

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>Zukunftstag</title>
  <link rel="icon" type="image/x-icon" href="Fotos/M365.png">
  <style>
    <!-- * { padding: 0; margin: 0; } canvas { background: #eee; display: block; margin: 0 auto; } -->
    body {
      background-image: url("Fotos/Netzwerk1.jpg");
      background-size: cover;
      background-repeat: no-repeat;
      display: flex;
      align-items: center;
      justify-content: center;
    }

    canvas {
      background-image: url("Fotos/Netzwerk1.jpg");
      background-size: cover;
      background-repeat: no-repeat;
      display: block;
      margin: 0 auto;
    }
  </style>
</head>
<body>
<canvas id="myCanvas" width="1700" height="900"></canvas>

<script>
  var canvas = document.getElementById("myCanvas"); //Diese Zeile ruft das Canvas-Element
  mit der ID "myCanvas" auf.
  var ctx = canvas.getContext("2d"); //Diese Zeile gibt den Kontext des Canvas-Elements
  zurück.
  var drohneRadius = 25; //Diese Zeile definiert den Radius der Drohne.
  var x = canvas.width/2; //Diese Zeile definiert die Startposition der Drohne.
  var y = canvas.height/2; //Diese Zeile definiert die Startposition der Drohne.
  var dx = 4; //Diese Zeilen definieren die Geschwindigkeit der Drohne.
  var dy = -4; //Diese Zeilen definieren die Geschwindigkeit der Drohne.
  var rollbrettHeight = 65; //Diese Zeilen definieren die Höhe und Breite des Rollbretts.
  var rollbrettWidth = 115;
  var rollbrettImage = new Image(); //neues Image-Objekt erstellen
  rollbrettImage.src = "Fotos/AGV.png"; //Pfad zum Bild
  var rollbrettX = (canvas.width-rollbrettWidth); //Diese Zeilen definieren die
  Startposition des Rollbretts.
  var rollbrettY = (canvas.height-rollbrettHeight);
  var rightPressed = false; //Diese Zeilen definieren, ob die Pfeiltasten gedrückt werden.
  var leftPressed = false;
  var containerRowCount = 8; //Diese Zeilen definieren die Anzahl der Containerreihen und
  -spalten.
  var containerColumnCount = 4;
  var containerWidth = 195; //Diese Zeilen definieren die Breite und Höhe der Container.
  var containerHeight = 70;
  var containerPadding = 10; //Diese Zeile definiert den Abstand zwischen den Containern.
  var containerOffsetTop = 60; //Diese Zeilen definieren den Abstand zwischen dem oberen
  linken Rand des Canvas-Elements und dem oberen linken Rand des ersten Containers.
  var containerOffsetLeft = 40;
  var containerImage = new Image(); //neues Image-Objekt erstellen
  containerImage.src = "Fotos/Container_1.png"; //Pfad zum Bild
  var score = 0; //Diese Zeile definiert den Punktestand des Spielers.
  var lives = 3; //Diese Zeile definiert die Anzahl der Leben des Spielers.
  var gameStarted = false; //Diese Zeile definiert, ob das Spiel gestartet wurde oder nicht.

  var containers = [];
  for(var c=0; c<containerColumnCount; c++) {
    containers[c] = [];
    for(var r=0; r<containerRowCount; r++) {
      containers[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.code == "ArrowRight") {
        rightPressed = true;
    }
    else if(e.code == 'ArrowLeft') {
        leftPressed = true;
    }
}
function keyUpHandler(e) {
    if(e.code == 'ArrowRight') {
        rightPressed = false;
    }
    else if(e.code == 'ArrowLeft') {
        leftPressed = false;
    }
}
function mouseMoveHandler(e) {
    var relativeX = e.clientX - canvas.offsetLeft;
    if(relativeX > 0 && relativeX < canvas.width) {
        rollbrettX = relativeX - rollbrettWidth/2;
    }
}

function startGame() {
    if (!gameStarted) {
        gameStarted = true;
        draw();
    }
}

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

function drawDrohne() { //Diese Funktion zeichnet die Drohne auf dem Canvas-Element.
    ctx.beginPath();
    var drohneImg = new Image();
    drohneImg.src = "Fotos/Drohne.png";
    ctx.drawImage(drohneImg, x, y, drohneRadius*2, drohneRadius*2);
    ctx.closePath();
}

function drawRollbrett() { //Diese Funktion zeichnet das Rollbrett auf dem Canvas-Element.
    ctx.beginPath();

```

```

var rollbrettImg = new Image();
rollbrettImg.src = "Fotos/AGV.png";
ctx.drawImage(rollbrettImg, rollbrettX, rollbrettY, rollbrettWidth, rollbrettHeight);
ctx.closePath();
}

function drawContainers() { //Diese Funktion zeichnet die Container auf dem
Canvas-Element.
    for(var c=0; c<containerColumnCount; c++) {
        for(var r=0; r<containerRowCount; r++) {
            if(containers[c][r].status == 1) {
                var containerX = (r*(containerWidth+containerPadding))+containerOffsetLeft
                ;
                var containerY = (c*(containerHeight+containerPadding))+containerOffsetTop
                ;
                containers[c][r].x = containerX;
                containers[c][r].y = containerY;
                ctx.beginPath();
                // ctx.rect(containerX, containerY, containerWidth, containerHeight);
                //ctx.fillStyle = "#0095DD";
                //ctx.fill();
                var containerImg = new Image();
                containerImg.src = "Fotos/Container_2.png";
                ctx.drawImage(containerImg, containerX, containerY, containerWidth,
                containerHeight);
                ctx.closePath();
            }
        }
    }
}

function drawScore() {
    ctx.font = "22px ArialBold";
    ctx.fillStyle = "white";
    ctx.fillText("Score: "+score, 8, 20);
}

function drawLives() {
    ctx.font = "22px ArialBold";
    ctx.fillStyle = "white";
    ctx.fillText("Lives: "+lives, canvas.width-150, 20);
}

function resetDrohne() {
x = canvas.width/2;
y = canvas.height/2;
}

function draw() {
    ctx.clearRect(0, 0, canvas.width, canvas.height);
    drawContainers();
    drawDrohne();
    drawRollbrett();
    drawScore();
    drawLives();
    collisionDetection();

if(x + dx > canvas.width - drohneRadius || x + dx < drohneRadius) {
    dx = -dx;
}
if(y + dy < drohneRadius) {
    dy = -dy;
} else if(y + dy > canvas.height - drohneRadius) {
    if(x > rollbrettX && x < rollbrettX + rollbrettWidth) {
        dy = -dy;
    } else {
        lives--;
        if(!lives) {
            alert("Du hast leider verloren ☹️");
            document.location.reload();
        } else {

```

```
        resetDrohne();
    }
}
if(y + dy > canvas.height - rollbrettHeight - drohneRadius) {
    if(x + drohneRadius > rollbrettX && x - drohneRadius < rollbrettX + rollbrettWidth) {
        dy = -dy;
    }
}

if(rightPressed && rollbrettX < canvas.width-rollbrettWidth) {
    rollbrettX += 7;
}
else if(leftPressed && rollbrettX > 0) {
    rollbrettX -= 7;
}

x += dx;
y += dy;
requestAnimationFrame(draw);
}

draw();
</script>

</body>
</html>
```