

02

O botão de Play/Stop

Transcrição

A primeira funcionalidade que vamos implementar no Alura Timer é **marcar o tempo**, ou seja, quando clicarmos no *play*, queremos que o aplicativo comece a contar o tempo (inicie o *timer*) do curso selecionado, que por padrão é o curso de Lógica de Programação. E assim que o botão de *play* for clicado, ele deve ser substituído pelo botão de *stop*.

Trocando a imagem do botão

Se queremos detectar um clique no *play*, assim como já vimos diversas vezes no curso, precisamos escutar o evento `click` do botão, que tem a classe `botao-play`. Fazemos isso no `renderer.js`:

```
// renderer.js

let botaoPlay = document.querySelector('.botao-play');

botaoPlay.addEventListener('click', function() {

});
```

A primeira coisa que vamos fazer é substituir a imagem, toda vez que o botão for clicado, a imagem é alterada. Se clicarmos no botão de *play*, a imagem é alterada para o botão de *stop*, que se for clicado, tem sua imagem alterada de volta para o botão de *play*.

Para realizarmos essa troca de imagens, basta alterarmos o `source` da imagem, a sua propriedade `src`. Na pasta `ap/img`, já temos as duas imagens, `play-button.svg` e `stop-button.svg`. Vamos colocar o caminho dessas imagens dentro de um array:

```
// renderer.js

let botaoPlay = document.querySelector('.botao-play');

let imgs = ['img/play-button.svg', 'img/stop-button.svg'];
botaoPlay.addEventListener('click', function() {

});
```

Agora, quando o botão for clicado, modificamos o `src` da imagem. Para isso, basta alterar a propriedade `src` do botão:

```
// renderer.js

let botaoPlay = document.querySelector('.botao-play');

let imgs = ['img/play-button.svg', 'img/stop-button.svg'];
botaoPlay.addEventListener('click', function() {
    botaoPlay.src = imgs[0];
});
```

Agora, ao clicar no botão, a sua imagem será trocada para o primeiro item do array. Só que essas imagens precisam ser alternadas, uma hora o primeiro item do array deve ser a imagem de *play*, outra hora de *stop*, se não a imagem será sempre fixa.

Invertendo os itens de um array

Para isso, sempre que clicarmos no botão, podemos inverter o array, garantindo assim a alternância das imagens, pois o primeiro item do array ficará alternando. Fazemos isso através da função `reverse()` :

```
// renderer.js

let botaoPlay = document.querySelector('.botao-play');

let imgs = ['img/play-button.svg', 'img/stop-button.svg'];
botaoPlay.addEventListener('click', function() {
    imgs = imgs.reverse();
    botaoPlay.src = imgs[0];
});
```

Pronto, agora as imagens estão sendo trocadas corretamente! Agora falta implementarmos o *timer*, pois ele deve ser inicializado quando o botão de *play* for clicado, e parado quando o botão de *stop* for clicado.

Para manter o nosso código organizado, deixaremos o script `renderer.js` como o responsável por manipular o DOM, interagir com os objetos, e criaremos um outro script dentro de `app.js`, o `timer.js`, responsável por interagir com o *timer*, iniciando-o, parando-o, tudo relacionado ao seu controle.

Começaremos a implementá-lo a partir do próximo vídeo :)