

HTML 8

Começamos este exercício apresentando o conceito de canvas. A tradução de canvas é tela a óleo e no HTML ele é uma área retangular da tela onde podem ser feitas operações gráficas: preencher, desenhar linhas e círculos, usar gradientes, e assim por diante. A tag é:

```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;"></canvas>
```

Praticamente todas as operações são feitas com Javascript razão pela qual este assunto será deixado para o futuro.

Outra ferramenta de desenho é o SVG (Scalable Vector Graphics) que é um padrão web para desenhos escaláveis. Desde o HTML5 surgiu o elemento <svg> que é um contendor para desenhos SVG. Eis um exemplo desta tag

```
<svg width="100" height="100">
<circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow" /> </svg>
```

Exercício 1

Crie um arquivo desenh.html e defina as duas tags acima. Veja o resultado.

As diferenças entre SVG e canvas podem ser resumidas

- SVG é uma linguagem para descrever gráficos 2D em XML
- em SVG cada forma é lembrada como um objeto. Se atributos do objeto SVG são alterados, o browser automaticamente redesenha o objeto.
- o canvas é renderizado pixel a pixel. Quando o canvas é desenhado o browser esquece dele. Se algum atributo muda tudo é redesenhado incluindo objetos que estão por trás do canvas
- em resumo canvas é mais eficiente mas menos preciso.

Multimídia

O assunto agora é multimídia. Hoje a Internet está repleta de conteúdos deste tipo: fotos, música, sons, vídeos, filmes, animações etc. Note-se que os primeiros browsers só suportavam texto limitados a um único fonte em uma única cor. Mais tarde passaram a suportar fotos (além de fontes e cores). O suporte de sons, animação e vídeos, é manuseado diferentemente pelos vários browsers. Diversos formatos são suportados e muitos deles exigem programas extra conhecidos como *plug-ins*. O HTML5 se propõe a padronizar a multimídia.

Vídeo

A seguir, os principais padrões de vídeo:

MPEG Arquivos terminados em .mpg Desenvolvido pelo MPEG. O primeiro formato popular na web. Suportado por todos os browsers, mas não suportado em HTML5.

AVI Desenvolvido pela Microsoft. Não é suportado por todos os browsers.

WMV Idem acima.

MOV Desenvolvido pela Apple. Roda bem em computadores desta marca, mas não em todos os browsers

RM Real Vídeo. Usado para vídeo online (streaming), mas não roda em todos os browsers

Flash Desenvolvido pela Macromédia. Com plug-ins roda nos browsers

OGG Formato livre desenvolvido pela Xiph.Org. Padrão HTML5

webm Desenvolvido pela Mozilla, Opera, Adobe e Google. Padrão HTML5

MP4 Desenvolvido pelo MPEG. Suportado pelo HTML5 e recomendado pelo youtube

Som

Já os formatos de som são:

MIDI Não é som e sim notas digitais. Não é suportado pelos browsers.

RM Real Audio. Usado para audio online (streaming), mas não roda em todos os browsers

WMA Desenvolvido pela Microsoft. Roda legal em windows, mas não em todos os browsers

AAC Desenvolvido pela Apple para o iTunes. Não roda em todos os browsers

WAV Desenvolvido pela IBM e Microsoft. Suportado pelo HTML5

OGG Formato livre desenvolvido pela Xiph.Org. Padrão HTML5

MP3 O som do MP4. É suportado por todos os browsers. Padrão HTML5.

Exercício 1

Crie o arquivo video.html incluindo nele

```
<video width="400" controls>
<source src="mov_bbb.webm" type="video/webm">
  Seu browser não suporta video HTML5.
</video> <p>
Video cortesia de
<a href="http://www.bigbuckbunny.org/" target="_blank">Big Buck Bunny</a>.
</p>
```

Veja o resultado em um browser. Lembre que o IE 8 e anteriores não suportam a tag video. O atributo controls acrescenta os botões play, pause e volume. O tamanho do video deve ser fornecido senão a construção da página pode sofrer flicker. O texto entre <video> e </video> só é apresentado se o browser não suporta a tag <video>. Pode-se colocar múltiplos source para diferentes arquivos de vídeo. O browser usa o primeiro formato reconhecido. O atributo autoplay começa o vídeo assim que a página for carregada.

```
<video width="320" height="240" autoplay>
...
```

Como anda o suporte de vídeo (em jan/2015)

| Browser | MP4 | WebM | Ogg |
|---------|--------------|------|-----|
| IE | SIM | NAO | NAO |
| Chrome | SIM | SIM | SIM |
| Firefox | SIM | SIM | SIM |
| Safari | SIM | NAO | NAO |
| Opera | SIM(Opera25) | SIM | SIM |

O assunto agora é áudio. Para executar um arquivo mp3 a tag é <audio>. Veja

```
<audio controls>
<source src="Horse.mp3" type="audio/mpeg">
Browser não suporta o elemento audio </audio>
```

Eis a situação atual em suporte a áudio

| Browser | MP3 | Wav | Ogg |
|---------|-----|-----|-----|
| IE | SIM | NAO | NAO |
| Chrome | SIM | SIM | SIM |
| Firefox | SIM | SIM | SIM |
| Safari | SIM | SIM | NAO |
| Opera | SIM | SIM | SIM |

Os plug-ins, também conhecidos como HTML helpers são programas que estendem as capacidades dos browsers. Um exemplo bem conhecido são as applets Java. Plug-ins podem ser adicionados à páginas web usando-se os elementos <object> ou <embed>. Lembre que no padrão HTML5 vídeo e áudio não devem ser tratados via plug-in e sim via as tags <video> e <audio>.

O elemento <object> é suportado por todos os browsers. Ele é usado para inserir objetos tais como java applets, leitores de PDF, ou tocadores flash nas páginas web. A tag object pode ser usada para incluir HTML dentro do HTML ou até mesmo uma imagem.

Veja um exemplo de um objeto flash.

```
<object width="100%" height="500px" data="snippet.html"></object> ou
<object data="carro.jpeg"></object>
```

Já a tag <embed> tem a mesma funcionalidade de <object> mas ela não faz parte de HTML5.

Exercício 2

Veja um exemplo de um objeto flash. Crie um arquivo chamado flash.html e inclua nele

```
<object width="400" height="400" data="tom.swf"></object>
```

e veja o resultado em um browser.

Ainda é difícil garantir que qualquer browser mostre qualquer vídeo. Assim uma possibilidade interessante é deixar o youtube mostrar seus vídeos. Para tanto, você precisa "erguer" (upload) seu vídeo e daí (usando iFrame)

```
<iframe width="420" height="315" src="http://www.youtube.com/embed/XGSy3_Czz8k">
</iframe>
```

A mesma coisa usando <object>

```
<object width="420" height="315" data="http://www.youtube.com/v/XGSy3_Czz8k">
</object>
```

e usando <embed>

```
<embed width="420" height="315" src="http://www.youtube.com/v/XGSy3_Czz8k">
```

Note que o youtube vai mostrar o código (como XGSy3_Czz8k) quando você clicar em share sob o vídeo.

Geolocalização

Há uma API HTML para obter a posição geográfica do internauta. Já que este assunto invade a privacidade dele tal informação só é submetida se o mesmo aprovar o fato. Deve-se notar que a informação é mais acurada quando emitida por dispositivos móveis. A seguir, um exemplo de uso

```
<script>
var x = document.getElementById("demo");
function getLocation() {
if (navigator.geolocation) {
navigator.geolocation.
getCurrentPosition(showPosition);
} else {
x.innerHTML = "Geolocalizacao nao suportada";
}
}
function showPosition(position) {
x.innerHTML = "Latitude: " +
position.coords.latitude +
"<br>Longitude: " + position.coords.longitude;
}
</script>
```

Exercício 3

Inclua no arquivo flash.html a apresentação de sua geolocalização. Observação: veja em www.w3schools.com no tutorial sobre HTML/ Geolocation como mostrar sua localização em um mapa do google maps.

Para você fazer

Note que todos os arquivos citados nesta aula estão disponíveis pelo professor. Imprima os arquivos pedidos (video e flash.html) grampeie-o nesta folha e devolva tudo ao professor.

Ou, se for mais fácil mostre-os ao professor no computador da sala.

