

CURSO HTML5

Curso em Vídeo (Gustavo Guanabara)

https://www.youtube.com/watch?v=epDCjksKMok&list=PLHz_AreHm4dlAnJ_jtV29RFxnPHDuk9o

Resumo do curso feito por Roberto Pinheiro

ESTRUTURA BÁSICA DE UM DOCUMENTO HTML5

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

Doctype e o elemento HTML

O Document Type Definition (DTD, ou simplesmente Doctype) é uma instrução que informa ao navegador qual é a especificação do código que está sendo usada no documento, e deve ser declarado antes da tag <html>.

Na versão anterior do HTML, a declaração do Doctype era mais extensa e difícil de decorar, havendo a necessidade de referenciar para o navegador o arquivo DTD com as definições daquela especificação:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

No HTML5 a inserção do Doctype foi simplificada, e a responsabilidade de buscar as definições da especificação fica por conta do próprio navegador:

```
<!DOCTYPE html!>
```

Após a declaração do Doctype, iniciamos o código HTML. Na árvore de elementos do código, a tag principal é a <html>, que comporta todos os outros elementos filhos.

É na tag <html> que declaramos o idioma principal do documento, através do atributo (que pode ser usado também em outras tags do documento).

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

Metadados

Os Metadados são um conjunto de informações a respeito da página e do conteúdo nela publicado. Essas informações são usadas pelos navegadores e user-agents em geral, sendo invisíveis para os usuários. Todos os Metadados ficam contidos na tag <head>:

```
<head>
  <meta charset="utf-8">
  <title>Entendo a estrutura e semântica do HTML5</title>
</head>
```

A metatag Charset é usada para indicar a codificação de caracteres que a nossa página está utilizando.

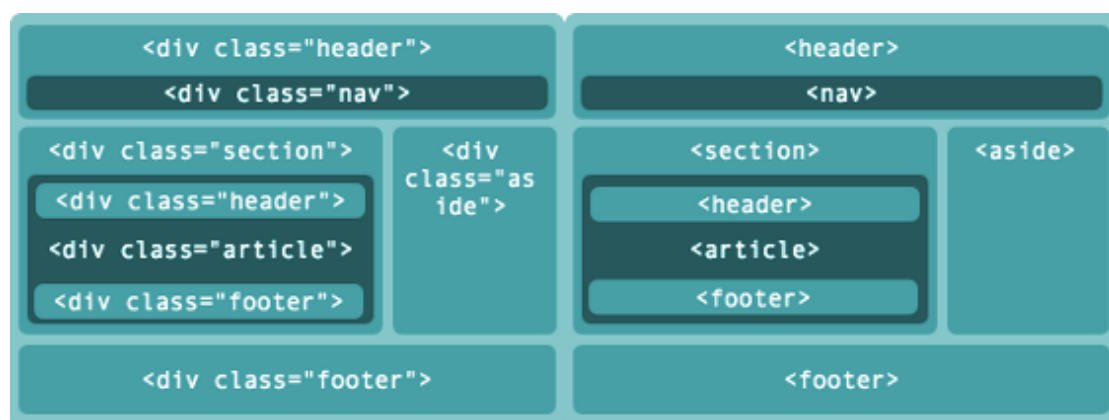
O uso da tag <meta charset="utf-8"> permite a visualização correta das palavras acentuadas.

A semântica das novas marcações do HTML5

Em todas as páginas da Web existem divisões básicas referentes aos tipos de conteúdo que são colocados em cada parte do layout, como cabeçalho, rodapé ou menu de navegação.

Nas versões anteriores do HTML não haviam tags com uma semântica apropriada para cada uma dessas divisões. Dessa forma, os desenvolvedores acabavam usando a tag <div> para todas as situações, e criando seus próprios padrões de nomeclaturas através dos atributos id ou class.

No HTML5 foram criadas diversas tags semânticas para indicar aos user-agents quais conteúdos estão sendo inseridos em cada uma das divisões da página, organizando e padronizando o desenvolvimento.



PRIMEIRO PROGRAMA - Olá Mundo!

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8" />
  <title>Curso de HTML5</title>
  <style>
    h1 {
      font-family: Arial;
      font-size: 30pt;
      color: blue;
      text-shadow: 2px 2px 2px black;
    }
  </style>
</head>
<body>
  <h1>Olá, Mundo!</h1>
  <h2>Me livrando da maldição</h2>
  <h1>Oba! Me livre!</h1>
</body>
</html>
```

TAGS BÁSICAS EM HTML5

<hgroup></hgroup>

Dentro do <header> podemos também inserir o elemento <hgroup>, que serve para agrupar dois ou mais elementos de títulos (h1 até h6), organizando-os hierarquicamente.

<header></header>

O novo elemento <header> do HTML5 é usado para definir o cabeçalho de uma página ou sessão, e pode conter logo, títulos, menu de navegação, campo de busca, etc.

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

<head>
  <meta charset="UTF-8"/>
  <title>Tudo sobre Google Glass</title>
</head>
```

```
<body>

<div id="interface">
  <header id="cabecalho">

    <hgroup>
      <h1>Google Glass</h1>
      <h2>A revolução do Google está chegando</h2>
    </hgroup>

  </header>

</div>
</body>
</html>
```

PARÁGRAFOS, QUEBRAS E SÍMBOLOS ESPECIAIS

Hierarquia de títulos em HTML5 com HGROUP: entenda que a hierarquia de títulos não depende apenas de ordem em um mesmo grupo de títulos. Tudo é uma questão de significado ou semântica HTML5.

Parágrafos em HTML5 com P: Os parágrafos delimitam textos agrupados em HTML5. Todo parágrafo deve estar limitado com a tag P e /P.

Quebras de linha usando HTML5 com BR: A quebra de linhas vai desviar a reprodução do texto no início da linha abaixo da atual. A tag BR não possui fechamento.

Cada quebra de linha é um
 e cada espaço em branco é um

Quebra de palavras em HTML5: Usando a nova tag WBR em HTML5, você consegue criar quebra de palavras em seu site. Qualquer palavra muito longa pode ser feita utilizando a tag WBR.

ex: super-<wbr/>humano

Hifenização em HTML5: No caso de criar uma palavra grande, você pode utilizar a marca SHY para gerar a separação de sílabas com hifenização. É uma das maneiras mais simples de separar sílabas em HTML5.

ex: video­conferência

Caracteres especiais e símbolos em HTML5: Os caracteres especiais permitem criar alguns símbolos em seu site feito em HTML5, como nbsp lt gt le ge pound yen euro copy reg trade pernil sum infin times plusmn oplus radic ne delta lambda omega phi larr rarr uarr darr harr spades clubs hearts dia

FORMATAÇÃO DE TEXTO COM HTML5 E CSS3

HTML5 é uma tecnologia nova e que rompe com alguns conceitos estabelecidos na versão 4.0 da linguagem. Agora, tudo aquilo que é feito em HTML5 tem um caráter semântico, enquanto a parte visual e de formatação ficará por conta das folhas de estilo CSS.

Semântica em HTML5: a partir de agora, as tags HTML5 possuem uma função semântica, o que diz para o navegador e mecanismos de busca que cada área do documento tem um significado.

Itálico em HTML5: Utilizando a tag I e a tag EM

Tags depreciadas HTML: Antigamente, a tag U era utilizada para sublinhar trechos de texto. Agora, tudo é realizado via CSS, com text-decoration: underline em uma SPAM. As tags S ou STRIKE foram substituídas pela tag DEL.

Tipos de text-decoration com CSS: Entenda os tipos de decoração para textos em CSS como undeline, overline, line-through e none.

Herança de características em CSS3

Formatações de Negrito em HTML5 com CSS3: O parâmetro text-weight como normal, bold e bolder.

Alinhamento de textos em HTML5 com CSS: A formatação com o align no código HTML também foi depreciada. A partir de agora, utilizaremos o text-align do CSS com os valores left, right, center e justify.

Indentação de parágrafos com HTML5: Utilizando text-indent em CSS você pode causar deslocamentos laterais no início de parágrafos.

Configuração global nas CSS: Utilizando a tag STYLE do HTML5, você pode criar formatações de folha de estilo que vão funcionar para todos os objetos em uma página.

Tags de formatação em HTML5: Entenda o funcionamento de tags como SUP, SUB, CODE e PRE.

CÓDIGOS DE CORES UTILIZANDO O HTML5 e CSS3.

Algumas tags e parâmetros foram descontinuadas do HTML4 para o HTML5, incluindo o FONT COLOR e o BGCOLOR da tag BODY. A partir de agora, devemos utilizar o parâmetro STYLE de qualquer tag e utilizar as configurações de background-color e color do CSS.

A utilização de cores através de seus nomes (como red, green, blue, etc) é algo bastante limitado. Por conta disso, vamos aprender como utilizar o código para cores RGB.

Para o código de cores #60453E, por exemplo, temos as quantidades de vermelho/red (60), verde/green (45) e azul/blue (3E). A junção desses códigos vai gerar uma cor única, em um espectro que possui milhões de combinações. Cada valor é representado em base hexadecimal.

ex:

```
body{
    color: #0000ff;
}
```

Outra maneira de utilizar cores em CSS é utilizando o modificador rgb(), passando as quantidades de vermelho, verde e azul em valores representados na base decimal.

ex:

```
body{
    background-color: rgb(35, 137, 195);
}
```

Em HTML5 e CSS3, utilizando o rgba() nas CSS3, podemos ainda incluir a configuração do canal alpha, que vai de 0 até 1.

ex:

```
body{
    background-color: rgba(35, 137, 195, 0.5);
}
```

Em HTML5 e CSS3, também podemos criar cores utilizando a representação percentual de Matiz (Hue), saturação (Saturation) e luminosidade (Brightness) utilizando a função hsla() das folhas de estilo.

ex:

```
body{
    background-color: hsl(165, 81%, 93%);
}
```

De maneira similar, temos também a função `hsla()` para indicar também o canal alpha.

```
body{
    background-color: hsla(165, 81%, 93%, 0.5);
}
```

Podemos também colocar imagens no fundo do site utilizando o parâmetro `background` e utilizar a função `url()` do CSS para carregar uma imagem.

ex:

```
body{
    background-image: url("fundo.jpg");
}
```

IMAGENS EM HTML5

Qual a diferença entre PNG e JPG? O formato PNG é ligeiramente maior, mas possibilitam transparências. O formato JPEG é mais otimizado, pois suporta compressão sem uma perda perceptível na qualidade visual.

ex:

```

```

Utilizando os parâmetros `WIDTH` ou `HEIGHT`, você pode redimensionar uma imagem pelo HTML. Porém, a melhor maneira de se diminuir o tamanho do arquivo é utilizar um editor de imagens para realizar o redimensionamento.

Outra coisa muito importante é utilizar a opção de “Exportar para Web” do seu editor de imagem. Isso gera um arquivo menor ainda, mais otimizado para ser utilizado em sites construídos com HTML5.

Como carregar uma imagem em HTML5 que esteja em uma outra pasta?

Novos parâmetros `FIGURE` e `FIGCAPTION`. O HTML5 ganhou um caráter semântico, e utilizar legendas de imagens em HTML5 é um ótimo recurso para otimizar seu site para mecanismos de busca, técnicas conhecidas como SEO.

ex:

```
<figure>

<figcaption>
  <h3>Google Glass</h3>
  <p>Uma nova maneira de ver o mundo.</p>
</figcaption>
</figure>
```

FORMATAÇÃO DE IMAGENS COM CSS3

Formatar imagens utilizando folhas de estilo CSS é algo muito simples e requer o conhecimento de alguns parâmetros específicos em HTML5 e CSS3.

Estilos CSS em arquivos separados

Quando as formatações com folhas de estilo CSS começam a ficar grandes demais, o ideal é trabalhar com arquivos CSS externos. Para isso, utilizamos a tag LINK com o parâmetro REL de valor STYLE SHEET.

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

Comentários em CSS

Para comentar pedaços de código em CSS, utilizamos os caracteres /* e logo após a descrição do comentário, colocamos */

Referenciando CLASS e ID

A diferença entre CLASS e ID dentro das CSS é na referência. Para usar o CLASS, utilizamos o caractere de ponto (.) e para o ID, utilizaremos a cerquilha (#)

Tamanho de objetos em CSS

Utilizando os parâmetros WIDTH (largura) e HEIGHT (altura), você pode modificar o tamanho de um objeto utilizando CSS.

Posição de objetos em CSS

Utilizando os parâmetros TOP (topo) e LEFT (esquerda)

Utilizando o POSITION em CSS – Diferença entre RELATIVE e ABSOLUTE

Existem dois tipos de posicionamento para CSS: com o POSITION RELATIVE, ele terá a localização feita em relação à sua posição atual. Para POSITION ABSOLUTE, ele terá a localização feita em relação a todo o conteúdo.

CSS com BOX-SIZING com o parâmetro BORDER-BOX.

Modificando opacidade com CSS utilizando o parâmetro OPACITY.

Utilizando o HOVER em um objeto com CSS vai fazer com que uma formatação específica

TRANSITION com CSS adiciona efeitos de transição ao modificar características visuais em um objeto.

MENUS E LISTAS

A tag OL (Ordered lists) vai criar uma lista ordenada. Utilizando os valores do parâmetro TYPE, você pode criar uma lista ordenada numérica (1), alfabética (a/A) ou em algarismos romanos (i/I). O parâmetro START da tag OL vai indicar onde vai começar a contagem dos itens. O parâmetro START aceita apenas valores numéricos, independente da maneira utilizada para numerar.

A tag UL (Unordered lists) vai criar uma lista com demarcadores ou lista não-ordenada. O parâmetro TYPE para OL suporta os valores CIRCLE, SQUARE ou DISC. Para a tag UL, o parâmetro START simplesmente não funciona.

Podemos também criar hierarquia com listas, utilizando tags OL e UL aninhadas.

LINKS EM HTML5

Para criar um link em HTML5, basta criar uma âncora utilizando a tag A.

O parâmetro Hypertext Reference, ou HREF. Ele cria uma referência hipertexto para outras páginas.

O parâmetro TARGET da tag A vai permitir indicar em qual janela o conteúdo será aberto. O valor _blank vai permitir abrir em uma janela em branco, enquanto o parâmetro _self vai abrir o conteúdo na janela atual.

link externo:

ex: Project Glass

link interno:

ex: Especificações

MENUS COM CSS3

A partir do HTML5, a tag NAV é utilizada para criar áreas de navegação. Com o CSS3, podemos configurar posicionamento, cores, efeitos de transição e outros comportamentos. Veremos como transformar uma lista com demarcadores em um menu flutuante para o nosso site.

ex:

```
<nav id="menu">
  <h1>Menu Principal</h1>
  <ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="specs.html">Especificações</a></li>
    <li><a href="fotos.html">Fotos</a></li>
    <li><a href="multimidia.html">Multimídia</a></li>
    <li><a href="fale-conosco.html">Fale conosco</a></li>
  </ul>
</nav>
```

Para realizar essas configurações, aprenderemos alguns parâmetros de CSS3, como:

Modificar o formato da lista, eliminando os demarcadores com o LIST-STYLE

ex:

```
nav#menu ul {
  list-style: none;
}
```

Utilizando o TEXT-TRANSFORM com o valor UPPERCASE, alteramos todas as letras de cada item para maiúsculas, independente da maneira que foram escritas.

ex:

```
nav#menu ul {
  list-style: none;
  text-transform: uppercase;
}
```

Com a configuração DISPLAY com o valor INLINE-BLOCK, você vai conseguir alterar a posição de cada item da lista.

ex:

```
nav#menu li {  
  display: inline-block;  
}
```

As propriedades BACKGROUND-COLOR e COLOR modificam a cor do objeto.

ex:

```
nav#menu li {  
  display: inline-block;  
  background-color: #dddddd;  
}
```

Com o PADDING, vamos configurar o espaço interno de cada objeto, enquanto o MARGIN vai definir o espaço externo de cada um deles.

ex:

```
nav#menu li {  
  display: inline-block;  
  background-color: #dddddd;  
  padding: 10px;  
  margin: 2px;  
}
```

O POSITION do CSS3, vai configurar qual será a base de movimentação em relação ao seu objeto container. Os valores aceitos são ABSOLUTE e RELATIVE.

As propriedades LEFT e TOP vão configurar a posição do objeto em suas grandezas em pixels para o lado esquerdo e borda superior, respectivamente.

```
nav#menu ul {  
  list-style: none;  
  text-transform: uppercase;  
  position: absolute;  
  top: -20px;  
  left: 300px;  
}
```

Utilizando a propriedade DISPLAY com o valor NONE, podemos esconder determinados objetos utilizando CSS3.

```
nav#menu h1 {  
    display: none;  
}
```

EFEITOS VISUAIS EM CSS3

Utilizando o modificador :HOVER aplica o efeito quando movemos o mouse por cima de um objeto.

ex:

```
nav#menu li:hover {  
    background-color: #606060;  
}
```

Com o seletor TRANSITION, podemos monitorar alterações nos objetos e fazer efeitos de transição.

ex:

```
nav#menu li {  
    display: inline-block;  
    background-color: #dddddd;  
    padding: 10px;  
    margin: 2px;  
    transition: background-color 1s;  
}
```

FORMATAÇÃO DE INTERFACES COM HTML5

A construção de interfaces em HTML5 necessita da organização das áreas do nosso site. Podemos fazer isso em qualquer tag, incluindo DIV, HEADER, ARTICLE, FOOTER e muitas outras.

O modificador WIDTH vai configurar a largura de um objeto.

ex:

```
div#interface {  
    width: 900px;  
}
```

O posicionamento em CSS pode ser ABSOLUTE ou RELATIVE.

ex:

```
header#cabecalho img#icone {  
  position: absolute;  
  left: 980px;  
  top: 40px;  
}
```

A centralização de um objeto em CSS3 é utilizando o parâmetro MARGIN.

ex:

```
div#interface {  
  width: 900px;  
  background-color: #ffffff;  
  margin: -20px auto 0px auto;  
  box-shadow: 0px 0px 10px rgba(0,0,0,.5);  
  padding: 10px;  
}
```

CABEÇALHO EM CSS3

Veremos como criar cabeçalhos para sites HTML5 utilizando CSS3 e alguns parâmetros adicionais para realizar uma formatação.

– Os modificadores BORDER-TOP, BORDER-BOTTOM, BORDER-LEFT e BORDER-RIGHT configura linhas em volta do objeto e aceita valores como SOLID, DOTTED, DASHED, etc.

ex:

```
header#cabecalho {  
  border-bottom: 1px solid #606060;  
}
```

Para carregar imagens via CSS, podemos utilizar o modificador BACKGROUND usando a função URL para carregar o arquivo no fundo dos objetos. Utilizando o parâmetro NO-REPEAT para que ele não repita o objeto.

```
header#cabecalho {  
  border-bottom: 1px solid #606060;  
  height: 150px;  
  background: url("../_imagens/glass-logo-peq.jpg") no-repeat 0px 80px;  
}
```

Para a configuração de texto, utilizaremos FONT-FAMILY, FONT-SIZE, COLOR e TEXT-SHADOW.

```
header#cabecalho h1 {  
  font-family: Arial, sans-serif;  
  font-size: 30pt;  
  color: #606060;  
  text-shadow: 1px 1px 1px rgba(0, 0, 0, .6);  
}
```

Além disso, configuramos os espaços internos e periféricos dos objetos com os parâmetros MARGIN e PADDING e suas variações MARGIN-BOTTOM e MARGIN-TOP.

```
header#cabecalho h1 {  
  font-family: Arial, sans-serif;  
  font-size: 30pt;  
  color: #606060;  
  text-shadow: 1px 1px 1px rgba(0, 0, 0, .6);  
  padding: 0px;  
  margin-bottom: 0px;  
}
```

FONTES PERSONALIZADAS COM CSS3

Como mudar a fonte de um site usando CSS3 e um arquivo True Type Font (TTF) com fontes personalizadas em CSS3.

Utilizando o modificador @FONT-FACE das CSS3, com o FONT-FAMILY e o SRC, poderemos criar fontes personalizadas utilizando CSS3.

O usuário que visita o seu site não precisa ter a fonte instalada. Ela é instalada automaticamente.

ex:

```
@font-face {  
  font-family: 'FonteLogo';  
  src: url("../_fonts/bubblegum-sans-regular.ottf");  
}
```

Como usar Google Fonts para inserir uma fonte personalizada do seu site utilizando o @IMPORT.

É possível utilizar fontes disponíveis no Google Fonts (www.google.com/fonts).

```
@import url('https://fonts.googleapis.com/css?family=Titillium+Web');
```

```
header#cabecalho h2 {  
  font-family: 'Titillium Web', sans-serif;  
  font-size: 15pt;  
  color: #888888;  
  padding: 0px;  
  margin-top: 0px;  
}
```

SEMÂNTICA EM HTML5

O HTML5 possui algumas tags que dão significado ao conteúdo da página.

SECTION: Cria uma sessão que vai poder conter vários artigos (ARTICLE) ou objetos.

ex:

```
<section id="corpo"></section>
```

ASIDE: São conteúdos periféricos, que estão diretamente relacionados ao conteúdo do artigo principal.

ex:

```
<aside="lateral"></section>
```

FOOTER: É o rodapé da página ou do conteúdo específico.

ex:

```
<footer id="rodape"></section>
```

Para configurar essas tags e utilizar o posicionamento correto, utilizaremos o CSS3 com os modificadores DISPLAY: BLOCK e o FLOAT para gerar uma flutuação.

ex:

```
section#corpo {  
  display: block;  
  width: 520px;  
  float: left;  
  border-right: 1px solid #606060;  
  padding-right: 15px;  
}
```

```
aside#lateral {  
    display: block;  
    width: 350px;  
    float: right;  
}
```

Caso seja necessário interromper a flutuação, utilizaremos o modificador CSS CLEAR:BOTH para eliminá-la.

```
footer#rodape {  
    clear: both;  
    border-top: 1px solid #606060;  
}
```

```
footer#rodape p {  
    text-align: center;  
}
```

TABELAS

Cada linha da tabela em HTML será uma Table Row, ou TR.

Cada célula de dados será uma Table Data, ou TD.

```
<table id="tabelaspec">  
    <caption>Tabela Técnica do Google Glass Mar/2013</caption>  
    <tr><td>Tela</td><td>Resolução equivalente a tela de 25"</td></tr>  
    <tr><td>Camera</td><td>5MP para fotos</td></tr>  
    <tr><td>720p para vídeos</td></tr>  
    <tr><td>Conectividade</td><td>Wi-Fi</td></tr>  
    <tr><td>Bluetooth</td></tr>  
    <tr><td>Memória Interna</td><td>12GB</td></tr>  
</table>
```

Observação: A partir do HTML5, os fechamentos das tags /TR e /TD são opcionais e podem ser omitidas.

Utilizando CSS3, podemos utilizar alguns modificadores para formatar a tabela, usando BORDER para criar bordas sólidas para TABLE, TR e TD.

O modificador BORDER-SPACING das CSS modifica o espaço entre as células de uma tabela.


```
table#tabelaspec {
  border: 1px solid #606060;
  border-spacing: 0px;
  margin-left: auto;
  margin-right: auto;
}
```

```
table#tabelaspec td {
  border: 1px solid #606060;
  padding: 10px;
}
```

No HTML, podemos realizar ROWSPAN, que vai criar uma expansão de linha nas células.

```
<table id="tabelaspec">
  <caption>Tabela Técnica do Google Glass Mar/2013</caption>
  <tr><td>Tela</td><td>Resolução equivalente a tela de 25"</td></tr>
  <tr><td rowspan="2">Camera</td><td>5MP para fotos</td></tr>
  <tr><td>720p para vídeos</td></tr>
  <tr><td rowspan="2">Conectividade</td><td>Wi-Fi</td></tr>
  <tr><td>Bluetooth</td></tr>
  <tr><td>Memória Interna</td><td>12GB</td></tr>
</table>
```

De maneira similar, utilizando COLSPAN, teremos uma expansão de coluna nas células de uma tabela HTML5.

ESTILOS CSS PARA TABELAS EM HTML5

Formataremos a tabela criada utilizando CSS3, com modificadores como VERTICAL-ALIGN.

```
table#tabelaspec td.ce {
  color: #ffffff;
  background-color: #606060;
  text-align: right;
  vertical-align: top;
  font-weight: bold;
}
```

Podemos formatar cada coluna de uma tabela com cores diferentes, utilizando o parâmetro CLASS diferentes para cada coluna..

```
table#tabelaspec td.ce {  
    color: #ffffff;  
    background-color: #606060;  
    text-align: right;  
    vertical-align: top;  
    font-weight: bold;  
}
```

```
table#tabelaspec td.cd {  
    background-color: #cecece;  
}
```

INTERFACES EM HTML5

Na versão 5, temos várias tags semânticas como HEADER, SECTION, ASIDE, NAV, FOOTER e ARTICLE.

Para criar um artigo em HTML5, utilize a tag ARTICLE. Os conteúdos relacionados a esse artigo podem ser tratados como ASIDE.

É importante saber que podemos usar a tag HEADER para qualquer tipo de cabeçalho, não apenas para o cabeçalho do site. Nessa aula, veremos que um ARTICLE também pode possuir um cabeçalho HEADER.

Para formatar os componentes, utilizamos alguns modificadores das CSS3, como:

margin, padding, color, background-color, font-family e muito mais.

INTRODUÇÃO AO JAVASCRIPT

A linguagem JS foi criada e popularizada pela empresa Netscape.

Há alguns comandos específicos e introdutórios do JavaScript, como o alert, document.write, Date

História do JavaScript: em 1992, a OpenWave estava criando a linguagem Cmm, rebatizada para ScriptEase, quando foi comprada pela Netscape. O projeto então passou a se chamar Mocha e logo em seguida (1995) teve seu nome modificado para LiveScript. Por conta de uma negociação com a Sun Microsystems (criadora do Java), uma jogada de marketing mudou o nome da linguagem para JavaScript.

JavaScript não é Java. Ela apenas usa nomes semelhantes por conta da jogada explicada anteriormente. Ambas se parecem por conta da origem, baseada em Linguagem C (C-like).

Atualmente, existe uma versão padronizada do JavaScript, o ECMAScript que está na versão 5.0. O JavaScript atualmente está na versão 1.8.5.

O JavaScript se popularizou e temos várias tecnologias relacionadas, como Ajax, jQuery e CommonJS.

JAVASCRIPT COM OBJETOS

Trabalhe com FUNCTION para criar funções que são disparadas de acordo com alguns eventos no HTML5, como ONCLICK e ONMOUSEMOVE.

Referencie objeto dentro do JavaScript com DOCUMENT.GETELEMENTBYID().

ex:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title>Teste JavaScript</title>
<script>
    function acendeLampada() {
        document.getElementById("luz").src="_imagens/lampada-acesa.jpg";
    }

    function apagaLampada() {
        document.getElementById("luz").src="_imagens/lampada-apagada.jpg";
    }

    function quebraLampada() {
        document.getElementById("luz").src="_imagens/lampada-quebrada.jpg";
    }
</script>
</head>

<body>
    <h1>Acenda a lâmpada</h1>
    
</body>
</html>
```

FUNÇÕES JAVASCRIPT

Como criar funções em JavaScript que recebem parâmetros.

ex:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title>Teste JavaScript</title>
<script>
    var quebrada = false;
    function mudaLampada(tipo) {
        if(!quebrada) {
            document.getElementById("luz").src = "_imagens/" + tipo + ".jpg";
            if (tipo == 'lampada-quebrada') {
                quebrada = true;
            }
        }
    }
</script>
</head>

<body>
    <h1>Acenda a lâmpada</h1>
    
</body>
</html>
```

INTEGRANDO JAVASCRIPT E HTML5

```
<script language="JavaScript" src="_javascript/funcões.js"></script>
<nav id="menu">
  <h1>Menu Principal</h1>
  <ul>
    <li onmouseover="mudaFoto('_imagens/home.png')"
onmouseout="mudaFoto('_imagens/glass-oculos-preto-peq.png')"><a
href="index.html">Home</a></li>
    <li onmouseover="mudaFoto('_imagens/especificacoes.png')"
onmouseout="mudaFoto('_imagens/glass-oculos-preto-peq.png')"><a
href="specs.html">Especificações</a></li>
    <li onmouseover="mudaFoto('_imagens/fotos.png')"
onmouseout="mudaFoto('_imagens/glass-oculos-preto-peq.png')"><a
href="fotos.html">Fotos</a></li>
    <li onmouseover="mudaFoto('_imagens/multimidia.png')"
onmouseout="mudaFoto('_imagens/glass-oculos-preto-peq.png')"><a
href="multimidia.html">Multimídia</a></li>
    <li onmouseover="mudaFoto('_imagens/contato.png')"
onmouseout="mudaFoto('_imagens/glass-oculos-preto-peq.png')"><a href="fale-
conosco.html">Fale conosco</a></li>
  </ul>
</nav>
```

Arquivo funcoes.js

```
function mudaFoto(foto) {
  document.getElementById("icone").src = foto;
}
```

DOCUMENTOS ADICIONAIS

Nos editores JetBrains, as teclas de atalho Ctrl + Alt + T (Win) ou Cmd + Opt + T (MacOS) servem para criar tags que podem delimitar trechos de código.

IFRAME EM HTML5

É possível adaptar as suas páginas HTML5 ao padrão definido pela sua interface, seguindo o mesmo esquema em todos os documento.

Vamos inserir uma imagem com a tag IMG, que será utilizada juntamente com um mapa de imagem.

Insira IFRAME em HTML5, utilizando os parâmetros NAME e ID.

ex:

```
<section id="conteudo">
  
  <iframe src="google-glass.html" name="janela" id="frame-spec"></iframe>
</section>
```

Faça as configurações de folha de estilo para formatar o IFRAME.

ex:

```
iframe#frame-spec {
  width: 380px;
  height: 280px;
  border: none;
  overflow: hidden;
}

iframe#frame-spec::-webkit-scrollbar {
  display: none;
}
```

MAPAS DE IMAGEM

Mapa de Imagem é uma técnica utilizada para criar múltiplos links em uma mesma imagem, criando áreas sensíveis ao clique em imagens para documentos HTML5.

Dentro da imagem criada com a tag IMG do HTML5, devemos usar um parâmetro USEMAP.

A tag utilizada para criar um mapa de imagem é a tag MAP no HTML5. A tag AREA do HTML5 permite que você defina o formato SHAPE e as coordenadas COORDS. Além disso, utilizaremos o TARGET para indicar o local onde o link será aberto.

Os SHAPE compatíveis com mapas MAP em HTML5 são RECT, CIRCLE e POLY.

ex:

```
<section id="conteudo">
  
  <map name="meumapa">
    <area shape="rect" coords="179,202,260,270" href="google-glass.html#tela"
    target="janela"/>
    <area shape="circle" coords="158,243,12" href="google-glass.html#camera"
    target="janela"/>
    <area shape="circle" coords="73,52,12" href="google-glass.html#gadgets"
    target="janela"/>
    <area shape="poly" coords="28,143,83,216,84,249,27,169" href="google-
    glass.html#sensores" target="janela"/>
  </map>
  <iframe src="google-glass.html" name="janela" id="frame-spec"></iframe>
</section>
```

ÁLBUM DE FOTOS EM HTML5

Há uma maneira bem criativa de criar um álbum de fotos utilizando apenas listas e folhas de estilo.

Utilizando as tags UL e LI, criaremos listas em HTML5. Cada um dos LI serão utilizados para carregar fotos. Utilizando a tag SPAN, criaremos legendas interativas para as fotos.

Em CSS3, utilizaremos modificadores como WIDTH, PADDING, MARGIN, OVERFLOW e LIST-STYLE para configurar características da lista.

ÁUDIO E VÍDEO EM HTML5

A nova versão da linguagem suporta a inserção de conteúdo multimídia com HTML5.

Buscar e baixar conversor de audio e vídeo em <http://www.getmiro.com/download/> e converter o arquivo na pasta "_media" em formato MP3 para OGG.

Usar a ferramenta iTunes para converter o vídeo do formato MP4 para x-aac (ficam com a extensão m4a).

Com a tag AUDIO é possível adicionar um áudio a um site com HTML5.

A tag VIDEO permite adicionar um vídeo a um site com HTML5.

```
<div id="tv-radio">
  <audio id="musica" controls="controls">
    <source src="_media/2009-lovers-carvings-bibio.mp3" type="audio/mpeg"/>
    <source src="_media/2009-lovers-carvings-bibio.m4a" type="audio/x-acc"/>
    <source src="_media/2009-lovers-carvings-bibio.ogg" type="audio/ogg"/>
    Desculpe, mas não foi possível carregar o áudio!
  </audio>

  <video id="filme" controls="controls" poster="_imagens/video-mini03.jpg">
    <source src="_media/getting-started.mp4" type="video/mp4"/>
    <source src="_media/getting-started.webm" type="video/webm"/>
    <source src="_media/getting-started.ogv" type="video/ogg"/>
    Desculpe, mas não foi possível carregar o vídeo!
  </video>
</div>
```

Há técnicas avançadas para a configuração de áudio e vídeo em um site em HTML5. Na verdade, aplica-se um estilo CSS para tornar a visualização mais agradável.

FORMULÁRIOS EM HTML5

Conheça as tags novas para formulários no HTML5, como os tipos NUMBER, DATE e EMAIL.

A tag FORM serve para criar um formulário, onde o usuário vai poder interagir com seu site e fornecer dados.

A tag FIELDSET em HTML cria conjuntos de campos agrupados dentro de uma linha container que vai identificar cada uma das sessões.

A tag LEGEND cria uma legenda para cada FIELDSET, um texto que fica sobre as linhas.

O parâmetro PLACEHOLDER configura um conteúdo informativo dentro de um objeto de formulário. Ao clicar sobre o objeto, o PLACEHOLDER desaparecerá.

Para criar uma caixa de senha em um formulário HTML, utilizaremos a tag INPUT com TYPE = "password"

Caixas de senha também suportam SIZE, MAXLENGTH e PLACEHOLDER.

Para solicitar um e-mail, utilizaremos INPUT com TYPE = "email", que é exclusivo da HTML5.

Para criar opções a serem selecionadas na forma de botões de rádio, utilizaremos INPUT com parâmetro TYPE = "radio". Para deixar uma opção previamente marcada, utilizamos o parâmetro CHECKED.

OBS: Para objetos do tipo RADIO que fazem parte de um mesmo grupo, todos devem ter o mesmo NAME. Isso é muito importante.

Para criar palavras relacionadas a objetos de formulário, utilizamos a tag LABEL com um ID. Para criar a relação entre objetos e LABEL, utilizamos o parâmetro FOR do objeto.

A tag INPUT com parâmetro TYPE = "date", cria uma caixa de seleção de data, com um calendário (depende do navegador utilizado).

Para permitir a entrada de valores numéricos, utilizamos INPUT com parâmetro TYPE="number". Nesse caso, os parâmetros MIN e MAX definem os valores mínimo e máximo para caixas desse tipo.

ex:

```
<form>

  <fieldset id="usuario">
    <legend>Identificação do Usuário</legend>
    <p><label for="cNome">Nome:</label> <input type="text" name="tNome"
id="cNome" size="20" maxlength="30" placeholder="Nome Completo"/></p>
    <p><label for="cSenha">Senha:</label> <input type="password" name="tSenha"
id="cSenha" size="8" maxlength="8" placeholder="8 dígitos"/></p>
    <p><label for="cMail">E-mail:</label> <input type="email" name="tMail"
id="cMail" size="20" maxlength="40"/></p>
  </fieldset>
  <fieldset>
    <legend>Sexo:</legend>
    <input type="radio" name="tSexo" id="cMasc"/> <label
for="cMasc">Masculino</label><br />
    <input type="radio" name="tSexo" id="cFem"/> <label
for="cFem">Feminino</label>
  </fieldset>
  <p>Data de Nascimento: <input type="date" name="tNasc" id="cNasc"/></p>
</fieldset>

  <fieldset id="endereco">
    <legend>Endereço do Usuário</legend>
    <p><label for="cRua">Logradouro:</label><input type="text" name="tRua"
id="cRua" size="13" maxlength="80" placeholder="Rua Av. Trav., etc."/></p>
    <p><label for="cNum">Número:</label><input type="number" name="tNum"
id="cNum" min="0" max="99999"/></p>
    Estado:
    Cidade:
  </fieldset>
</form>
```

Para a tag FORM, o parâmetro METHOD aceita os valores GET e POST:

- O método GET envia dados via URL (endereço do site). É um método mais rápido e menos seguro.

- O método POST envia dados por pacotes. É um método um pouco mais lento, mas possui uma segurança maior.

O parâmetro ACTION indica o local para onde os dados serão enviados. Quem utiliza MAILTO, deve ter um cliente de e-mail previamente configurado.

ex:

```
<form method="post" id="fContato" action="mailto:contato@cursoemvideo.com">
```

A tag SELECT cria caixas combinadas, utilizando a tag OPTION para criar as opções.

O parâmetro SELECTED serve para tornar um OPTION padrão. Os valores podem ser definidos com o parâmetro VALUE.

```
<p><label for="cEst">Estado: </label>
  <select name="tEst" id="cEst">
    <optgroup label="Região Sudeste">
      <option value="RJ">Rio de Janeiro</option>
      <option value="SP" selected>São Paulo</option>
      <option value="MG">Minas Gerais</option>
    </optgroup>
    <optgroup label="Região Sul">
      <option value="PR">Paraná</option>
      <option value="SC">Santa Catarina</option>
      <option value="RS">Rio Grande do Sul</option>
    </optgroup>
  </select>
</p>
```