

Aquesta vegada desenvoluparem un joc sencer, evidentment molt senzill, podreu seguir-lo sense cap problema. El joc proposat és l'Arkanoid.

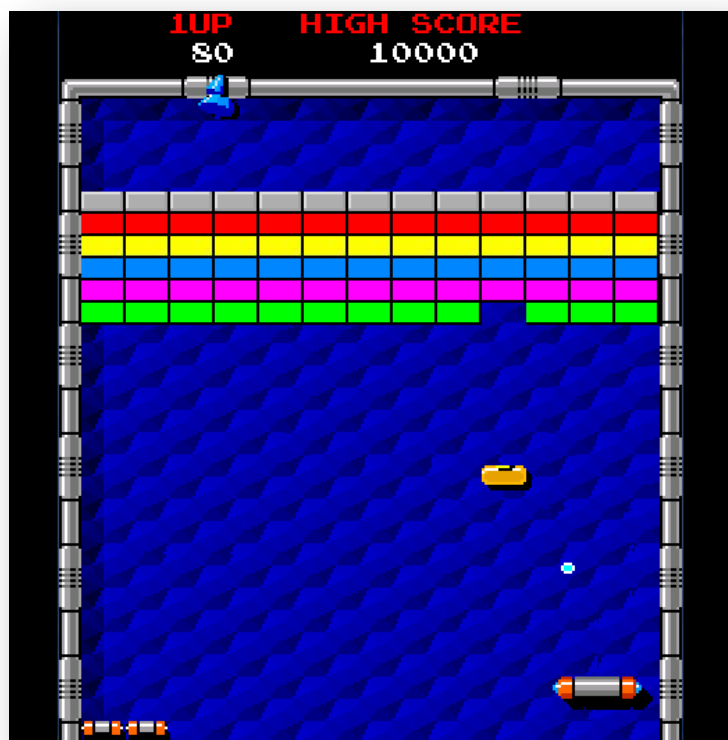
Arkanoid

L'**Arkanoid** és un videojoc del gènere **breakout** o de "destruir maons". Creat en 1986 per l'empresa **Taito Corporation**, es basa en una nau que llença una bola, la qual va rebotant com si fos un frontó per les parets de la pantalla, destruint al seu pas uns maons. Alguns d'ells contenen uns "**bonus**" especials que augmenten la jugabilitat, com augmentar la mida de la nau, fer que la bola reboti més lenta o destruir massivament els maons com si es tractés d'una pistola. L'èxit del joc fou enorme, fins al punt que inaugurà una saga. Millorava les prestacions dels jocs pioners d'**Atari** en aquest gènere. L'habilitat i la coordinació són les qualitats necessàries per vèncer en aquest joc i superar les 33 pantalles fins a derrotar el monstre final.



Desenvolupament o sistema de joc

El jugador controla una menuda plataforma, coneguda com "**Nau Espacial Vaus**", que impedeix que una bola surti de la zona de joc, fent-la rebotar. En la part superior hi ha "**rajoles**" o "**blocs**", que desapareixen al ser tocats per la bola.



Quan no queda cap rajola, el jugador passa al següent nivell, on apareix un altre patró de blocs. Existeixen diferents variacions (rajoles que cal copejar diverses vegades perquè desapareguin, naus enemigues, etc) i càpsules que milloren a la **Vaus** (expandint-la, equipant-la amb un canó làser, passant directament al següent nivell, o augmentant el nombre de boles) Tots els nivells són veritablement colorits i tenen el seu propi estil. Existeixen menudes figures que pots copejar per alguns punts i fins i tot són diferents per cada nivell. Tens 3 naus al principi, i després que les perds totes, has de començar de nou des del inici.

En la pantalla nombre 33, l'últim nivell, el jugador s'enfronta al principal enemic del joc, **Doh**. Una vegada que el jugador arriba a aquest nivell, ha de vèncer a **Doh** amb el nombre de "Vauses" que tingui en reserva, si no, el joc acaba i el jugador ha perdut.

Història

A causa de la popularitat del joc, quatre versions es van desenvolupar per al mercat de les màquines arcade:
Arkanoid, Tournament Arkanoid i Revenge of Doh (Arkanoid II), ambdues en 1987, i **Arkanoid Returns** en 1997.

La majoria de computadors de 8 bits (**ZX Spectrum, Amstrad CPC 464, Commodore 64, MSX, Atari 8-bit, Apple II...**) eren molt populars en Europa en els 80. Un port de consola en la NES també era popular, de manera que el joc va ser portat per a computadors de 16 bits com **Commodore Amiga, Atari ST, Apple IIGS o IBM-PC**. Es va desenvolupar un port pel TRS-80 en 1989.

www.old-computers.com



Una versió per a **SNES**, dita **Arkanoid: Doh it Again**, es va llançar en 1997. **Arkanoid Returns** i la seua seqüela, **Arkanoid Returns 2000**, es llança en Japó per a **PlayStation**. Les versions de **16 bits** tenien exactament els mateixos gràfics que el joc original. La conversió de **Arkanoid** per a **Commodore 64** és coneguda per ser el primer joc per a aquest sistema que incloïa música que usava **"samples"** digitalitzats (composats per Martin Galway).

Els controls usats difereixen entre les màquines, i algunes conversions permetien múltiples mètodes de control. Els dos mètodes bàsics de control eren digital i analògic. Els controls digitals (molts maneta de jocs i teclats) són considerats menys convenients que els analògics (com ratolins i trackballs): mentre que els digitals limiten al jugador a una única velocitat, els analògics permeten moure la **Vaus** quasi a qualsevol velocitat desitjada a través de la pantalla.

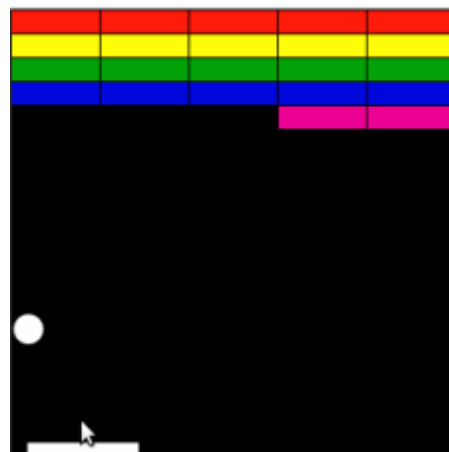


Utilitzem el CANVAS

Evidentment, nosaltres no el farem tant sofisticat, ni molt menys, en comptes de **33** nivells o pantalles, tindrem només una i en comptes de tenir una nau, utilitzarem una paleta rectangular. Però això sí, podrem jugar.

Aquí a la dreta teniu una imatge de com quedarà el joc, com veieu es força simple. Però funciona!

Podem utilitzar la llibreria **jQuery** com una ajuda. Nosaltres, de fet, l'hem utilitzat, encara que no és imprescindible, podríem fer-ho perfectament sense aquesta llibreria, però pot facilitar el desenvolupament.



El primer que hem de fer és crear una instància de l'element **<canvas>** en el nostre document **HTML 5**, perquè puguem començar a dibuixar-hi.

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

Aquesta declaració crea el llenç sobre el qual anem a dibuixar el joc.

Ara que tenim un llenç per dibuixar, fem-ho, escrivim el codi. Anirem pas a pas, i explicarem el que anem fent. Potser sembli una mica repetitiu, però cal. De totes maneres, això seria molt fàcil, anar copiant el codi, per això desmanem el joc mitjançant un marc de treball. Podeu escollir un entre els tres principals **Fabric.js**, **Paper.js** i **bHive**

Nosaltres farem el joc amb el **Canvas** normal i afegirem una captura de pantalla per cada pas. Vosaltres l'he d'interpretar segon el marc de treball escollit. Sembla que teniu feina.

Recordeu que l'objectiu d'aquesta pràctica és l'aprenentatge, per tant, intenteu jugar amb el codi, canviant dades, per aprendre.

Primer pas: Dibuixar un cercle

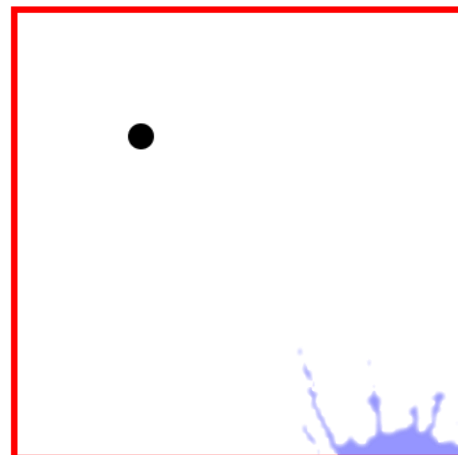
```
// obtenir una referència al llenç
var ctx = $('#canvas')[0].getContext("2d");

//dibuixar un cercle
ctx.beginPath();
ctx.arc(75, 75, 10, 0, Math.PI*2, true);
ctx.closePath();
ctx.fill();
```

És possible que vulgueu crear un comentari en l'arc (...) i tractar de cridar **ctx.rect(x, y, ample, alçada)**.

Podeu mantenir **arc** i **rect** entre les crides a **beginPath()** i a **closePath()**, igual que l'exemple anterior. També podríem tractar de substituir **stroke()** per **fill()**.

Be, ja tenim la piloteta!.



Segon pas: Afegir una mica de color

També podem donar-li a la bola colors diferents. Canviar el valor de **ctx.fillStyle** canviarà el color actual, podem establir el seu valor en una cadena hexadecimal del format **'#rrggbb'** o en una cadena **'rgba (r, g, b, a)'** on a és un valor entre **0** i **1** que representa la transparència del color.

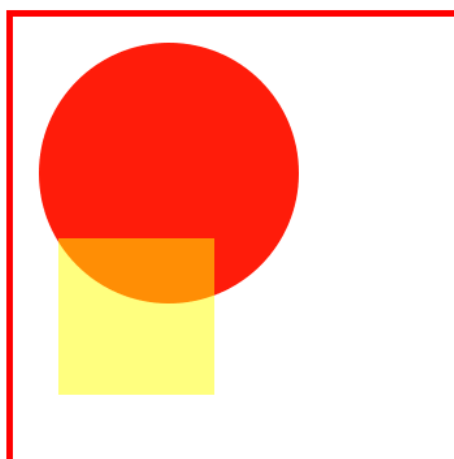
```
// obtenir una referència al llenç
var ctx = $('#canvas')[0].getContext("2d");

//dibuixar un cercle
ctx.fillStyle = "#00A308";
ctx.beginPath();
ctx.arc(220, 220, 50, 0, Math.PI*2, true);
ctx.closePath();
ctx.fill();

ctx.fillStyle = "#FF1C0A";
ctx.beginPath();
ctx.arc(100, 100, 100, 0, Math.PI*2, true);
ctx.closePath();
ctx.fill();

//el rectangle es mig transparent
ctx.fillStyle = "rgba(255, 255, 0, .5)";
ctx.beginPath();
ctx.rect(15, 150, 120, 120);
ctx.closePath();
ctx.fill();
```

Jugueu una mica amb els colors i el valor alfa del rectangle per veure com responen els objectes .



Tercer pas: Acció

Ja hem fet una bola, ara farem que es mogui. Per això, crearem una funció **dibuixa()** que neteja la pantalla, assenyal a la pilota, i a continuació, actualitza la seva posició actual. Utilitzarem una crida a **setInterval(funció, temps d'espera)** en la funció **init()** per dir-li al navegador que executi la nostra funció cada **10 milisegons**, creant la il·lusió de moviment.

Heu de refer tot el que hem fet fins ara i deixar-lo així:

```
var x = 150;
var y = 150;
var dx = 2;
var dy = 4;
var ctx;

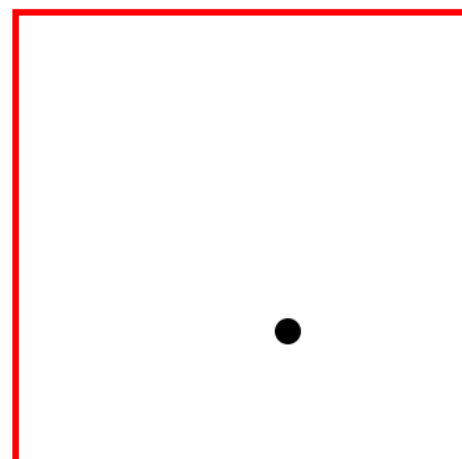
function init() {
  ctx = $('#canvas')[0].getContext("2d");
  return setInterval(dibuixa, 10);
}

function dibuixa() {
  ctx.clearRect(0,0,300,300);
  ctx.beginPath();
  ctx.arc(x, y, 10, 0, Math.PI*2, true);
  ctx.closePath();
  ctx.fill();
  x += dx;
  y += dy;
}

init();
```

Ara és molt més compacte.

Proveu de canviar els valors de **dx** i **dy** per canviar la direcció de la pilota, o canviar les variables **x** i **y** per canviar el lloc on la pilota s'iniciarà. No deixeu de provar els valors negatius de **dx** i **dy**.

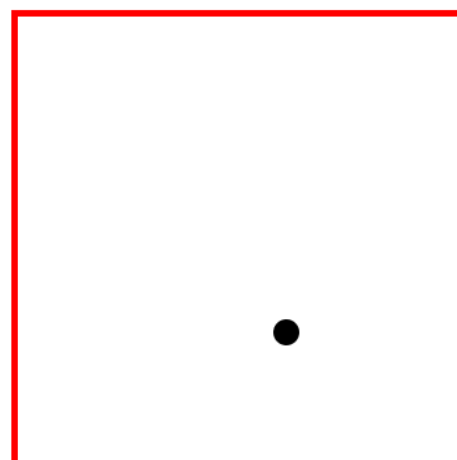


Quart pas: Fem una llibreria

Ara que estem arribant a algun lloc, el nostre codi s'està fent massa gran per a una sola pantalla, així que començarem a refer-lo una mica en una llibreria de funcions per facilitar-nos la vida. Una vegada fet això, ens podem centrar en la nostra funció **dibuixa()**.

Cada vegada que veiem més codi i el provem, l'anirem afegint a la nostra llibreria.

```
//INICI DEL CODI DE LA LLIBRERIA  
var x = 150;  
var y = 150;  
var dx = 2;  
var dy = 4;  
var AMPLE;  
var ALTURA;  
var ctx;  
  
function init() {  
    ctx = $('#canvas')[0].getContext("2d");  
    AMPLE = $("#canvas").width();  
    ALTURA = $("#canvas").height();  
    return setInterval(dibuixa, 10);  
}  
  
function cercle(x,y,r) {  
    ctx.beginPath();  
    ctx.arc(x, y, r, 0, Math.PI*2, true);  
    ctx.closePath();  
    ctx.fill();  
}  
  
function rect(x,y,w,h) {  
    ctx.beginPath();  
    ctx.rect(x,y,w,h);  
    ctx.closePath();  
    ctx.fill();  
}  
  
function clear() {  
    ctx.clearRect(0, 0, AMPLE, ALTURA);  
}  
  
//FINAL DEL CODI DE LA LLIBRERIA  
  
function dibuixa() {  
    clear();  
    cercle(x, y, 10);  
  
    x += dx;  
    y += dy;  
}  
  
init();
```



Vegeu com ara és més simple la funció **dibuixa()**.

Cinquè pas: Rebotar

La nostra bola es mou, però s'escapa massa ràpid, anem a contenir el rebot dins les parets de la nostra caixa.

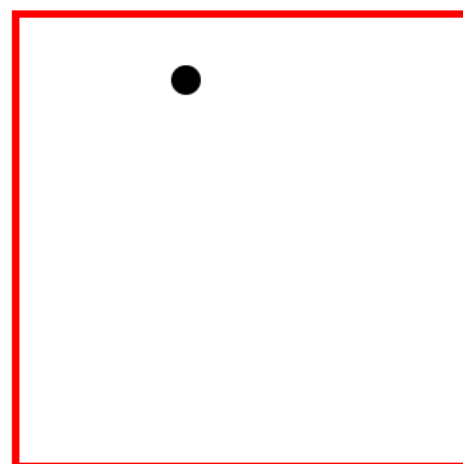
Afegim a la funció **dibuixa()**.

```
function dibuixa() {
  clear();
  cercle(x, y, 10);

  if (x + dx > AMPLE || x + dx < 0)
    dx = -dx;
  if (y + dy > ALTURA || y + dy < 0)
    dy = -dy;

  x += dx;
  y += dy;
}

init();
```



Proveu de canviar la funció **dibuixa()** de manera que la pilota s'acceleri o desacceleri cada vegada que colpeja una paret.

Evidentment aquesta es una imatge fixa, però vosaltres, mitjançant el navegadors veureu una piloteta que rebotar per les quatre cantonades



Sisè pas: Afegim una pala (o una nau!)

Ara podem començar a pensar en fer del nostre joc una mica com un joc real. Anem a afegir una pala, a la part inferior, i només permetrem que la bola reboti en el fons quan se li colpegi amb la pala.

Retoqueu amb el següent codi.

```
var paddlex;
var paddleh;
var paddlew;

function init_paddle() {
    paddlex = AMPLE / 2; // lloc x inicial de la pala
    paddleh = 10; // alçada de la pala
    paddlew = 75; // amplada de la pala
}

function dibuixa() {
    clear();
    cercle(x, y, 10);
    rect(paddlex, ALTURA-paddleh, paddlew, paddleh);

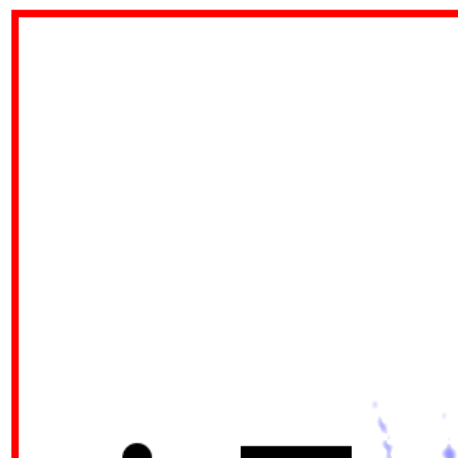
    if (x + dx > AMPLE || x + dx < 0)
        dx = -dx;
    if (y + dy < 0)
        dy = -dy;
    else if (y + dy > ALTURA) {
        if (x > paddlex && x < paddlex + paddlew)
            dy = -dy;
        else
            //game over, final del joc per tant l'animació s'atura
            clearInterval(intervalId);
    }

    x += dx;
    y += dy;
}

init();
init_paddle();
```

Probablement voldrem ser capaços de moure la pala - fem-ho seguidament.

Les constants **paddlex**, **paddleh**, i **paddlew** les mourem a la funció **init()** principal, dins la llibreria.



Refarem la biblioteca de la següent manera: (canvieu de lloc la funció `init()`)

```
//INICI DEL CODI DE LA LLIBRERIA

var x = 150;
var y = 150;
var dx = 2;
var dy = 4;
var ctx;
var AMPLE;
var ALTURA;
var paddlex;
var paddleh = 10;
var paddlew = 75;
var intervalId = 0;

function cercle(x,y,r) {
    ctx.beginPath();
    ctx.arc(x, y, r, 0, Math.PI*2, true);
    ctx.closePath();
    ctx.fill();
}

function rect(x,y,w,h) {
    ctx.beginPath();
    ctx.rect(x,y,w,h);
    ctx.closePath();
    ctx.fill();
}

function clear() {
    ctx.clearRect(0, 0, AMPLE, ALTURA);
}

function init() {
    ctx = $('#canvas')[0].getContext("2d");
    AMPLE = $("#canvas").width()
    ALTURA = $("#canvas").height()
    paddlex = AMPLE / 2;
    intervalId = setInterval(dibuixa, 10);
    return intervalId;
}

//FINAL DEL CODI DE LA LLIBRERIA
```



Amb això hem eliminat, també, la funció `init_paddle()`.

Setè pas: El teclat

Per afegir l'entrada de teclat per controlar la nostra paleta, hem de fer dues coses: saber quan les fletxes esquerra i dreta han estat pressionades i moure la pala amb elles.

Per tal de rebre els esdeveniments clau, crearem una funció anomenada **onKeyUp** i una altre anomenada **onKeyDown**, a continuació, utilitzarem una mica de la màgia de **jQuery** per unir-los als esdeveniments adequats.

Aleshores, en el nostre camí a través de la funció **dibuixa()**, anem a comprovar per veure si qualsevol de les fletxes quan se les pressiona, mouen, consegüentment, la paleta.

```
fletxaDreta = false;
fletxaEsquerra = false;

//Establir fletxaDreta o fletxaEsquerra si les tecles dreta o esquerra són polsades
function onKeyDown(evt) {
    if (evt.keyCode == 39) fletxaDreta = true;
    else if (evt.keyCode == 37) fletxaEsquerra = true;
}

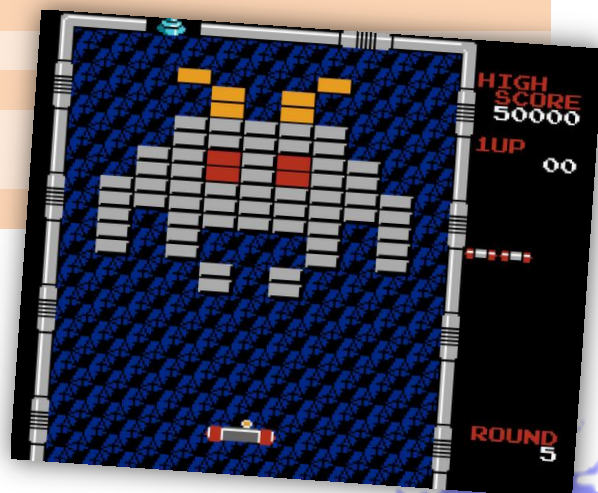
//i Les desactivem quan les tecles dreta o esquerra són alliberades
function onKeyUp(evt) {
    if (evt.keyCode == 39) fletxaDreta = false;
    else if (evt.keyCode == 37) fletxaEsquerra = false;
}
$(document).keydown(onKeyDown);
$(document).keyup(onKeyUp);

function dibuixa() {
    clear();
    cercle(x, y, 10);

    //moure la pala si les tecles dreta o esquerra son polsades
    if (fletxaDreta) paddlex += 5;
    else if (fletxaEsquerra) paddlex -= 5;
    rect(paddlex, ALTURA-paddleh, paddlew, paddleh);

    if (x + dx > AMPLE || x + dx < 0)
        dx = -dx;
    if (y + dy < 0)
        dy = -dy;
```

continua a la següent pàgina



```

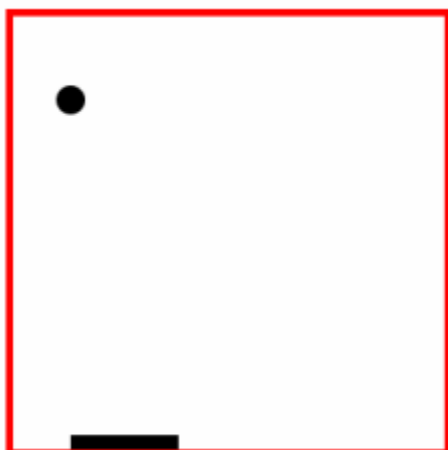
else if (y + dy > ALTURA) {
    if (x > paddlex && x < paddlex + paddlew)
        dy = -dy;
    else
        //game over, final del joc per tant l'animació s'atura
        clearInterval(intervalId);
}

x += dx;
y += dy;
}

init();

```

Ara que tenim una pala (o nau, com vosaltres vulgueu) que funciona, i una pilota que rebotja, tenim una mena de joc.



A continuació, anem a moure el suport per el teclat a la llibreria i afegirem el suport per el ratolí d'una manera molt similar a com ho hem fet amb el teclat.

En l'apartat variables, afegim a la biblioteca les dues variables de les tecles dreta i esquerra, i per què no, també dues més pel ratolí:

```

var x = 150;
var y = 150;
var dx = 2;
var dy = 4;
var AMPLE;
var ALTURA;
var ctx;

```

continua a la següent pàgina

```
var paddlex;
var paddleh = 10;
var paddlew = 75;
var fletxaDreta = false;
var fletxaEsquerra = false;
var canvasMinX = 0;
var canvasMaxX = 0;
var intervalId = 0;
```

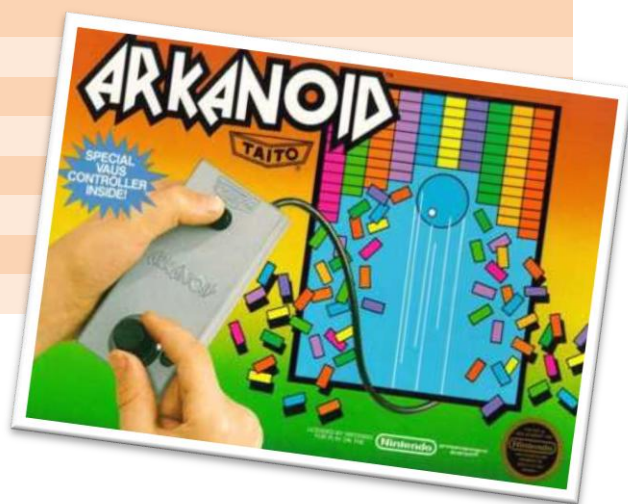
Afegim també les dues noves funcions del teclat, mes els esdeveniments, just abans de la funció `init()`.

```
//Activem fletxaDreta o fletxaEsquerra si les tecles dreta o esquerra són polsades
function onKeyDown(evt) {
    if (evt.keyCode == 39) fletxaDreta = true;
    else if (evt.keyCode == 37) fletxaEsquerra = true;
}

//i les desactivem quan les tecles dreta o esquerra són alliberades
function onKeyUp(evt) {
    if (evt.keyCode == 39) fletxaDreta = false;
    else if (evt.keyCode == 37) fletxaEsquerra = false;
}
$(document).keydown(onKeyDown);
$(document).keyup(onKeyUp);
```

Finalment afegim les dues noves variables que utilitzarem amb el ratolí

```
function init() {
    ctx = $('#canvas')[0].getContext("2d");
    AMPLE = $("#canvas").width();
    ALTURA = $("#canvas").height();
    paddlex = AMPLE / 2;
    canvasMinX = $("#canvas").offset().left;
    canvasMaxX = canvasMinX + AMPLE;
    intervalId = setInterval(dibuixa, 10);
    return intervalId;
}
```



Vuitè pas: El ratolí

Afegir el suport per el ratolí del nostre jocs encara és més simple que el del teclat, tot el que hem de fer és enviar l'esdeveniment `MouseMove` a una funció `onMouseMove`, veure si el ratolí és dins dels límits del joc, i moure la pala si ho és. Fixeu-vos en el que hem afegit.

Simple, no!


```

var canvasMinX;
var canvasMaxX;

function init_mouse() {
    canvasMinX = $("#canvas").offset().left;
    canvasMaxX = canvasMinX + AMPLE;
}

function onMouseMove(evt) {
    if (evt.pageX > canvasMinX && evt.pageX < canvasMaxX) {
        paddlex = evt.pageX - canvasMinX;
    }
}

$(document).mousemove(onMouseMove);

function dibuixa() {
    clear();
    cercle(x, y, 10);

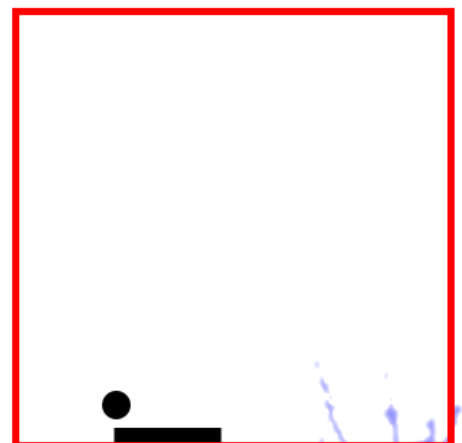
    //moure la pala si les tecles dreta o esquerra son polsades
    if (fletxaDreta) paddlex += 5;
    else if (fletxaEsquerra) paddlex -= 5;
    rect(paddlex, ALTURA-paddleh, paddlew, paddleh);

    if (x + dx > AMPLE || x + dx < 0)
        dx = -dx;
    if (y + dy < 0)
        dy = -dy;
    else if (y + dy > ALTURA) {
        if (x > paddlex && x < paddlex + paddlew)
            dy = -dy;
        else
            //game over, final del joc per tant l'animació s'atura
            clearInterval(intervalId);
    }

    x += dx;
    y += dy;
}

init();
init_mouse();

```



Novè pas: Els maons (o totxos)

Ara anem a crear una matriu de 2 dimensions per sostenir els maons, utilitzeu un parell de nanses per treure els que no s'han trencat, i assegureu-vos de treure els maons quan han estat colpejats.

Primer afegim el ratolí a la llibreria, sota mateix de l'última entrada de teclat.

```
$(document).keydown(onKeyDown);
$(document).keyup(onKeyUp);

//Movem el ratolí
function onMouseMove(evt) {
    if (evt.pageX > canvasMinX && evt.pageX < canvasMaxX) {
        paddlex = evt.pageX - canvasMinX;
    }
}

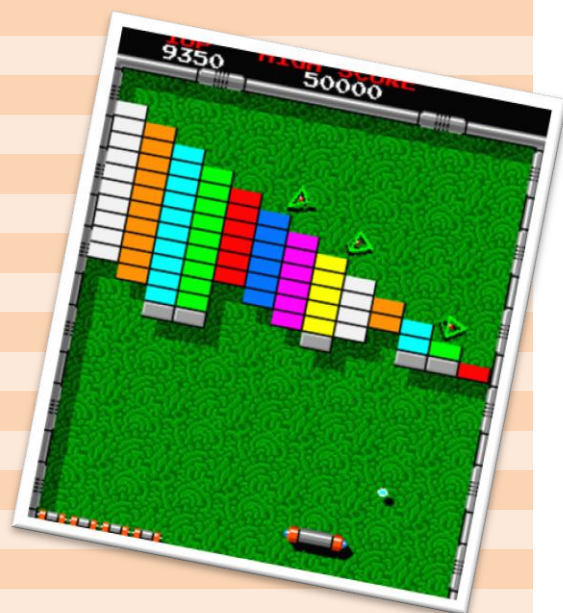
$(document).mousemove(onMouseMove);
```

Ja hem afegit la funció `init_mouse()` al `init()` principal abans. Eliminem el `init_mouse()`, l'esdeveniment i la seva crida i llestos. Ja podem continuar. Anem pel maons.

```
var maons;
var NFILES;
var NCOLS;
var MAONSAMPLE;
var MAONSALTURA;
var FARCIT;

function init_maons() {
    NFILES = 5;
    NCOLS = 5;
    MAONSAMPLE = (AMPLE/NCOLS) - 1;
    MAONSALTURA = 15;
    FARCIT = 1;

    maons = new Array(NFILES);
    for (i=0; i < NFILES; i++) {
        maons[i] = new Array(NCOLS);
        for (j=0; j < NCOLS; j++) {
            maons[i][j] = 1;
        }
    }
}
```



continua a la següent pàgina

```
function dibuixa() {
    clear();
    cercle(x, y, 10);

    //moure la pala si les tecles dreta o esquerra son polsades
    if (fletxaDreta) paddlex += 5;
    else if (fletxaEsquerra) paddlex -= 5;
    rect(paddlex, ALTURA-paddleh, paddlew, paddleh);

    //dibuixa els maons
    for (i=0; i < NFILES; i++) {
        for (j=0; j < NCOLS; j++) {
            if (maons[i][j] == 1) {
                rect((j * (MAONSAMPLE + FARCIT)) + FARCIT,
                    (i * (MAONSALTURA + FARCIT)) + FARCIT,
                    MAONSAMPLE, MAONSALTURA);
            }
        }
    }

    //hem colpejat un maó?
    alturafila = MAONSALTURA + FARCIT;
    amplecol = MAONSAMPLE + FARCIT;
    fila = Math.floor(y/alturafila);
    col = Math.floor(x/ amplecol);
    // si és així, inverteix el sentit de la pilota i marca'l com trencat
    if (y < NFILES * alturafila && fila >= 0 && col >= 0 && maons[fila][col] == 1) {
        dy = -dy;
        maons[fila][col] = 0;
    }

    //mou i comprova
    if (x + dx > AMPLE || x + dx < 0)
        dx = -dx;
    if (y + dy < 0)
        dy = -dy;
    else if (y + dy > ALTURA) {
        if (x > paddlex && x < paddlex + paddlew)
            dy = -dy;
        else
            //game over, final del joc per tant l'animació s'atura
            clearInterval(intervalId);
    }

    x += dx;
    y += dy;
}
```



```
init();
init_maons();
```

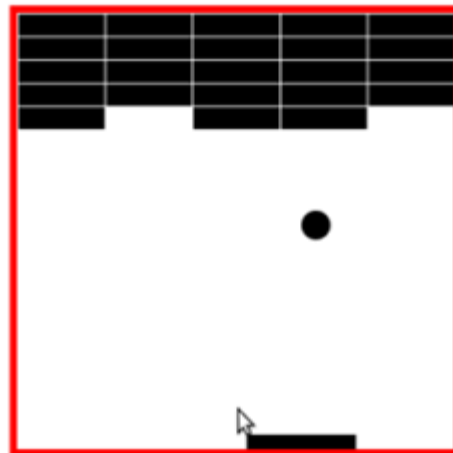
Podem Intentar afegir codi per fer que la pilota reboti en diferents direccions segons el lloc de la pala on colpegi.

Pel que es veu, a cada pas, tenim més a prop el de assemblar-se a un joc real, realment, no és gaire més enllà.

Però una cosa que si és segura, és que aquesta versió del joc es força lletja.

En l'últim pas anem a intentar millorar-lo, estèticament, una mica.

ARKANOID
El joc legendari

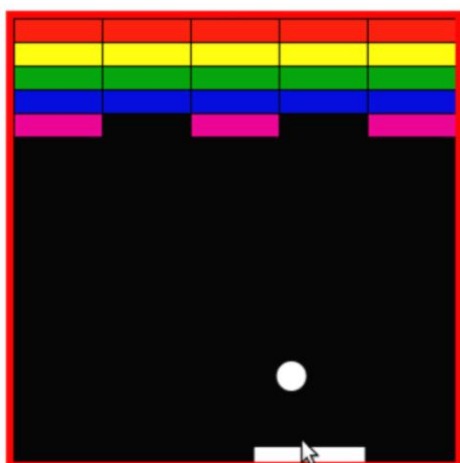


Deu i últim pas: Tocs finals

Si heu arribat fins aquí i encara no us heu perdut, es que alguna cosa hem fet malament. Us portarà un temps familiaritzar-vos i comprendre totalment tot el joc i estar en condicions de d'afegir o millorar qualsevol part, però creiem que la seva realització portarà més beneficis que altre cosa.

Pràctica DAW0922-PR07

ARKANOID
El joc legendari



Departament Informàtica - INS. La Pineda-2013

Ara anem a afegir uns tocs finals al nostre joc, com els colors, i una equació de col·lisió amb la pala més divertida. Hi ha una quantitat infinita de millores que es podrien fer, així que endavant, porteu-les a terme, si us atreviu i teniu temps, clar!.

En la propera pàgina donem tot el codi acabat. Afegiu allò que no tingueu.

Fixeu-vos be en el nom de les variables, qualsevol equivocació, per mínima que sigui, no funcionarà.

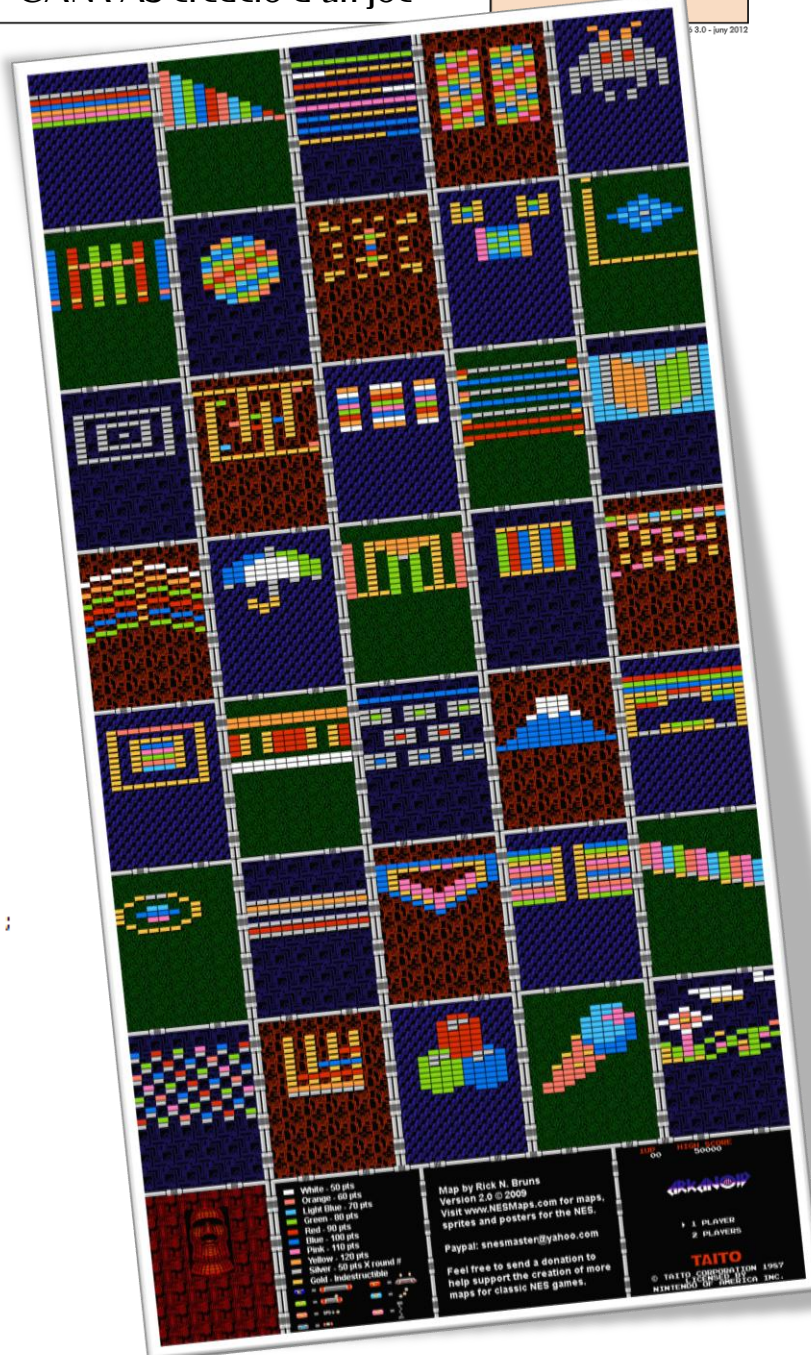
El codi està comprovat, i funciona!

Així es com quedarà amb les millores del codi.


```

1  //INICI DEL CODI DE LA BIBLIOTECA
2
3  var x = 25;
4  var y = 250;
5  var dx = 1.5;
6  var dy = -4;
7  var AMPLE;
8  var ALTURA;
9  var ctx;
10 var paddlex;
11 var paddleh = 10;
12 var paddlew = 75;
13 var fletxaDreta = false;
14 var fletxaEsquerra = false;
15 var canvasMinX = 0;
16 var canvasMaxX = 0;
17 var intervalId = 0;
18 var maons;
19 var NFILES = 5;
20 var NCOLS = 5;
21 var MAONSAMPLE;
22 var MAONSALTURA = 15;
23 var FARCIT = 1;
24
25 function cercle(x,y,r) {
26     ctx.beginPath();
27     ctx.arc(x, y, r, 0, Math.PI*2, true);
28     ctx.closePath();
29     ctx.fill();
30 }
31
32 function rect(x,y,w,h) {
33     ctx.beginPath();
34     ctx.rect(x,y,w,h);
35     ctx.closePath();
36     ctx.fill();
37 }
38
39 function clear() {
40     ctx.clearRect(0, 0, AMPLE, ALTURA);
41     rect(0,0, AMPLE, ALTURA);
42 }
43
44 //Establir fletxaDreta o fletxaEsquerra si les tecles dreta o esquerra són polsades
45 function onKeyDown(evt) {
46     if (evt.keyCode == 39) fletxaDreta = true;
47     else if (evt.keyCode == 37) fletxaEsquerra = true;
48 }
49
50 //i les desactivem quan les tecles dreta o esquerra són alliberades
51 function onKeyUp(evt) {
52     if (evt.keyCode == 39) fletxaDreta = false;
53     else if (evt.keyCode == 37) fletxaEsquerra = false;
54 }
55
56 $(document).keydown(onKeyDown);
57 $(document).keyup(onKeyUp);
58

```



```

58
59 //Movem el ratolí
60 function onMouseMove(evt) {
61     if (evt.pageX > canvasMinX && evt.pageX < canvasMaxX) {
62         paddlex = Math.max(evt.pageX - canvasMinX - (paddlew/2), 0);
63         paddlex = Math.min(AMPLE - paddlew, paddlex);
64     }
65 }
66
67 $(document).mousemove(onMouseMove);
68
69 //Inicialitza els maons
70 function init_maons() {
71     maons = new Array(NFILES);
72     for (i=0; i < NFILES; i++) {
73         maons[i] = new Array(NCOLS);
74         for (j=0; j < NCOLS; j++) {
75             maons[i][j] = 1;
76         }
77     }
78 }
79
80 //dibuixa els maons
81 function dibuixaMaons(){
82     for (i=0; i < NFILES; i++) {
83         ctx.fillStyle = colorslinia[i];
84         for (j=0; j < NCOLS; j++) {
85             if (maons[i][j] == 1) {
86                 rect((j * (MAONSAMPLE + FARCIT)) + FARCIT, (i * (MAONSALTURA + FARCIT)) + FARCIT,
87                     MAONSAMPLE, MAONSALTURA);
88             }
89         }
90     }
91 }
92
93 function init() {
94     ctx = $('#canvas')[0].getContext("2d");
95     AMPLE = $('#canvas').width();
96     ALTURA = $('#canvas').height();
97     paddlex = AMPLE / 2;
98     MAONSAMPLE = (AMPLE/NCOLS) - 1;
99     canvasMinX = $('#canvas').offset().left;
100     canvasMaxX = canvasMinX + AMPLE;
101     intervalId = setInterval(dibuixa, 10);
102     return intervalId;
103 }
104
105 //FINAL DEL CODI DE LA BIBLIOTECA

```




```

103
106 var radipilota = 10;
107 var colorslinia = ["#FF1C0A", "#FFFD0A", "#00A308", "#0008DB", "#EB0093"];
108 var colorpala = "#FFFFFF";
109 var colorpilota = "#FFFFFF";
110 var colorfons = "#000000";
111
112
113
114 function dibuixa() {
115     ctx.fillStyle = colorfons;
116     clear();
117     ctx.fillStyle = colorpilota;
118     cercle(x, y, radipilota);
119
120     //moure la pala si les tecles dreta o esquerra son polsades
121     if (fletxaDreta) paddlex += 5;
122     else if (fletxaEsquerra) paddlex -= 5;
123     ctx.fillStyle = colorpala;
124     rect(paddlex, ALTURA-paddleh, paddlew, paddleh);
125
126     dibuixaMaons();
127
128     //hem colpejat un maó?
129     alturafila = MAONSALTURA + FARCIT;
130     amplecol = MAONSAMPLE + FARCIT;
131     fila = Math.floor(y/alturafila);
132     col = Math.floor(x/ amplecol);
133     // si és així, inverteix el sentit de la pilota i marca'l com trencat
134     if (y < NFILES * alturafila && fila >= 0 && col >= 0 && maons[fila][col] == 1) {
135         dy = -dy;
136         maons[fila][col] = 0;
137     }
138
139     // mou i comprova
140     if (x + dx + radipilota > AMPLE || x + dx - radipilota < 0)
141         dx = -dx;
142
143     if (y + dy + radipilota < 0)
144         dy = -dy;
145     else if (y + dy + radipilota > ALTURA - paddleh) {
146         if (x > paddlex && x < paddlex + paddlew){
147             //Mouem la pilota de manera diferent segons el lloc de la pala on es colpeja.
148             dx = 8 * ((x-(paddlex+paddlew/2))/paddlew);
149             dy = -dy;
150         }
151         else if (y + dy + radipilota > ALTURA)
152             //game over, final del joc per tant l'animació s'atura
153             clearInterval(intervalId);
154     }
155
156     x += dx;
157     y += dy;
158 }
159
160 init();
161 init_maons();

```



Això és tot. El codi presentat aquí és plagiat, la seva major part, d'un manual trobat per Internet, i que sembla ha estat elaborat per un tal Bill Mill.

ACTIVITATS i EXERCICIS:

Llegiu-vos tota la pràctica fins el final.



Paper.js



PREMISSES:

1. Llegiu-vos completament la pràctica i si cal construïu-lo. Aneu en compte amb els possibles errors, arregleu-los. Si voleu no cal utilitzar **jQuery**.
2. Escriviu el codi del joc utilitzant el **marc de treball** que heu escollit en un arxiu que anomenareu **"arkanoid.js"**. No si val lliurar-lo amb el **canvas** normal, o sigui copiant el codi d'aquesta pràctica.
3. Utilitzeu el document **HTML5**, anomenat **"arkanoid.html"**, que subministrem, canvieu tot allò que necessiteu. Respecteu les mides del **canvas**.
4. Heu de fer una d'aquestes dues millores, o canvieu el fons del llenç per un degradat o també una imatge o reemplaça la pala per una imatge de la nau **"Vaus"**. Us en donem una. Pels més esbojarrats podeu fer-ne totes dues millores, si us atreviu!. Fixeu-vos en el temps que teniu.

LLIURAMENT:

Comprimiu el document **HTML5** si l'heu retocat junt amb el codi i les imatges (si les utilitzeu), en un arxiu amb el format **7z**, anomeu-lo **"DAW0922-PR10_vostresinicials.7z"** i envieu-lo a l'aula virtual.

Recordeu que **"vostresinicials"**, són les vostres inicials de nom i cognoms o el vostre cognom real.

Per exemple, si fos la primera pràctica:

El vostre nom i cognoms foren: **Ferran Martínez i Olivé**

Nom de l'arxiu de la vostra pràctica seria: "daw0922-pr10_fmo.7z" o també "daw0922-pr10_martinezolive.zip"

