

Sobre o carousel

Transcrição

A partir de agora focaremos no carrossel que aparece na parte do blog - vocês devem ter testado e percebido que de fato ele funciona, e seu código em JavaScript nada mais é que uma forma de alterar qual conjunto de imagem e texto está ativo no momento.

Isto acontece pela inserção de uma classe, e por pegarmos todos os botões, executando-se a função de *callback* do `forEach`, e remoção da classe `listaDeArtigos-slider-item--ativo`, a ser adicionada justamente quando o botão for pressionado.

Vamos testar a página sem o `carousel.js`, indo à `index.html` e comentando sua chamada, assim:

```
<!-- <script src="js/carousel.js"></script>-->
```

Atualizaremos a página e veremos que os links continuam funcionando, pois utilizamos **links âncora**, um código pronto.

Vamos ativar o leitor de telas para navegar usando o "Tab". O NVDA lerá:

Continuar lendo o post 1

Continuar lendo o post 1

linque

marca navegação

lista com 3 itens

Notícia 1

visitado linque

Notícia 2

visitado linque

Notícia 3

visitado linque

Quando damos "Enter" no terceiro link/*bullet*, e depois usamos "Shift + Tab" para voltarmos, o leitor vai ao primeiro post. Percebam que a hierarquia está meio estranha. E se usarmos dois "Shift + Tab"s, o foco se alterna entre o primeiro e o terceiro posts. Isto está bem esquisito, e não parece um carrossel.

É possível fazer muito apenas utilizando HTML e CSS puros, sem termos que recorrer ao JavaScript, mas às vezes teremos que fazê-lo, infeliz ou felizmente.

A seção `secaoBlog` container contém a `ul listaDeArtigos`, mas como informaremos que esta lista possui um nome? Poderíamos rotular com o `aria-label`, no entanto o `h2` que vem acima já não indica do que se trata esta lista?

Sendo assim, poderemos usar `aria-labelledby` em seu lugar. Outra opção seria utilizamos `id`, porém, "blog" é muito genérico; trocaremos por "Lista de artigos", tornando o `h2` mais descritivo. Porém, isto não será muito apreciado por *copywriters* ou designers, por não ser atraente ou chamativo visualmente.

Poderíamos também inserir um `<h3>Lista de artigos</h3>` entre o `h2` e a `ul`, utilizando um `id` como `tituloBlog`, por exemplo. O código de `ul` ficaria `<ul class="listaDeArtigos" aria-labelledby="tituloBlog">`. Vamos salvar e testar no navegador.

O texto "Lista de artigos" aparece no topo do carrossel, o que não é interessante visualmente. Deixaremos isto visível apenas para o leitor de telas com uma classe feita para isto, o `escondeVisualmente`:

```
<h2 class="secaoBlog-titulo tituloPadrao">fique bem  
  BLOG  
</h2>  
  
<h3 id="tituloBlog" class="escondeVisualmente">Lista de artigos</h3>
```

Vamos testar na nossa página com o leitor de telas?

A alteração funciona perfeitamente, não implicando diretamente na experiência do usuário que enxerga.

Continuando, o atalho "Shift + Tab" está estranho demais, e não funciona como esperado. Basicamente, faremos algo muito parecido com o [carrossel que se encontra na página da W3C \(https://www.w3.org/WAI/tutorials/carousels/working-example/\)](https://www.w3.org/WAI/tutorials/carousels/working-example/), sem os botões exatamente como aparecem aqui.

Conforme se vai navegando com o "Tab", o NVDA vai lendo os botões, inclusive indicando o item atual, o que é bem interessante:

botão

News1 (Current Item)

botão

News2

botão

Com o "Enter", ele lê um trecho da notícia selecionada. E com o "Tab" continuamos navegando no menu.