



**WEB ACADEMY**

# Front-end за Напреднали

## HTML5

### New Markup

<!DOCTYPE html>

<Canvas>

<Audio>

<Video>

<section>, <article>,  
<header>, <nav> ...

### New APIs

File APIs  
Drag & Drop  
Web Messaging  
Browser history management  
Web Storage  
Web Sockets  
Web Workers  
.....

Touch

Animation

CSS3

SVG

more...

**HTML**

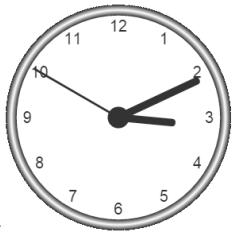


MathML

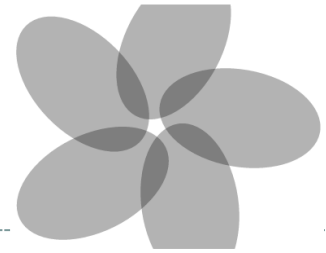
Web Open Font

Geo Location

Selectors



# Canvas & SVG

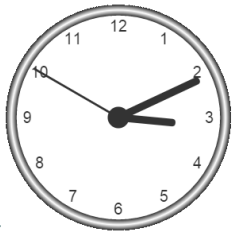


Какво е общото и какво е различното между тях?

- SVG е език за описание на 2D графики посредством XML
- Canvas изчертава 2D графики посредством JavaScript
- При SVG всеки елемент е отделен обект, достъпен през DOM към който могат да се свързват JavaScript събития, а при промяна на атрибутите браузъра го преизчертава
- Canvas се изчертава самостоятелно пиксел по пиксел
- Сравнение между Canvas и SVG

Canvas	SVG
<ul style="list-style-type: none"><li>• Resolution dependent</li><li>• No support for event handlers</li><li>• Poor text rendering capabilities</li><li>• You can save the resulting image as .png or .jpg</li><li>• Well suited for graphic-intensive games</li></ul>	<ul style="list-style-type: none"><li>• Resolution independent</li><li>• Support for event handlers</li><li>• Best suited for applications with large rendering areas (Google Maps)</li><li>• Slow rendering if complex (anything that uses the DOM will be slow)</li><li>• Not suited for game applications</li></ul>



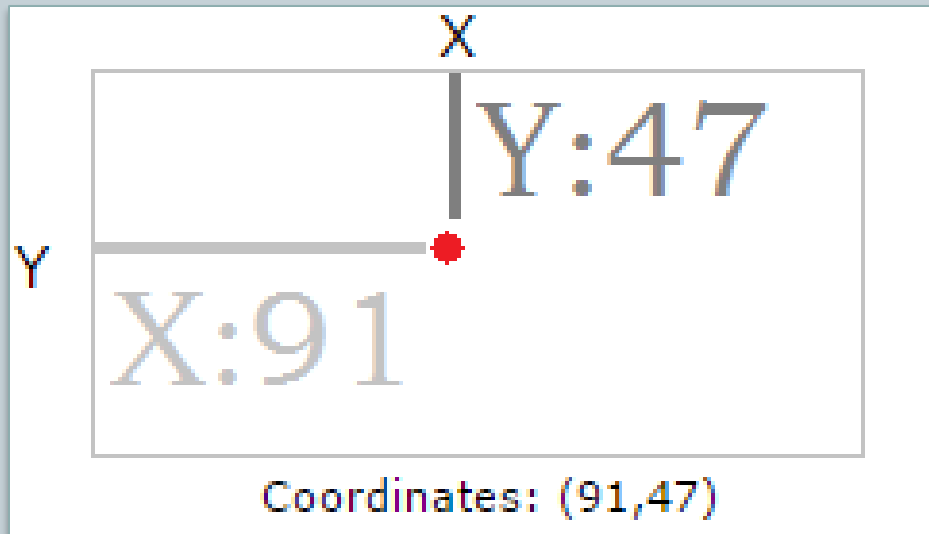


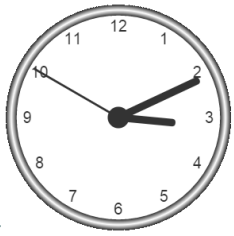
# Координати



Няколко думи за Canvas координати и навигация:

- използва двумерна декартова координатна система
- горния ляв ъгъл има координати 0:0
- за изчертаване на даден обект е необходимо правилно да се придвижва маркера върху координатната система





# Canvas



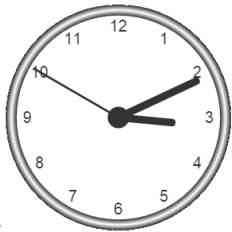
## Какво представлява Canvas?

- Canvas изчертава 2D графики посредством JavaScript
- Canvas се изчертава самостоятелно пиксел по пиксел
- Canvas е просто контейнер за графиката
- Canvas трябва да има `id`, `width` и `height` атрибути

## Какво може да прави Canvas?

- да изчертава цветен текст с/без анимация
- да изчертава графики и таблици
- да движи обекти и да създава анимации
- може да реагира на JavaScript събития като клавиши, бутони и други, т.е. има интерактивност
- може да се използва при разработката на уеб игри





# Изчертаване

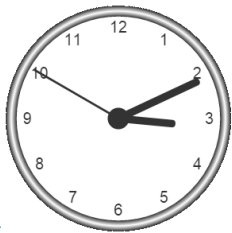


За работа с Canvas се нуждаем от няколко стъпки:

1. намиране на <canvas> елемента по id, чрез JavaScript
2. създаване на обект с който ще работим
3. чертаене върху <canvas> обекта

```
<!-- Изчертаване на червен правоъгълник с позиция 10:10 и размери 150:75 -->  
<canvas id="myCanvas" width="200" height="100" style="border:1px solid red;">  
  Your browser does not support the canvas element.  
</canvas>  
<script>  
  var canvas = document.getElementById("myCanvas"); // Стъпка 1  
  var ctx = canvas.getContext("2d");                // Стъпка 2  
  ctx.fillStyle = "#FF0000";                        // Цвят на запълване  
  ctx.fillRect(10,10,150,75); /* x, y, width, height */ // Чертаем фигура  
</script>
```





# Как да изчертаем обекти?



Чертаене на линия - необходими методи:

- moveTo(x,y) - задава позиция на началната точка
- lineTo(x,y) - задава позиция на крайната точка
- stroke() - изчертава линия свързваща горните точки

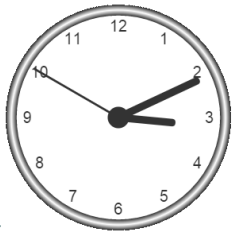
Чертаене на кръг - необходими методи:

- beginPath() - започва път или рестартира текущия път
- arc(x,y,r,start,stop) - изчертава кръг или част от него

Чертаене на правоъгълник - нужни методи:

- rect(x, y, w, h) - задава правоъгълник, но не го чертае
- fillRect(x, y, w, h) - чертае запълнен правоъгълник
- strokeRect(x, y, w, h) - чертае празен правоъгълник
- clearRect() - изчиства пикселите в зададен правоъгълник





# Преливки

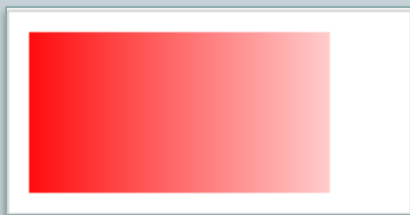


Преливки могат да се задават на различни обекти.  
Съществуват 2 различни вида преливки:

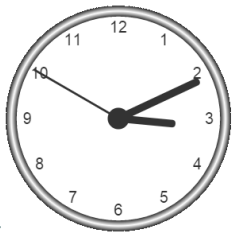
- линейна преливка (linear gradient)
- радиална преливка (radial gradient)

Необходими методи за работа с преливки:

- [createLinearGradient](#)(x,y,x1,y1) - задава linear gradient
- [createRadialGradient](#)(x,y,r,x1,y1,r1) - задава radial gradient
- [addColorStop\(\)](#)
  - задава позицията на която даден цвят от преливката трябва да спре
  - позицията може да има стойност между 0 и 1







# Чертаене на Текст



## Необходимии методи за работа с текст:

- `font()` - задава шрифт на текста
- `fillText(text, x, y)` - чертае запълнен текст
- `strokeText(text, x, y)` - чертае незапълнен текст

```
var canvas =  
document.getElementById("myC");  
var ctx = canvas.getContext("2d");  
ctx.font = "30px Arial";  
ctx.strokeText("Hello World",10,50);
```

//В ляво виждате резултата

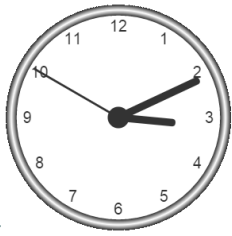
Hello World

Hello World

```
var canvas =  
document.getElementById("myC");  
var ctx = canvas.getContext("2d");  
ctx.font = "30px Comic Sans MS";  
ctx.fillStyle = "red";  
ctx.textAlign = "center";  
ctx.fillText("Hello World",  
canvas.width/2, canvas.height/2);
```

// <- Тук в дясно виждате резултата





# Изображения



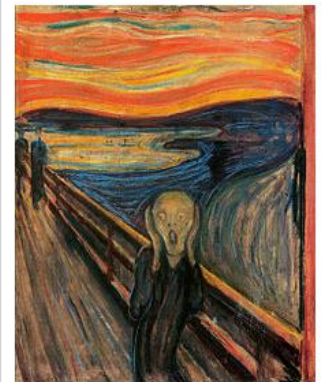
```
<p>Image to use:</p>

<p>Canvas:</p>
<canvas id="myCanvas" width="240" height="297"
style="border:1px solid #d3d3d3;">
Your browser does not support the HTML5 canvas tag.
</canvas>
<script>
window.onload = function() {
    var canvas = document.getElementById("myCanvas");
    var ctx = canvas.getContext("2d");
    var img = document.getElementById("scream");
    ctx.drawImage(img, 10, 10);
};
</script>
```

Image to use:



Canvas:



# ВЪПРОСИ

82

