

- ❖ Basta invertermos a ordem de chamamento das funções, se chamarmos o círculo vermelho maior primeiro, e o círculo vermelho menor por último, da seguinte forma:
- ❖ Aprimoraremos nosso código, usando o raio inicial de 10 como parâmetro, guardando-o em uma variável. Em seguida, acrescentaremos os valores necessários para compor os raios das demais circunferências. Criaremos a variável `raio = 10` a seguir:
- ❖

```
function limpaTela() {
```
- ❖

```
    pincel.clearRect(0,
```
- ❖

```
    0, 600, 400);
```
- ❖

```
}
```
- ❖

```
    desenhaCirculo(100,
```
- ❖

```
    100, raio+20, 'red');
```
- ❖

```
    desenhaCirculo(100,
```
- ❖

```
    100, raio+10, 'white');
```
- ❖

```
    desenhaCirculo(100,
```
- ❖

```
    100, raio, 'red');
```
- ❖
- ❖
- ❖ Criaremos uma função chamada `sorteiaPosicao()`, que receberá um valor `maximo` que pode sortear entre 0 a 600 (para o eixo X), e 0 a 400 (para o eixo Y):
- ❖ Criaremos uma variável `xAleatorio` que recebe `sorteiaPosicao`, já que estamos trabalhando com a variável X, então o valor máximo

será 600. Do mesmo modo, criaremos uma variável `yAleatorio` que receberá `sorteiaPosicao`, cujo máximo será 400

- ❖

```
function
```
- ```
 sorteiaPosicao(maximo)
```
- ```
{
```
- ❖

```
}
```
- ❖
- ❖

```
    var xAleatorio =
```
- ```
 sorteiaPosicao(600);
```
- ❖ 

```
 var yAleatorio =
```
- ```
        sorteiaPosicao(400);
```
- ❖
- ❖

```
    desenhaAlvo(50,
```
- ```
 200);
```
- ❖
- ❖
- ❖ Ainda precisamos programar a função `sorteiaPosicao()`. Ela nos dará um retorno - `return` - de `Math.floor`, que arredonda o número para baixo, diferente do `Math.round`, que o arredonda para cima. Em seguida temos `'Math.random() * maximo)`:
- ❖ 

```
function
```
- ```
    sorteiaPosicao(maximo)
```
- ```
{
```
- ❖
- ❖ 

```
 return
```
- ```
        Math.floor(Math.random(
```
- ```
) * maximo);
```
- ❖
- ❖ 

```
}
```
- ❖
- ❖ 

```
function
```
- ```
    atualizaTela() {
```

```
❖ limpaTela();
❖ var xAleatorio =
sorteiaPosicao(600);
❖ var yAleatorio =
sorteiaPosicao(400);
❖
desenhaAlvo(xAleatorio,
yAleatorio);
❖ }
❖
❖
setInterval(atualizaTel
a, 1000);
```

❖ Vimos no vídeo que ao clicar no canvas, chamamos a função `dispara()`, e assim teremos acesso à posição do clique do mouse

Porém não saberemos a posição do alvo (`xAleatorio` e `yAleatorio`).

❖ Selecione a opção com a função que vai fazer essa troca de posições de tempos em tempos, quando for necessário.

❖ `setInterval()`.

❖ Com o `setInterval()` podemos definir um intervalo de tempo para executar a função novamente.

❖

Selecione a alternativa que resolve esse problema.

- Alternativa correta
- Declarar as variáveis `xAleatorio` e `yAleatorio` fora das funções.
- Se declararmos as variáveis fora das funções, poderemos acessar seu valor em qualquer função, inclusive em `dispara()`.