

03

Disparando contra o alvo

Transcrição

Excelente, a cada um segundo temos um alvo desenhado em posições aleatórias da nossa tela. Agora precisamos implementar a lógica de colisão, ou seja, aquela que detecta se o clique do botão do mouse é no centro do alvo:

```
<canvas width="600" height="400"></canvas>

<script>

var tela = document.querySelector('canvas');
var pincel = tela.getContext('2d');

pincel.fillStyle = 'lightgray';
pincel.fillRect(0, 0, 600, 400);

var raio = 10;

function desenhaCirculo(x, y, raio, cor) {

    pincel.fillStyle = cor;
    pincel.beginPath();
    pincel.arc(x, y, raio, 0, 2 * Math.PI);
    pincel.fill();
}

function limpaTela() {
    pincel.clearRect(0, 0, 600, 400);
}

function desenhaAlvo(x, y) {

    desenhaCirculo(x,y, raio + 20, 'red');
    desenhaCirculo(x,y, raio + 10, 'white');
    desenhaCirculo(x,y, raio, 'red');
}

function sorteiaPosicao(maximo) {
    return Math.floor(Math.random() * maximo);
}

function atualizaTela() {

    limpaTela();
    var xAleatorio = sorteiaPosicao(600);
    var yAleatorio = sorteiaPosicao(400);
    desenhaAlvo(xAleatorio, yAleatorio);
}

setInterval(atualizaTela, 1000);

```

```
// cria a função dispara que será chamado quando a tela for clicada
function dispara(evento) {

    // descobre onde o usuário clicou para comparar a seguir com a posição atual do alvo
    var x = evento.pageX - tela.offsetLeft;
    var y = evento.pageY - tela.offsetTop;

    if((x > xAleatorio - raio)
        && (x < xAleatorio + raio)
        && (y > yAleatorio - raio)
        && (y < yAleatorio + raio)) {

        alert('Acertou!');
    }

}

// associa a função `dispara` ao clique do canvas, ou seja, da nossa tela.
tela.onclick = dispara;

</script>
```

Lidando com o escopo de variáveis

Isso ainda não é suficiente. Isso porque a lógica de `dispara` precisa saber em que local da tela o alvo atual está para comparar sua posição com a posição do clique do mouse. Contudo, não temos acesso às variáveis `xAleatorio` muito menos `yAleatorio` na função `dispara`. A razão disso é que essas variáveis foram definidas com `var` dentro da função `AtualizaTela`. Variáveis declaradas com `var` dentro de uma função só podem ser acessadas dentro da função na qual foram declaradas. Sendo assim, jamais a função `dispara` terá acesso à essas variáveis declaradas em `AtualizaTela`. E agora?

Para que as variáveis sejam acessíveis dentro de qualquer função do nosso programa podemos declará-las antes, fora de qualquer função. Uma variável declarada com `var` fora de uma função pode ser acessível por qualquer outra função:

```
<canvas width="600" height="400"></canvas>

<script>

var tela = document.querySelector('canvas');
var pincel = tela.getContext('2d');

pincel.fillStyle = 'lightgray';
pincel.fillRect(0, 0, 600, 400);

var raio = 10;

// variáveis declaradas fora de uma função. São acessíveis por todas as funções
var xAleatorio;
var yAleatorio;

function desenhaCirculo(x, y, raio, cor) {

    pincel.fillStyle = cor;
    pincel.beginPath();
    pincel.arc(x, y, raio, 0, 2 * Math.PI);
    pincel.fill();
}
```

```
pincel.beginPath();
pincel.arc(x, y, raio, 0, 2 * Math.PI);
pincel.fill();
}

function limpaTela() {

    pincel.clearRect(0, 0, 600, 400);
}

function desenhaAlvo(x, y) {

    desenhaCirculo(x,y, raio + 20, 'red');
    desenhaCirculo(x,y, raio + 10, 'white');
    desenhaCirculo(x,y, raio, 'red');
}

function sorteiaPosicao(maximo) {
    return Math.floor(Math.random() * maximo);
}

function atualizaTela() {

    limpaTela();
    // veja que foi removido a palavra var, porque as variáveis já foram declaradas anteriormente
    // O que fazemos aqui é apenas atualizar os valores das variáveis xAleatorio e yAleatorio
    xAleatorio = sorteiaPosicao(600);
    yAleatorio = sorteiaPosicao(400);

    desenhaAlvo(xAleatorio, yAleatorio);
}

setInterval(atualizaTela, 1000);

function dispara(evento) {

    var x = evento.pageX - tela.offsetLeft;
    var y = evento.pageY - tela.offsetTop;

    if((x > xAleatorio - raio)
        && (x < xAleatorio + raio)
        && (y > yAleatorio - raio)
        && (y < yAleatorio + raio)) {

        alert('Acertou!');
    }
}

tela.onclick = dispara;

</script>
```

Perfeito. Agora nosso programa funciona como o esperado!

