

JavaScript Game Development

github.com/eaaa/jsgames

The “Catch” Game

0



Highscore

Player	Score
--------	-------



How-to

How-to

- Planning - break down in manageable pieces

How-to

- Planning - break down in manageable pieces
- code - test - code - test...

How-to

- Planning - break down in manageable pieces
- code - test - code - test...
- Debug

How-to

- Planning - break down in manageable pieces
- code - test - code - test...
- **Debug**

Planning

Planning

Set the stage

Planning

Set the stage

Create player

Planning

Set the stage

Create player

Create objects

Planning

Set the stage

Create player

Create objects

Create game

Planning

Set the stage

Create player

Create objects

Create game

Score

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Sound

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Sound

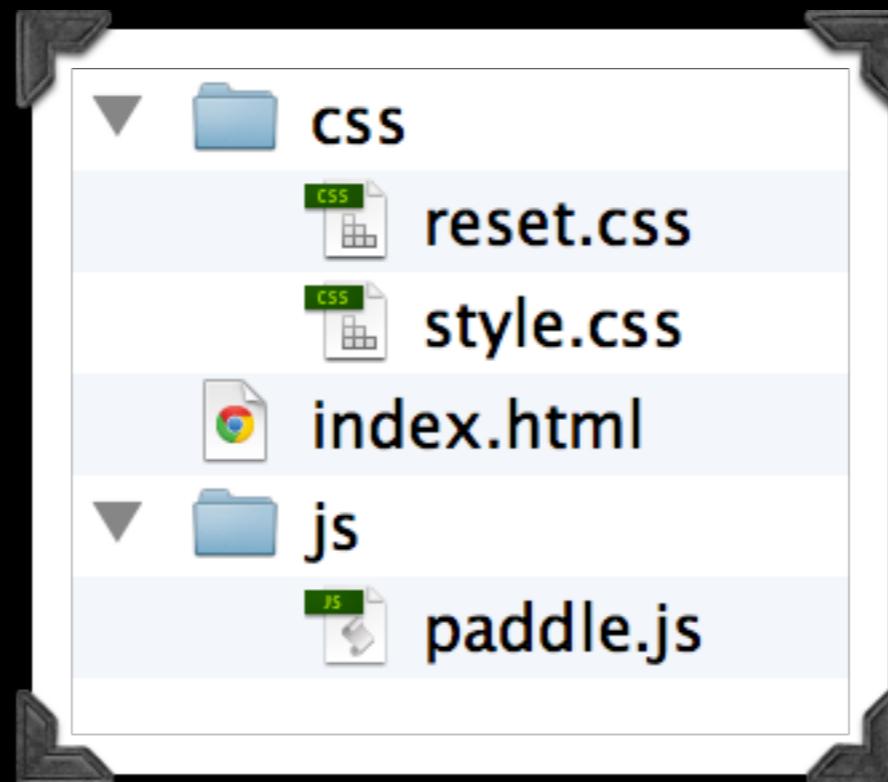
Replay



Setting The Stage

The Stage

- The files



The Stage

```
<!doctype html>
<html>

    <head>
        <meta charset="utf-8">
        <title>Paddle game</title>
        <link rel="stylesheet" href="css/reset.css" />
        <link rel="stylesheet" href="css/style.css" />
    </head>

    <body onLoad="pageReady()">
        <script src="js/paddle.js"></script>
    </body>

</html>
```

JavaScript

```
function pageReady(){
    console.log("I am ready");
}
```

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Sound

Replay

The Player

```
var paddle,
    keyPressed = false,
    paddleSpeed = 10;

function pageReady(){
    paddle = document.getElementById("paddle");
    window.addEventListener("keydown", keyDown);
    window.addEventListener("keyup", keyUp);
    setInterval(movePaddle, 5);
}

function keyDown(e){
    keyPressed = e.keyCode;
}

function keyUp(e){
    keyPressed = false;
}

function movePaddle(){
    if(keyPressed){
        if(keyPressed == 39){
            paddle.style.left = paddle.offsetLeft + paddleSpeed + "px";
        }
        else if(keyPressed == 37){
            paddle.style.left = paddle.offsetLeft - paddleSpeed + "px";
        }
    }
}
```

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Sound

Replay

The Object

```
var animationId,
    stageBottom,
    stageRight,
    ball,
    ballStart,
    ballSpeed = 5,
    fps = 40;

function pageReady(){
    // Stage dimensions
    stageBottom = window.innerHeight;
    stageRight = window.innerWidth;

    // Setup ball
    ball = document.getElementById("ball");
    var ballWidth = ball.offsetWidth;
    ballStart = -ballWidth + "px";
    ball.style.top = ballStart;
    ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth);

    // Animate the ball
    animationId = setInterval/animateBall, 1000/fps);
}

function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
    }
    else{
        // Move ball
        ball.style.top = ball.offsetTop + ballSpeed + "px";
    }
}

function randomRange(min, max){
    return(Math.random()*(max-min)+min);
}
```

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Sound

Replay

The Game

```
function animateBall(){
    // Start animation and wait for next frame
    setTimeout(function(){
        // Tell the browser to animate continuously
        window.requestAnimationFrame(animateBall);
        if(ball.offsetTop > stageBottom){
            // Return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
        }
        else{
            if(collisionTest(ball, paddle)){
                // if collision between ball and paddle: return to top
                ball.style.top = ballStart;
                ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            }
            else{
                // Move ball
                ball.style.top = ball.offsetTop + ballSpeed + "px";
            }
        }
    }, 1000/fps)
}
```

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Sound

Replay

Score

```
function animateBall(){
    // Start animation and wait for next frame
    setTimeout(function(){
        // Tell the browser to animate continuously
        window.requestAnimationFrame(animateBall);
        if(ball.offsetTop > stageBottom){
            // Return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
        }
        else{
            if(collisionTest(ball, paddle)){
                // if collision between ball and paddle: return to top
                ball.style.top = ballStart;
                ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
                score++;
                updateScoreBoard();
            }
            else{
                // Move ball
                ball.style.top = ball.offsetTop + ballSpeed + "px";
            }
        }
    }, 1000/fps)
}
```

Score

```
function animateBall(){
    // Start animation and wait for next frame
    setTimeout(function(){
        // Tell the browser to animate continuously
        window.requestAnimationFrame(animateBall);
        if(ball.offsetTop > stageBottom){
            // Return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth)
        }
        else{
            if(collisionTest(ball, paddle)){
                // if collision between ball and paddle: return to top
                ball.style.top = ballStart;
                ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
                score++;
                updateScoreBoard();
            }
            else{
                // Move ball
                ball.style.top = ball.offsetTop + ballSpeed + "px";
            }
        }
    }, 1000/fps)
}
```

```
var animationId,
    stageBottom,
    stageRight,
    ball,
    ballStart,
    ballSpeed = 5,
    ballWidth,
    fps = 40,
    keyPressed = false,
    paddleSpeed = 10,
    paddleWidth,
    scoreBoard,
    score = 0;
```

Score

```
function animateBall(){
    // Start animation and wait for next frame
    setTimeout(function(){
        // Tell the browser to animate continuously
        window.requestAnimationFrame(animateBall);
        if(ball.offsetTop > stageBottom){
            // Return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth)
        }
        else{
            if(collisionTest(ball, paddle)){
                // if collision between ball and paddle: return to top
                ball.style.top = ballStart;
                ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
                score++;
                updateScoreBoard();
            }
            else{
                // Move ball
                ball.style.top = ball.offsetTop + paddleSpeed;
            }
        }
    }, 1000/fps)
}
```

```
var animationId,
    stageBottom,
    stageRight,
    ball,
    ballStart,
    ballSpeed = 5,
    ballWidth,
    fps = 40,
    keyPressed = false,
    paddleSpeed = 10,
    paddleWidth,
    scoreBoard,
    score = 0;
```

```
function updateScoreBoard(){
    scoreBoard.innerHTML = score;
}
```

Score

```
function pageReady(){
    // Stage dimensions
    stageBottom = window.innerHeight;
    stageRight = window.innerWidth;

    // Setup ball
    ball = document.getElementById("ball");
    ballWidth = ball.offsetWidth;
    ballStart = -ballWidth + "px";
    ball.style.top = ballStart;
    ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

    // Animate the ball
    animateBall();

    // Setup paddle
    paddle = document.getElementById("paddle");
    paddleWidth = paddle.offsetWidth;
    window.addEventListener("keydown", keyDown);
    window.addEventListener("keyup", keyUp);
    setInterval(movePaddle, 5);

    // Setup scoreboard
    scoreBoard = document.getElementById("scoreBoard");
    updateScoreBoard();
}
```

Score

```
function pageReady(){
    // Stage dimensions
    stageBottom = window.innerHeight;
    stageRight = window.innerWidth;

    // Setup ball
    ball = document.getElementById("ball");
    ballWidth = ball.offsetWidth;
    ballStart = -ballWidth + "px";
    ball.style.top = ballStart;
    ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

    // Animate the ball
    animateBall();

    // Setup paddle
    paddle = document.getElementById("paddle");
    paddleWidth = paddle.offsetWidth;
    window.addEventListener("keydown", keyDown);
    window.addEventListener("keyup", keyUp);
    setInterval(movePaddle, 5);

    // Setup scoreboard
    scoreBoard = document.getElementById("scoreBoard");
    updateScoreBoard();
}
```

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Sound

Replay

Life

```
function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

        // Remove a life
        document.getElementById("life" + lifeCounter).style.display = "none";
        lifeCounter--;
        if(lifeCounter == 0){
            // Gameover
            clearInterval(animationId);
        }
    } else{
        if(collisionTest(ball, paddle)){
            // if collision between ball and paddle: return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            score++;
            updateScoreBoard();
        } else{
            // Move ball
            ball.style.top = ball.offsetTop + ballSpeed + "px";
        }
    }
}
```

Life

```
function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

        // Remove a life
        document.getElementById("life" + lifeCounter).style.display = "none";
        lifeCounter--;
        if(lifeCounter == 0){
            // Gameover
            clearInterval(animationId);
        }
    }  
    else{
        if(collisionTest(ball, paddle)){
            // if collision between ball and paddle: return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            score++;
            updateScoreBoard();
        }
        else{
            // Move ball
            ball.style.top = ball.offsetTop + ballSpeed + "px";
        }
    }
}
```

Life

```
function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

        // Remove a life
        document.getElementById("life" + lifeCounter).style.display = "none";
        lifeCounter--;
        if(lifeCounter == 0){
            // Gameover
            clearInterval(animationId);
        }
    } else{
        if(collisionTest(ball, paddle)){
            // if collision between ball and paddle: return
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            score++;
            updateScoreBoard();
        } else{
            // Move ball
            ball.style.top = ball.offsetTop + ballSpeed + "px";
        }
    }
}
```

```
var animationId,
    stageBottom,
    stageRight,
    ball,
    ballStart,
    ballSpeed = 15,
    ballWidth,
    fps = 40,
    keyPressed = false,
    paddleSpeed = 10,
    paddleWidth,
    scoreBoard,
    score = 0,
    lifeCounter = 3;
```

Life

```
function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

        // Remove a life
        document.getElementById("life" + lifeCounter).style.display = "none";
        lifeCounter--;
        if(lifeCounter == 0){
            // Gameover
            clearInterval(animationId);
        }
    } else{
        if(collisionTest(ball, paddle)){
            // if collision between ball and paddle: return
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            score++;
            updateScoreBoard();
        } else{
            // Move ball
            ball.style.top = ball.offsetTop + ballSpeed + "px";
        }
    }
}
```

```
var animationId,
    stageBottom,
    stageRight,
    ball,
    ballStart,
    ballSpeed = 15,
    ballWidth,
    fps = 40,
    keyPressed = false,
    paddleSpeed = 10,
    paddleWidth,
    scoreBoard,
    score = 0,
    lifeCounter = 3;
```

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Sound

Replay

Sound

```
<!doctype html>
<html>

  <head>
    <meta charset="utf-8">
    <title>Paddle game</title>
    <link rel="stylesheet" href="css/reset.css" />
    <link rel="stylesheet" href="css/style.css" />
  </head>

  <body onLoad="pageReady()">
    <div class="ball" id="ball"></div>
    <div id="paddle"></div>
    <div id="scoreBoard"></div>
    <div id="life">
      <div id="life1" class="lifemarker"></div>
      <div id="life2" class="lifemarker"></div>
      <div id="life3" class="lifemarker"></div>
    </div>
    <audio id="bgSound" src="sound/bg_sound.mp3"></audio>
    <audio id="gameOverSound" src="sound/fail-trombone-03.mp3"></audio>
    <audio id="hitSound" src="sound/hit_sound.mp3"></audio>
    <audio id="missSound" src="sound/miss_sound.mp3"></audio>
    <script src="js/collisionTest.js"></script>
    <script src="js/requestAnimationFrame.js"></script>
    <script src="js/paddle.js"></script>
  </body>

</html>
```

Sound

```
<!doctype html>
<html>

  <head>
    <meta charset="utf-8">
    <title>Paddle game</title>
    <link rel="stylesheet" href="css/reset.css" />
    <link rel="stylesheet" href="css/style.css" />
  </head>

  <body onLoad="pageReady()">
    <div class="ball" id="ball"></div>
    <div id="paddle"></div>
    <div id="scoreBoard"></div>
    <div id="life">
      <div id="life1" class="lifemarker"></div>
      <div id="life2" class="lifemarker"></div>
      <div id="life3" class="lifemarker"></div>
    </div>
    <audio id="bgSound" src="sound/bg_sound.mp3"></audio>
    <audio id="gameOverSound" src="sound/fail-trombone-03.mp3"></audio>
    <audio id="hitSound" src="sound/hit_sound.mp3"></audio>
    <audio id="missSound" src="sound/miss_sound.mp3"></audio>
    <script src="js/collisionTest.js"></script>
    <script src="js/requestAnimationFrame.js"></script>
    <script src="js/paddle.js"></script>
  </body>

</html>
```

Sound

```
function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

        // Remove a life
        document.getElementById("life" + lifeCounter).style.display = "none";
        document.getElementById("missSound").play();
        lifeCounter--;
        if(lifeCounter == 0){
            // Gameover
            clearInterval(animationId);
            document.getElementById("bgSound").pause();
            document.getElementById("gameOverSound").play();
        }
    }else{
        if(collisionTest(ball, paddle)){
            // if collision between ball and paddle: return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            score++;
            updateScoreBoard();
            document.getElementById("hitSound").play();
        }else{
            // Move ball
            ball.style.top = ball.offsetTop + ballSpeed + "px";
        }
    }
}
```

Sound

```
function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

        // Remove a life
        document.getElementById("life" + lifeCounter).style.display = "none";
        document.getElementById("missSound").play();
        lifeCounter--;
        if(lifeCounter == 0){
            // Gameover
            clearInterval(animationId);
            document.getElementById("bgSound").pause();
            document.getElementById("gameOverSound").play();
        }
    } else{
        if(collisionTest(ball, paddle)){
            // if collision between ball and paddle: return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            score++;
            updateScoreBoard();
            document.getElementById("hitSound").play();
        } else{
            // Move ball
            ball.style.top = ball.offsetTop + ballSpeed + "px";
        }
    }
}
```

Sound

```
function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

        // Remove a life
        document.getElementById("life" + lifeCounter).style.display = "none";
        document.getElementById("missSound").play();
        lifeCounter--;
        if(lifeCounter == 0){
            // Gameover
            clearInterval(animationId);
            document.getElementById("bgSound").pause();
            document.getElementById("gameOverSound").play();
        }
    }else{
        if(collisionTest(ball, paddle)){
            // if collision between ball and paddle: return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            score++;
            updateScoreBoard();
            document.getElementById("hitSound").play();
        }else{
            // Move ball
            ball.style.top = ball.offsetTop + ballSpeed + "px";
        }
    }
}
```

Sound

```
function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

        // Remove a life
        document.getElementById("life" + lifeCounter).style.display = "none";
        document.getElementById("missSound").play();
        lifeCounter--;
        if(lifeCounter == 0){
            // Gameover
            clearInterval(animationId);
            document.getElementById("bgSound").pause();
            document.getElementById("gameOverSound").play();
        }
    }else{
        if(collisionTest(ball, paddle)){
            // if collision between ball and paddle: return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            score++;
            updateScoreBoard();
            document.getElementById("hitSound").play();
        }else{
            // Move ball
            ball.style.top = ball.offsetTop + ballSpeed + "px";
        }
    }
}
```

Sound

```
function animateBall(){
    if(ball.offsetTop > stageBottom){
        // Return to top
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";

        // Remove a life
        document.getElementById("life" + lifeCounter).style.display = "none";
        document.getElementById("missSound").play();
        lifeCounter--;
        if(lifeCounter == 0){
            // Gameover
            clearInterval(animationId);
            document.getElementById("bgSound").pause();
            document.getElementById("gameOverSound").play();
        }
    } else{
        if(collisionTest(ball, paddle)){
            // if collision between ball and paddle: return to top
            ball.style.top = ballStart;
            ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
            score++;
            updateScoreBoard();
            document.getElementById("hitSound").play();
        } else
    }
}

// Start background music
document.getElementById("bgSound").play();
```

Planning

Set the stage

Create player

Create objects

Create game

Score

Life

Sound

Replay

Replay

```
<body onLoad="pageReady()">
  <div class="ball" id="ball"></div>
  <div id="paddle"></div>
  <div id="scoreBoard"></div>
  <div id="life">
    <div id="life1" class="lifemarker"></div>
    <div id="life2" class="lifemarker"></div>
    <div id="life3" class="lifemarker"></div>
  </div>
  <div id="gameOver">
    <p>Game Over</p>
    <button id="replay">Play it again, Sam</button>
  </div>
  <audio id="bgSound" src="sound/bg_sound.mp3"></audio>
  <audio id="gameOverSound" src="sound/fail-trombone-03.mp3"></audio>
  <audio id="hitSound" src="sound/hit_sound.mp3"></audio>
  <audio id="missSound" src="sound/miss_sound.mp3"></audio>
  <script src="js/collisionTest.js"></script>
  <script src="js/requestAnimationFrame.js"></script>
  <script src="js/paddle.js"></script>
</body>
```

Replay

```
<body onLoad="pageReady()">
  <div class="ball" id="ball"></div>
  <div id="paddle"></div>
  <div id="scoreBoard"></div>
  <div id="life">
    <div id="life1" class="lifemarker"></div>
    <div id="life2" class="lifemarker"></div>
    <div id="life3" class="lifemarker"></div>
  </div>
  <div id="gameOver">
    <p>Game Over</p>
    <button id="replay">Play it again, Sam</button>
  </div>
  <audio id="bgSound" src="sound/bg_sound.mp3"></audio>
  <audio id="gameOverSound" src="sound/fail-trombone-03.mp3"></audio>
  <audio id="hitSound" src="sound/hit_sound.mp3"></audio>
  <audio id="missSound" src="sound/miss_sound.mp3"></audio>
  <script src="js/collisionTest.js"></script>
  <script src="js/requestAnimationFrame.js"></script>
  <script src="js/paddle.js"></script>
</body>
```

Replay

```
<body onLoad="pageReady()">
  <div class="ball" id="ball"></div>
  <div id="paddle"></div>
  <div id="scoreBoard"></div>
  <div id="life">
    <div id="life1" class="lifemarker"></div>
    <div id="life2" class="lifemarker"></div>
    <div id="life3" class="lifemarker"></div>
  </div>
  <div id="gameOver">
    <p>Game Over</p>
    <button id="replay">Play it again, Sam</button>
  </div>
  <audio id="bgSound" src="sound/bg_sound.mp3"></audio>
  <audio id="gameOverSound" src="sound/fail-trombone-03.mp3"></audio>
  <audio id="hitSound" src="sound/hit_sound.mp3"></audio>
  <audio id="missSound" src="sound/miss_sound.mp3"></audio>
  <script src="js/collisionTest.js"></script>
  <script src="js/requestAnimationFrame.js"></script>
  <script src="js/paddle.js"></script>
</body>
```

```
#gameOver{
  position: absolute;
  top: -250px;
  width: 400px;
  height: 200px;
  background-color: #906;
  -moz-border-radius: 25px;
  -webkit-border-radius: 25px;
  border-radius: 25px;
  -webkit-transition-property: top;
  -webkit-transition-duration: 2s;
  -webkit-transition-timing-function: ease-in;
  font-size: 50px;
  color: #fff;
  text-align: center;
  padding-top: 40px;
}

#replay{
  background-color: #fff;
  width: 300px;
  height: 50px
  -moz-border-radius: 25px;
  -webkit-border-radius: 25px;
  border-radius: 25px;
  border: 0;
  padding: 20px;
  margin-top: 40px;
  font-size: 20px;
  cursor: pointer;
}
```


pageReady()

```
// Setup game over screen
gameOver = document.getElementById("gameOver");
gameOver.style.left = stageRight/2 - gameOver.offsetWidth/2 + "px";
document.getElementById("replay").addEventListener("click", replay);
```

pageReady()

```
// Setup game over screen
gameOver = document.getElementById("gameOver");
gameOver.style.left = stageRight/2 - gameOver.offsetWidth/2 + "px";
document.getElementById("replay").addEventListener("click", replay);

function replay(){
    // Slide game over screen up
    gameOver.style.top = "-250px";

    setTimeout(function(){

        // Reset life
        lifeCounter = 3;
        var lifemarkers = document.getElementsByClassName("lifemarker"),
            lifemarkersLength = lifemarkers.length,
            i;
        for(i=0; i<lifemarkersLength; i++){
            var lifemarker = lifemarkers[i];
            lifemarker.style.display = "inline";
        }

        // Reset score
        score = 0;
        updateScoreBoard();

        // Animate ball again
        ball.style.top = ballStart;
        ball.style.left = randomRange(ballWidth/2, stageRight-ballWidth) + "px";
        animationId = setInterval/animateBall, 1000/fps);

        document.getElementById("bgSound").play();
    }, 2500);
}
```

Bonus! The Highscore

Bonus! The Highscore

HTML5 localStorage

Bonus! The Highscore

HTML5 localStorage

Go and pick the 09-highscore folder

0



Highscore

Player	Score
--------	-------





Shop Amok!!!

[Start shopping](#)

WEEEEEE!!! Du skal shoppe!

Saml penge til den super smøkke Sex In The City boks
Brug højre og venstre piletast for at undgå de MEGA
irriterende facere der vil tage dine penge
og la' vær' med at falde over bænkene, girlfriend :-(
God fornøjelse! MYS MYS Love Ya!



niha@eaaa.dk

github.com/eaaa/jsgames