

**Objetivo:** assimilar conceitos básicos para construção de páginas web usando HTML5 e CSS. Tais ferramentas serão úteis no desenvolvimento do trabalho de programação. HTML5 e CSS propiciarão o desenvolvimento da interface gráfica cuja funcionalidade será implementada em Prolog.

**Desenvolvimento:** o procedimento a ser seguido consiste na realização de **vários passos** a serem seguidos na **ordem especificada**. Tais passos são descritos a seguir.

### **Passo 1:** Criando sua **primeira página usando HTML5**

Para criar a página web você pode fazer uso do **notepad do windows**. Como estamos usando o SWI-Prolog use o **emacs** que apresenta facilidades para a edição do código. Observe que existem ferramentas muitas sofisticadas para a criação de páginas web tais como o popular *DreamWeaver*. Quando você entender os princípios básicos do processo de construção de páginas você poderá fazer uso desta ou de outras ferramentas sofisticadas. Inicialmente, usando seu **editor emacs** abra um **novo arquivo com tipo html** (por exemplo, pag1.html) e digite:

```
<!DOCTYPE HTML>
<html lang="pt-br">
<head>
  <title>Minha Primeira Página</title>
</head>

<body>

<!-- Conteúdo principal -->
<h1>Minha Primeira Página Web</h1>

<p>Seja bem-vindo!

<p> Ainda não sei o que colocar nesta página.

<!-- dados relativos a criação e autor! -->
<address>Construída em Novembro/2011<br>
  por José da Silva.</address>

</body>
</html>
```

Como você provavelmente deve ter pensado (e até mesmo já feito) não é necessário escrever cada linha do código acima no seu editor. Use o **procedimento copy & paste**.

A primeira linha do arquivo HTML informa ao seu navegador que tipo de documento (DOCTYPE da expressão *DOCument TYPE*) é usado neste programa. **Neste caso é HTML.**

As palavras que aparecem entre **< e >** são chamadas **tags** e como você pode ver o documento está contido dentro das tags **<html> e </html>**. Entre **<head> e </head>** existe espaço para se inserir **vários tipos de informações**, o que não é mostrado na tela. Até o momento encontra-se especificado o **título da página**. Os **elementos podem conter atributos**. Neste exemplo o **elemento html possui atributo LANG**. O atributo LANG é necessário para que os usuários saibam qual a linguagem principal do documento. No caso do código acima o valor do atributo LANG é “pt-br” por se tratar da língua portuguesa falada no Brasil.

A tag `<body>` é onde o texto do documento será especificado. Toda informação colocada entre as tags será mostrada ao visitante da sua página, exceto o texto que aparece entre `<!--` e `-->`, que refere-se a um comentário. O navegador não mostrará tal informação ao nosso visitante.

A tag `<p>` especifica a inserção de um parágrafo.

E quanto à tag `<h1>`? Alguma ideia da sua finalidade? *título*

A seguir salve o arquivo e use um navegador web para visualizar o seu conteúdo.

## Passo 2: Adicionando mais informações à página.

Vamos inserir informações para o nosso visitante de tal modo que ele possa ter acesso a nossas músicas e filmes prediletos. O que desejamos fazer neste momento é estabelecer links para estas informações. Os links (ou melhor hiperlinks) são visualizados por palavras em destaque que quando selecionadas nos levam a uma outra página web com a informação desejada. Vejamos como fazer isto. Para isto modifique o código especificado anteriormente para que ele fique da seguinte forma:

```
<!DOCTYPE HTML>
<html lang="pt-br">
<head>
  <title>Minha Primeira Página</title>
</head>

<body>

<!-- menu -->
<ul class="navbar">
  <li><a href="musicas.html">Minhas Músicas Prediletas </a>
  <li><a href="filmes.html">Meus Filmes Preferidos </a>
</ul>

<!-- Conteúdo principal -->
<h1>Minha Primeira Página Web</h1>

<p>Seja bem-vindo!

<p> Ainda não sei o que colocar nesta página.

<!-- dados relativos a criação e autor! -->
<address>Construída em Novembro/2011<br>
  por José da Silva.</address>

</body>
</html>
```

*nome da ul*

Observe que a informação em destaque está em negrito. Com relação às tags que aparecem no código inserido temos que a tag `<ul>` introduz uma lista não ordenada, que é uma lista em que os itens não são numerados. A tag `<li>` especifica um item da lista enquanto a tag `<a>` permite criar um *hiperlink*.

O objetivo neste momento é mostrar que `ul` pode servir como um menu, mais especificamente um menu de navegação que permite ligar esta página a outras páginas. Isto significa que todas as nossas páginas terão um menu similar a este. Os elementos `h1` e `p` formam o conteúdo único desta página enquanto a assinatura na parte inferior será a mesma em todas as páginas. Da mesma como feito no item anterior gere esta página no editor e visualize seu conteúdo no navegador.

### Passo 3: Adicionando cores e fontes a página

Neste passo apresentaremos a você um outro padrão web, que é **CSS**. Veremos como adicionar estilo a uma página web, ou mais especificamente uma **folha de estilo**. Você ao visualizar o conteúdo da página especificada anteriormente percebeu o texto em cor preta sobre um fundo de cor branca. Vamos adicionar algumas cores a esta página de modo a torná-la mais atrativa.

Inicialmente vamos iniciar com a **folha de estilos dentro do arquivo HTML**. Posteriormente iremos colocá-la num **arquivo separado**. Colocar as informações de estilo em um arquivo separado é uma boa prática pois podemos usá-la para diferentes arquivos HTML.

É preciso adicionar um elemento `<style>` ao arquivo HTML. A folha de estilo será **especificada dentro do elemento**. Abra o editor e altere o conteúdo HTML para o conteúdo que aparece especificado a seguir. A folha de estilos aparece em destaque no cabeçalho do arquivo HTML.

```
<!DOCTYPE HTML>
<html lang="pt-br">
<head>
  <title>Minha Primeira Página</title>
  <style type="text/css">
    body {
      color: purple;
      background-color: #d8da3d }
  </style>
</head>

<body>
[etc.]
```

A primeira linha diz que esta é uma **folha de estilo e que é escrita em CSS**. A segunda linha especifica que se **adicionou estilo ao elemento *body***. A terceira linha faz com que **a cor do texto se torne lilás e a próxima faz com que o fundo se torne amarelo**.

As folhas de estilo em **CSS são construídas usando-se regras**. Cada regra tem três partes:

- **seletor**: informa ao navegador que parte será afetada pela regra (no exemplo *body*);
- **propriedade**: especifica que aspecto do layout será alterado (no exemplo correspondem a *color* e *background-color*);
- **valor**: especificam o valor da propriedade (no exemplo correspondem a *purple* e *#d8da3d*).

O exemplo mostra que **regras podem ser combinadas**. As propriedades *color* e *background-color* poderiam ser especificadas da seguinte forma:

```
body {color: purple}
body {background-color: #d8da3d}
```

Salve o conteúdo do arquivo e o visualize por meio do navegador para verificar o resultado,

### Passo 4: Adicionando um menu de navegação

A lista especificada no topo da página HTML foi colocada para tornar-se um menu de navegação conforme mencionado anteriormente. Muitos websites têm algum tipo de menu que aparece no topo

ou na lateral da página. Faremos isto para a página que estamos criando. Vamos colocar o menu na lateral esquerda da página.

O menu já se encontra especificado na página pela *tag ul*. O que precisamos fazer é mover a lista para a esquerda e mover o resto do texto para a direita. As propriedades CSS para isto são '*padding-left*' (para mover o texto) e *position, left e top* (para mover o menu). Existem outras formas de se fazer isto. Se você procurar por *column* ou *layout* em [Learning CSS](#) você encontrará vários *templates* prontos para serem usados. A seguir apresenta-se o código especificando o menu na lateral. Mais uma vez salve o arquivo e veja o seu conteúdo no navegador.

```
<!DOCTYPE HTML>
<html lang="pt-br">
<head>
  <title>Minha Primeira Página</title>
  <style type="text/css">
    body {
      padding-left: 11em;
      font-family: Georgia, "Times New Roman",
        Times, serif;
      color: purple;
      background-color: #d8da3d }
    ul.navbar {
      position: absolute;
      top: 2em;
      left: 1em;
      width: 9em }
    h1 {
      font-family: Helvetica, Geneva, Arial,
        SunSans-Regular, sans-serif }
  </style>
</head>
<body>
...
```

A especificação *position: absolute* informa que o elemento *ul* é posicionado de forma independente de qualquer texto que venha antes ou depois. Os atributos *left* e *top* indicam a posição, que neste exemplo 2em do topo e 1em do lado esquerdo da janela. O valor 2em corresponde a duas vezes o tamanho da fonte corrente. Por exemplo, se o menu é visualizado com uma fonte de 12 pontos então 2em corresponde a 24 pontos. A medida *em* é uma unidade muito útil em CSS pois pode adaptar-se a fonte que o leitor possa estar usando. A maioria dos navegadores apresenta uma funcionalidade para aumentar ou diminuir o tamanho da fonte. Experimente fazer isto e verificar que o menu aumenta em tamanho quando a fonte aumenta. Isto não seria obtido se tivéssemos usado pixels como medida.

### Passo 5: Criando um arquivo CSS

Até o momento temos um arquivo HTML com uma folha de estilos. Observe, contudo, que se o nosso sítio aumenta provavelmente teremos muitas páginas compartilhando de um mesmo estilo. Para se obter isto um primeiro raciocínio é copiar a folha de estilo em cada página. Contudo existe um método melhor. A ideia é colocar a folha de estilos num arquivo separado com cada página web fazendo referência a este arquivo. Para isto *cut e paste* tudo que esteja dentro do elemento *<style>* e coloque dentro de um arquivo a ser criado com tipo css, por exemplo *estilo.css*. Lembre-se que a tag *<style>* não deve fazer parte do conteúdo. Portanto, o arquivo *estilo.css* terá o seguinte código:

```
body {
```

```
padding-left: 11em;
font-family: Georgia, "Times New Roman",
    Times, serif;
color: purple;
background-color: #d8da3d }
ul.navbar {
    position: absolute;
    top: 2em;
    left: 1em;
    width: 9em }
h1 {
    font-family: Helvetica, Geneva, Arial,
        SunSans-Regular, sans-serif }
```

Já o arquivo HTML fica reduzido a:

```
<!DOCTYPE HTML>
<html lang="pt-br">
<head>
    <title>Minha Primeira Página</title>
    <link rel="stylesheet" href="estilo.css">
</head>

<body>
...
```

O elemento *link* que agora aparece informa ao navegador que a folha de estilos está especificada no arquivo *estilo.css* e desde que nenhum diretório é mencionado isto significa que ele estará no mesmo diretório do arquivo HTML.

## Passo 6: criando um formulário

Neste passo vamos criar uma página web que contenha um formulário. Um formulário é um componente de uma página web que possui elementos tais como campos de texto, botões, *checkboxes* etc. Um usuário pode interagir com um formulário por fornecer dados que serão enviados a um servidor para processamento posterior. Neste passo vamos criar um formulário para a especificação de certa pizza a ser entregue.

Todo formulário começa com um elemento *form*, dentro dos quais são colocados os controles. A maioria destes controles são representados por um elemento *input*, que corresponde a um campo de texto de uma linha. Para rotular um controle o elemento *label* é usado. O texto do rótulo e o elemento *input* são especificados dentro do elemento *label*. Cada parte de um formulário é considerado um parágrafo e é tipicamente separado dos outros pelo uso de elementos *p*. Observe a seguir a especificação do formulário para solicitação de pizzas.

```
<form>
  <p><label>Nome do Cliente: <input></label></p>
  <p><label>Telefone: <input type=tel></label></p>
  <p><label>E-mail: <input type=email></label></p>
  <fieldset>
    <legend> Tamanho da Pizza </legend>
    <p><label> <input type=radio name=size> Pequena </label></p>
    <p><label> <input type=radio name=size> Média </label></p>
    <p><label> <input type=radio name=size> Grande </label></p>
  </fieldset>
  <fieldset>
    <legend> Ingredientes </legend>
    <p><label> <input type=checkbox> Tomate </label></p>
    <p><label> <input type=checkbox> Presunto </label></p>
```

```

<p><label> <input type=checkbox> Cebola </label></p>
<p><label> <input type=checkbox> Queijo </label></p>
</fieldset>
<p><label>Horário de entrega desejado: <input type=time min="11:00"
max="21:00" step="900"></label></p>
<p><label>Instruções para entrega: <textarea></textarea></label></p>
<p><button>Enviar Pedido</button></p>
</form>

```

Crie um arquivo de tipo HTML insira o conteúdo especificado anteriormente e visualize a página por meio do navegador.

Vamos descrever os demais elementos que aparecem dentro de *form* e que não foram mencionados anteriormente. Com a visualização do exemplo feito anteriormente esta tarefa ficará mais fácil. O elemento *fieldset* serve principalmente para agrupar elementos *input*. No exemplo percebemos agrupamentos de tamanho e sabores de pizza. No agrupamento de tamanho a seleção é feita em função de *inputs* associados a botões de rádio (*radio buttons*). No agrupamento de sabores a entrada está agrupada por meio de *checkboxes*. Os demais elementos do formulário constituem-se de um elemento *input* associado a tempo, uma área de texto (*textarea*) para descrição de instruções para entrega e um botão (*button*) para submeter o pedido. Cabe aqui um detalhamento sobre o *input* associado a tempo. Nele podemos especificar um valor mínimo e um valor máximo para tempo em horas. Além disso podemos especificar um incremento de tempo (em segundos). No nosso exemplo a pizzeria poderá entregar pizzas no horário das 11 às 23 horas em intervalos de 15 minutos (900 segundos).

A especificação HTML para o formulário ainda não está completa. Precisamos acrescentar informações necessárias para que o servidor possa processar o conteúdo especificado. Para cada elemento *input* do formulário é necessário fornecer um nome que será usado para identificar os dados na submissão. Nós já demos um nome para o grupo de *radio buttons*. O atributo (*name*) também especifica o nome da submissão. Os botões podem ser diferenciados por dar a eles diferentes valores usando o atributo *value*. Em função destas observações passará a ter a seguinte especificação:

```

<form>
<p><label>Nome do Cliente: <input name="nomeCliente" ></label></p>
<p><label>Telefone: <input type=tel name="telCliente"></label></p>
<p><label>E-mail: <input type=email name="emailCliente"></label></p>
<fieldset>
<legend> Tamanho da Pizza </legend>
<p><label> <input type=radio name=size value="p"> Pequena </label></p>
<p><label> <input type=radio name=size value="m"> Média </label></p>
<p><label> <input type=radio name=size value="g"> Grande </label></p>
</fieldset>
<fieldset>
<legend> Ingredientes </legend>
<p><label> <input type=checkbox name="ing" value="tom"> Tomate </label></p>
<p><label> <input type=checkbox name="ing" value="pre"> Presunto</label></p>
<p><label> <input type=checkbox name="ing" value="ceb"> cebola </label></p>
<p><label> <input type=checkbox name="ing" value="que"> Queijo </label></p>
</fieldset>
<p><label>Horário de entrega desejado: <input type=time min="11:00" max="21:00"
step="900" name="tempo"></label></p>
<p><label>Instruções para entrega: <textarea name="obs"></textarea></label></p>
<p><button>Enviar Pedido</button></p>
</form>

```

Posteriormente será apresentado o procedimento para se associar ações aos elementos do formulário.