

DESAFIO: Sistema de navegação

Forma de entrega: link do programa salvo no Gist do Github

Linguagens aceitas: Javascript, Java, C#, Python

Você foi contratado para implementar um componente de navegação. Para isso, você deverá implementar a classe Browser, com as funções de acessar uma página, voltar e avançar, conforme projeto ao lado.

Quando esse componente for instanciado, por padrão ele começa na página inicial da navegação, identificada pela string "home".

Quando o usuário informar a URL para acessar uma página, a função `access(url)` deve ser chamada. Essa página passa então a ser a página atual (`currentPage`) da navegação. A página atual da navegação pode ser acessada pela função `getCurrentPage()`.

Browser
- <code>currentPage</code> : String
- <code>backStack</code> : Stack
- <code>forwardStack</code> : Stack
+ <code>getCurrentPage()</code> : String
+ <code>access(page : String)</code> : void
+ <code>back()</code> : void
+ <code>forward()</code> : void

powered by Astah

Deve ser armazenado um histórico das páginas anteriores, pois quando o usuário acionar a funcionalidade "voltar" em seu navegador, a função `back()` deve ser chamada. Esta função faz a navegação voltar para a última página anteriormente acessada. Se a função `back()` for chamada e não houver páginas anteriores no histórico, deve ocorrer um erro "Back error".

Quando o usuário opta por "voltar" em pelo menos uma página, deve ser também armazenado um histórico de páginas futuras, pois o usuário pode acionar a funcionalidade de "avançar", que deve chamar a função `forward()`. Esta função faz a navegação avançar para a última página futura a partir da qual o usuário voltou. Se a função `forward()` for chamada e não houver páginas futuras no histórico, deve ocorrer um erro "Forward error".

Repare que, se o usuário acessar uma nova página diretamente informando sua URL, o histórico de páginas futuras deve ser esvaziado, pois o acesso a uma nova página invalida as páginas futuras das quais eventualmente o usuário tenha voltado.

Você deve criar a classe Browser conforme especificação acima.

Você deve também criar um programa para testar a classe Browser. Este programa deve instanciar um objeto Browser, e executar uma função `testNavigation` (veja assinatura da função ao final deste documento). Esta função deve receber um objeto browser e um array de strings representando uma sequência de comandos a serem chamados no objeto browser. A função `testNavigation` deve executar a sequência de comandos no objeto browser e, sempre que for dado um comando "get-current", deve ser exibido na saída a página atual (`currentPage`) da navegação. Veja os exemplos a seguir.

Entrada 1	Saída 1
<pre>["get-current", "access,https://amazon.com", "access,https://cnn.com", "get-current", "back", "get-current", "back", "get-current", "back"]</pre>	<pre>home https://cnn.com https://amazon.com home Error: Back error</pre>

Entrada 2	Saída 2
<pre>["access,https://amazon.com", "access,https://cnn.com", "get-current", "forward"]</pre>	<pre>https://cnn.com Error: Forward error</pre>

Entrada 3	Saída 3
<pre>["access,https://amazon.com", "access,https://cnn.com", "access,https://gmail.com", "access,https://outlook.com", "get-current", "back", "back", "back", "get-current", "forward", "forward", "get-current"]</pre>	<pre>https://outlook.com https://amazon.com https://gmail.com</pre>

Entrada 4	Saída 4
<pre>["access,https://amazon.com", "access,https://cnn.com", "access,https://gmail.com", "access,https://outlook.com", "get-current", "back", "back", "back", "get-current", "forward", "forward", "get-current", "access,https://devsuperior.com.br", "back", "forward", "get-current", "forward"]</pre>	<pre>https://outlook.com https://amazon.com https://gmail.com https://devsuperior.com.br Error: Forward error</pre>

Assinaturas:

Javascript:

```
function testNavigation(browser, commands)
```

Java:

```
static void testNavigation(Browser browser, String[] commands)
```

C#:

```
static void TestNavigation(Browser browser, string[] commands)
```

Python:

```
def test_navigation(browser, commands)
```