
- 4. top center
- 5. top right
- 6. center left
- 7. center center
- 8. center right
- 9. bottom left
- 10. bottom center
- 11. bottom right
- 12. top
- 13. right
- 14. bottom
- 15. left

# Múltiplas imagens de fundo

- Podemos definir quantas imagens de fundo quisermos, em um mesmo box, posicionando e dimensionando cada um deles individualmente
- Acessem o site para testar a propriedade background das CSS:  
<https://www.maujor.com/tutorial/interativo/ibackNOVO.html>
- Considere a regra CSS a seguir, destinada a definir três imagens como fundo de uma DIV:

# Múltiplas imagens de fundo

```
div {
 background-image:
 url(triangulo.png),
 url(circulo.png),
 url(retangulo.png);
 background-repeat:
 repeat-x,
 no-repeat,
 repeat-y;
 background-position:
 right 120px,
 center,
 0 0;
 background-color:
 #f0c;
}
```

# Vamos testar

- Abra o arquivo *curiosidades.html*
- Ajuste as telas para que fique exibido as duas janelas: navegador e notepad++
- Abaixo de </title> adicione as tags <style type="text/css"></style>

# Estilo de fundo – Atividade

- Ao elemento h2 adicione a cor de fundo #C0262C;
- Adicione a imagem de fundo back-bolinha.gif que está dentro da pasta multimídia;
- Faça com que a imagem se repita por todo fundo do h2;
- Coloque na posição centralizado;
- Atribua a cor #FFFFFF ao elemento h2;
- Verifique as alterações que aconteceram;

# Estilo de fundo – Atividade

- Referências: cor de fundo do título 2: #c0262c; cor do texto do título 2: #ffffff; imagens do fundo do título 2: back-bolinha.gif; a imagem deve se repetir por todo o fundo do título 2 e estar na posição centralizado

# Resultado

**Antes de mais nada é importante que você tenha em mente a resposta para algumas questões:**

# Resultado

- O procedimento anterior acrescenta uma imagem de fundo que se repete ao longo de toda a área do elemento `h2` a partir da posição central
- Foi alterada a cor do texto, portanto foi necessário alterar também a cor de fundo, uma medida de segurança caso a imagem não carregue, o que tornaria a leitura impossível

# Estilo de fundo com mais de uma img

- Ainda no código anterior, adicione a imagem importante-saber.jpg antes da outra imagem
  - background-image: url(img.jpg), url (img.jpg)
- Alinhe esta imagem centralizada a direita
  - background-position: direita, esquerda...
- Adicione a não repetição na imagem adicionada
- Deixe o tamanho da fonte em 2.5em
- Adicione um espaçamento entre as letras de .1em

# Resultado

Antes de mais nada é importante que você tenha em mente a resposta para algumas questões:



# Resultado

- Na img foi acrescentado uma imagem antes da imagem de repetição, para que a img que não se repetirá fique na frente da imagem que se repete
- Foi adicionado estilos de tamanho de fonte e espaçamento entre caracteres para aumentar a legibilidade

# Espaçamento

- Normalmente, quando se acrescenta uma imagem de fundo, é necessário aumentar o espaçamento interno do elemento que está sendo alterado para que a imagem se acomode e seja mais bem visualizada
- O recurso *padding* é utilizado para manipular esse espaçamento entre o conteúdo do elemento e sua borda

# Espaçamento

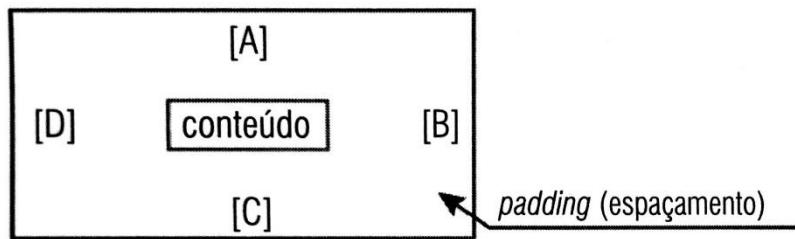
- Acrescente ao código anterior a largura: 470 px;
- Adicione o espaçamento: 61px 400px 61px 100px;



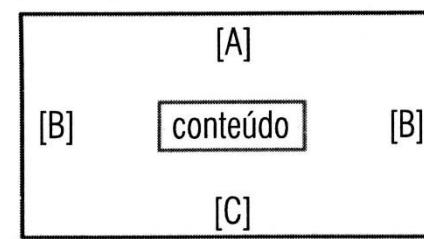
# Padding (espaçamento)

- Essa propriedade pode ser escrita de 4 formas diferentes: usando 4 valores; 3 valores; 2 valores; 1 valor

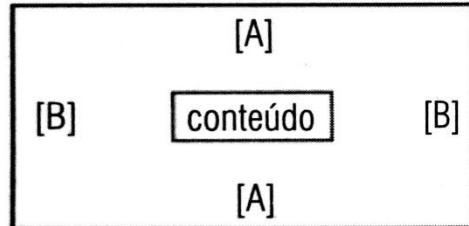
*padding: A B C D;*



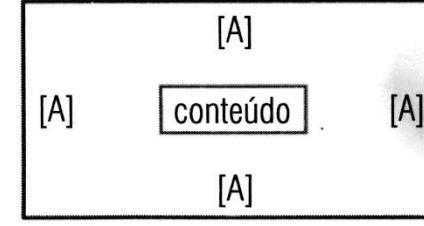
*padding: A B C;*



*padding: A B;*



*padding: A;*

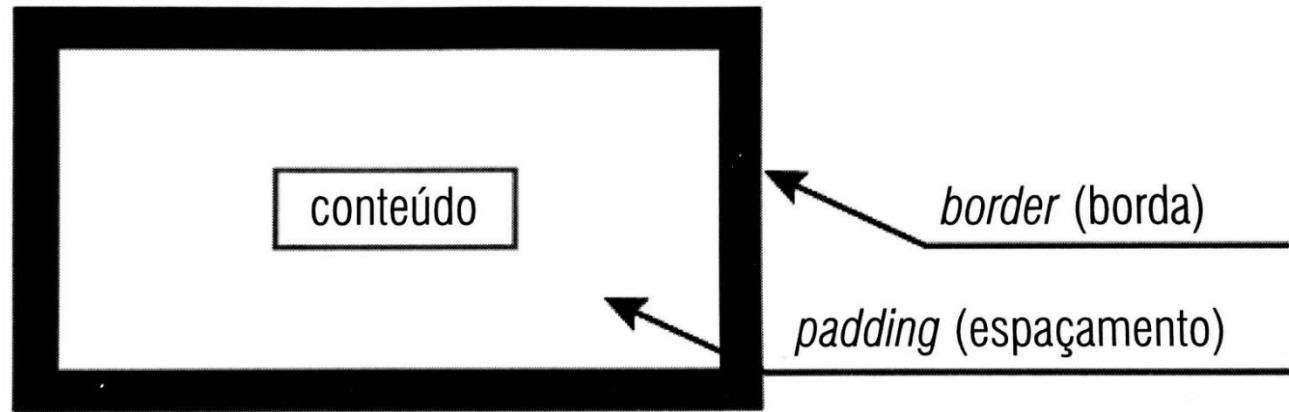


# Estilo - *figcaption*

- Vamos criar um estilo para o elemento *figcaption*
  - cor de fundo: #C0262C;
  - espaçamento: 7px 0; (não precisa declarar o tipo quando o valor é 0)
  - cor: #FFFFFF;
  - tamanho da fonte: 1.5em;
- Esse elemento altera o elemento de título das imagens

# Bordas

- Também é possível trabalhar com bordas de um elemento. Se considerar o conteúdo como referência e “caminhar” para fora dele, a primeira área é determinada como *padding*, em seguida vem a borda



# Estilo - figure

- Vamos criar um estilo para o elemento *figure*
  - Adicione a cor #C0262C a borda;
  - Adicione o estilo sólido a borda;
  - Coloque a espessura fina as 4 bordas;
  - Crie o efeito raio das bordas com os seguintes valores: 35px 35px 0 0;
  - Adicione espaçamento com os valores: 20px 0 0;
  - Alinhe os textos/img centralizado

# Resultado



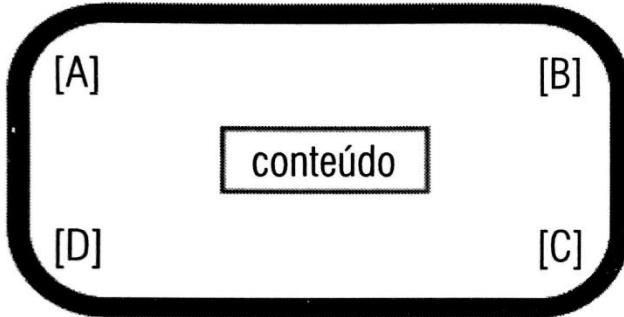
Para brincar ou esquentar os seus pés

# Resultado

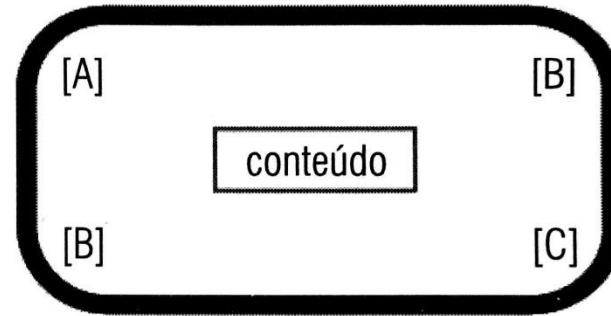
- O resultado é uma borda de cor vermelha, sólida e fina
- Da mesma forma que as propriedades *padding* e *border*, a propriedade *border-radius* tem 4 formas possíveis de declaração conforme imagem:

# Estilo border-radius

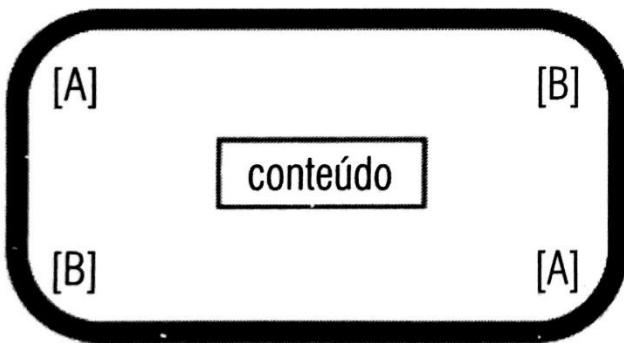
*border-radius: A B C D;*



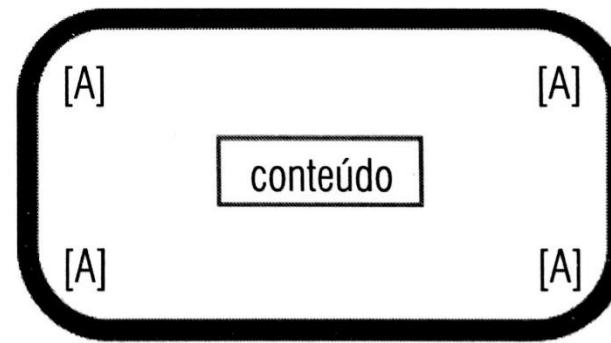
*border-radius: A B C;*



*border-radius: A B;*



*border-radius: A;*

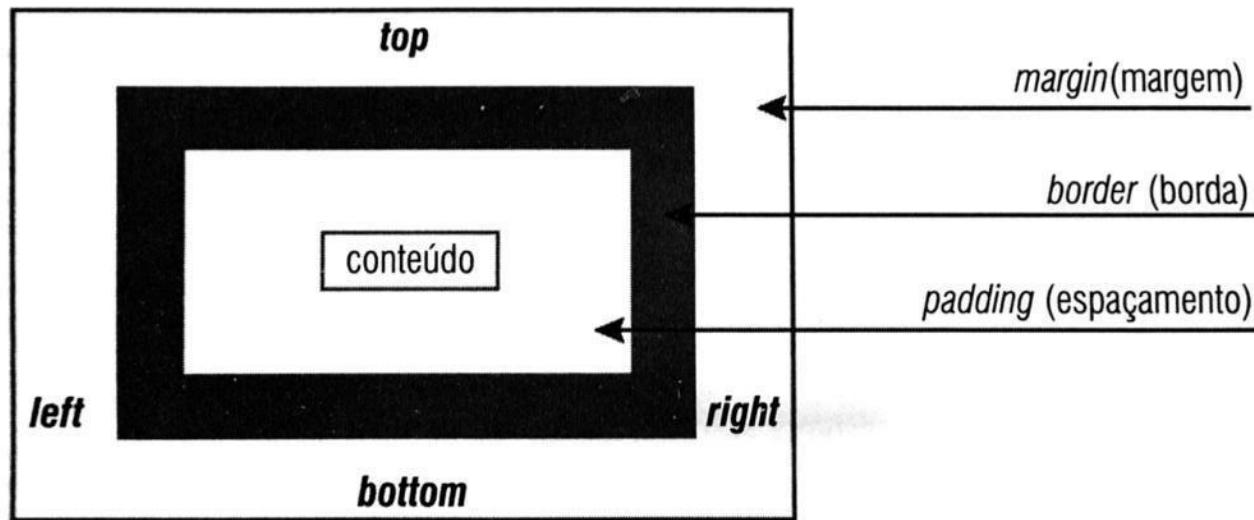


# Margens

- Para finalizar o que é conhecido como Box Model (modelo de caixa) do CSS, só falta a propriedade *margin* (margem), a qual afeta os elementos que estão ao redor da caixa
- Todo elemento HTML é uma caixa retangular e respeita estas regras
- Existem algumas variações entre tipos de exibição de conteúdo de elementos, como elementos em bloco, de linha, de tabela, etc.

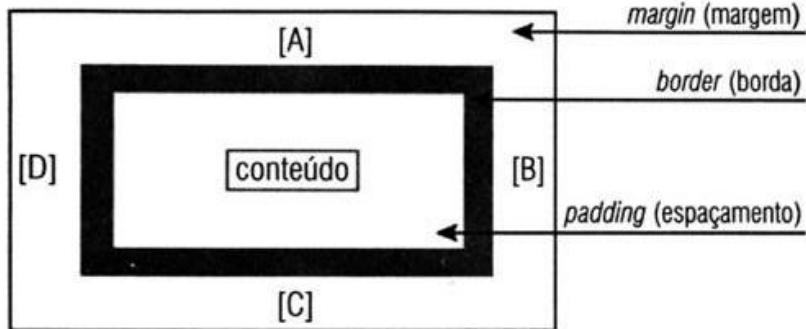
# Margens

- Da mesma forma que as propriedades *padding* e *border*, também existem diferentes maneiras para se declarar a propriedade *margin* e seus valores

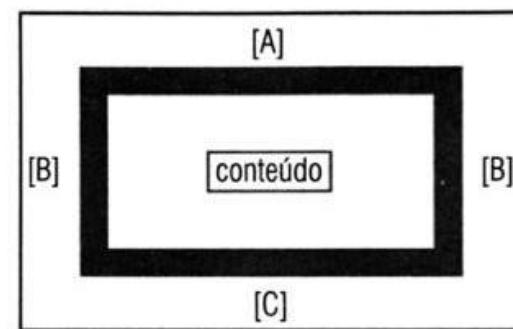


# Margens

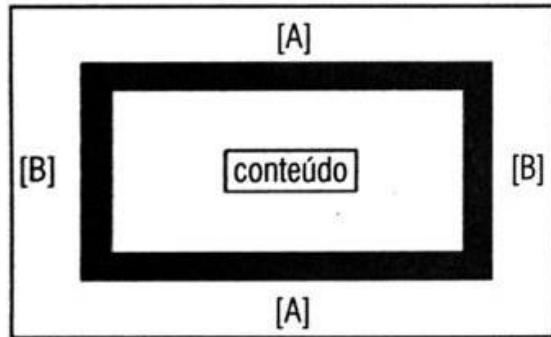
*margin: A B C D;*



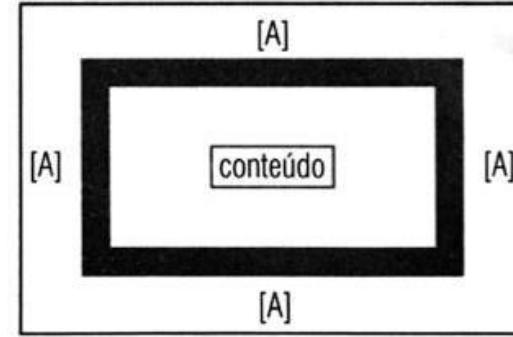
*margin: A B C;*



*margin: A B;*



*margin: A;*



# Listas

- Existem 3 propriedades específicas para as listas:
  - list-style-type
  - list-style-image
  - list-style-position
- Somadas as outras propriedades do CSS, as listas podem se transformar em praticamente tudo que você quiser

# Lista com imagem

- list-style-image: configura uma imagem em vez do marcador de lista
- valores: none, url

# Posição do marcador na lista

- list-style-position: configura onde o marcador de lista será posicionado
- valores: inside (dentro), outside (fora)

# Tipo de marcador de lista

- list-style-type: configura o tipo de marcador de lista
- valores: none, disc, circle, square, decimal, decimal-leading-zero, lower-roman, upper-roman, lower-alpha, upper-alpha, lower-latin, upper-latin, hebrew, armenian, georgian, cjk-ideographic, hiragana, katakana, hiragana-iroha, katakana-iroha

# Listas

- Lembre-se de que essas propriedades podem ser usadas apenas em elementos de lista, não afetando outros tipos de elementos

# Exemplo

```
ul li {
 list-style-type: "valor";
 list-style-image: url("img.png");
 list-style-position: "valor";
 margin: "30px 0";
}
```

# Atividade

- Na lista onde contem as raças crie os seguintes elementos CSS
  - Retire a marcação original do elemento de lista
  - Adiciona a imagem ossinho.png no elemento da lista
  - deixe posicionado dentro da lista
  - Adicione uma margem de 30px 0 para aumentar o espaçamento entre os itens da lista

# Resultado

Veja algumas raças:

 [Bulldog](#)

 [Beagle](#)

 [Bull Terrier](#)

 [Collie](#)

 [Labrador](#)

 [Lhasa Apso](#)

# Atividade

- Neste estilo foi retirada a marcação original do elemento lista (none) e foi adicionada uma imagem em seu lugar
- Para que a posição indentada (um recuo do texto em relação à sua margem) do item seja preservada, é necessário definir a imagem para dentro da marcação (inside)
- Como o tamanho da imagem é maior do que a marcação da ul, foi preciso aumentar a margem do elemento

# Lista de definição (dl, dt, dd)

- Vamos aplicar algumas formatações aos elementos da lista de definição da página curiosidades.html

# Lista de definição (dl, dt, dd)

- Adicione ao elemento dt as seguintes características
  - cor: #fff;
  - cor de fundo: #C0262C;
  - espaçamento: 7px;

# Lista de definição (dl, dt, dd)

- Adicione ao elemento dd as seguintes características
  - cor da borda: #C0262C;
  - espessura da borda: fina;
  - estilo da borda: sólido;
  - adicione raio nas bordas: 0 0 25px 25px;
  - margem superior: -3px;
  - margem inferior: 30px;
  - espaçamento: 20px 20px 0;

# Resultado

## Bulldog

Embora a aparência do Bulldog Inglês pode ser um pouco intimidante, está entre o mais gentil dos cães. Apenas o mesmo que vai ver fora qualquer intruso, e poucos arriscariam um encontro próximo com um cão corajoso o suficiente para atrair um touro. É descrito como um animal muito carinhoso e confiável, gentil com as crianças, mas conhecido por sua coragem e suas excelentes habilidades de guarda.

[Tenha mais informações](#)

[Voltar](#)

## Beagle

O Beagle é amoroso, doce e gentil, feliz de ver todos, cumprimentá-los com uma cauda abanando. É sociável, corajoso e inteligente. O Beagle é excelente com crianças e geralmente bem com outros cães, mas por causa de seus instintos de caça, não deve ser confiado com animais de estimação não-caninos, a não ser socializados com gatos e outros animais domésticos quando jovens

[Tenha mais informações](#)

[Voltar](#)

# Resultado

- Neste exemplo foi selecionado cada um dos elementos de forma pontual, sem usar o dl, que é o elemento pai
- No caso do dt foi alterada a cor do texto, a cor de fundo e o espaçamento
- No elemento dd foi alterada a borda e sua curvatura, as margens e o espaçamento

# Contadores – Lista ordenada

- Vamos aplicar algumas formatações aos elementos da lista ordenada da página curiosidades.html

# Contadores – Lista ordenada

- Adicione ao elemento ol as seguintes características:
  - Estilo do marcador da lista: nenhum;
  - ao texto configure os espaços em branco internos para: pre
  - Distância/altura entre as linhas: .9em;
  - Definir uma variável/nome de contagem para personalizar os contadores com texto (1,2,3...):
    - counter-reset: contador;

# Contadores – Lista ordenada

- Adicione ao elemento lista propriamente dito (ol li) o pseudoelemento *:before* para selecionar a posição antes do elemento indicado no seletor, ou seja, a posição antes do elemento li
  - Altere a cor para: #999999;
  - Incrementar a variável em 1:
    - counter-incremente: contador;
  - Usar o número do contador como conteúdo:
    - content: “Questão “ counter(contador) “ \00BB “;

# Resultado

Questão 1 » Mora em casa ou apartamento?

Questão 2 » Quer um cachorro para ser parceiro em atividades? Brincar? Terapia?

Questão 3 » Você mora no campo ou na cidade?

Questão 4 » Alguém da família é alérgico a cães?

Questão 5 » Está disposto a cuidar do cão com frequência?

Questão 6 » É rígido com limpeza?

Questão 7 » Fica preocupado com bagunça?

# Resultado

- Nesse procedimento, foi alterado o aspecto do elemento *ol* e seus itens *li*
- No elemento pai da lista, *ol*, foi retirada do tipo padrão de contagem dos elementos (1,2,3...) e a entrelinha
- Foi definida a propriedade *counter-reset*, que é a responsável por indicar o nome do contador a ser utilizado

# Resultado

- Em seguida, no item da lista propriamente dito, foi utilizado o pseudoelemento *:before* para selecionar a posição antes do elemento *li*
- Para essa posição, alterou-se a cor, criou-se um contador que se auto incrementa em uma unidade e inseriram-se um conteúdo, que é a palavra “Questão”, o contador e o sinal “>>”
- Caso queiram outros caracteres especiais, como o exemplo utilizado do “>>”, use a calculadora conversora de valores hexadecimais para CSS em <http://www.evotech.net/articles/testjsentities.html>

# Links

- Os links são um caso especial dentro das opções de estilo, visto que os navegadores já se encarregam, por padrão, de adicionar estilo para os links: no geral, azul e sublinhado e, após ser visitado, roxo e sublinhado
- São elementos especiais também por carregarem o conceito de interatividade, portanto, possuem estados diferentes que variam conforme a ação do usuário

# Links

- É possível selecionar cada um dos estados indicados por meio das pseudoclasses:
  - *a:link*: link padrão, antes de ser visitado
  - *a:visited*: link após ser visitado
  - *a:hover*: link quando o mouse passa por cima
  - *a:active*: link ativo
- ATENÇÃO: quando for usar os 4 seletores juntos, procure manter a ordem indicada acima, para que o funcionamento, principalmente o da passagem de mouse sobre o link, continue correto

# Atividade

- Alterar o link da lista não ordenada (*ul*) que contém as raças dos cachorros:
  - retirar o sublinhado do link
  - alterar a cor para: #C0262C
  - aumentar o peso da fonte para negrito
  - aumentar o espaçamento em 7px
- Alterar o layout quando o mouse passa por cima do link
  - exibir o sublinhado novamente

# Resultado

 **Bulldog**

 **Beagle**

 **Bull Terrier**

 **Collie**

 **Labrador**

 **Lhasa Apso**

# Resultado

- Foi alterado o link que está contido na lista não ordenada (ul):
  - o sublinhado do link foi retirado, a cor passou a ser vermelha, o texto está em negrito e foi acrescentado um pequeno espaçamento para aumentar a área do clique no texto
- Foi alterado o layout quando o mouse passa por cima do link, que passa a exibir novamente o sublinhado

# Escolhendo links específicos

- É possível ainda escolher links de uma forma bastante específica utilizando um outro tipo de seletor chamado seletor de atributo
- Desta forma, a seleção deverá ser bastante específica, conforme exemplos a seguir:

# Escolhendo links específicos

- Código HTML:

```
<dd>
```

“conteúdo da lista”

```
 “link”
```

```
</dd>
```

- CSS para alterar o link que abrirá em uma nova aba:

```
dd a[href="top"] {
```

“propriedades a serem alteradas no css”

```
}
```

# Escolhendo links específicos

- Código HTML:

```
<dd>
```

“conteúdo da lista”

```

```

```
</dd>
```

- CSS para alterar o link que abrirá em uma nova aba:

```
dd a[target="_blank"] {
```

“propriedades a serem alteradas no css”

```
}
```

# Atividades

- Vamos alterar as propriedades do link que abre em uma nova aba (tenha mais informações):
  - retirar a decoração do texto;
  - alterar a cor para: #C0262C;
  - adicionar o peso na fonte para deixa-la negrito
  - adicionar espaçamento de 7px;
  - display: inline-block;

# Atividades

- Vamos alterar as propriedades do link quando o mouse passa por cima e abre em uma nova aba (tenha mais informações):
  - adicionar o sublinhado ao passar o mouse por cima do texto

# Atividades

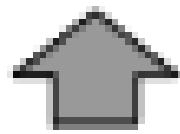
- Vamos alterar as propriedades do link que retorna para o menu das raças dos cães(voltar):
  - retirar a decoração do texto;
  - alterar a cor para: #C0262C;
  - adicionar o peso na fonte para deixa-la negrito
  - adicionar espaçamento de 7px 0 7px 30px
  - adicionar a imagem “seta-voltar.png”, na posição centralizado e a esquerda, de forma que não se repita

# Atividades

- Vamos alterar as propriedades do link quando o mouse passa por cima e abre em uma nova aba (tenha mais informações):
  - adicionar o sublinhado ao passar o mouse por cima do texto  
a[target="\_blank"]:hover{propriedades}
  - adicionar a imagem “seta-voltar-hover.png”, na posição centralizado e a esquerda, de forma que não se repita

# Resultado

Tenha mais informações



*VOLTAR*

# Combinando folhas de estilo

- Mesmo trabalhando com folhas de estilo incorporada, é possível acrescentar uma folha de estilo vinculada
- Vamos acrescentar primeiro a folha de estilo externa e depois a folha de estilo incorporada que já esta criada

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

# Atividade

- Acrescente as seguintes características ao elemento h1 da folha de estilo externa
  - altere a cor da borda inferior para: #C0262C;
  - adicione o estilo sólido a borda;
  - deixe a espessura fina;
  - adicione um espaçamento inferior em 7px;

# Exercício propostos

- A função destes exercícios é rever os conteúdos anteriores
- Em todos eles deve-se criar o arquivo CSS externo e fazer a sua ligação ao respectivo arquivo HTML
- Será disponibilizados os valores de referência e imagens de cada exercícios

# Exercício 1

- Nome do arquivo: *cabecalhos.html*
- Crie um único arquivo CSS externo e insira os estilos e os códigos necessários para que tenha o aspecto da imagem disponibilizada da página
- Referências: imagem – *cabecalhos\_back.png*; fundo: `#fff6d8`; títulos: `#906718`; caixa importante: `#d48b00`; e régua: `3px, tracejada (dashed) e #d48b00`

# Exercício 2

- Nome do arquivo: *imagens.html*
- Crie um único arquivo CSS externo e insira os estilos e os códigos necessários para que tenha o aspecto da imagem disponibilizada da página
- Referências: cor de fundo: `#fcfcfc`; cor do título, da barra de título e dos fios: `#764e36`

# Exercício 3

- Nome do arquivo: *links\_1.html* e *links\_2.html*
- Crie um único arquivo CSS externo e insira os estilos e os códigos necessários para que tenha o aspecto da imagem disponibilizada da página
- Referências: fundo do subtítulo: #7faeb7; link: #004552; link mouse-over: #ffa700; link visitado: #c9e0e5; régua: #7faeb7

# Exercício 4

- Nome do arquivo: *torta\_limao.html*
- Crie um único arquivo CSS externo e insira os estilos e os códigos necessários para que tenha o aspecto da imagem disponibilizada da página
- Referências: fundo: `#ffffcb`; imagens do fundo: *torta\_limao\_backS.png* e *torta\_limao\_backL.png*; título e cor da barra do título 2: `#FF9900`; fundo da lista: `#daff95`; itens de lista: *torta\_limao\_item1.gif*, *torta\_limao\_item2.gif* e *torta\_limao\_item3.gif*



# HTML 5 – Elementos Técnicos

# Objetivo

- Conhecer novos elementos do HTML5

# Novos Elementos

- O HTML5 possui uma série de elementos que permitem definir setores principais na página do código HTML
- Por meio desses elementos, é possível diferenciar diretamente pelo código HTML5 do site, como:
  - Rodapé
  - Cabeçalho
  - Barra lateral

# Novos Elementos

- Desta forma, conseguimos nos reportar à área do site indicando exatamente onde deve estar o conteúdo mencionado
- Fazendo uma comparação: quando você está lendo um livro, é muito fácil diferenciar um capítulo, um título ou um parágrafo apenas pelo tamanho da letra, da fonte ou da cor
- Isso seria muito complicado para um mecanismo de busca utilizando apenas o código, pois esses detalhes passariam despercebidos

# Novos Elementos

- Com o HTML5, os buscadores têm seu trabalho simplificado e conseguem obter do código informações mais exatas, levando menos tempo para estocar essa informação
- É o que chamamos de **SEMÂNTICA** ☺, que vocês estão cansados de ouvir ☹
- Ela pode ser compreendida como o estudo do significado, ou seja, esses novos elementos representam um significado para o navegador e o desenvolvedor

# Novos Elementos

- Com a inserção desses elementos, o HTML5 oferece novos elementos semânticos ☺ para definir diferentes partes de uma página para web
- Vejamos a seguir quais seriam essas partes de uma página web:

# Novos Elementos

<header>

<nav>

<section>

<aside>

<article>

<footer>

# Elemento <header>

- Especifica um cabeçalho de um documento ou uma seção
- Deve ser utilizada quando seu conteúdo tiver o significado introdutório para a página
- A página pode ter vários elementos <header> em um mesmo documento

# Elemento <nav>

- Define um conjunto de links de **navegação**
- Esse elemento destina-se a grandes blocos de links de navegação
- No entanto, nem todos os links em um documento precisam estar dentro de um elemento <nav>

# Elemento <section>

- Define uma seção do documento, um agrupamento temático de um conteúdo que geralmente vem com um título
- A página pode ser dividida em seções nas quais o desenvolvedor pode inserir informações como: introdução, conteúdo e informações de contato

# Elemento <article>

- Especifica um conteúdo próprio e independente de outros conteúdos
- Toda vez que a página tiver uma informação que faça sentido por si própria de forma independente do resto do site, esse conteúdo deve ser colocado nesse elemento da página
- Por exemplo: postagem de blogs, artigos para jornal, fóruns de discussão

# Elemento <aside>

- Define um conteúdo secundário que pode ser colocado como uma barra lateral, um conteúdo que pode agregar ou não informações relativas à página

# Elemento <footer>

- Esse elemento especifica um rodapé de um documento ou de uma seção
- Deve conter informações sobre o autor do documento, os direitos autorais, os links para termos de uso, as informações de contato, etc.
- É possível ter vários elementos <footer> em uma página

# Atividade 1 – Elementos para a página “Quem somos”

- Objetivo: inserir novos elementos na página *quem-somos.html*
- Tarefas:
  - Inserir os elementos *main* e *footer*
  - Salvar a sua página

# Elemento <main>

- O elemento <main> define o conteúdo principal dentro do <body> em seu site
- Em 2012 a W3C incluiu o elemento <main>, anteriormente os desenvolvedores utilizavam o elemento <div>, e essa inclusão tornou o novo elemento principal do corpo do documento
- Não devem haver mais do que um elemento <main> em uma mesma página
- O elemento <main> **não** deve ser utilizado dentro de outros elementos, como <article>, <aside>, <footer>, <header> ou <nav>

# Atividade 01 – Elementos para página

## “Quem somos”

- Inserir o elemento <main> a página *quem-somos.html*

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

# Atividade 01 – Elementos para página

## “Quem somos”

- Inserir o elemento <footer></footer> e os dados do atendimento após o fechamento do </main>

<footer>

<p>Atendimento:  
atendimento@meuamigocaopetshop.com  
(0XX) 99999-9999 / (0XX) 88888-8888 </p>

</footer>

# Atividade 02 – Elementos para página “Produtos”

- Objetivo: Inserir novos elementos na página *produtos.html*
- Tarefas:
  - Inserir os elementos *main* e *footer*
  - Salvar sua página

# Atividade 03 – Elementos para página “Curiosidades”

- Objetivo: Inserir novos elementos na página *curiosidades.html*
- Tarefas:
  - Inserir os elementos *main* e *footer*
  - Salvar a sua página

# Atividade 04 – Elementos para página “Banho e Tosa”

- Objetivo: Inserir novos elementos na página *banho-e-tosa.html*
- Tarefas:
  - Inserir os elementos *main* e *footer*
  - Salvar a sua página

# Atividade 05 – Elementos para página “Boletim”

- Objetivo: Inserir novos elementos na página *boletim.html*
- Tarefas:
  - Inserir os elementos *main* e *footer*
  - Salvar a sua página

# Atividade 06 – Elementos para página “Obrigado”

- Objetivo: Inserir novos elementos na página *obrigado.html*
- Tarefas:
  - Inserir os elementos *main* e *footer*
  - Inserir uma imagem e um desritivo
  - Salvar a sua página

# Atividade 07 – Elementos para página “Index”

- Objetivo: Inserir novos elementos na página *index.html*
- Tarefas:
  - Inserir os elementos *header*, *section*, *nav*, *aside*, *main* e *footer*
  - Inserir **imagens e descritivos**
  - Salvar a sua página

# HTML5Shiv

- O HTML5 trouxe novas tags com o intuito de melhorar a semântica por meio do código que escrevemos, mas nem todos os navegadores reconhecem os novos elementos (ex.: IE8 e anteriores)
- Uma saída para esse problema foi criar um elemento usando uma linguagem que é muito utilizada nos navegadores web: o JavaScript

# HTML5Shiv

- A linguagem JavaScript permite que, com o uso de scripts, o usuário possa alterar o conteúdo do documento que é apresentado
- <script>: é usada para definir uma sequência de comandos por meio de uma linguagem como o JavaScript. O elemento pode conter declarações ou apontar para um arquivo externo
- Seus uso mais comum é o de manipular imagem, validar formulário e efetuar mudanças dinâmicas no conteúdo

# Atividade 8 – HTML5Shiv

- Objetivo: Atualizar todas as páginas para que ocorra a execução do HTML5Shiv na pasta *interação*
- Tarefas:
  - Inserir o elemento <script> nas páginas dos seus projetos na
  - Salvar os seus arquivos
- Sintaxe: <script> comandos; </script>

# Atividade 8 – HTML5Shiv

```
<head>
<script src="pasta/nome_arquivo.min.js"></script>
</head>
```

# Exercícios extras

- Exercício 1: elemento <header></header>
- Exercício 2: elemento <footer></footer>
- Exercício 3: elemento <nav></nav>
- Exercício 4: elemento <section></section>, <article></article>, <aside></aside>
- Criar tabela com o resumo de tudo que foi estudado e aprendido nesse capítulo



# 5 - CSS Estilos Aprimorados

# Objetivos

- Conhecer mais tipos de seletores:
  - ID
  - Class
  - Seletores especiais
- Conhecer outros tipos de marcação semântica
  - Div
  - Span
- Conhecer os conceitos de cascata, herança e especificidade

# Objetivos

- Aprimorar a combinação de seletores, declarações de estilos e marcação semântica
- Manipular as propriedades:
  - Dimensão
  - Posicionamento
  - Apresentação
  - Tabelas

# Objetivos

- Utilizar recursos especiais:
  - Menu com abas
  - Layouts com posicionamento, flutuação e flexíveis
- Considerações finais sobre o CSS

# Seletores de Elementos

- São aqueles baseados nos próprios elementos do HTML, como: *p*, *h1*, *img*, *li*, etc.
- São a opção mais abrangente e prática para criar uma declaração de estilo, pois automaticamente mudam todos os elementos aos quais fazem referência
- Por esse motivo, essa opção deve ser utilizada para criar estilos mais genéricos

# Seletores de ID

- Esse tipo de seletor é utilizado para marcar elementos únicos ID (identificador) em suas páginas

```
#menu{
background-color: #f0e14c;
color: #59cc0d;
width: 400px;
}
```

# Seletores de ID

- REGRA DE BOAS PRÁTICAS: ao criar seus IDs, utilizem NOMES que fazem referência ao tipo de conteúdo e não ao aspecto, pois é muito comum haver alterações ao longo do projeto

# Seletores de ID

- Para nomear seus Ids:
  - Não utilizar caracteres especiais ou espaços, pois são permitidos apenas letras (maiúsculas e minúsculas), números e os sinais “-” e “\_”;
  - Não comece com números ou com um sinal “-” seguido de número;
  - Escreva exatamente da mesma forma, pois “**Nome**” é diferente de “**nome**”

# Seletores de ID

- Em CSS, a declaração do seletor do tipo ID é feita com o sinal “#” e o seu uso no código HTML é feito com o atributo de elemento *id*

# Seletores de classe

- Nesse tipo de seletor você também cria um nome, mas poderá usar esse nome quantas vezes forem necessárias

```
.destaque {
background-color: #f0e14c;
font-weight: bolder;
}
```

# Seletores de classe

- Ao criar suas classes, use nomes que fazem referência ao tipo do conteúdo e não ao aspecto

# Seletores de classe

- Para nomear suas classes:
  - Não utilizar caracteres especiais ou espaços, pois são permitidos apenas letras (maiúsculas e minúsculas), números e os sinais “-” e “\_”;
  - Não comece com números ou com um sinal “-” seguido de número;
  - Escreva exatamente da mesma forma, pois “Nome” é diferente de “nome”

# Seletores de classe

- Em CSS, a declaração do seletor do tipo classe é feita com o sinal “.” e o seu uso no código HTML é feito com o atributo de elemento *class*

# Seletor universal

- Funciona como um curinga
- É como selecionar todos os elementos, um a um, mas de uma única vez

```
* {
color: #f00;
}
```

# Combinando seletores

- São feitas com diversos tipos de seletores:
  - de elemento (p, h1, img...)
  - ID
  - classe
  - universal

# Agrupamento de seletores

```
h1, #especial, .destaque {
color: #f00;
}
•
```

- Essa regra diz que todos os elementos *h1*, mais os indicadores com o ID *#especial* e a classe *.destaque* terão sua cor alterada para vermelho

# Seletor contextual

```
p .destaque {
color: #f00;
}
•
```

- Essa regra diz que apenas os elementos marcados com a classe *.destaque* que estiverem DENTRO do elemento *p* terão suas cores alteradas para vermelho
- Elementos com a classe *.destaque* que estejam dentro de quaisquer outros elementos NÃO serão alterados

# Seletor contextual

```
p.destaque {
color: #f00;
}
• Cuidado com a grafia! É importante tomar
cuidado ao escrever seletores, pois a
AUSÊNCIA ou a PRESENÇA do espaço em
branco muda o seletor
```

# Seletor contextual

```
p.destaque {
color: #f00;
}
}
```

- Essa regra diz que **APENAS** os elementos *p* que tiverem a classe *destaque* terão a sua cor alterada
- Outros elementos *p* ou outros elementos com a classe *destaque* não sofrerão qualquer alteração

# Seletor contextual

- Existem diversos seletores, mas nem todos funcionam da mesma forma nos diversos navegadores e em suas versões
- Segue uma pequena lista de seletores e a relação de navegadores incompatíveis

# Seletor contextual

Exemplo	Seletor	Significado	Navegador
<b>Seletores de associação</b>			
$e > f$	Elemento filho	Seleciona qualquer elemento $f$ descendente direto do elemento $e$	–
$e + f$	Elemento irmão adjacente	Seleciona o elemento $f$ que estiver imediatamente após o elemento $e$	–
$e \sim f$	Elemento irmão genérico	Seleciona qualquer elemento $f$ que estiver após o elemento $e$ , desde que compartilhem a mesma ascendência	IE8, parcial

# Seletor contextual

Seletores de elemento por atributo			
e[atrib]	Atributo	Seleciona o elemento <i>e</i> que contém o atributo <i>atrib</i>	—
e[atrib="vlr"]	Valor de atributo	Seleciona o elemento <i>e</i> que contém o atributo <i>atrib</i> de valor <i>vlr</i>	—
Seletores do tipo pseudoelemento			
e:first-letter	Primeira letra	Seleciona a primeira letra do elemento <i>e</i>	
e:before	Before (antes)	Insere um conteúdo antes do elemento <i>e</i>	IE7
e:after	After (depois)	Insere um conteúdo depois do elemento <i>e</i>	IE7

# Seletor contextual

Seletores do tipo pseudoclasse			
e:hover	Ação de usuário	Seleciona quando o mouse estiver sobre o elemento <i>e</i>	
e:first-child	Primeiro filho	Seleciona o primeiro elemento <i>e</i> descendente do elemento pai	
e:nth-child()	Enésimo filho	Seleciona o enésimo filho do elemento <i>e</i>	IE8, FF3

# Pseudelementos

- São abstrações da árvore de um documento
- Possibilitam acesso à primeira letra ou à primeira linha do conteúdo, bem como algo que esta antes ou depois de um elemento
- Exemplo:
  - *p:first-line* (primeira linha), *p:first-letter* (primeira letra), *p:before* (aplicar conteúdo antes), *p:after* (aplicar conteúdo depois)

# Pseudoclasse

- Foi criado para permitir seleções baseadas em informações que estão fora da árvore do documento ou que não poderiam ser escritas usando seletores simples
- Entre elas estão as pseudoclasses dinâmicas, relativas as ações do usuário, as pseudoclasses de estado do elemento, relativas ao estado do elemento na interface do usuário, e as pseudoclasses estruturais, relativas a posição do elemento na árvore, mas de uma forma mais complexa

# Pseudoclasse

- Exemplos.:
  - *e:first-child* (primeiro-filho)
  - *e:lang* (atributo *lang*)
  - *a:link* (link)
  - *a:visited* (link visitado)
  - *e:hover* (mouse em cima do elemento)
  - *e:active* (elemento ativo)
  - *e:focus* (elemento quando recebe o foco)

# Marcação semântica

- É a marcação pelo significado
- É uma das razões de existir o HTML5
- Exemplos:
  - Um parágrafo é delimitado pelo elemento *p*
  - Um título muito importante recebe o elemento *h1*
  - Um título de menor importância o elemento *h2*
  - O conjunto principal do conteúdo fica entre os elementos *main*
  - A navegação fica entre os elementos *nav*

# Marcação semântica

- Cada elementos do HTML5 é usado para agrupar, classificar e indicar o sentido do conteúdo
- Dentro dessa marcação, devemos acrescentar mais dois elementos:
  - *div*: delimitador genérico de um conteúdo de bloco
  - *span*: delimitador genérico para um conteúdo de linha

# Marcação semântica

- Tais elementos devem agrupar conteúdos em bloco ou em linha quando não for possível utilizar um outro elemento HTML semântico
- Geralmente, são usados apenas para acrescentar um apelo visual ou um recurso de interação

# Marcação semântica

Diferença entre elementos de bloco e de linha:

- **Elemento de bloco:** é também chamado de elemento de parágrafo (*p, h1 a h6, hr*); seu aspecto é de um bloco que ocupa toda a largura disponível da janela do navegador (ou do bloco ao qual pertence) e cria blocos antes e depois de si

# Marcação semântica

Diferença entre elementos de bloco e de linha:

- **Elemento de linha:** é uma pequena parte dentro de um parágrafo ou de um bloco, por exemplo, um trecho em negrito (*strong*) ou em itálico (*em*). Possui apenas a própria largura e não cria bloco nem antes nem depois de si

# Marcação semântica

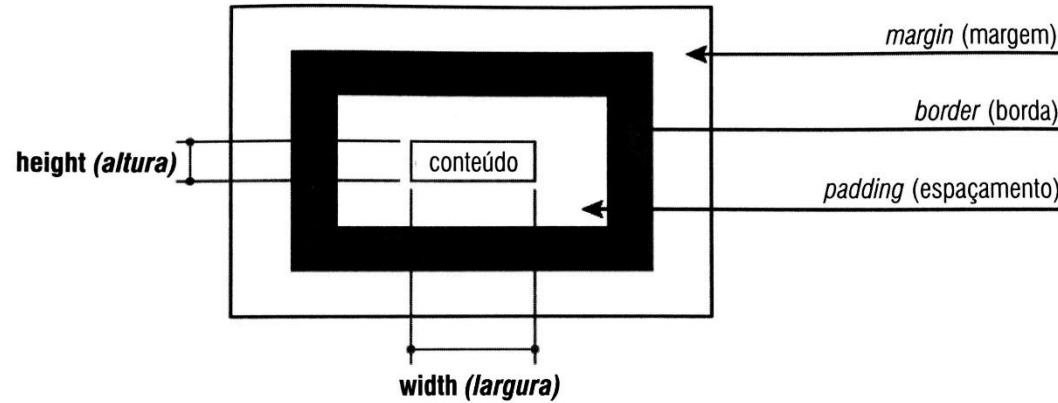
- Com estilos CSS você pode alterar esses comportamentos e dizer que um elemento de linha terá comportamento em bloco
- Elementos de linha não possuem as propriedades de estilo *width* (largura) nem *height* (altura)
- A altura de um elemento *inline* é definida pela propriedade *line-height* (entre linha)
- **Vamos fazer um teste para verificar!!!**

# Marcação semântica

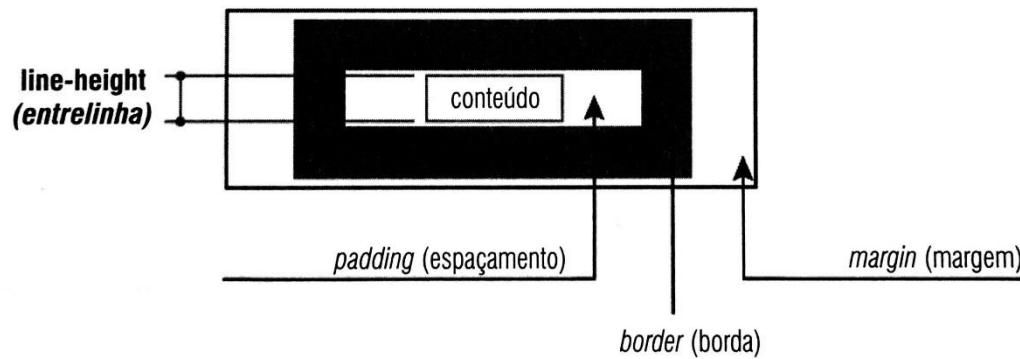
- As propriedades *padding* (espaçamento) e *margin* (margem) funcionam corretamente para valores *left* e *right*, abrindo espaço à esquerda e à direita do elemento, mas **não funcionam** para valores *top* e *bottom*
- Os valores até podem ser criados, mas não produzem descolamento acima ou abaixo, pois não há blocos nem antes nem depois de elementos de linha

# Marcação semântica

Modelo de caixa de um elemento de bloco:



Modelo de caixa de um elemento de linha:



# Herança

- Devido a quantidade de seletores e combinações diferentes, além das marcações *div* e *span*, em alguns momentos poderá correr o risco de criar mais de uma regra de estilo para o mesmo conteúdo
- Existem 3 características básicas do funcionamento dos estilos CSS:
  - Herança
  - Cascata
  - Especificidade

# Herança

- No capítulo 3, na atividade referente as fontes, utilizamos o seletor do elemento *body*:

```
body{
font-family: Helvetica, Arial, sans-serif;
}
```
- Tal declaração fez todo o texto da página ser exibido com a família fonte Helvética, ou seja, os elementos de parágrafo e de título tiveram seu aspecto alterado, pois eles são elementos **filhos** do elemento *body*

# Herança

- A propriedade *font-family*, aplicada ao elemento *body*, foi transmitida para seus elementos filhos
- Em um esquema simplificado, sua estrutura, ou sua árvore do documento (*document tree*), seria:

# Herança

html

|\_ head

| |\_ title

|

|\_ body

|\_ h1

|\_ p

# Herança

- Pensando em um esquema de caixa, no qual toda a caixa que abre tem que ser fechada, a caixa principal, chamada raiz, contém as caixas filhas *head* e *body*
- O elemento *body* também é uma caixa que contém outras, que são suas filhas, nesses casos os elementos *h1* e *p*
- Essa transmissão de propriedade do elemento pai para elementos filhos é chamada de herança

# Herança\*

- Alguns cuidados especiais devem ser tomados com relação ao conceito de herança:
  - Propriedades específicas para elementos de bloco não são herdadas por elementos de linha (???)
  - Algumas propriedades não são transmitidas por herança
  - Você pode forçar a hereditariedade de uma propriedade usando o valor *inherit*

# Cascata

- O efeito cascata é o primeiro conceito utilizado pelo navegador para a exibição de estilos e para resolução de conflitos quando há sobreposição ou duplicidade de regras:

# Cascata

1. Folha de estilo padrão do navegador do usuário
2. Folha de estilo do usuário
3. Folha de estilo do desenvolvedor:
  1. Estilo externo (importado ou linkado)
  2. Estilo incorporado/interno (definido na seção *head* do documento)
  3. Estilo inline (dentro de um elemento HTML)
4. Declarações do desenvolvedor com *!importante*
5. Declarações do usuário com *!important*

# Cascata

- Isso equivale dizer que a primeira regra válida é a do **navegador**, ou seja, quando nenhuma regra de estilo é criada, valem as regras do navegador
- Por isso, mesmo quando você não define nenhum estilo, um título *h1* aparece maior que um *h3*
- Isso está definido na folha de estilos do navegador

# Cascata

- A segunda regra válida é a definida pelo **usuário**, ele pode alterar a folha de estilos do navegador sobrepondo seus valores
- A ação mais comum é aumentar e diminuir o tamanho das fontes do navegador
- Curiosidades
  - No IE11, o usuário pode alterar as regras por meio da opção *menu: Página, Ferramenta de texto*
  - No Firefox, deve usar o botão *menu, Opções, aba Conteúdo*
  - No Chrome utiliza-se o botão *Personalizar, Configurações, Configurações Avançadas*

# Cascata

- A terceira regra válida é a definida pelo **desenvolvedor**, ele cria suas próprias folhas de estilos e sobrescreve as regras anteriores
- Nos casos que o desenvolvedor não declarou nenhuma regra para o elemento, vale o que disse o usuário; se o usuário não disse nada, vale o que disse o navegador

# Cascata

- Como vimos nas aulas anteriores, existem 3 posições diferentes para o desenvolvedor inserir as folhas de estilos, sendo a de menor valor a do estilo **externo** e a de maior valor a do estilo **inline**
- Não é só questão de importância, mas também de posicionamento (a ordem das declarações no *style*)

# Cascata

- Para que o estilo externo tenha menor valor que o estilo incorporado, o desenvolvedor deve declarar primeiro o elemento de link do estilo externo e depois o elemento de estilo incorporado

```
<link href="estilo.css" rel="stylesheet" type="text/css">
<style>
 [declarações]
</style>
```

# Cascata

- Caso contrário, em regras conflitantes, o estilo externo sobrescreverá o estilo incorporado

<style>

[declarações]

</style>

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

# Cascata

- A quarta regra válida é aquela que o desenvolvedor marcar como *!importante*
- Independente da posição da declaração ou dos desejos do navegador e do usuário, essa é a regra que vai ter o maior valor no caso de regras conflitantes

# Cascata

- A marcação *!importante* deve aparecer antes do “;” da declaração:

```
p{
color: #0F0 !important;
}
```

# Cascata

- Por último, a regra mais importante de todas, aquela que sobrescreve qualquer uma das regras anteriores em caso de regras conflitantes, é a marcada como *!importante pelo usuário*
- Nesse caso, o usuário deve criar sua própria folha de estilos marcando com *!importante* as regras com valor imperativo sobre as outras e também indicar ao navegador sua posição

# Cascata

- Dessa forma, ele sobrescreverá, em caso de conflito, todas as regras indicadas pelo navegador ou desenvolvedor (raro)
  - No IE11, o usuário *Ferramentas, Opções da Internet*, aba *Geral*, botão *Acessibilidade*, ativar a caixa *Formatar documentos usando minha folha de estilos* e localizar o arquivo com as modificações
  - No Firefox, o usuário deve instalar um add-on chamado *User Style Manager* e seguir suas instruções
  - No Chrome o usuário deve usar uma das extensões: *Stylish* ou *Stylebot*

# Especificidade

- Para ser realmente preciso em uma determinada regra, deve-se conhecer o conceito de especificidade de cada seletor
- É com base na especificidade que o desenvolvedor consegue escolher a forma mais precisa o tipo de seletor em função do resultado desejado

# Especificidade

- Quando há um conflito (sobreposição) de regras, o desempate é decidido primeiro pelas regras de importância
- 1º *!important* marcado pelo usuário
- 2º *!importante* pelo desenvolvedor
- E por fim a especificidade do seletor
- A especificidade estabelece uma relação de autoridade entre os tipos de seletores

# Especificidade

- Como calcular a especificidade:
  1. Conte 1 se a declaração é de um estilo *inline* (aquela que usa o atributo *style*); caso contrário, conte 0
  2. Conte o número de seletores ID
  3. Conte o número de seletores de classe, de pseudoclasse e de atributo
  4. Conte o número de seletores de elemento e de pseudoelemento

# Especificidade

- Concatene (junte, sem somar) os quatro números ABCD (em um sistema numérico de base alta) e você terá o valor da especificidade

# Especificidade

Seletor	A estilo inline [1], caso contrário [0]	B seletores tipo ID	C classes, pseudoclasses* e atributos	D elementos e pseudoelementos**	ABCD
* {}	0	0	0	0	0000
p {}	0	0	0	1	0001
p:first-line {}	0	0	0	1+1	0002
ul li {}	0	0	0	1+1	0002
ul ol+li {}	0	0	0	1+1+1	0003
li a:link {}	0	0	1	1	0011
h1 + *[rel=up] {}	0	0	1	1	0011
ul ol li.carro {}	0	0	1	1+1+1	0013
.carro li.novo{}	0	0	2	1	0021
#chamada {}	0	1	0	0	0100
style="color:#00F;"	1	2	0	0	1000

\* Exemplos de pseudoclasses: *e:first-child* (primeiro filho), *a:link* (link), *a:visited* (link visitado), *e:hover* (mouse acima do elemento), *e:focus* (elemento quando recebe o foco), *e:lang* (idioma).

\*\* Exemplos de pseudoelementos: *e:first-line* (primeira linha), *e:first-letter* (primeira letra), *e:before* (aplicar conteúdo antes), *e:after* (aplicar conteúdo depois).

# Especificidade

- Conforme o seletor escolhido para um determinado conjunto de regras de estilo, podemos ter valores mais altos ou mais baixos de especificidade
- Olhando a tabela, podemos verificar que o seletor menos específico é o universal (\*) e o mais específico é o de atributo de estilo (*style="propriedade:valor;"*)

# Especificidade

- Isso equivale dizer que seletores de atributo de estilo mandam mais que qualquer um outro
- Em seguida, mandam mais os seletores de ID
- Depois os de classe
- E por último os de elemento

# Especificidade

- Não existe uma regra para a criação de folhas de estilos, muito menos para a escolha dos seletores, existem apenas boas práticas, como:
  - Crie primeiro os estilos mais genéricos e depois os mais pontuais
  - Trabalhe sempre com a estrutura semântica do HTML5, indicando para cada tipo de conteúdo o seu elemento mais apropriado

# Especificidade

- Não deixe elemento de linha, como uma imagem, “flutuando” sem um elemento de bloco, como um parágrafo, *section*, *header* ou *figure*
- Crie elementos *div* apenas para fins de layout ou interação, mas seja econômico
- Não use sempre seletores de classe, ou sempre seletores de ID, varie suas escolhas com base nas necessidades do projeto

# Especificidade

- Faça primeiro o projeto de seu layout, e em seguida crie os estilos necessários
- Comente sempre o seu código CSS, utilizando /\*comentário\*/
- Mantenha um documentação como guia de estilos para sempre saber qual estilo usar em cada situação e, dessa forma, manter a consistência semântica e visual

# Especificidade

- Vamos fazer alguns exemplos para entender melhor todos esses conceitos
  - Cascata 1,2,3,4
  - Especificidade 1,2,3
  - Herança 1,2
- `atividades\cap05\teoria\conceitos_css`

# Atividade 2 – Estilos em tabela

- Objetivos:
  - Aprimorar a combinação de seletores, declarações de estilos e marcação semântica
  - Manipular estilos em tabelas
- Tarefas
  - Trabalhar com uma única folha de estilo externa
  - Criar e manipular seletores de ID
  - Combinar diversos seletores
  - Trabalhar com estilo em tabela

# Atividade 2 – Estilos em tabela

1. Abra o arquivo *produtos.html*
2. Insira o link da folha de estilos externa

# Identificando a página

- Para tornar a seleção de estilos mais pontual e manter sua folha de estilos organizada, uma prática comum é rotular todos os elementos de uma página com ID

# Identificando a página

1. Vamos atribuir esse identificador no elemento que é “pai” de todos os outros elementos: *body*

```
<body id="produtos">
```

2. No elemento *table*, remover o atributo *border*
3. Salvar o arquivo

# Seletor de ID

- Incluir os seguintes estilos a tabela
  - \* página produtos - tabela \*/

```
#produtos table {
 largura: 600px;
 margem: 50px auto 30px;
 alinhamento do texto: center;
}
```

# Bordas na tabela

- Uma tabela tem dois tipos de bordas: uma borda externa, definida pelo elemento *table*, e as bordas internas, criadas a partir das próprias células por meio dos elementos *td* e *th*

# Bordas na tabela

- Adicionar os elementos abaixo aos elemento *th* e *td* da tabela
- Cor da borda: #999
- Estilo da borda: sólido
- Largura da borda: 1px
- Espaçamento: 5px;

# Bordas na tabela

Produtos			
Porte	Marca		Preço
Pequeno	Vestibulum		R\$ 5,00
	Nunc		R\$ 4,80

# Bordas na tabela

- Para remover os espaços entre as bordas é necessário usar a propriedade:  
*border-collapse: collapse;*

Produtos			
Porte	Marca		Preço
Pequeno	Vestibulum		R\$ 5,00
	Nunc		R\$ 4,80

# Título da tabela

- Alterar o título da tabela com os seguintes valores:
  - ✓ cor de fundo: #999;
  - ✓ cor das letras:#fff;
  - ✓ espaçamento: 15px 0;
  - ✓ tamanho da fonte: 1.5em;

# Título da tabela

Produtos			
Porte	Marca		Preço
Pequeno	Vestibulum		R\$ 5,00
	Nunc		R\$ 4,80

# Seletor de atributo em tabela

- Atribua a linha com *colspan* os seguintes atributos:
  - ✓ Tipo de fonte: itálico;
  - ✓ Fonte negrito;
  - ✓ cor: #999;

# Seletor de atributo em tabela

Grande	Lacinia		R\$ 6,90
	Sapien		R\$ 3,90
<i>Consulte nossos preços para grandes quantidades.</i>			

# Seletor do tipo pseudoclasse em tabela

- Alterar a cor de fundo da linha quando o mouse estiver em cima
- ✓ Cor de fundo: #FEFF76

# Seletor do tipo pseudoclasse em tabela

Produtos			
Porte	Marca		Preço
Pequeno	Vestibulum		R\$ 5,00
	Nunc		R\$ 4,80

# Atividade 3 – Estilos em formulários simples

- Objetivos:
  - Dar estilo a elementos de formulários simples
  - Manipular propriedades de visualização e flutuação
- Tarefas
  - Trabalhar com uma única folha de estilo externa
  - Criar e manipular seletores de ID e elementos de formulários
  - Trabalhar com estilos de formulário
  - Trabalhar com estilos especiais: cantos arredondados e sombra

# Atividade 3 – Estilos em formulários simples

- Abra o arquivo *boletim.html* e adicione a folha de estilo externa
- Adicione o identificador “boletim-f” ao elemento *body*

# Imagen de fundo em elemento de bloco

- Adicione ao elemento `h2` as seguintes propriedades de forma resumida
  - Cor de fundo: `#C0262C`
  - Imagem de fundo: `back-bolinha.gif`
  - Repetir do topo centralizado
  - Cor do letra: `#fff;`
  - Espaçamento entre as letras: `.1em;`
  - Espaçamento: `7px;`
  - Largura: `316px;`

# Imagen de fundo em elemento de bloco

## Boletim mensal

---

Assine já!

# Cor de fundo em formulários

- Adicione ao elemento *form* as seguintes propriedades
  - cor de fundo: #999;
  - margem: 30px 0;
  - espaçamento: 20px;
  - raio nas bordas: 0 15px;

# Cor de fundo em formulários

**Nome:**

**Email:**

# Transformações em um botão

- Alguns estudos já foram feitos comprovando que o usuário tende a perceber de forma mais efetiva um botão com cantos arredondados e um certa dose de profundidade, criada pelo efeito gradiente e sombra

# Transformações em um botão

- Adicione ao elemento *input* do tipo botão as seguintes propriedades:
  - cor de fundo: #C0262C;
  - retirar a borda padrão do botão
  - raio na borda: 30px;
  - tamanho da fonte: 1.2em;
  - cor da fonte: #fff;
  - espaçamento: 10px 30px;
  - margem: 20px 0;
  - sombra de texto: 2px 2px 5px #000;  
[\(https://www.maujor.com/tutorial/css3-text-shadow.php\)](https://www.maujor.com/tutorial/css3-text-shadow.php)
  - cursor: pointer;

# Transformações em um botão

- Adicione ao elemento *input* do tipo botão as seguintes propriedades:
  - Background-image: linear-gradient(to bottom, #6b1516 0%,#c0262c 30%,#c0262c 70%,#6b1516 100%);

<https://tableless.com.br/como-usar-gradient-no-css-de-forma-consciente/>

- cursor: ponteiro (mão);

[https://www.w3schools.com/cssref/playit.asp?filename=playcss\\_cursor&preval=pointer](https://www.w3schools.com/cssref/playit.asp?filename=playcss_cursor&preval=pointer)

# Transformações em um botão

**Nome:**

**Email:**

Enviar

# Estilizando elementos de formulário

- Alguns elementos são característicos de formulários: as etiquetas (elemento *label*) e a caixa de entrada (elemento *input*)
- O primeiro é tratado como um texto comum, mas o segundo recebe características especiais, como alteração da caixa e do conteúdo, incluindo o conteúdo temporário criado pelo atributo *placeholder*

# Estilizando elementos de formulário

- Adicione ao elemento *input* do nome e do endereço as seguintes propriedades:
  - Cor de fundo: transparente;
  - Retire a borda padrão
  - Borda inferior: #000 solid 1px;
  - Tamanho da fonte: 1.2em;
  - Cor: #fff;
  - Largura: 800px;

# Estilizando elementos de formulário

- Agora verifique se ocorre alguma diferença na aparência do input nos navegadores:
  - Chrome
  - Firefox
  - IE

# Estilizando elementos de formulário

- Logo, vamos configurar a aparência e personalizar para cada navegador, para isso devemos declarar no CSS os seguintes códigos:
- Abaixo do seletor *form* e antes do seletor *input*
  - ::-webkit-input-placeholder { /\* Chrome/Safari/Opera \*/}
  - ::-moz-placeholder { /\* Firefox \*/}
  - :-ms-input-placeholder { /\* IE \*/}

# Estilizando elementos de formulário

- Atribuir aos 3 elementos mencionados as seguintes características:
  - cor: #ddd;
  - estilo da fonte: italic;
  - tamanho da fonte: .8em;

# Propriedade CSS *display*

- Permite definir a maneira como determinado elemento HTML deve ser renderizado
- **display: block**
  - faz com que o elemento HTML seja renderizado como bloco, tal como parágrafos e cabeçalhos são
  - Um bloco contém espaços em branco tanto em cima como em baixo e não permite outros elementos HTML ao lado, exceto quando tiver sido declarado (declarar *float* para o elemento próximo ao bloco)

# Estilizando elementos de formulário

- Atribuir ao elemento *label* as seguintes características:
  - transformar o elemento rótulo em bloco;
  - margem superior: 7px;
  - flutuar à esquerda;
  - largura: 60px;

# Propriedade CSS *display*

- Esta propriedade permite que você defina a maneira como determinado elemento HTML deve ser renderizado
  - block
  - inline
  - none
  - inline-block
  - list-item
  - run-in
  - table

# Propriedade CSS *display*

- Vamos criar um exemplo para cada um dos elementos citados:
  - block
  - inline
  - none
  - inline-block
  - list-item
  - run-in
  - table

# Flutuação

- Ao utilizar a propriedade de flutuação, o elemento indicado sai do fluxo normal do conteúdo deixando de ocupar seu espaço na posição original e passando a flutuar para o lado esquerdo ou direito, até que sua borda externa toque a borda da caixa que contém ou a borda de outra caixa flutuante
- <https://developer.mozilla.org/pt-BR/docs/Web/CSS/clear>

# Flutuação de um único elemento

- Vamos criar nossa primeira página com um único elemento *float*, que está flutuando a direita do conteúdo da página dentro da seção cabeçalho
- Dar o nome do arquivo de *float-bloco.html*
- *Referências: largura: 270px; espaçamento: 5px 15px 20px; margem: 0 0 10px 30px; raio da borda esquerda inferior 25px; cor de fundo: #e5c5b0;*

# Flutuação de mais de um elemento

- Vamos criar esta página com mais de um elemento *float*, que está flutuando a direita do conteúdo da página dentro da seção cabeçalho
- Dar o nome do arquivo de *float-bloco-2.html*
- *Referências: largura do texto: 330px; flutuação: esquerda; margem-direita: 40px; cor de fundo do título: #bd4e15; cor do título: #fff; espaçamento: 5px; tamanho da fonte: 2em;*

# Flutuação com elementos de imagem

- Vamos adicionar a página as imagens do livro que fazem referência ao subtítulo do texto com seus respectivos nomes
- Dar o nome do arquivo de *float-bloco-3.html*
- *Referências: espaçamento à direta: 20px ao elemento img;*

# A propriedade *clear* (limpar)

- A propriedade *clear* especifica se um elemento pode ter elementos flutuantes ao seu lado ou se devem ser movidos para abaixo dele (*clear*)
- Dar o nome do arquivo de *float-bloco-4.html*
- *Referências: não permitir que o conteúdo principal tenha elementos flutuando a sua direita*

# A propriedade *overflow* (transbordar)

- A propriedade *overflow* especifica quando o conteúdo de um elemento de nível de bloco deve ser cortado, exibido com barras de rolagem ou se transborda do elemento
- Um último **cuidado** é necessário quando se usar a propriedade de flutuação com cor de fundo no elemento que contém os que estão flutuando

# A propriedade *overflow* (transbordar)

- Ao aplicar a propriedade *float*, o elemento flutuará até que sua borda externa toque a borda da caixa que o contém
- Isso que dizer que o elemento externo não reconhece mais todo o conteúdo das caixas flutuantes, mas apenas sua borda
- Para que esse elemento reconheça esse conteúdo, é necessário acrescentar a propriedade *overflow*

# A propriedade *overflow* (transbordar)

- Vamos aplicar as seguintes propriedades:
  - transbordar: auto;
  - adicionar cor de fundo a todo o conteúdo principal da página: #e3dacd;
  - adicionar margem no parágrafo: 20px;
- Dar o nome do arquivo de *float-bloco-5.html*

# Atividade 4 – Estilos em formulários complexos

- Objetivos:
  - Dar estilo a elementos de formulários complexos
  - Manipular propriedades de visualização e flutuação
- Tarefas:
  - Criar e manipular seletores de ID e de elementos de formulários
  - Trabalhar com flutuação
  - Trabalhar com sobreposição de regras em formulários e outros elementos
  - Trabalhar com flutuação em imagem

# Atividade 4 – Estilos em formulários complexos

- Abra o arquivo: *banho e tosa.html*
- Insira o link

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

antes da tag </head>
- Acrescente o identificador *banhoEtosa* ao elemento *body*
- Salve o arquivo
- Visualize o resultado dessa edição no navegador

# Atividade 4 – Estilos em formulários complexos

- O formulário atual recebe todo o layout indicado para o formulário anterior, que era bem mais simples
- Para fazer alterações no formulário atual sem alterar o anterior, trabalharemos com uma combinação de seletores que envolve sempre o identificados da página “Banho e Tosa”

# Agrupar elementos

- Após o fechamento do último seletor, acrescentar o comentário:  
/\* página banho e tosa - formulário \*/
- Acrescente as seguintes alterações:
  - Ao elemento de agrupamento:
    - cor de fundo: #fff;
    - retirar borda atual;
    - raio na borda: 0 15px;
    - margem inferior: 20px;

# Agrupar elementos

- Acrescente as seguintes alterações:
  - Ao elemento legenda do agrupamento:
    - cor de fundo: #C0262C;
    - cor do texto: #fff;
    - fonte: negrito;
    - espaçamento entre as letras: .05em;
    - espaçamento: 10px;
    - margem esquerda: -13px;

# Agrupar elementos

## Identificação do dono do animal

Nome:

Endereço:

CEP:

Email:

Tel/Cel:

# Itens de entrada

- Agora são os rótulos e os itens dos formulários que precisam ser alterados, agora serão utilizados os seletores de atributo, além do identificador

# Itens de entrada

- Acrescente as seguintes alterações:
  - Ao elemento rótulo:
    - renderize-o como bloco;
    - margem superior: 7px;
    - flutuar à esquerda;
    - largura: 115px;

# Itens de entrada

- Acrescente as seguintes alterações:
  - Ao elemento de entrada de texto:
    - retirar a borda;
    - borda inferior: #000 solid 1px;
    - tamanho da fonte: 1.2em;
    - cor do texto: #000;
    - largura: 770px;

# Itens de entrada

## Identificação do dono do animal

Nome: *Digite o nome completo do dono do animal* \_\_\_\_\_

Endereço: *Digite o endereço* \_\_\_\_\_

CEP: *XXXXX-XXX* \_\_\_\_\_

Email: *Digite o endereço de email* \_\_\_\_\_

Tel/Cel: *99 99999-9999* \_\_\_\_\_

# Entrada de texto de área

- Ao elemento de entrada de texto do tipo área:
  - largura: 770px;
  - altura: 100px;
  - tamanho da fonte: 1.2em;
  - borda: #000 solid 1px;

# Elementos de formulário e classe

- Em formulário mais complexos é bom tomar cuidado com os elementos *label* vinculados a elementos do tipo *radio* e *checkbox*
- Deve-se escolher, entre os tipos de seletor, aquele que melhor atende à estrutura do seu código

# Elementos de formulário e classe

- Nos elementos *label* e *checkbox* referentes aos itens de tipo *radio*, inclua a classe *labelSimples*
- Acrescente as seguintes alterações:
  - Ao elemento de rótulo através da classe que criamos:
    - renderize-o como elemento em linha;
    - remova a flutuação;

# Elementos de formulário e classe

- Para quem criou parágrafo para cada um dos elementos ao invés de usar o <br>, adicione as seguintes alterações:
  - Ao elemento parágrafo:
    - alinhamento do texto: esquerda
  - Ao elemento p com pseudoclasse de alteração da primeira linha faça as seguintes alterações:
    - Tamanho de fonte: inherit;
    - Variação de fonte: inherit;
    - peso da fonte: inherit;
    - estilo de fonte: inherit;

# Elementos de formulário e classe

- Adicionando o valor *inherit* as propriedades dos elementos, ele volta a herdar do próprio elemento as propriedades originais de tamanho, variação, peso e estilo

# Elementos de formulário e classe

## Animal

Nome: *Digite o nome do animal*

Porte do cão:  pequeno  médio  grande

Tipo de tosa:

*Digite alguma informação sobre o animal se necessário*

Observações:

## Horário - (Não atendemos de domingo e segunda-feira)

Indique os dias da semana de sua preferência:

Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira  Sábado

Primeira opção de horário:

Segunda opção de horário:

# Sobrepondo regras

- Assim como foi feito no passo anterior, é possível sobrescrever qualquer regra manipulando os conceitos de herança, cascata e especificidade

# Sobrepondo regras

- Vamos adicionar a imagem do postit.jpg e colocar o subtítulo do h2 dentro do postit
  - adicionar imagem *postit.jpg*, colocar a cor #FEF84F e alinhar no topo à esquerda;
  - cor do subtítulo: #000;
  - Espaçamento: 35px;
  - Largura: 207px;
  - Altura: 115px;
  - Tamanho da fonte: 1.3125em;
  - Espaçamento das letras: .07em;

# Agendamento para banho e tosa

---

Faça o seu  
agendamento  
com pelo menos  
48 horas de  
antecedência

# Sobrepondo regras

- Na primeira vez que o elemento *h2* foi estilizado, foi usado apenas o seletor do elemento
- Agora foram utilizado mais dois seletores: o do identificador e o do elemento *main*
- Isso aumentou a especificidade dessa regra e alterou o estilo
- Outro benefício do uso do identificador é que apenas esse título foi alterado, pois apenas eles é filho de um identificador “banhoEtosa”

# Flutuação em imagem

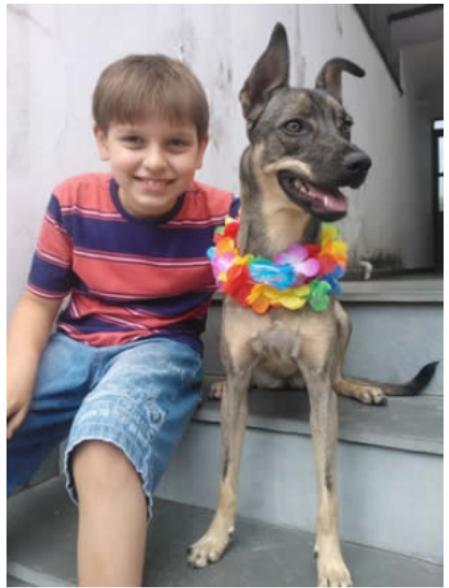
- Abra o arquivo *quem-somos.html*
- Acrescente o identificador *quemSomos* ao elemento *body*
- Adicione o comentário abaixo após o fechamento do último seletor

```
/* página quem somos */
```

# Flutuação em imagem

- Vamos editar a exibição da imagem meuamigocao.jpg
  - Vamos adicionar as seguintes mudanças:
    - flutuar à esquerda;
    - margem: 0 20px 20px 0;
    - espaçamento: 10px;
    - borda: solida #C0262C fina;
    - cor de fundo: #fff;

# Flutuação em imagem



## Quem somos

### O QUE NÓS ACREDITAMOS

Na Meu amigo cão, o nosso sucesso como uma organização é conduzida pelo cuidado que temos com nossos animais.

*CRAS SEMPER MAGNA IN VELIT TEMPUS TRISTIQUE QUIS NON LOREM. MAURIS VITAE FRINGILLA SAPIEN, SED volutpat elit. Fusce molestie euismod odio, ut blandit nulla bibendum ac. Fusce posuere turpis sed sem tincidunt, sed auctor tellus faucibus. Proin ultrices enim vitae ipsum luctus, sed malesuada orci ornare. Vivamus eget molestie magna, sed aliquam risus. Aliquam adipiscing laoreet felis, eget tincidunt purus sollicitudin sit amet. Nulla facilisi.*

*DONEC DAPIBUS NIBH EU IPSUM ELEMENTUM, NON CONSECTETUR MASSA FACILISIS. NAM COMMODO AT LEO non commodo. In blandit arcu sit amet sem convallis, rutrum gravida quam elementum. Cras accumsan orci vel lectus blandit, at dictum tellus rhoncus. Etiam vitae urna nec leo facilisis auctor a quis lorem. Integer dolor urna, luctus quis consequat vitae, vestibulum non lectus. In turpis mi, tincidunt non iaculis ac, feugiat eget purus.*

# Atividade 5 – Estruturação do conteúdo

- Objetivos:
  - Dar estrutura à página usando os elementos HTML5 estruturais
- Tarefas
  - Criar e manipular seletores de ID e de elementos estruturais
  - Combinar diversos seletores
  - Trabalhar com as propriedades de espaçamento, largura e flutuação
  - Ajustar todas as páginas do projeto

# Atividade 5 – Estruturação do conteúdo

- Abra o arquivo *index.html*
- Acrescente o identificador *inicial* ao elemento *body*

# Identificando a área principal

- Nas aulas anteriores, criamos diversos estilos sem a preocupação de elementos estruturais (*main, header, footer...*) que foram adicionados posteriormente nas aulas seguintes
- Agora é hora de organizar a folha de estilos para que tais estilos cumpram a função de organizar seu conteúdo, não apenas de forma semântica, mas também visivelmente
- Na folha de estilo *principal.css* inclua o comentário abaixo:  
`/* textos da área principal */`

# Identificando a área principal

- Nos seletores de *h1* a *hr*, inclua o elemento *main*
  - main h1 {
  - main h2 {
  - main p {
  - main p:first-line {
  - main hr {
- No seletor *p:first-line*, inclua *h2*
  - main h2+p:first-line {

# Identificando o corpo da área estrutural

- Na folha de estilos, inclua o comentário:  
/\* estrutura \*/
- Abaixo acrescente as seguintes propriedades ao elemento *body*:
  - largura: 1000px;
  - margem esquerda: auto;
  - margem direita: auto;

# Cabeçalho: título principal

- Vamos adicionar as seguintes propriedades:
  - largura: 240px;
  - flutuar à esquerda;
  - imagem de fundo: meu-amigo-cao-petshop-header.png;
  - posição da imagem de fundo: center center;
  - altura: 200px;
  - indentação do texto: -1000px; (irá sumir da tela)
  - margem inferior: 0px;

# Cabeçalho: título principal



**PETSHOP**  
**meuamigocão**

[Assine o boletim mensal](#)

nos acompanhe:

- [Home](#)
- [Quem somos](#)
- [Produtos](#)
- [Banho e Tosa](#)
- [Curiosidades](#)



**cantinho feliz**  
hotel para cachorro  
[www.hotelpetcantinhofeliz.com](http://www.hotelpetcantinhofeliz.com)  
55 11 1234-5678 / 9 9999 9999

# Cabeçalho: área de anúncio

- Dentro do elemento *header*, ao redor do elemento imagem *anuncio-cantinho-feliz-h.jpg*, acrescente o elemento *div* com identificador *anuncio*
- Assim, você acrescenta uma área para inserir anúncios e manipular seu posicionamento
- Como é área visual, foi utilizado o elemento *div*, que não tem qualquer valor semântico

# Cabeçalho: área de anúncio

- Agora vamos alterar suas propriedades no css:
  - largura: 510px;
  - flutuar à esquerda
  - margem: 80px auto auto;
  - espaçamento superior: 0px;

# Cabeçalho: área de anúncio

- O elemento *div* recebeu as propriedades de flutuação e largura para poder assumir sua localização ao lado do logotipo



**PETSHOP**  
**meuamigocão**



**cantinho feliz**  
hotel para cachorro  
[www.hotelpetcantinhofeliz.com](http://www.hotelpetcantinhofeliz.com)  
55 11 1234-5678 / 9 9999 9999

[Assine o boletim mensal](#)

nos acompanhe:

- [Home](#)
- [Quem somos](#)
- [Produtos](#)
- [Banho e Tosa](#)
- [Curiosidades](#)

# Cabeçalho: mídias sociais

- Ao elemento seção dentro do elemento cabeçalho, vamos adicionar a seguinte propriedade
  - espaçamento superior: 70px;

# Cabeçalho: mídias sociais

- Ao elemento link com o texto “Assine o boletim mensal”, vamos adicionar as seguintes propriedades:
  - borda: sólida #C0262C fina;
  - espaçamento: 15px;
  - raio nas bordas: 15px;
  - cor: #C0262C;
  - retirar o formato padrão do link
  - tamanho da fonte: 1.1em;

# Cabeçalho: mídias sociais

- Ao elemento parágrafo com o texto “nos acompanhe”, vamos adicionar as seguintes propriedades:
  - cor: #999999;
  - margem: 33px auto 7px;

# Cabeçalho: mídias sociais

- Ao elemento figura que contém as imagens de mídias sociais, vamos adicionar as seguintes propriedades:
  - largura: 100%;
  - margem: 0px;

# Cabeçalho: mídias sociais

- Ao elemento link que está dentro da tag de figuras que contém as imagens de mídias sociais, vamos adicionar as seguintes propriedades:
  - largura: 100%;
  - margem: 0px;

# Cabeçalho: mídias sociais

- Ao elemento link primeiro filho que está dentro da tag de figuras que contém as imagens de mídias sociais, vamos adicionar as seguintes propriedades:
  - espaçamento: 25px 5px 0 0;

# Cabeçalho: mídias sociais

- Ao elemento link último filho que está dentro da tag de figuras que contém as imagens de mídias sociais, vamos adicionar as seguintes propriedades:
  - espaçamento: 25px 0 0 5px;

# Cabeçalho: mídias sociais



**cantinho feliz**  
hotel para cachorro

[www.hotelpetcantinhofeliz.com](http://www.hotelpetcantinhofeliz.com)  
55 11 1234-5678 / 9 9999 9999

Assine o boletim mensal

nos acompanhe:



# Cabeçalho: mídias sociais

- A própria estrutura semântica auxilia na criação visual
- Observe como os elementos *header*, *section* e *figure* são combinados para que não seja necessário acrescentar marcações não semânticas à estrutura
- Somado a isso, o conhecimento de seletores complexos como o de atributo e o de pseudoclasse também contribui nesse sentido

# Navegação

- Na folha de estilos, inclua o comentário:  
/\* estrutura de navegação \*/
- Abaixo acrescente as seguintes propriedades ao elemento *nav*:
  - cor de fundo: #000;
  - transbordar: auto;
  - mover o elemento para baixo, para limpar ambos os elementos flutuantes da esquerda ou direita;
  - margem superior: 10px;

# Navegação

- Abaixo acrescente as seguintes propriedades ao elemento *lista não ordenada* da navegação:
  - retirar o estilo padrão da lista;
  - espaçamento esquerdo: 0px;
  - margem superior: 1px;

# Navegação

- Abaixo acrescente as seguintes propriedades ao elemento item da *lista não ordenada da navegação*:
  - flutuar à esqueda;

# Navegação

- Abaixo acrescente as seguintes propriedades ao elemento link da *lista não ordenada da navegação*:
  - cor: #fff;
  - retirar padrão de link atual;
  - tamanho da fonte: 1.2em;
  - espaçamento: 8px 50px;
  - altura da linha: 2em;

# Navegação

- Abaixo acrescente as seguintes propriedades ao elemento link com href da *lista não ordenada da navegação*:
  - cor de fundo: #C0262C;

# Estrutura da página inicial

- Vamos organizar as imagens adicionadas (anuncio\_pet1.jpg, anuncio\_pet2.jpg, anime-index.jpg) da estrutura da página com o conteúdo principal
- Na folha de estilos, inclua o comentário:  
/\* estrutura página inicial \*/

# Estrutura da página inicial

- Altere os elementos de publicidade com as seguintes propriedades:
  - Largura: 230px;
  - Flutuar à esquerda
  - Margem superior: 30px;

# Estrutura da página inicial

- Altere os elementos do conteúdo principal da página com as seguintes propriedades:
  - Largura: 490px;
  - Flutuar à esquerda;
  - Margem: 30px 25px 10px;

# Estrutura da página inicial



**cantinho feliz**  
hotel para cachorro

[www.hotelpetcantinhofeliz.com](http://www.hotelpetcantinhofeliz.com)  
55 11 1234-5678 / 9 9999 9999

Assine o boletim mensal

nos acompanhe:

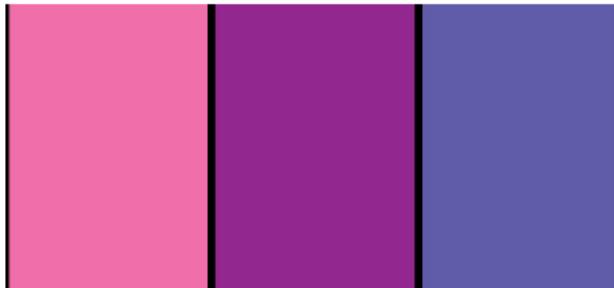
Home

Quem somos

Produtos

Banho e Tosa

Curiosidades



Atendimento:

## O melhor para o nosso amigo cão

Quisque venenatis urna ac nulla. Maecenas quis turpis nec nulla auctor mattis. Curabitur mi libero, scelerisque eu, cursus semper, tempor quis, ante. Etiam iaculis, nisl eu imperdiet tincidunt, neque lacus ornare leo, non ultrices arcu justo ut

# Rodapé

- Acrescente ao código CSS a formatação ao elemento rodapé das páginas:
  - Mova o element para baixo de forma que não tenha elementos em seu lado direito ou esquerdo;
  - borda superior: 7px solid #999;
  - cor de fundo: #000;
  - cor do texto: #fff;
  - alinhamento do texto: center;
  - espaçamento: 10px;

# Rodapé

- Acrescente ao código CSS a formatação ao elemento parágrafo presente no rodapé das páginas:
  - margem: 0px;

# Atividade 6 - Acabamento

- Objetivos:
  - Fazer o acabamento das páginas estruturais
  - Manipular propriedades de posicionamento
- Tarefas:
  - Ajustar todas as páginas do projeto
  - Criar e manipular seletores
  - Trabalhar com propriedades de posicionamento

# Ajuste de estrutura em todas as páginas

1. Abra o arquivo *index.html*
2. Selecione da abertura do elemento *header* até o fechamento do elemento *nav*
3. Copie (Ctrl + C)
4. Abre a página *quem-somos.html* e cole (Ctrl + V) o conteúdo copiado imediatamente antes do elemento *main*
5. Remova o elemento *a* que contém o link provisório para a página *index.html*

# Ajuste de estrutura em todas as páginas

6. Verifique se o elemento *footer* e seu conteúdo estão no rodapé da página
7. Salve seu trabalho e visualize a página no navegador

# Ajuste de estrutura em todas as páginas

6. Verifique se o elemento *footer* e seu conteúdo estão no rodapé da página
7. Salve seu trabalho e visualize a página no navegador

# Ajuste de estrutura em todas as páginas



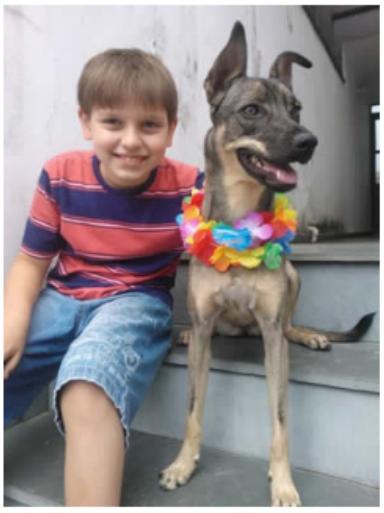
**cantinho feliz**  
hotel para cachorro  
[www.hotelpetcantinhofeliz.com](http://www.hotelpetcantinhofeliz.com)  
55 11 1234-5678 / 9 9999 9999

Assine o boletim mensal

nos acompanhe:

f t YouTube p g+

Home Quem somos Produtos Banho e Tosa Curiosidades



## Quem somos

O que nós acreditamos  
Na **Meu amigo cão**, o nosso sucesso como uma organização é conduzida pelo cuidado que temos com nossos animais.

Cras semper magna in velit tempus tristique quis non lorem. Mauris vitae fringilla sapien, sed volutpat elit. Fusce molestie euismod odio, ut blandit nulla bibendum ac. Fusce posuere turpis sed sem tincidunt, sed auctor tellus faucibus. Proin ultrices enim vitae ipsum luctus, sed malesuada orci ornare. Vivamus eget molestie magna, sed aliquam risus. Aliquam adipiscitur laoreet felis, eget tincidunt purus sollicitudin sit amet. Nulla facilisi.

Donec dapibus nibh eu ipsum elementum, non consectetur massa facilisis. Nam commodo at leo non commodo. In blandit arcu sit amet sem convallis, rutrum gravida quam elementum. Cras accumsan orci vel lectus blandit, at dictum tellus rhoncus. Etiam vitae urna nec leo facilisis auctor a quis lorem. Integer dolor urna, luctus quis consequat vitae, vestibulum non lectus. In turpis mi, tincidunt non iaculis ac, feugiat eget purus.

# Ajuste de estrutura em todas as páginas

8. Repita os passos de 1 a 7, colocando os conteúdo de *header* e *nav* em cada uma das páginas do projeto (*produtos.html*, *banho-e-tosa.html*, *curiosidades.html*, *boletim.html* e *obrigado.html*), lembrando de remover o link provisório para a página “Index” e verificando se o rodapé e o seu conteúdo estão no final da página

# Ajuste de estrutura em todas as páginas

## 9. Na página *curiosidades.html*:

- acrescente o identificador ao elemento *body* com o nome de “curiosidades”
- Acrescente o seletor *#curiosidades main* antes de todos os seletores que estão incorporados a essa página (h2, figure, figcaption, ul li, dt, dd, ol, ol li:before, ul li a, ul li a:hover, dd a[target=\_blank], dd a[target=\_blank]:hover, dd a[href="#topo"], dd a[href="#topo"]:hover

# Ajuste de estrutura em todas as páginas

10. Nas páginas *boletim.html* e *obrigado.html*, não há o link provisório para a página “Index”

11. Na página *obrigado.html*:

- Insira o link da folha de estilos externa
- Acrescente o identificador ao elemento *body* com o nome de “obrigado”
- Salve as alterações e navegue por todas as páginas observando como a estrutura se comporta com os respectivos conteúdos

# Posicionamento de elementos

1. Abra a página *obrigado.html*
2. Antes do parágrafo que contém a imagem *anuncio\_pet3.jpg*, acrescente a imagem *lacinho.png* e coloque-a dentro de uma `<div>` com identificador “lacinho”
3. Antes da *div* que foi adicionar, abra uma nova *div* com identificador “destaqueObrigado”
4. Feche a *div* aberta anteriormente antes do elemento de fechamento `</main>`

# Posicionamento de elementos

**Seu formulário foi  
enviado.**

*NOSSA EQUIPE IRÁ CONFERIR SUA SOLICITAÇÃO E ENVIARÁ UMA CONFIRMAÇÃO PARA SEU EMAIL.*

Aproveite agora mesmo a nossa dica em destaque:



*Lorem ipsum dolor sit amet,*

*consectetur adipiscing elit. Aliquam fringilla lacinia ultricies. Nam elit lectus, interdum eu dictum et, hendrerit in tellus. Fusce tempor fringilla massa vitae consequat. Vivamus aliquet, neque sed tempus rhoncus, nisl turpis malesuada ipsum, vel imperdiet arcu mauris ut metus.*

*Aliquam luctus venenatis porttitor. Vivamus vitae aliquam arcu, sed auctor lacus. Fusce placerat, odio at congue suscipit, lectus purus accumsan justo, at dapibus tortor risus nec libero. Nullam orci arcu, lacinia dignissim diam et, adipiscing tempus velit. Vivamus nec interdum sem, ac ullamcorper arcu.*

*Morbi risus diam, cursus ac dapibus sit amet, feugiat vitae mauris.*

# Posicionamento de elementos

- Na folha de estilos, inclua o comentário:  
`/* página obrigado */`
- Abaixo acrescente as seguintes propriedades ao elemento parágrafo que contém a imagem(anuncio\_pet3.jpg):
  - *Flutuar: esquerda;*
  - *Margem: 0 20px 20px 0;*
  - *Espaçamento: 10px;*
  - *borda: sólida #C0262C fina;*
  - *Cor de fundo: #fff;*

# Posicionamento de elementos

- Abaixo acrescente as seguintes propriedades ao elemento div destaqueObrigado:
  - *Posição: relativa;*
  - *borda: solida #C0262C 1px;*
  - *Raio da borda: 0 25px;*
  - *Espaçamento: 5px 25px;*
  - *Margem: 20px 0;*

# Posicionamento de elementos

- Aos elementos com id obrigado e id lacinho adicione as seguintes propriedades:
  - *position: absolute;*
  - *top: -20px;*
  - *right: -20px;*
  - *-webkit-transform: rotate(30deg);*
  - *-moz-transform: rotate(30deg);*
  - *-ms-transform: rotate(30deg);*
  - *-o-transform: rotate(30deg);*
  - *transform: rotate(30deg);*

# Posicionamento de elementos

**NOSSA EQUIPE IRÁ CONFERIR SUA SOLICITAÇÃO E ENVIARÁ UMA CONFIRMAÇÃO PARA SEU EMAIL.**

Aproveite agora mesmo a nossa dica em destaque:



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam fringilla lacinia ultricies. Nam elit lectus, interdum eu dictum et, hendrerit in tellus. Fusce tempor fringilla massa vitae consequat. Vivamus aliquet, neque sed tempus rhoncus, nisl turpis malesuada ipsum, vel imperdiet arcu mauris ut metus.

Aliquam luctus venenatis porttitor. Vivamus vitae aliquam arcu, sed auctor lacus. Fusce placerat, odio at congue suscipit, lectus purus accumsan justo, at dapibus tortor risus nec libero. Nullam orci arcu, lacinia dignissim diam et, adipiscing tempus velit. Vivamus nec interdum sem, ac ullamcorper arcu.

Morbi risus diam, cursus ac dapibus sit amet, feugiat vitae mauris.



# Posicionamento

- É a propriedade capaz de posicionar um elemento em qualquer lugar da página
- Atualmente, são 4 tipos de posicionamento:
  - Static;
  - Fixed;
  - Relative;
  - Absolute.

# Position: static

- No posicionamento estático (*position: static*) é o padrão de todo elemento
- É sua posição natural no fluxo de elementos, respeitando sua ordem na árvore do documento
- As propriedades *top*, *right*, *bottom* e *left* (topo, direita, inferior e esquerda) **NÃO** se aplicam
- Arquivo: *static.html*

# Position: fixed

- Com o valor *fixed*, o elemento é retirado do seu fluxo normal, ou seja, ele deixa de existir na posição que ocupava e passa a não mais ficar lado a lado com qualquer outro elemento presente na árvore do documento
- Além disso, ele passa a ficar **FIXO** na página, sem se mover com a barra de rolagem
- A nova posição é indicada pelo valores *top*, *right*, *bottom* e *left* do elemento em relação ao *viewport* – janela ou outra área de visualização na tela por meio da qual o usuário está vendo o conteúdo

# Position: fixed

- Observe que não há nenhum espaço reservado, nenhum “buraco” indicando a ausência de um elemento acima dos parágrafos dentro do elemento *section*, isso porque o elemento é retirado visualmente do fluxo normal dos elementos da página e passa a ocupar a nova posição
- Arquivo: *fixed.html*

# Position: relative

- Quando recebe este valor, o elemento permanece em seu fluxo normal, ou seja, não afeta os próximos elementos na árvore do documento
- Ele é deslocado de sua posição original conforme os valores de *top*, *right*, *bottom* e *left*
- Agora ao acionar a barra de rolagem, todos os elementos movem-se juntos
- Arquivo: *relative.html*

# Position: absolute

- Quando recebe este valor, o comportamento é muito semelhante ao comportamento do posicionamento *fixed*
- A primeira diferença é que o elemento continua a se mover com a barra de rolagem
- A segunda é que a nova posição é indicada pelas distâncias do elemento em relação ao elemento ancestral posicionado `<html>`, por mais que a declaração ocorra no elemento `<figure>`
- Foi o que utilizamos na página *obrigado.html*, na div principal declararamos *position: relative* e na div do lacinho, *position: absolute* e esta respeitou seu elemento ancestral
- Arquivo: *absolute.html*

# Position: absolute2

- Foi explicitamente declarado o posicionamento para o elemento *body*
- Como o posicionamento *relative* não altera a posição dos elementos subsequentes na árvore do documento, nada mudou de lugar
- A sua posição passou a ser a borda do elemento *body*
- Arquivo: *absolute2.html*

# Position: absolute3

- Foi explicitamente declarado o posicionamento para o elemento *section*
- Como o posicionamento *relative* não altera a posição dos elementos subsequentes na árvore do documento, nada mudou de lugar
- A sua posição passou a ser a borda do elemento *section*
- Arquivo: *absolute3.html*

# Árvore do documento

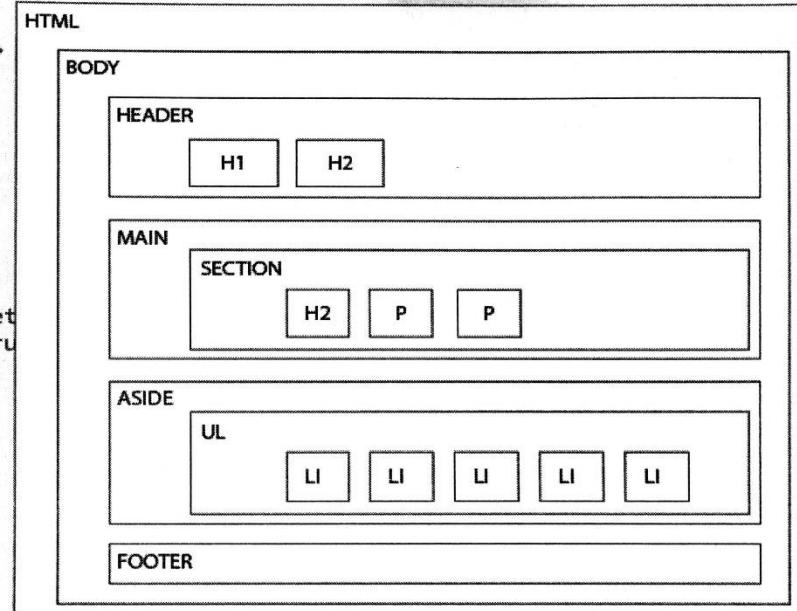
- É importante conhecer a árvore do documento e saber distinguir elementos ancestrais/pais e irmãos
- Uma estrutura utilizando corretamente os elementos HTML5 é fundamental
- Veja a estrutura do exemplo indicado e como fica sua árvore do documento em relação a elementos ancestrais e irmãos

# Árvore do documento

```
<!doctype html>
<html lang="pt-br">
<head>
 <meta charset="UTF-8">
 <title>Estrutura simples</title>
</head>
<body>
 <header>
 <h1>Título</h1>
 <h2>Subtítulo</h2>
 </header>
 <main>
 <section>
 <h2>Subtítulo 2</h2>
 <p>Lorem ipsum dolor sit amet</p>
 <p>Doloribus, adipisci, corru</p>
 </section>
 </main>
 <aside>

 Lorem ipsum
 Repudiandae quidem
 Quisquam veniam
 Repudiandae quod
 Praesentium magnam

 </aside>
 <footer>Quibusdam, earum, culpa sed harum laboriosam!</footer>
</body>
</html>
```



# Atividade 7

- Objetivo:
  - Adicionar uma pequena transição, um vídeo, um áudio e um mapa
- Tarefa:
  - Criar uma transição
  - Adicionar um vídeo
  - Adicionar um áudio
  - Adicionar um mapa

# Criar uma transição

- Abra o arquivo *index.html*
- No lugar da *<img anime-index.jpg>*, vamos substituí-la por um elemento *<figure>*, que possui 3 divs com classe *boxAnuncio* e dentro de cada uma delas as imagens *anuncio1.jpg*, *anuncio2.jpg*, *anuncio3.jpg* e seus respectivos parágrafos

# Criar uma transição

```
<figure class="anuncioAnime">
 <div class="boxAnuncio">

 <p>BRINCADEIRAS
Lorem ipsum dolor sit amet, consectetur adipisicing elit.</p>
 </div>
 <div class="boxAnuncio">

 <p>CUIDADOS
Facilis, suscipit, expedita, sed, architecto ex illo excepturi blanditiis.</p>
 </div>
 <div class="boxAnuncio">

 <p>SAÚDE
Aliquam qui at fugit magnam possimus praesentium repudiandae sint.</p>
 </div>
</figure>
```

# Criar uma transição

- Vamos alterar o elemento `<figure>` `anuncioAnime` com as seguintes propriedades:
  - Margem: 0;
  - Transbordar: auto;
- Essa é a propriedade da caixa maior (*figure*) que contém as 3 caixas menores
- Trata-se de um pequeno ajuste de margem e espaçamento

# Criar uma transição

- Vamos alterar as propriedades das *div* que possuem classe boxAnuncio:
  - Largura: 163px;
  - Altura: 230px;
  - Borda inferior: sólida fina #C0262C
  - Flutuar: esquerda;
  - Posição: relativa;
  - Transbordar: hidden; (escondida)

# Criar uma transição

- É um ajuste para cada uma das caixas que contêm o grupo *foto+parágrafo*: flutuação para que eles fiquem lado a lado e posicionamento para servirem de “ancestral posicionado” do parágrafo que vai aparecer por cima da foto

# Criar uma transição

- Vamos alterar as propriedades dos parágrafos que estão dentro das *div* *boxAnuncio*:
  - Alinhamento do texto: centralizado;
  - Tamanho da fonte: .95em;
  - Altura da linha: 1.3em;
  - Cor de fundo: rgba(255, 255, 255, 1);
  - Borda esquerda: solida fina #C0262C;
  - Borda direita: solida fina #C0262C;
  - Margem:0;
  - Espaçamento: 7px 15px;
  - Posição: absoluta;
  - Transição: 1s linear 0.1s;
  - inferior: -80px;
  - esquerda: 0;
  - cursor: default;

# Criar uma transição

- É o parágrafo propriamente dito e sua posição sobre a foto
- Possui instrução *transition* e seus valores: duração da transição, tipo de *timing* (de que forma a transição progride ao longo do tempo) e *delay* (tempo de espera entre a ação e o início da transição)
- Transição:  
<https://matthewlein.com/tools/ceaser>

# Criar uma transição

- Vamos alterar as propriedades da primeira linha dos parágrafos que estão dentro das *div* `boxAnuncio`:
  - Tamanho da fonte: 1.1em;
  - fonte-variante: inherit;
  - Estilo da fonte: normal;
  - Peso da fonte: bolder;
  - cor: #C0262C;
  - espaçamento: 0;

# Criar uma transição

- Definimos apenas o visual da primeira linha do parágrafo

# Criar uma transição

- Vamos alterar as propriedades dos parágrafos que estão dentro das *div* boxAnuncio ao passar o cursor do mouse em cima da img:
  - background-color: rgba(255, 255, 255, 0.7);
  - bottom: 0;

# Criar uma transição

- Define apenas o que será alterado quando o visitante passar o mouse por cima do elemento *p* da caixa
- É exatamente nesse grupo que será visto o efeito da transição

# Criar uma transição

Home

Quem somos

Produtos

Banho e Tosa

Curiosidades



O melhor para o nosso  
amigo cão

Quisque venenatis urna ac nulla. Maecenas quis turpis nec

# Transição

- Para aplicar efeitos de transição nos elementos da página é necessário conhecer algumas propriedades que causam esses efeitos
  - transition-property
  - transition-duration
  - Transition-timing-function
  - Transition-delay

# transition-property

- É dizer nada mais que em qual propriedade receberá o efeito de transição, como por exemplo: largura, altura, tamanho da fonte e etc
- Exemplo de aplicação da propriedade:
  - transition-property: width;

# transition-duration

- É a duração do efeito da transição, que pode ser em segundos
- Exemplo de aplicação da propriedade:
  - transition-duration: 3s;

# transition-timing-function

- De que forma a transição progride ao longo do tempo, ou seja, a curva de velocidade da transição
- Pode ser de vários modos: ease, linear, ease-in, ease-out, ease-in-out, step-start, step-end
- Exemplo de aplicação da propriedade:
  - transition-timing-function: linear;

# transition-timing-function

- `ease`: especifica um efeito de transição com início lento, depois rápido e, a seguir, fim lento (é o padrão)
- `linear`: especifica um efeito de transição com a mesma velocidade do início ao fim
- `ease-in`: especifica um efeito de transição com um início lento
- `ease-out`: especifica um efeito de transição com um final lento
- `ease-in-out`: especifica um efeito de transição com um início e fim lentos
- `cubic-bezier(n, n, n, n)`: permite definir seus próprios valores em uma função de bezier cúbico

# transition-delay

- É o intervalo de tempo entre a ação do usuário e o início do efeito de transição
- Exemplo de aplicação da propriedade:
  - transition-delay: .3s;

# transition

- Forma condensada que resume todas as propriedades de transição entre valores de propriedades [*property / duration / timing / delay*]
- Exemplo de aplicação da propriedade:
  - transition: width 1s linear .3s;

# Animação

- Da mesma forma que o efeito de transição, é possível acrescentar mais riqueza e experiência do usuário com o efeito animação
- Seu processo de escrita é um pouco mais trabalhoso, mas igualmente fácil de manipular e entender
- A principal diferença entre uma e outra, é que a transição é controlado pelos estados do elementos, enquanto a animação é controlado pelo tempo

# Animação

- A forma mais comum na criação da animação, é realizada em 2 etapas
- Na primeira é declarada a animação, utilizando as palavras-chave *from* e *to*, indicando respectivamente as propriedades iniciais e finais da animação

# Animação

- Primeira parte, ou seja, declaração da animação

```
@keyframes anime {
 from {
 propriedade: valor;
 }
 to {
 propriedade: valor;
 }
}
```

# Animação

- A segunda etapa, são indicado as características da animação: o seu nome, o tempo de duração, o tipo de progressão, o intervalo antes do início, o contador de repetições, a direção (só ida, ida e volta) e o estado da animação (em funcionamento ou em pausa)

# animation-name

- Nome da animação conforme @keyframe
- Exemplo de aplicação da propriedade:
  - animation-name: anime;

# animation-duration

- Tempo de duração da animação
- Exemplo de aplicação da propriedade:
  - animation-duration: 5s;

# animation-timing-function

- De que forma a transição progride ao longo do tempo
- Pode ser de vários modos: ease, linear, ease-in, ease-out, ease-in-out, step-start, step-end
- Exemplo de aplicação da propriedade:
  - animation-timing-function: ease;

# animation-delay

- Quanto tempo a animação demora para iniciar
- Exemplo de aplicação da propriedade:
  - animation-delay: .3s;

# animation-iteration-count

- Quantas vezes a animação se repete
- Exemplo de aplicação da propriedade:
  - animation-iteration-count : infinite;

# animation-direction

- De que forma a animação se reinicia
- Pode ser dos modos normal, reverso, alternando
- Exemplo de aplicação da propriedade:
  - animation-direction: alternate;

# animation-play-state

- Determina o estado da animação
- Pode ser running | paused
- Exemplo de aplicação da propriedade:
  - animation-play-state: running;

# animation

- Forma resumida de declarar todos os valores da animação
- name, duration, timing-function, delay, iteration-count, direction, play-state

# Animação

- Segunda parte, ou seja, declaração da animação
  - .animeGo {

```
animation-name: nome;
animation-duration: 5s;
animation-timing-function: ease;
animation-delay: .3s;
animation-iteration-count: infinite;
animation-direction: alternate;
animation-play-state: running;
```

}

# Adicionar elemento de áudio

- Abra o arquivo *index.html*
- Localize o primeiro conjunto *aside*, que contém a imagem *anuncio\_pet1.jpg*
- Adicione a imagem *brincando.jpg*
- Adiciona abaixo o áudio (*brincar\_com\_cuidado.ogg*) do veterinário
- Insira o parágrafo: *Ouça a dica do veterinário!*
- E mantenha a imagem(*anuncio\_pet1.jpg*) já adicionada anteriormente no final

# Adicionar elemento de áudio

- Adicione ao estilo na arquivo principal o comentário abaixo:  
`/* controle de áudio */`
- Adicione ao elemento áudio a seguinte propriedade:
  - Largura: 230px;

# Adicionar elemento de áudio

Home

Quem somos

Produtos

Banho e Tosa

Curiosidades



▶ 0:00 / 0:12 🔍

Ouça a dica do veterinário!



## O melhor para o nosso amigo cão

Quisque venenatis urna ac nulla. Maecenas quis turpis nec nulla auctor mattis. Curabitur mi libero, scelerisque eu, cursus semper, tempor quis, ante. Etiam iaculis, nisl eu imperdiet tincidunt, neque lacus ornare leo, non ultrices arcu justo ut enim. Morbi pulvinar sagittis sapien. Sed adipiscing purus nec tortor. Cras nisi sapien, ultricies nec, dapibus ut, vehicula ac, pede. Donec venenatis lorem sit amet erat vehicula hendrerit.



# Adicionar elemento de vídeo

- Abra o arquivo *index.html*
- Localize o primeiro conjunto *aside*, que contém a imagem *anuncio\_pet2.jpg*
- A tag vídeo controls adicione o poster=“brincando2.jpg”
- Adicione o vídeo *cão\_brincando* com os dois formatos *mp4* e *ogg*
- Adicione o parágrafo logo abaixo: “*Brinque sempre com seu cãozinho*”
- Mantenha a imagem *anuncio\_pet2* adicionada anteriormente logo abaixo

# Adicionar elemento de vídeo

- Adicione ao estilo na arquivo principal o comentário abaixo:  
`/* controle de vídeo */`
- Adicione ao elemento video a seguinte propriedade:
  - Largura: 230px;

# Adicionar elemento de vídeo

Home      Quem somos      Produtos      Banho e Tosa      Curiosidades

BRINCADEIRAS      CUIDADOS      SAÚDE

▶ 0:00 / 0:12      [volume icon]      [dots icon]

Ouça a dica do veterinário!

Quisque venenatis urna ac nulla. Maecenas quis turpis nec nulla auctor mattis. Curabitur mi libero, scelerisque eu, cursus semper, tempor quis, ante. Etiam iaculis, nisl eu imperdiet tincidunt, neque lacus ornare leo, non ultrices arcu justo ut enim. Morbi pulvinar sagittis sapien. Sed adipiscing purus nec tortor. Cras nisi sapien, ultricies nec, dapibus ut, vehicula ac, pede. Donec venenatis lorem sit amet erat vehicula hendrerit.

Brinque sempre com seu cãozinho!

# Adicionar um mapa

- Acesse o site <http://maps.google.com.br>
- Pesquise pelo endereço “Rua Santa Eulália, 86, São Paulo”
- No canto inferior direito, clique no ícone da engrenagem e em seguida, em *Compartilhar ou incorporar mapa*

# Adicionar um mapa



# Adicionar um mapa

- No quadro seguinte, clique na aba *Incorporar mapa* e adicione o tamanho abaixo

Compartilhar X

[Enviar um link](#) [Incorporar um mapa](#)

Tamanho personalizado 1000 x 300 [VISUALIZAR TAMANHO REAL](#)

`<iframe src="https://www.google.com/maps/embed?pb=" width="1000" height="300"` [COPIAR HTML](#)

Ao incorporar este mapa, você concorda com os [Termos de Serviço](#).

# Adicionar um mapa

- Copie o código <iframe> e no editor de código do arquivo *quem-somos.html* e adicione os elementos abaixo antes da tag de fechamento </main>
  - Crie uma régua
  - Crie um subtítulo nível 2 com o nome: Local
  - Cole o código copiado do <iframe>
  - Salve a página

# Media Queries e unidade relativas

- O projeto Meu Amigo Cão Petshop que desenvolvemos no decorrer da UC foi feito com medidas específicas
- Definimos como largura da página o valor de 1.000px
- Essa é uma forma segura de trabalhar, mas, com a evolução da linguagem e dos dispositivos com acesso à web, esse formato apenas não é o suficiente

# Media Queries e unidade relativas

- É importante conhecer e trabalhar com Media Queries e unidades relativas

# Unidades relativas

- São aquelas que estabelecem alguma relação com algum outro elemento
- Os dois exemplos mais utilizados são **%** e **em**

# Unidade Relativa (%)

- Uma medida usando unidade em porcentagem (%) significa que o valor é uma porcentagem do elemento pai
- Por exemplo, podemos ter um box com 80% de largura contendo duas menores com 50% de largura cada
- A primeira caixa ocupar 80% da largura do elemento pai e as duas caixas menores ocupam a metade da nova área
- Atividade: *unidade-percent.html*, *unidade-percent-text.html*

# Unidade Relativa (*em*)

- Uma media usando unidade *em* significa que o valor é uma relação com a fonte padrão do navegador, que em geral equivale a 16px
- Dessa forma, se uma caixa deve ter 500px, seu valor será calculado assim:  $500 / 16 = 31,25\text{em}$
- Para colocar duas caixas iguais dentro dessa medida, o cálculo é  $(500 / 2 = 15,625\text{em})$  (que é o mesmo que  $31,25 / 2$ )
- Atividade: *unidade-em.html*, *unidade-em-text.html*

# Media Queries

- Em conjunto com as unidade relativas, está o uso da declaração Media Queries, que é um novo conjunto de declarações criado com o CSS3 para permitir ao desenvolvedor estabelecer uma faixa de atuação para um determinado conjunto de estilos

# Media Queries

- Chamada do HTML:
  - <meta name="viewport" content="width=device-width, initial-scale=1">
- Chamada dentro do CSS delimitado pelo valor menor e maior da tela com estilos agrupados em uma única folha de estilo:

```
***** pequeno *****/
@media screen and (max-width: 419px){
 [declarações do tamanho pequeno]
}
```

# Media Queries

```
***** médio *****
```

```
@media screen and (min-width: 420px) and (max-width: 679px) {
```

```
 [declarações do tamanho médio]
```

```
}
```

```
***** grande *****
```

```
@media screen and (min-width: 680px) {
```

```
 [declarações do tamanho grande]
```

```
}
```

# Media Queries

- Exercícios propostos:
  - *media-queries-1.html* (fazer utilizando três folhas de estilo CSS)

```
<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="stylesheet" href="estilo-pq.css" />

<link rel="stylesheet" media="screen and (min-width: 420px)" href="estilo-
md.css" />

<link rel="stylesheet" media="screen and (min-width: 680px)" href="estilo-
gd.css" />
```

# Media Queries

- Exercícios propostos:
  - *media-queries-2.html* (fazer utilizando uma única folha de estilo CSS)

```
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="estilos-agrupados.css" />
```

# pequeno – até 419px

## Tela pequena: até 419px



### **Lorem ipsum dolor sit amet**

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quo voluptatibus deleniti asperiores atque laudantium dolorem, ad qui voluptate repellat velit, fuga quaerat ipsa dicta quam cupiditate labore corporis enim tenetur.



# médio – entre 420px e 679px

Tela média: min. 420px



## **Lore ipsum dolor sit amet**

Lore ipsum dolor sit amet, consectetur adipisciing elit. Quo voluptatibus deleniti asperiores atque laudantium dolorem, ad qui voluptate repellat velit, fuga quaerat ipsa dicta quam cupiditate labore corporis enim tenetur.



## **Libero iure reiciendis**

Libero iure reiciendis, nam ullam deserunt sint fugit maxime, quasi officiis itaque, eos modi deleniti! Atque et nobis libero, ratione, ut enim amet animi modi ducimus adipisci officiis neque quasi!



## **Quidem quas reiciendis**

Quidem quas reiciendis, doloremque molestias adipisci, laboriosam aliquid, perspiciatibus ullam maiores quia numquam velit reprehenderit, dolorem vel! Qui veniam adipisci earum numquam voluptate, modi velit porro officiis laborum iusto itaque.

# grande – mínimo 680px

Tela grande: min. 680px



## **Lorem ipsum dolor sit amet**

*Lorem ipsum dolor sit amet, consectetur adipisciing elit. Quo voluptatibus deleniti asperiores atque laudantium dolorem, ad qui voluptate repellat velit, fuga quaerat ipsa dicta quam cupiditate labore corporis enim tenetur.*



## **Libero iure reiciendis**

*Libero iure reiciendis, nam ullam deserunt sint fugit maxime, quasi officiis itaque, eos modi deleniti! Atque et nobis libero, ratione, ut enim amet animi modi ducimus adipisci officiis neque quasi!*



## **Quidem quas reiciendis**

*Quidem quas reiciendis, doloremque molestias adipisci, laboriosam aliquid, perspiciatis ullam maiores quia numquam velit reprehenderit, dolorem vel! Qui veniam adipisci earum numquam voluptate, modi velit porro officiis laborum iusto itaque.*

# Exercícios propostos Cap. 05

- Em todos os exercícios criar o arquivo CSS externo e fazer sua ligação ao respectivo arquivo HTML
- cap-05/\_propostos

# Exercício 1

- Nome do arquivo: *formulario.html*
- Crie um único arquivo CSS externo e insira os estilos e os códigos necessários para que tenha o aspecto da imagem disponibilizada da página
- Referências: imagem – *form1\_back.jpg*; títulos: #909; caixas, borda e botão: #FCC.

# Exercício 1

formulário

## Dados da conta

Dados Pessoais:

Nome:

Email:

Dados da Conta:

Senha:

Frase  
opcional:

enviar

# Exercício 2

- Nome do arquivo: *tabelas.html*
- Crie um único arquivo CSS externo e insira os estilos e os códigos necessários para que tenha o aspecto da imagem disponibilizada da página
- Referências: fundo e borda: #036

# Exercício 2

MODELO	ALTURA	LARGURA	ACESSÓRIOS	
			USB	WI-FI
Xk-7	20	20	<input checked="" type="radio"/>	<input type="radio"/>
WZ8	10	15	<input type="radio"/>	<input type="radio"/>
X15	30	30	<input checked="" type="radio"/>	<input checked="" type="radio"/>

**Dados técnicos do equipamento**

# Exercício 3

- Nome do arquivo: *layout.html*
- Crie um único arquivo CSS externo e insira os estilos e os códigos necessários para que tenha o aspecto da imagem disponibilizada da página
- Referências: trabalhe com as unidades em “*em*”; verde: #c4d6a8; vermelho: #900;

# Exercício 3



## Portal da Floresta

[link 1](#)

[link 2](#)

[link 3](#)

[link 4](#)

[link 5](#)

### Título Principal

autor e data

*Lorem ipsum dolor sit amet,  
consectetur adipiscing elit. Nulla vitae  
varius nunc.*



*Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla vitae varius nunc. Aliquam sed nulla urna, porta porttitor libero. Duis molestie rhoncus neque, quis posuere turpis ac congue non. In mattis, turpis ac rhoncus ultrices, nisl lacus fermentum lectus, in vulputate purus erat in nisl. Phasellus tristique, nunc et eleifend viverra, felis dolor sollicitudin nunc, et semper nisi magna id elit. Sed nulla turpis, tempus vitae rhoncus et, iaculis at est. Aliquam erat volutpat. Nullam orci eros, volutpat in congue vitae, ornare ut felis. Sed venenatis lorem non arcu venenatis a placerat leo ullamcorper. Ut imperdiet pharetra mi ac sagittis. Mauris a nibh massa. Morbi consequat faucibus velit vel viverra. Donec quis nulla in sem rutrum accumsan nec et ipsum. Praesent viverra est at tortor euismod at interdum augue interdum.*

#### notícia 1

00h00 / 01.01

*Lore ipsum dolor  
sit amet, consectetur  
adipiscing elit. Nulla  
vitae varius nunc.*

[» leia mais](#)

#### notícia 2

00h00 / 01.01

*Aenean ut massa et  
erat sagittis fringilla  
vel sed nibh. Sed  
lacinia volutpat  
consectetur.*

[» outras](#)

#### notícia 3

00h00 / 01.01

*Fusce tempor sapien  
at nunc congue  
cursus. Aenean lacus  
felis, volutpat.*

[» siga o exemplo](#)

**rua vitae rhoncus, 123 - são paulo, sp - 01234-000 - 11 1234.5678**

# Como última dica: treine!

- Nada melhor do que fazer e refazer para treinar
- Visite um site de que você goste e tente reproduzi-lo
- Aos poucos, você entenderá o funcionamento do CSS em seus mínimos detalhes e poderá fazer novas combinações e layouts

# Finalização

- Seu projeto está pronto! Navegue por suas páginas e faça os reajustes necessários antes de entregar
- Parabéns!