

HTML5 Canvas

Canvas

```
<!doctype html>
<html>
  <head>
    <title>canvasExample</title>
    <meta charset='utf-8' />
  </head>
  <body>
    <canvas height='320' width='480' id='example'>Обновите браузер</canvas>
    <script>
      var example = document.getElementById("example"),
          ctx     = example.getContext('2d');
      ctx.fillRect(0, 0, example.width, example.height);
    </script>
  </body>
</html>
```

Прямоугольники

```
strokeRect(x, y, ширина, высота) // Рисует прямоугольник
fillRect(x, y, ширина, высота)   // Рисует закрашенный прямоугольник
clearRect(x, y, ширина, высота)  // Очищает область на холсте размер с прямоугольник заданного размера
```

```
var example = document.getElementById("example"),
    ctx     = example.getContext('2d');
example.width = 640;
example.height = 480;
ctx.strokeRect(15, 15, 266, 266);
ctx.strokeRect(18, 18, 260, 260);
ctx.fillRect(20, 20, 256, 256);
for (i = 0; i < 8; i += 2)
  for (j = 0; j < 8; j += 2) {
    ctx.clearRect(20 + i * 32, 20 + j * 32, 32, 32);
    ctx.clearRect(20 + (i + 1) * 32, 20 + (j + 1) * 32, 32, 32);
  }
```

Линии и дуги

```
moveTo(x, y) // перемещает "курсор" в позицию x, y и делает её текущей
lineTo(x, y) // ведёт линию из текущей позиции в указанную, и делает в последствии указанную текущей
arc(x, y, radius, startAngle, endAngle, anticlockwise) // рисование дуги, где x и y центр окружности, далее началь-
ный и конечный угол, последний параметр указывает направление
```

```
beginPath()
closePath()
stroke()
fill()
```

```
ctx.beginPath();
ctx.arc(80, 100, 56, 3/4 * Math.PI, 1/4 * Math.PI, true);
ctx.fill(); // *14
ctx.moveTo(40, 140);
ctx.lineTo(20, 40);
ctx.lineTo(60, 100);
ctx.lineTo(80, 20);
ctx.lineTo(100, 100);
ctx.lineTo(140, 40);
ctx.lineTo(120, 140);
ctx.stroke(); // *22
```

Цвета

```
fillStyle = color // определяет цвет заливки
strokeStyle = color // цвет линий цвет задается точно так же как и css, на примере все четыре способа задания цвет
```

```
// все четыре строки задают оранжевый цвет заливки
```

```
ctx.fillStyle = "orange";
```

```
ctx.fillStyle = "#FFA500";
```

```
ctx.fillStyle = "rgb(255,165,0)";
```

```
ctx.fillStyle = "rgba(255,165,0,1)"
```

Изображения

- Добавление изображения условно можно разделить на два шага:

- создание JavaScript объекта Image

```
var img = new Image(); // Создание нового объекта изображения
img.src = 'image.png'; // Путь к изображению которое необходимо
нанести на холст
```

- отрисовка изображения на холсте при помощи функции drawImage

```
drawImage(image, x, y) // Где x и y это координаты левого
верхнего угла изображения, а первый параметр это изображение
```

```
<!doctype html>
<html>
  <meta charset='utf-8'>
  <head>
    <title>imgExample</title>
  </head>
  <body>
    <canvas id='example'>Обновите браузер</canvas>
    <script>
      var example = document.getElementById("example"),
          ctx      = example.getContext('2d'), // Контекст
          pic      = new Image();           // "Создаём" изображение
          pic.src   = 'example.png';        // Источник изображения
          pic.onload = function() {          // Событие onLoad, ждём момента пока
загрузится изображение
            ctx.drawImage(pic, 0, 0); // Рисуем изображение от точки с
координатами 0, 0
          }
    </script>
  </body>
</html>
```