

Yokoscroll game

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>横スクロールゲーム</title>
</head>
<body>
  <script>
    // HTML Structure
    const body = document.body;
    body.style.margin = '0';
    body.style.overflow = 'hidden';

    const canvas = document.createElement('canvas');
    canvas.width = window.innerWidth;
    canvas.height = window.innerHeight;
    body.appendChild(canvas);

    const ctx = canvas.getContext('2d');

    // Game Variables
    let player = {
      x: 50,
      y: canvas.height / 2 - 25,
      width: 50,
      height: 50,
      color: 'blue',
      speed: 5
    };

    let obstacles = [];
    let score = 0;
    let gameSpeed = 3;

    // Helper Functions
    function createObstacle() {
      const height = Math.random() * (canvas.height / 2) + 50;
      const gap = 150;
      obstacles.push({
        x: canvas.width,
        y: 0,
        width: 50,
```

```

    height: height
  });
  obstacles.push({
    x: canvas.width,
    y: height + gap,
    width: 50,
    height: canvas.height - height - gap
  });
}

function detectCollision(player, obstacle) {
  return (
    player.x < obstacle.x + obstacle.width &&
    player.x + player.width > obstacle.x &&
    player.y < obstacle.y + obstacle.height &&
    player.y + player.height > obstacle.y
  );
}

// Event Listener
window.addEventListener('keydown', (e) => {
  if (e.key === 'ArrowUp') {
    player.y -= player.speed * 10;
  } else if (e.key === 'ArrowDown') {
    player.y += player.speed * 10;
  }
});

// Game Loop
function update() {
  ctx.clearRect(0, 0, canvas.width, canvas.height);

  // Draw Player
  ctx.fillStyle = player.color;
  ctx.fillRect(player.x, player.y, player.width, player.height);

  // Draw Obstacles
  ctx.fillStyle = 'red';
  for (let i = obstacles.length - 1; i >= 0; i--) {
    const obs = obstacles[i];
    obs.x -= gameSpeed;
    ctx.fillRect(obs.x, obs.y, obs.width, obs.height);

    // Remove off-screen obstacles
    if (obs.x + obs.width < 0) {
      obstacles.splice(i, 1);
      score++;
    }
  }
}

```

```
    }

    // Check for collision
    if (detectCollision(player, obs)) {
        alert(`Game Over! Your score: ${score}`);
        document.location.reload();
    }
}

// Add new obstacles
if (obstacles.length === 0 || obstacles[obstacles.length - 1].x < canvas.width
- 300) {
    createObstacle();
}

// Display Score
ctx.fillStyle = 'black';
ctx.font = '20px Arial';
ctx.fillText(`Score: ${score}`, 10, 30);

requestAnimationFrame(update);
}

// Start Game
update();

</script>
</body>
</html>
```