

Aula 07 – Folhas de Estilo (CSS)

As folhas de estilo (*Cascading Stylesheets*) são uma forma bem conhecida de manipular a aparência de páginas HTML. As folhas de estilo permitiram uma separação entre conteúdo e apresentação das informações contidas nos arquivos HTML.

Utilizando CSS com HTML

No passado vimos que a separação entre conteúdo e apresentação para arquivos HTML resultou em um ganho de tempo na administração de sites e uma possibilidade de mudança rápida na aparência de diversas páginas ao mesmo tempo.

Usando um arquivo css externo podíamos mapear características de formatação para determinadas tags nesse arquivo (como no exemplo abaixo):

```
h1 {  
    font-size: 24pt;  
    font-family: arial;  
    color: blue  
}
```

Arquivo teste.css

Depois era só usarmos a tag <link> dentro da seção <head> do html para chamar esse arquivo:

```
<html>  
  <head>  
    <link rel="stylesheet" type="text/css" href="teste.css">  
    <title>Teste CSS</title>  
  </head>  
  <body>  
    <h1> Teste de HTML com CSS </h1>  
  </body>  
</html>
```

Utilizando CSS com XML

Sabemos que a princípio o XML não é um formato utilizado para apresentar dados, para tornarmos os dados do XML “apresentáveis” precisamos recorrer a outros formatos para que o resultado final seja mais compreensível pelo usuário. Para apresentarmos os dados podemos usar o CSS aplicado a arquivos XML.

Ao visualizarmos um arquivo XML sem nenhum tipo de folha de estilo anexada à ele veremos apenas a estrutura em árvore do arquivo XML e suas tags como o exemplo abaixo da figura 1.

Como no exemplo as tags são auto-explicativas é possível ler sem problema o que consta no arquivo e ainda entendermos o seu significado, mas pode ser que as tags não sejam tão explicativas ou você queira ocultar as tags do usuários.

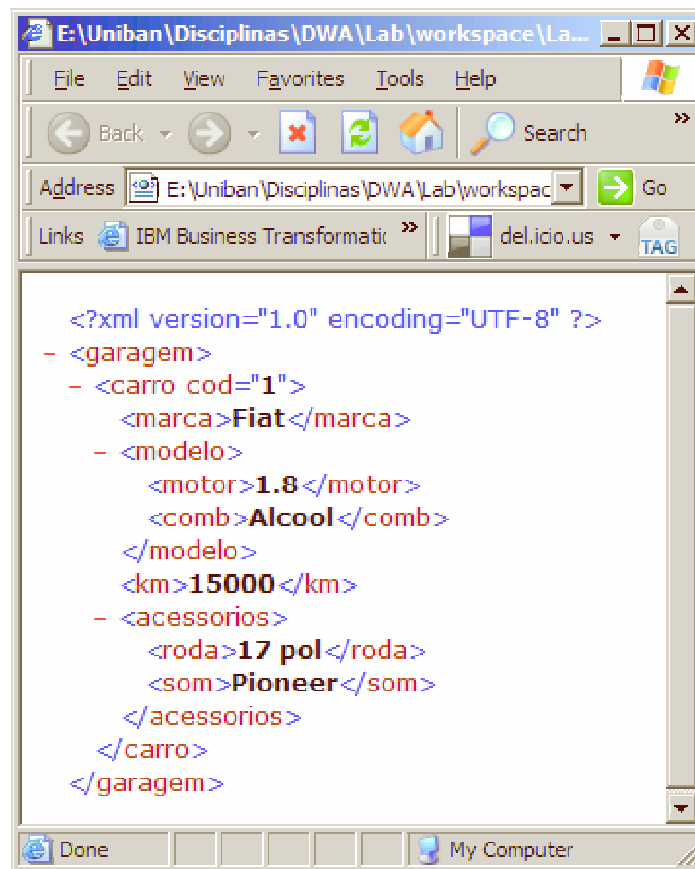


figura 1: Browser apresentando um arquivo XML sem CSS

O primeiro passo para formatar os dados para apresentação é anexar a declaração para o arquivo CSS no seu arquivo XML usando a seguinte linha após a declaração xml:

```
<?xml-stylesheet type="text/css" href="estiloGaragem.css" ?>
```

Vamos começar o nosso arquivo *estiloGaragem.css* com os seguintes dados:

```
* { display: block; }
```

Essa informação indica que todos os dados (*) do XML devem ser apresentados em seu respectivos blocos, o resultado prático dessa folha de estilo é o apresentado na figura 2.

Podemos ainda mostrar os dados referentes a uma tag usando **display: inline**, isso faz com que todos os dados da tag que está sendo tratada pelo estilo apareçam na mesma linha, como mostrado na figura 3. O browser Opera que é o mais aderente aos padrões ditados pelo W3C já assume automaticamente uma instrução `* { display: inline; }` mesmo que o arquivo xml não tenha nenhuma ligação com folhas de estilo.

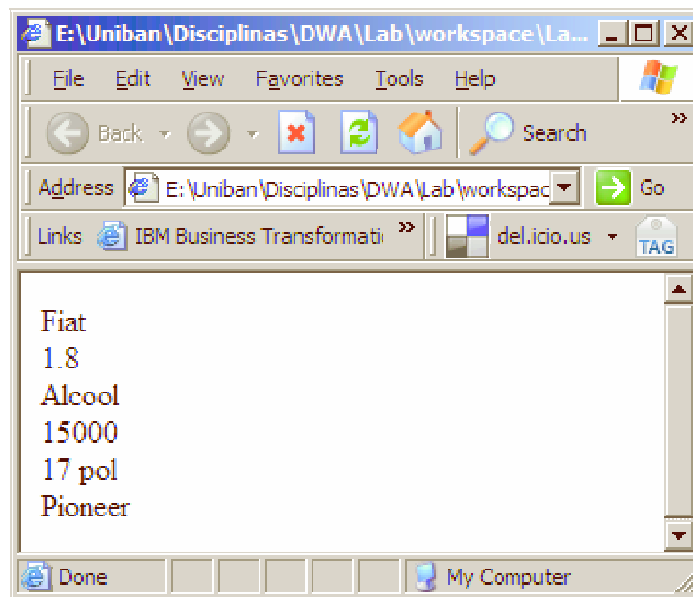


figura 2: Mostrando os dados em Blocos

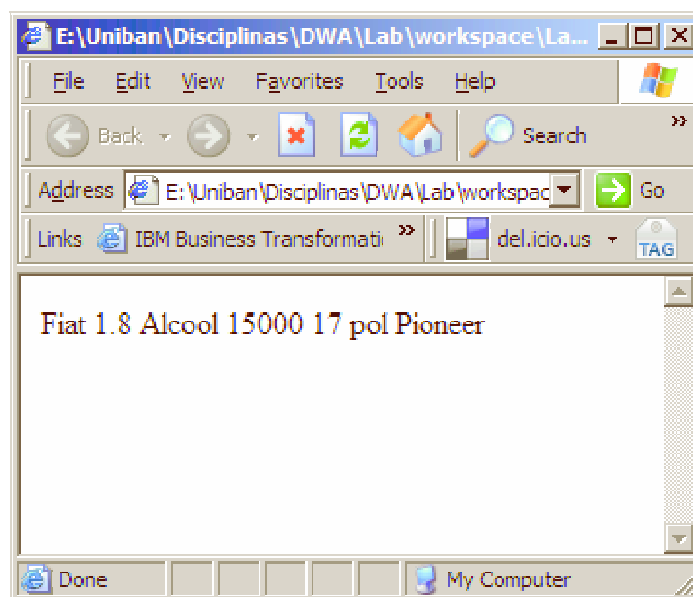


figura 3: Mostrando os dados "inline"

Podemos agora modificar o arquivo *estiloGaragem.xml* para mapear as tags específicas do arquivo xml, usando as linhas de css:

```
carro * { display: block; }  
modelo * { display: inline;  
           color: red; }
```

Isso fará com que todos os elementos carro e todos os elementos (*) que estão dentro dele sejam mostrado em seus respectivos blocos e apenas o modelo e os filhos dele sejam mostrados na mesma linha. (adicionei a cor vermelha para ser mais visível no exemplo). Obteremos o resultado mostrado na figura 4.

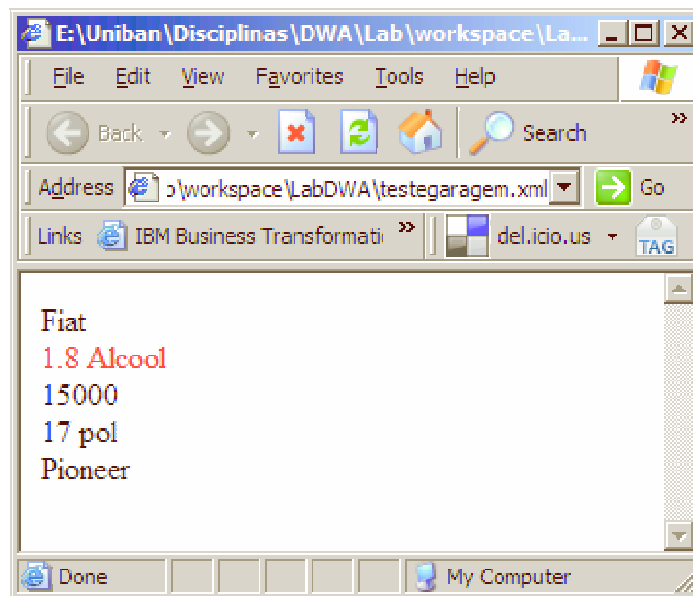


figura 4: Formatação do XML com block, inline e red

A propriedade *display* possui ainda os valores:

none: para não exibir os dados para determinado elemento

inherit: faz com que os elementos filhos herdem as propriedades definidas como *inherit*

Gerando conteúdo no XML usando CSS

Podemos ainda adicionar conteúdo que não estava presente originalmente no XML para melhorar a apresentação dos dados para o usuário. É possível inserir conteúdo antes ou depois de um elemento usando os especificadores **:before** ou **:after** no nome da tag (dentro do css) e dentro do trecho respectivo da tag no css é usada uma propriedade chamada **content** pra especificar o texto a ser inserido.

No exemplo abaixo os códigos CSS são responsáveis por inserir texto para os elementos `<marca>` e `<km>` do XML

```
marca:before {
    content: "Marca do Carro: ";
    color: green;
}
km:after {
    content: " kilometros";
    color: green;
}
```

Esse CSS combinado com o restante dos exemplos anteriores leva aos resultado da figura 5, note que agora estamos usando o browser Opera, pois este apresenta o melhor suporte aos padrões do W3C.

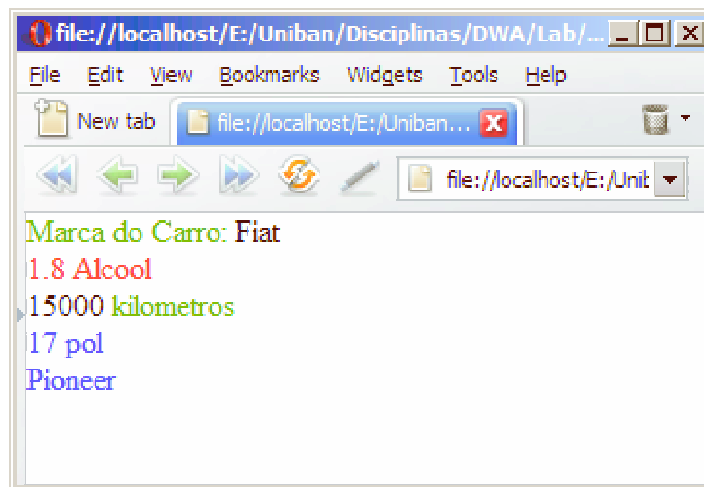


figura 5: Elementos em verse inseridos pelo CSS

Usando os atributos do XML

Podemos trabalhar ainda com os atributos de determinado elemento fazendo com que esses apareçam no conteúdo mostrado ao usuário:

```
carro:before {  
    content: "Cod: " attr(cod);  
}
```

No exemplo acima inserimos um conteúdo antes do elemento `<carro>` que irá mostrar o texto "Cod: " seguido do valor do atributo `cod` do elemento, como na figura 6.

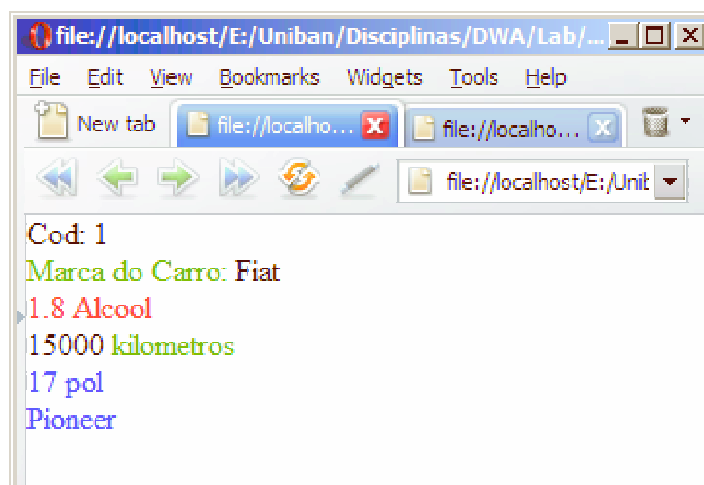


figura 6: A string "Cod: 1" foi inserida antes de `<carro>`

Contadores CSS

O browser Opera suporta ainda contadores CSS, contadores são úteis para arquivos XML muito extensos e que possuem muito elementos do mesmo tipo, usando contadores você consegue numerar esses elementos.

```
carro:before {  
    content: " Carro no." counter(carro);  
    counter-increment: carro;  
    color: blue;  
}
```

No exemplo acima adicionamos um contador para ser exibido sempre antes (:before) do elemento carro ser exibido, note o comando **counter(carro)** para a criação do contador e depois a propriedade **counter-increment: carro** que incrementa o contador, podemos ver o resultado do estilo aplicado no XML com o trecho em azul na figura 7.

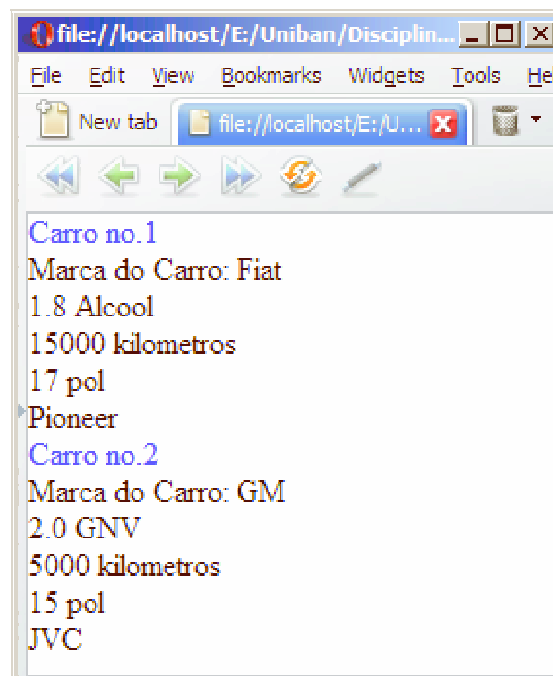


figura 7: Contadores CSS em azul

Existe ainda a propriedade **counter-reset: carro**, que é usada para resetar o contador que é passado como valor da propriedade (no exemplo usamos o contador carro, então esse contador será resetado)

Listas em CSS

Existe a possibilidade de exibir os elementos de um XML em forma de lista no browser usando o seguinte estilo:

```
carro {  
    display: list-item;  
    list-style: circle;  
    margin-left: 2em;  
}
```

Esse estilo faz com que todo elemento carro seja mostrado em forma de lista, usando como marcador um círculo, como podemos ver na figura 8.

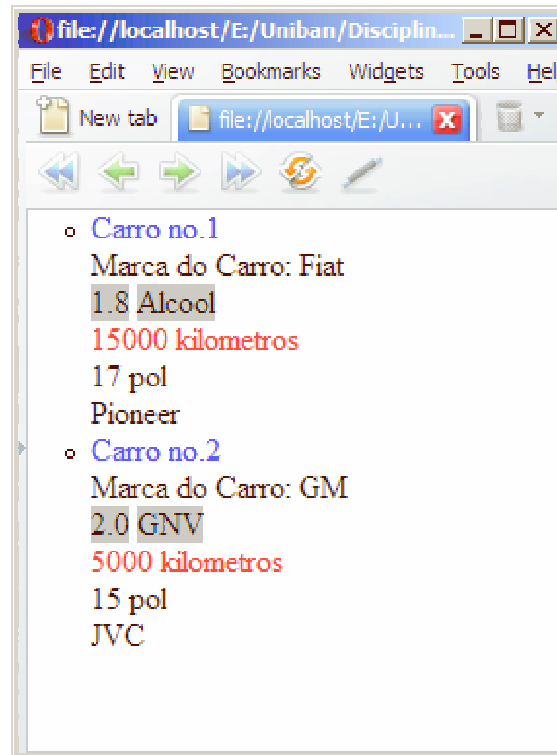


figura 8: Elementos exibidos em forma de lista