

# HTML5 - Canvas API

mai 08, 2012 às 11:42 por [DiegoBlanco](#)

2 classificações 20 comentários

## Descrição



Este treinamento é homologado pela organização W3C que trabalha no desenvolvimento de padrões para web.

### CANVAS API

#### O elemento canvas

A Canvas API permite a você desenhar na tela do navegador via Javascript. O único elemento HTML existente para isso é o elemento canvas, o resto todo é feito via Javascript. Veja como inserir o elemento canvas numa página:

```
<canvas id="x" width="300" height="300"></canvas>
```

Isso vai exibir um retângulo vazio. Para desenhar nele, primeiro obtemos o contexto de desenho, com Javascript:

```
context=document.getElementById('x').getContext('2d')
```

Agora que temos um contexto, podemos desenhar nele. Vamos começar com um simples retângulo:

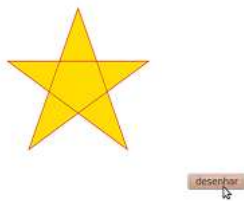
```
context.fillRect(10, 10, 50, 150)
```

Simples, não? Que tal tentarmos algo um pouco mais complexo? Dê uma olhada no exemplo:

```
<!DOCTYPE html>
<html>
<head>
<metacharset="UTF-8" />
<title>Canvas API</title>
</head>
<body>
<canvasid="x" width="300" height="300"></canvas>
<button onclick="desenhar()">desenhar</button>
<script>
function desenhar(){
// Obtemos o contexto
context=document.getElementById('x').getContext('2d')
//Iniciamos um novo desenho
context.beginPath()
```

```
//Movemos a caneta para o início do desenho
context.moveTo(150,50)
//Desenhamos as linhas
context.lineTo(220,250)
context.lineTo(50,125)
context.lineTo(250,125)
context.lineTo(80,250)
context.lineTo(150,50)
//O desenho não é de verdade enquanto você
//não mandar o contexto pintá-lo.
//Vamos pintar o interior de amarelo
context.fillStyle='#ff0'
context.fill()
//Vamos pintar as linhas de vermelho.
context.strokeStyle='#f00'
context.stroke()
}
</script>
</body>
</html>
```

E veja o que acontece quando se clica no botão:



Há muito mais para você estudar se quiser se aprofundar na Canvas API. Apenas para que você tenha uma idéia, é possível desenhar texto, sombras, gradientes, incluir imagens no canvas, manipular os pixels, rotacionar e transformar os objetos.

Canvas e SVG Uma dúvida muito comum é quando usar Canvas, quando usar SVG. Para saber escolher, é preciso entender as diferenças entre um e outro. SVG é vetorial, e baseado em XML, logo, acessível via DOM. Canvas é desenhado pixel a pixel, via Javascript.

Assim, as vantagens do SVG são:

- O conteúdo é acessível a leitores de tela

- O gráfico é escalável, não perde resolução ou serrilha ao redimensionar

- O conteúdo é acessível via DOM

E as vantagens do Canvas:

- A performance é muito superior ao SVG na maioria dos casos

- É fácil desenhar via Javascript. Em SVG, é preciso fazer seu script escrever XML para você. Com Canvas você só manda desenhar, e pronto.

Este treinamento faz parte do Microsoft Virtual Academy: <http://www.microsoftvirtualacademy.com>

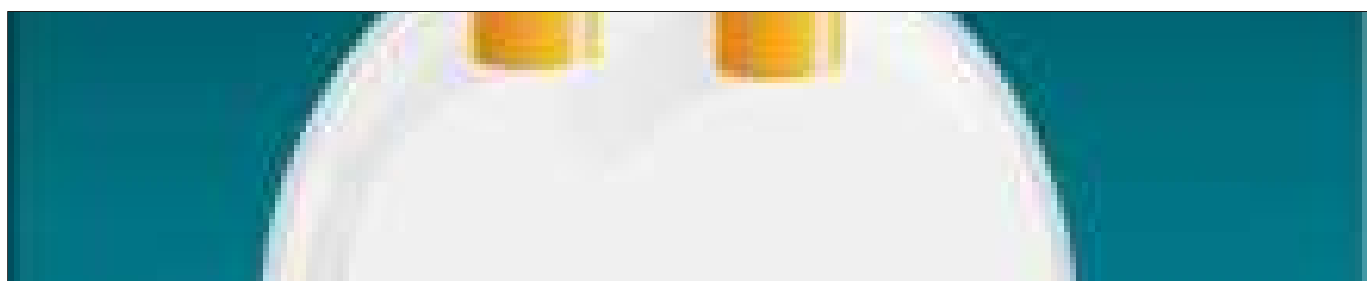
## Episódios relacionados



HTML5 - Undo



HTML5 - Geolocation API



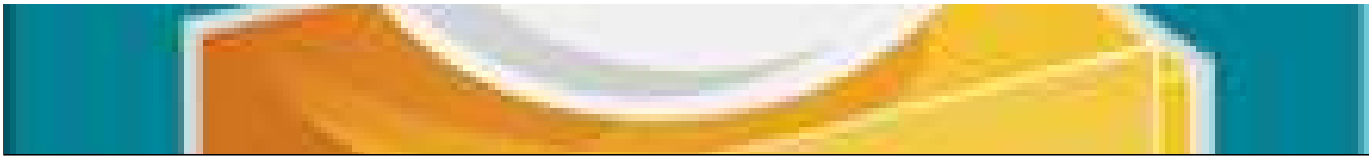


HTML5 - Scroll Into View e Hidden



HTML5 - Aplicações Offline





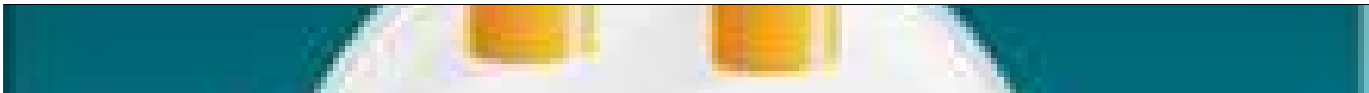
HTML5 - Histórico de Sessão e API Storage



HTML5 - Microdata



HTML5 - Tipos de Links





HTML5 - Novos Eventos DOM



HTML5 - DOM e HTML5

