

JAVASCRIPT BREAKOUT GAME

1. Create a new Notepad file, save as index.html, and type in the following html to make it a webpage:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title>My Javascript Breakout Game</title>
<style>    * { padding: 0; margin: 0; }    canvas { background:
#eee; display: block; margin: 0 auto; }
</style>
</head>
<body> <canvas id="myCanvas" width="480" height="320"></canvas>
<script>    // All JavaScript code goes here
</script>
</body>
</html>
```

2. Add the following Javascript after the <script> tag.

```
var canvas = document.getElementById("myCanvas");
var ctx = canvas.getContext("2d");
var ballRadius = 10;
var x = canvas.width/2;
var y = canvas.height-30;
var dx = 2;
var dy = -2;
var paddleHeight = 10;
var paddleWidth = 75;
```

```
var paddleX = (canvas.width-paddleWidth)/2;
var rightPressed = false;
var leftPressed = false;
var brickRowCount = 5;
var brickColumnCount = 3;
var brickWidth = 75;
var brickHeight = 20;
var brickPadding = 10;
var brickOffsetTop = 30;
var brickOffsetLeft = 30;
var score = 0;
var lives = 3;

var bricks = [];
for(var c=0; c<brickColumnCount; c++) {
  bricks[c] = [];
  for(var r=0; r<brickRowCount; r++) {
    bricks[c][r] = { x: 0, y: 0, status: 1 };
  }
}

document.addEventListener("keydown", keyDownHandler,
false);
document.addEventListener("keyup", keyUpHandler, false);
document.addEventListener("mousemove", mouseMoveHandler,
false);

function keyDownHandler(e) {
  if(e.key == "Right" || e.key == "ArrowRight") {
    rightPressed = true;
  }
  else if(e.key == "Left" || e.key == "ArrowLeft") {
    leftPressed = true;
  }
}
```

```

    }
}

function keyUpHandler(e) {
    if(e.key == "Right" || e.key == "ArrowRight") {
        rightPressed = false;
    }
    else if(e.key == "Left" || e.key == "ArrowLeft") {
        leftPressed = false;
    }
}

function mouseMoveHandler(e) {
    var relativeX = e.clientX - canvas.offsetLeft;
    if(relativeX > 0 && relativeX < canvas.width) {
        paddleX = relativeX - paddleWidth/2;
    }
}

function collisionDetection() {
    for(var c=0; c<brickColumnCount; c++) {
        for(var r=0; r<brickRowCount; r++) {
            var b = bricks[c][r];
            if(b.status == 1) {
                if(x > b.x && x < b.x+brickWidth && y > b.y && y
< b.y+brickHeight) {
                    dy = -dy;
                    b.status = 0;
                    score++;
                    if(score == brickRowCount*brickColumnCount) {
                        alert("YOU WIN, CONGRATS!");
                        document.location.reload();
                    }
                }
            }
        }
    }
}

```

```
    }  
  }  
}  
}
```

```
function drawBall() {  
  ctx.beginPath();  
  ctx.arc(x, y, ballRadius, 0, Math.PI*2);  
  ctx.fillStyle = "#0095DD";  
  ctx.fill();  
  ctx.closePath();  
}  
function drawPaddle() {  
  ctx.beginPath();  
  ctx.rect(paddleX, canvas.height-paddleHeight,  
paddleWidth, paddleHeight);  
  ctx.fillStyle = "#0095DD";  
  ctx.fill();  
  ctx.closePath();  
}  
function drawBricks() {  
  for(var c=0; c<brickColumnCount; c++) {  
    for(var r=0; r<brickRowCount; r++) {  
      if(bricks[c][r].status == 1) {  
        var brickX =  
(r*(brickWidth+brickPadding))+brickOffsetLeft;  
        var brickY =  
(c*(brickHeight+brickPadding))+brickOffsetTop;  
        bricks[c][r].x = brickX;  
        bricks[c][r].y = brickY;  
        ctx.beginPath();  
        ctx.rect(brickX, brickY, brickWidth,  
brickHeight);
```

```

        ctx.fillStyle = "#0095DD";
        ctx.fill();
        ctx.closePath();
    }
}
}
}
function drawScore() {
    ctx.font = "16px Arial";
    ctx.fillStyle = "#0095DD";
    ctx.fillText("Score: "+score, 8, 20);
}
function drawLives() {
    ctx.font = "16px Arial";
    ctx.fillStyle = "#0095DD";
    ctx.fillText("Lives: "+lives, canvas.width-65, 20);
}

function draw() {
    ctx.clearRect(0, 0, canvas.width, canvas.height);
    drawBricks();
    drawBall();
    drawPaddle();
    drawScore();
    drawLives();
    collisionDetection();

    if(x + dx > canvas.width-ballRadius || x + dx <
ballRadius) {
        dx = -dx;
    }
    if(y + dy < ballRadius) {
        dy = -dy;
    }
}

```

```

}
else if(y + dy > canvas.height-ballRadius) {
    if(x > paddleX && x < paddleX + paddleWidth) {
        dy = -dy;
    }
    else {
        lives--;
        if(!lives) {
            alert("GAME OVER");
            document.location.reload();
        }
        else {
            x = canvas.width/2;
            y = canvas.height-30;
            dx = 3;
            dy = -3;
            paddleX = (canvas.width-paddleWidth)/2;
        }
    }
}

if(rightPressed && paddleX < canvas.width-paddleWidth)
{
    paddleX += 7;
}
else if(leftPressed && paddleX > 0) {
    paddleX -= 7;
}
x += dx;
y += dy;
requestAnimationFrame(draw);
}
draw();

```