

Canvas

Canvas

- Rityta (målarduk) för att rita på
- Vektorbaserad grafik
- Ingår i HTML5
- Ritning görs med JavaScript
- Allt som behövs för enklare spel

Canvas vs SVG

Canvas	SVG
2D	2D
Ritar med JavaScript	Figurer beskrivs i HTML
Upplösningsberoende	Upplösningsoberoende
Ritar snabbt	Ritar långsamt om många figurer
Lämplig för spel	Inte lämplig för spel

```
<!DOCTYPE html>
<html lang="sv">
  <head>
    <meta charset="utf-8">
    <title>Canvas Demo</title>
  </head>
  <body>
    <canvas id="myCanvas" width="500" height="150"></canvas>

    <!-- sist i body länkas scriptet in -->
    <script src="canvas.js"></script>
  </body>
</html>
```

canvas.js

```
"use strict";  
window.addEventListener("load", init);  
  
function init()  
{  
    var c = document.getElementById("myCanvas");  
    var ctx = c.getContext("2d");  
  
    ctx.fillStyle = "#99FF99";  
    ctx.fillRect(0, 0, 500, 150);  
}
```



```
"use strict";
window.addEventListener("load", init);

function init()
{
    var c = document.getElementById("myCanvas");
    var ctx = c.getContext("2d");

    ctx.fillStyle = "#99FF99";
    ctx.fillRect(0, 0, 500, 150);
    drawCircle();
}
```

```
function drawCircle()
{
    ctx.beginPath();
    ctx.fillStyle = "#FF0000";
    ctx.arc(70, 40, 20, 0, Math.PI*2, true);
    ctx.fill();
}
```

Testa. Problem?

```
"use strict";  
window.addEventListener("load", init);
```

```
var ctx;
```

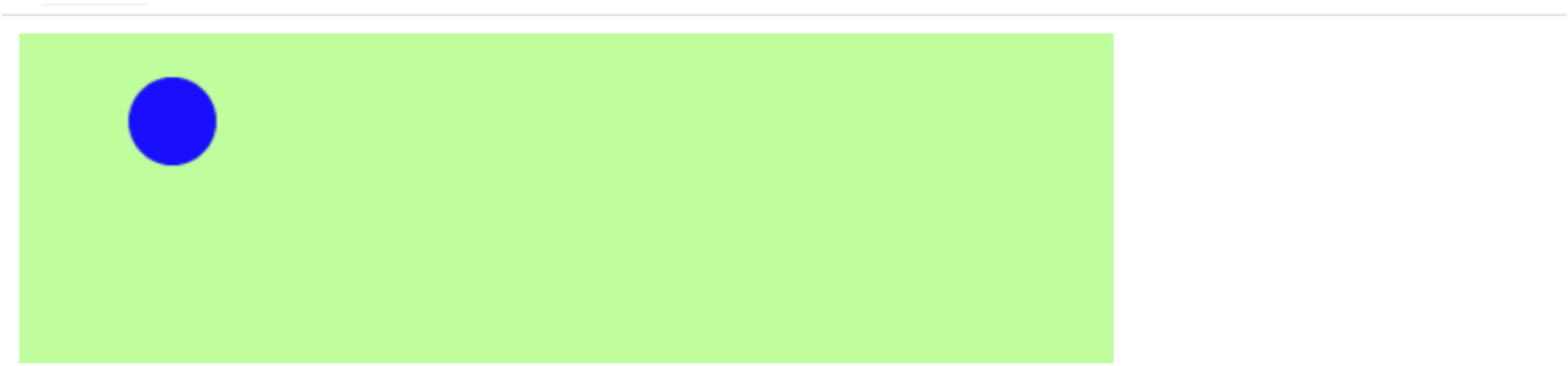
```
function init()  
{  
  var c = document.getElementById("myCanvas");  
  var ctx = c.getContext("2d");  
  ctx.fillStyle = "#99FF99";  
  ctx.fillRect(0, 0, 500, 150);  
  drawCircle();  
}
```

```
function drawCircle()  
{  
  ctx.beginPath();  
  ctx.fillStyle = "#0000FF";  
  ctx.arc(70, 40, 20, 0, Math.PI*2, true);  
  ctx.fill();  
}
```

Testa

Vad betyder parametrarna?





```
"use strict";
window.addEventListener("load", init);

var ctx;

function init()
{
    var c = document.getElementById("myCanvas");
    ctx = c.getContext("2d");

    ctx.fillStyle="#99FF99";
    ctx.fillRect(0, 0, 500, 150);

    drawCircle();
}
```

```
function drawLines()
{
    //linjer
    ctx.beginPath();
    ctx.strokeStyle = "#000";
    ctx.lineWidth = 4;
    ctx.moveTo(50, 30);
    ctx.lineTo(100, 80);
    ctx.lineTo(150, 30);
    ctx.stroke();
}
```

Rita linje, behöver inte skriva

Problem

Jag vill använda hela webbläsfönstret som canvas och jQuery

```
<!DOCTYPE html>
<html lang="sv">
  <head>
    <meta charset="utf-8">
    <title>Canvas Demo</title>
  </head>
  <body>
    <canvas id="myCanvas" width="500" height="150">
    </canvas>
    <!-- sist i body länkas scriptet in -->
    <script src="canvas.js"></script>
  </body>
</html>
```

```
"use strict";
window.addEventListener("load", init);

var ctx;

function init()
{
  var c = document.getElementById("myCanvas");
  ctx = c.getContext("2d");

  ctx.fillStyle = "#99FF99";
  ctx.fillRect(0, 0, 250, 75);

  drawCircle();
}
```

```
<!DOCTYPE html>
<html lang="sv">
  <head>
    <meta charset="utf-8">
    <title>Canvas Demo</title>
  </head>
  <body>
    <canvas id="myCanvas" width="500" height="150"></canvas>

    <!-- sist i body länkas scripten in -->
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
    <script src="canvas.js"></script>
  </body>
</html>
```

canvas.js

```
"use strict";

window.addEventListener("load", init);
$(document).ready(init);

var ctx;

function init()
{
    //var c = document.getElementById("myCanvas");
    var c = $("#myCanvas");
    //var ctx = c.getContext("2d");
    ctx = c[0].getContext("2d");

    ctx.canvas.width = $(window).width();
    ctx.canvas.height = $(window).height();

    ctx.fillStyle = "#99FF99";
    ctx.fillRect(0, 0, 500, 150);

    drawCircle();
}
```

```
"use strict";
$(document).ready(init);

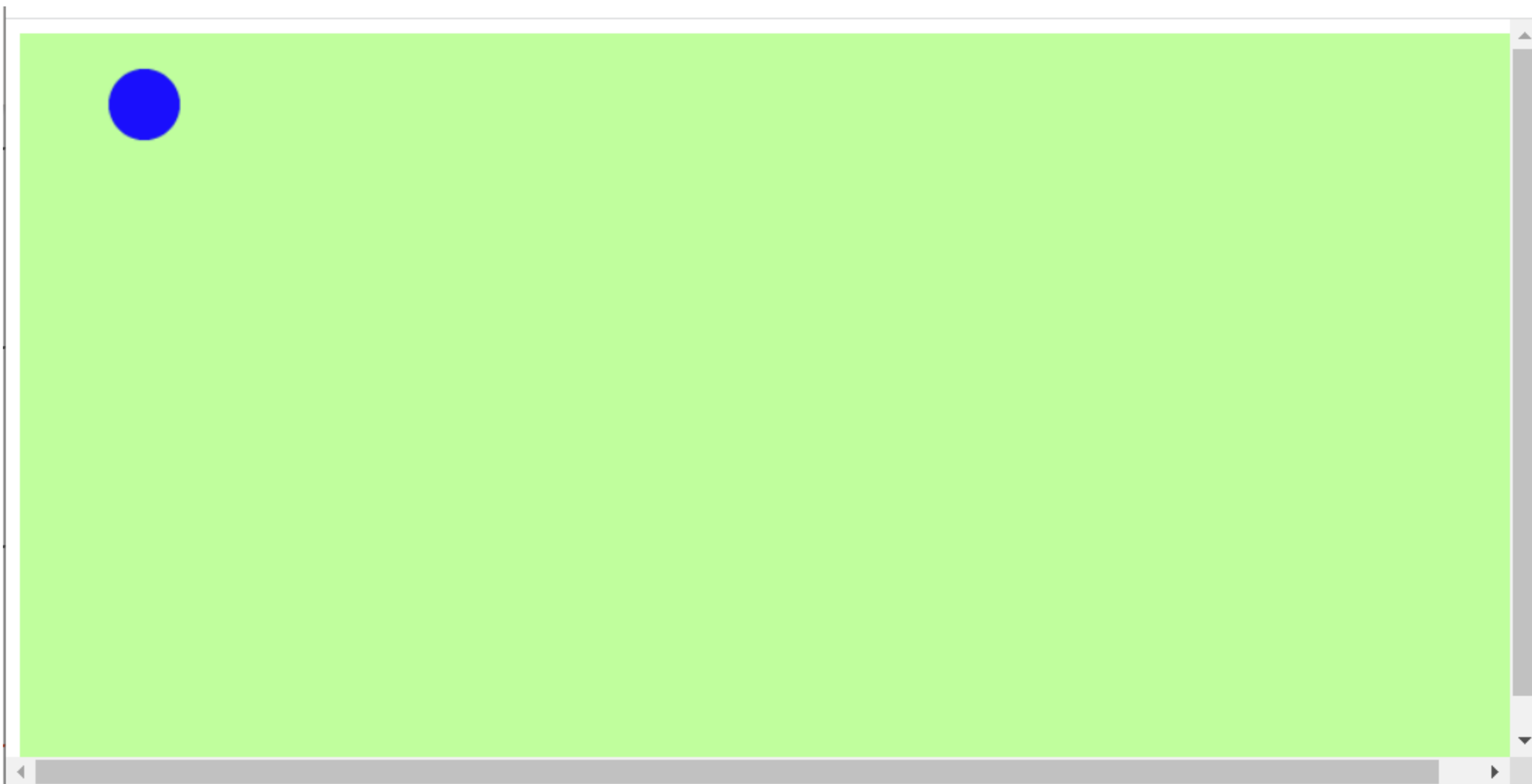
var ctx;

function init()
{
    var c = $("#myCanvas");
    ctx = c[0].getContext("2d");

    ctx.canvas.width = $(window).width();
    ctx.canvas.height = $(window).height();

    ctx.fillStyle = "#99FF99";
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);

    drawCircle();
}
```



```
"use strict";
$(document).ready(init);

var ctx;

function init()
{
var c = $("#myCanvas");
ctx = c[0].getContext("2d");
ctx = $("#myCanvas")[0].getContext("2d");

ctx.canvas.width = $(window).width();
ctx.canvas.height = $(window).height();

ctx.fillStyle = "#99FF99";
ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);

drawCircle();
}
```

Förkorta

```
"use strict";  
$(document).ready(init);
```

```
var ctx;
```

```
function init()  
{
```

```
    ctx = $("#myCanvas")[0].getContext("2d");
```

```
    $('body').css('margin', '0'); // No margins
```

```
    $('body').css('overflow', 'hidden'); // Hide scrollbars
```

```
    ctx.canvas.width = $(window).width();
```

```
    ctx.canvas.height = $(window).height();
```

```
    ctx.fillStyle = "#99FF99";
```

```
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);
```

```
    drawCircle();
```

```
}
```



Problem

Canvas ska följa webbläsarfönstrets storlek



```
"use strict";
$(document).ready(init);
var ctx;

function init()
{
    ctx = $("#myCanvas")[0].getContext("2d");
    $('body').css('margin', '0'); // No margins
    $('body').css('overflow', 'hidden'); // Hide scrollbars
    ctx.canvas.width = $(window).width();
    ctx.canvas.height = $(window).height();

    ctx.fillStyle = "#99FF99";
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);
    $(window).resize(setCanvas);
    drawCircle();
}
```

```
"use strict";
$(document).ready(init);
var ctx;
```

```
function init()
{
    ctx = $("#myCanvas")[0].getContext("2d");
    $('body').css('margin', '0'); // No margins
    $('body').css('overflow', 'hidden'); // Hide scrollbars
    ctx.canvas.width = $(window).width();
    ctx.canvas.height = $(window).height();

    ctx.fillStyle = "#99FF99";
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);

    $(window).resize(setCanvas);

    drawCircle();
}
```

```
function setCanvas()
```

```
{
```

```
    alert("resize");
```

```
}
```

```
"use strict";
$(document).ready(init);
var ctx;
```

```
function init() {
  var ctx = $("#myCanvas")[0].getContext("2d");
  $('body').css('margin', '0'); // No margins
  $('body').css('overflow', 'hidden'); // Hide scrollbars
```

```
  ctx.canvas.width = $(window).width();
  ctx.canvas.height = $(window).height();

  ctx.fillStyle = "#99FF99";
  ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);
```

```
  $(window).resize(setCanvas);
  drawCircle();
}
```

```
function setCanvas()
```

```
{
```

```
  alert("resize");
```

```
  ctx.canvas.width = $(window).width();
```

```
  ctx.canvas.height = $(window).height();
```

```
  ctx.fillStyle = "#99FF99";
```

```
  ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);
```

```
}
```



```
"use strict";
$(document).ready(init);

var ctx;
function init()
{
    ctx = $("#myCanvas")[0].getContext("2d");
    $('body').css('margin', '0'); // No margins
    $('body').css('overflow', 'hidden'); // Hide scrollbars
```

setCanvas();

```
$(window).resize(setCanvas);
drawCircle();
}
```

```
function setCanvas()
{
    ctx.canvas.width = $(window).width();
    ctx.canvas.height = $(window).height();

    ctx.fillStyle = "#99FF99";
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);
}
```

Canvas följer webbläsarfönstrets storlek!
Men vad händer med cirkeln?

Rita en boll

```
"use strict"; $(document).ready(init); var ctx;
```

```
function init() {
```

```
  ctx=$("#myCanvas")[0].getContext("2d");
```

```
  $('body').css('margin', '0'); // No margins
```

```
  $('body').css('overflow', 'hidden'); // Hide scrollbars
```

```
  setCanvas();
```

```
  $(window).resize(setCanvas);
```

```
drawCircle();
```

```
drawBall(100, 200);
```

```
}
```

```
function drawCircle()
```

```
function drawBall(x, y)
```

```
{
```

```
  ctx.beginPath();
```

```
  ctx.fillStyle = "#0000FF";
```

```
ctx.arc(70, 40, 20, 0, Math.PI*2, true);
```

```
ctx.arc(x, y, 20, 0, Math.PI*2, true);
```

```
ctx.fill();
```

```
}
```

```
function setCanvas() { ctx.canvas.width=$(window).width(); ctx.canvas.height=$(window).height(); ctx.fillStyle="#99FF99"; ctx.fillRect(0,0,ctx.canvas.width,ctx.canvas.height); }
```

Om bollen ska röra på sig?

```
"use strict";  
$(document).ready(init);
```

```
var ctx;
```

```
function init() {
```

```
    ctx = $("#myCanvas")[0].getContext("2d");  
    $('body').css('margin', '0'); // No margins  
    $('body').css('overflow', 'hidden'); // Hide scrollbars  
    setCanvas();  
    $(window).resize(setCanvas);
```

```
drawBall(100, 200);
```

```
setInterval(loop, 10);
```

```
}
```

```
function loop()
```

```
{
```

```
    drawBall(100, 200);
```

```
}
```

```
function drawBall(x, y) {ctx.beginPath(); ctx.fillStyle = "#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill(); }
```

```
function setCanvas() {ctx.canvas.width=$(window).width(); ctx.canvas.height=$(window).height(); ctx.fillStyle="#99FF99"; ctx.fillRect(0,0,ctx.canvas.width,ctx.canvas.height); }
```

Bollen ritas fortfarande på samma ställe!

```
"use strict";  
$(document).ready(init);
```

```
var ctx;
```

```
var x = 100;
```

```
var y = 200;
```

```
function init() {
```

```
    ctx = $("#myCanvas")[0].getContext("2d"); $('body').css('margin', '0'); //No margins $('body').css('overflow', 'hidden'); //Hide scrollbars  
    setCanvas(); $(window).resize(setCanvas);
```

```
    setInterval(loop, 10);
```

```
}
```

```
function loop()
```

```
{
```

```
    drawBall(x, y);
```

```
}
```

```
function drawBall(x, y) { ctx.beginPath(); ctx.fillStyle="#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill(); }
```

```
function setCanvas() { ctx.canvas.width=$(window).width(); ctx.canvas.height=$(window).height(); ctx.fillStyle="#99FF99"; ctx.fillRect(0,0,ctx.canvas.width,ctx.canvas.height); }
```

Var ska x och y-värdet ökas?

```
"use strict";
$(document).ready(init);

var ctx;
var x = 100;
var y = 200;
function init() {
  ctx = $("#myCanvas")[0].getContext("2d"); $('body').css('margin', '0'); //No margins $('body').css('overflow', 'hidden'); //Hide scrollbars
  setCanvas(); $(window).resize(setCanvas);
  setInterval(loop, 10);
}

function loop() {
  x = x + 5;
  y = y + 5;
  drawBall(x, y);
}

function drawBall(x, y) { ctx.beginPath(); ctx.fillStyle="#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill(); }

function setCanvas() { ctx.canvas.width=$(window).width(); ctx.canvas.height=$(window).height(); ctx.fillStyle="#99FF99"; ctx.fillRect(0,0,ctx.canvas.width,ctx.canvas.height); }
```

Testa olika värden? Problem?



```
"use strict"; $(document).ready(init); var ctx; var x = 100; var y = 200;
```

```
function init() { ctx=$("#myCanvas")[0].getContext("2d");$('body').css('margin','0');$('body').css('overflow','hidden');setCanvas(); $(window).resize(setCanvas); setInterval(loop, 10); }
```

```
function loop() {
```

```
    ctx.fillStyle = "#99FF99";
```

```
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);
```

```
    x = x + 5;
```

```
    y = y + 5;
```

```
    drawBall(x, y);
```

```
}
```

```
function drawBall(x, y) { ctx.beginPath(); ctx.fillStyle="#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill(); }
```

```
function setCanvas() { ctx.canvas.width=$(window).width();ctx.canvas.height=$(window).height();ctx.fillStyle="#99FF99"; ctx.fillRect(0,0,ctx.canvas.width,ctx.canvas.height); }
```

*Rensa före ritning.
Ser då ut som bollen rör sig.*

```
"use strict"; $(document).ready(init); var ctx; var x = 100; var y = 200;  
function init() { ctx=$("#myCanvas")[0].getContext("2d");$('body').css('margin','0');$('body').css('overflow','hidden');setCanvas(); $(window).resize(setCanvas); setInterval(loop, 10); }
```

```
function loop() {  
  ctx.fillStyle = "#99FF99";  
  ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);  
  x = x + 5;  
  y = y + 5;  
  drawBall(x, y);  
}
```

```
function drawBall(x, y) { ctx.beginPath(); ctx.fillStyle="#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill(); }
```

```
function setCanvas() {  
  ctx.canvas.width=$(window).width();  
  ctx.canvas.height=$(window).height();  
  ctx.fillStyle = "#99FF99";  
  ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);  
}
```

Ser du något med designen?
DRY

```
"use strict"; $(document).ready(init); var ctx; var x = 100; var y = 200;  
function init() { ctx=$("#myCanvas")[0].getContext("2d");$('body').css('margin','0');$('body').css('overflow','hidden');setCanvas(); $(window).resize(setCanvas); setInterval(loop, 10); }
```

```
function loop() {  
    ctx.fillStyle = "#99FF99";  
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);  
    x = x + 5;  
    y = y + 5;  
    drawBall(x, y);  
}
```

```
function drawBall(x, y) { ctx.beginPath(); ctx.fillStyle="#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill(); }
```

```
function setCanvas() {  
    ctx.canvas.width=$(window).width();  
    ctx.canvas.height=$(window).height();  
    ctx.fillStyle = "#99FF99";  
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);  
}
```

DRY

```
"use strict"; $(document).ready(init); var ctx; var x = 100; var y = 200;
```

```
function init() { ctx=$("#myCanvas")[0].getContext("2d");$('body').css('margin','0');$('body').css('overflow','hidden');setCanvas(); $(window).resize(setCanvas); setInterval(loop, 10); }
```

```
function loop() {
```

```
    ctx.fillStyle = "#99FF99";
```

```
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);
```

```
    drawBackground();
```

```
    x = x + 5;
```

```
    y = y + 5;
```

```
    drawBall(x, y);
```

```
}
```

```
function drawBall(x, y) { ctx.beginPath(); ctx.fillStyle="#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill(); }
```

```
function setCanvas() {
```

```
    ctx.canvas.width=$(window).width();
```

```
    ctx.canvas.height=$(window).height();
```

```
    ctx.fillStyle = "#99FF99";
```

```
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);
```

```
    drawBackground();
```

```
}
```

```
function drawBackground()
```

```
{
```

```
    ctx.fillStyle = "#99FF99";
```

```
    ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height);
```

```
}
```

```
function loop()
{
  drawBackground();
  x = x + 5;
  y = y + 5;
  drawBall(x, y);
}
```

*I vilken riktning rör sig bollen?
Om man vill att bollen
ska röra sig i olika riktningar?*

```
"use strict";  
$(document).ready(init);
```

```
var ctx;  
var x = 100;  
var y = 200;
```

```
var speedX = 5;
```

```
var speedY = 5;
```

```
function init() { ctx=$("#myCanvas")[0].getContext("2d");$('body').css('margin','0');$('body').css('overflow','hidden');setCanvas(); $(window).resize(setCanvas); setInterval(loop, 10); }
```

```
function loop() {  
  drawBackground();  
  x = x + speedX;  
  y = y + speedY;  
  drawBall(x, y);  
}
```

```
function drawBall(x, y) { ctx.beginPath(); ctx.fillStyle="#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill();}
```

```
function setCanvas() { ctx.canvas.width = $(window).width(); ctx.canvas.height = $(window).height(); drawBackground(); }
```

```
function drawBackground() { ctx.fillStyle="#99FF99"; ctx.fillRect(0,0,ctx.canvas.width,ctx.canvas.height); }
```

Kommer behöva olika värden
för olika riktningar!

*Testa olika värden på
speedX och speedY?*

```
"use strict";  
$(document).ready(init);
```

```
var ctx;  
var x = 100;  
var y = 200;
```

```
var speedX = 5;
```

```
var speedY = 5;
```

```
function init() { ctx=$("#myCanvas")[0].getContext("2d");$('body').css('margin','0');$('body').css('overflow','hidden');setCanvas(); $(window).resize(setCanvas); setInterval(loop, 10); }
```

```
function loop() {  
  drawBackground();  
  x = x + speedX;  
  y = y + speedY;  
  drawBall(x, y);  
}
```

```
function drawBall(x, y) { ctx.beginPath(); ctx.fillStyle="#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill();}
```

```
function setCanvas() { ctx.canvas.width = $(window).width(); ctx.canvas.height = $(window).height(); drawBackground(); }
```

```
function drawBackground() { ctx.fillStyle="#99FF99"; ctx.fillRect(0,0,ctx.canvas.width,ctx.canvas.height); }
```

Hur studsas mot kanterna?

```
"use strict";
$(document).ready(init);

var ctx;

var x = 100; var y = 200;
var speedX = 5; var speedY = 5;
function init() { ctx=$("#myCanvas")[0].getContext("2d");$('body').css('margin','0');$('body').css('overflow','hidden');setCanvas(); $(window).resize(setCanvas); setInterval(loop, 10); }
```

```
function loop() {
  drawBackground();
  x = x + speedX;
  y = y + speedY;
```

```
if (y >= ctx.canvas.height)
```

```
????;
```

```
  drawBall(x, y);
}
```

```
function drawBall(x, y) { ctx.beginPath(); ctx.fillStyle="#0000ff"; ctx.arc(x, y, 20, 0, Math.PI*2, true); ctx.fill(); }
```

```
function setCanvas() { ctx.canvas.width = $(window).width(); ctx.canvas.height = $(window).height(); drawBackground(); }
```

```
function drawBackground() { ctx.fillStyle="#99FF99"; ctx.fillRect(0,0,ctx.canvas.width,ctx.canvas.height); }
```

Inlämningsuppgift

En studsande boll. Kommentera hela koden!

Visa exempel

Ta hänsyn till bollens radie vid studs mot kanterna.

Extra 1: Få två bollar att studsa.

Extra 2: Få godtyckligt antal bollar att studsa genom att bara ändra värdet på en konstant.

Extra 3: Slumpa fram startpositionen för bollarna

Extra 4: Slumpa fram en färg för varje boll

Extra 5: Bollarna ska även studsa om de kolliderar med varandra

Ta hänsyn till bollens radie vid studs mot kanterna

//Ledtråd

// x och y motsvarar bollens mittpunkt

```
ctx.arc(x, y, 20, 0, Math.PI*2, true);
```

Extra-uppgift 1: Få två bollar att studsa

//Ledtråd

~~var x = 100;~~

~~var y = 200;~~

~~var speedX = 5;~~

~~var speedY = 5;~~

var x = [100, 200];

var y = [200, 100];

var speedX = [5, 5];

var speedY = [5, 5];

Sedan måste all annan kod också ändras så att de hanterar arrayerna

Se https://www.w3schools.com/js/js_arrays.asp

Ledtråd: I funktionen loop behövs en for-loop och använder värdena i arrayerna för att uppdatera positionen, kontrollera kanterna och rita för båda bollarna.

//Ledtråd

~~var x = [100, 200];~~

~~var y = [200, 100];~~

~~var speedX = [5, 5];~~

~~var speedY = [5, 5];~~

var x = [];

var y = [];

var speedX = [];

var speedY = [];

const COUNT = 3;

//I funktionen init måste du göra...

for (var i=0; i<COUNT; i++) {

var xCoord = //Slumpa fram ett lämpligt läge;

var yCoord = //Slumpa fram ett lämpligt läge;

x.push(xCoord);

y.push(yCoord);

speedX.push(5);

speedY.push(5);

}

Extra-uppgift 2:

Få godtyckligt antal bollar att studsas genom att bara ändra värdet på en konstant.

För att inte alla bollar ska börja på samma ställe måste startläget slumpas fram.

Se https://www.w3schools.com/js/js_random.asp

Se till att övrig kod fungerar med dessa ändringar.

Extra-uppgift 3:

Slumpa fram en färg på varje boll (se https://www.w3schools.com/js/js_random.asp)
Funktionen drawBall ska ha 3 parametrar, x, y och color.

```
//Ledtråd
```

```
var color = [];
```

```
//I funktionen init måste du göra något i stil med...
```

```
var colorR = vad ska det stå här? //Slumpa fram ett lämpligt läge;
```

```
var colorG = vad ska det stå här? //Slumpa fram ett lämpligt läge;
```

```
var colorB = vad ska det stå här?//Slumpa fram ett lämpligt läge;
```

```
var rgb = 'rgb('+ colorR + ',' + colorG + ',' + colorB + ')';
```

```
color.push(rgb);
```

Se till att övrig kod fungerar med dessa ändringar.

Extra-uppgift 4:

Slumpa fram en hastigheten (och riktningen) på varje boll.
Slumpa fram värdena för speedX och speedY.

Se till att övrig kod fungerar med dessa ändringar.

Extra-uppgift 5: Bollarna ska även studsa om de kolliderar med varandra

//Ledtråd, lägg till en funktion för att se om bollarna träffar varandra. Gör inga ändringar i funktionen!

```
function collisionDetection_circles(cx1, cy1, radius1, cx2, cy2, radius2)
{
    // cx1, cy1, radius1 = centre coordinates and radius for circle 1
    // cx2, cy2, radius2 = centre coordinates and radius for circle 1
    var distance = Math.sqrt((cx1 - cx2) * (cx1 - cx2) + (cy1 - cy2) * (cy1 - cy2))
    if (distance <= radius1 + radius2)
        return true;
    else
        return false;
}
```

Anropa funktionen på rätt ställe (efter bollarna ritats) och skriv kod till vad som ska hända vid kollision.

```
for (var i=0; i<COUNT-1; i++) {
    for (var j=i+1; j<COUNT; j++) {
        if (collisionDetection_circles(.....) == true) {
            //Skriv kod till vad som ska hända vid kollision
        }
    }
}
```

//Se till att övrig kod fungerar med dessa ändringar.