

Classe Alert!

Autor: Reinaldo M. R. Junior

HTML Alert Box:


```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Alert</h2>

<button onclick="myFunction()">Try it</button>

<script>
function myFunction() {
    alert("I am an alert box!");
}
</script>

</body>
</html>
```



Quando fazemos o click no botão, vai fazer a chamada do método onclick do elemento button, que dispara a função JavaScript ou seja o método alert.

HTML Prompt Box:

```
<!DOCTYPE html>
<html>
<body>

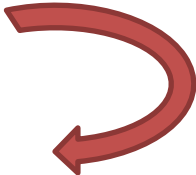
<p>Click the button to demonstrate the prompt box.</p>

<button onclick="myFunction()">Try it</button>

<p id="demo"></p>

<script>
function myFunction() {
    var person = prompt("Please enter your name", "Harry Potter");
    if (person != null) {
        document.getElementById("demo").innerHTML =
            "Hello " + person + "! How are you today?";
    }
}
</script>

</body>
</html>
```



HTML Confirm Box:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Confirm Box</h2>

<button onclick="myFunction()">Try it</button>

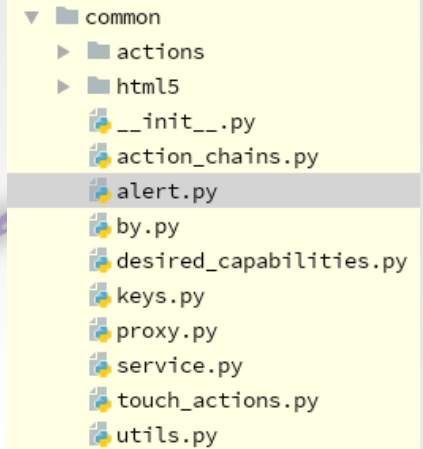
<p id="demo"></p>

<script>
function myFunction() {
    var txt;
    if (confirm("Press a button!") == true) {
        txt = "You pressed OK!";
    } else {
        txt = "You pressed Cancel!";
    }
    document.getElementById("demo").innerHTML = txt;
}
</script>

</body>
</html>
```



Alert Popup.



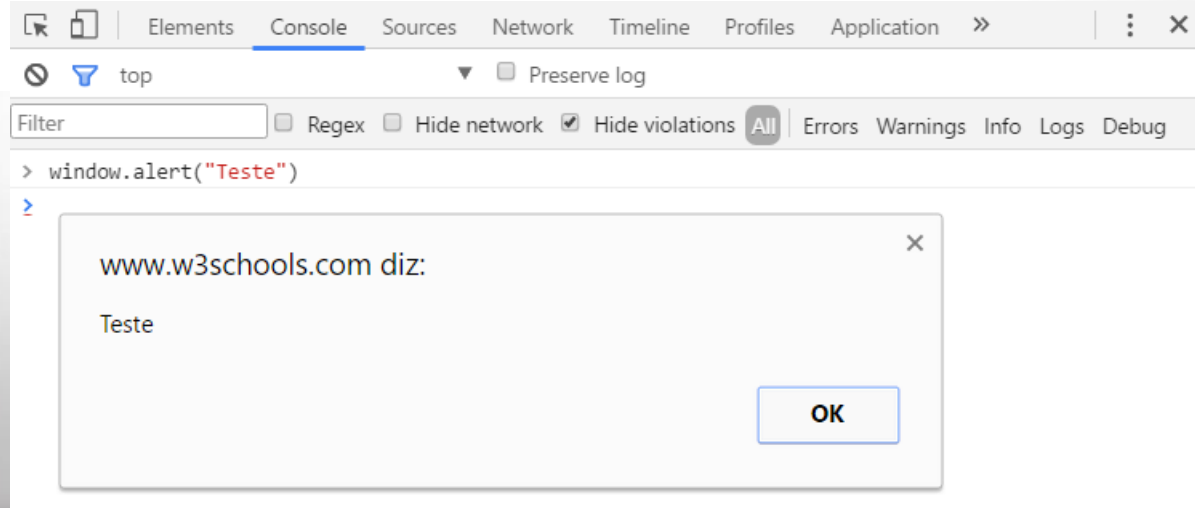
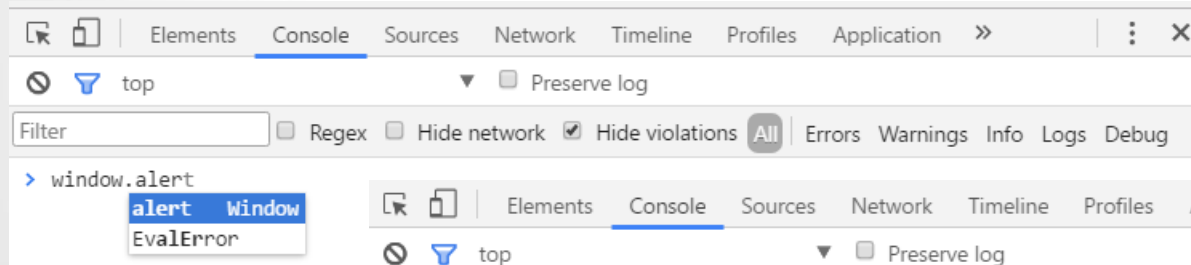
- O que é o Alert/Prompt/Confirm box? É uma caixa de diálogo que exibe notificações/mensagens na tela, para fornecer ao usuário algum tipo de informação, ou pedir permissão para executar determinado tipo de operação. Também pode ser usado para fins de advertência.
- Através do Selenium você pode confirmar ou cancelar uma mensagem de Alerta, ou até mesmo pegar o texto escrito nessas caixas de diálogo Web.

Metódo alert

Os métodos de caixa de diálogo pertence ao Window Objeto (*Window Object Methods*), ou seja não pertence ao body html. É por isso que no Selenium você precisa passar o foco para o alert. São geralmente chamado de *dialog box*. No javascript você não precisa referenciar o window, pode escrever direto os métodos, alert, prompt, confirm.

`window.alert("Error no Sistema."); alert("Error no Sistema.");`

>> Exibe um pop-up com a opção apenas de Ok, ele não tem um valor de retorno.

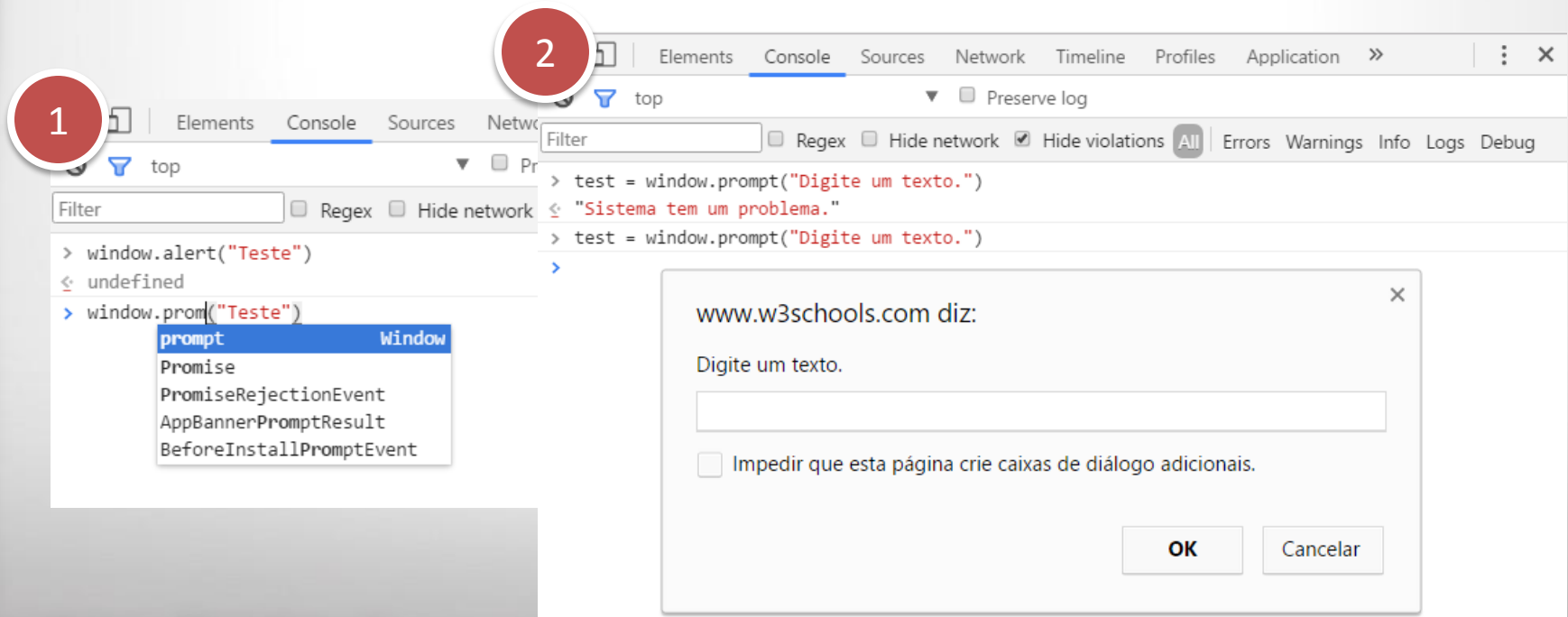


Metódo prompt

O método prompt pertence ao Window Objeto (Window Object Methods).

```
test = window.prompt("Digite um texto.")
```

>> No exemplo acima exibe um pop-up e habilita uma caixa de texto para a digitação, o retorno vai ser o próprio texto digitado.

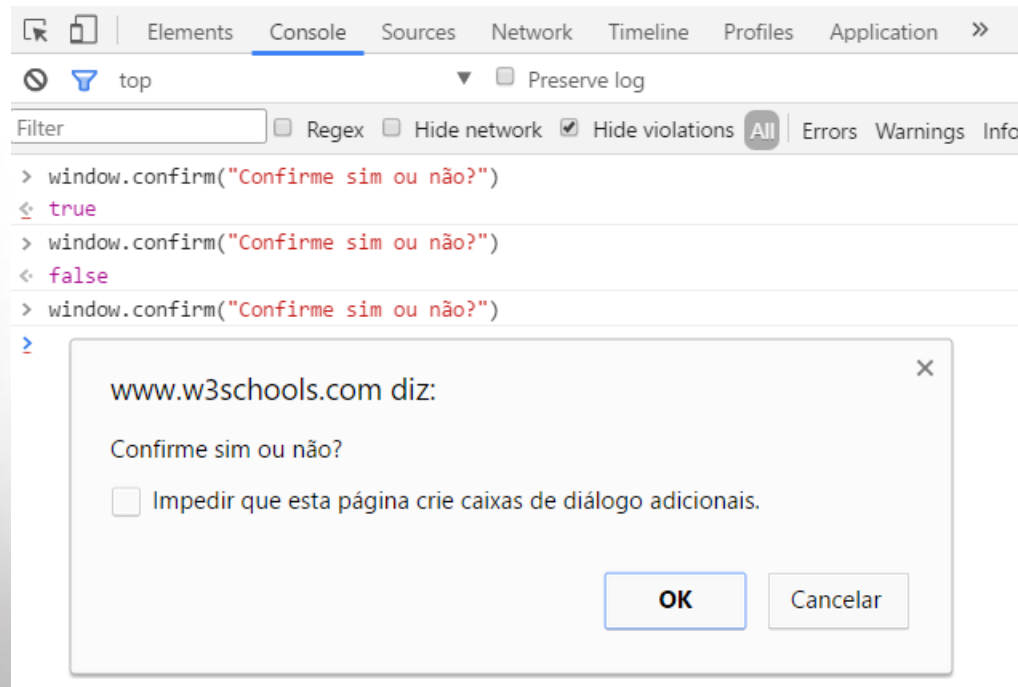


Metódo confirm

O método confirm pertence ao Window Objeto (Window Object Methods).

```
test = window.confirm("Confirme sim ou não?")
```

>> No exemplo acima exibe um pop-up com as opções Ok ou Cancelar, vai ter um retorno true se apertar no botão Ok, ou false se apertar no botão Cancelar.



Selenium Métodos

A classe alert permite a gente trabalhar com o alert/prompt/confirm box, ou seja interage com as caixas de diálogo web, temos várias opções, como confirmar, cancelar a caixa de dialogo ou até mesmo pegar e enviar um determinado texto.

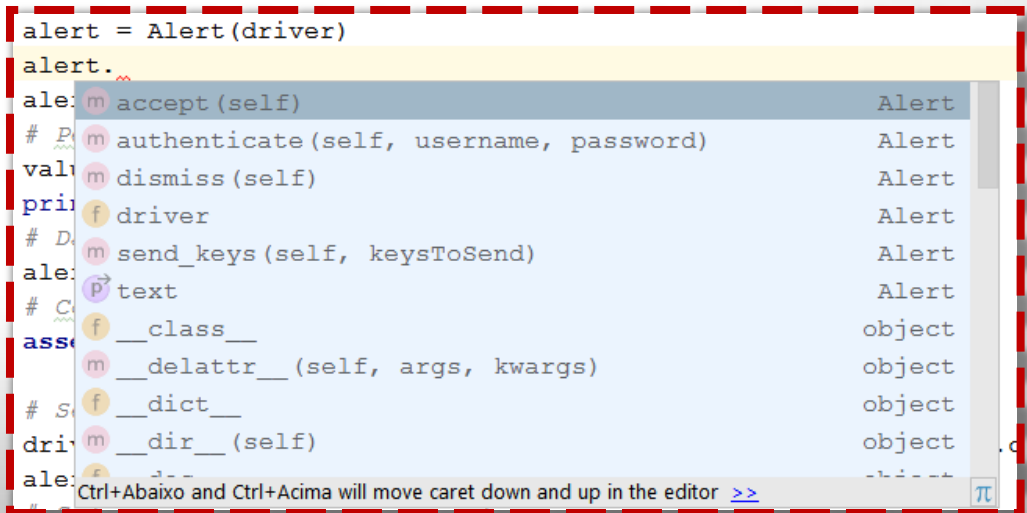
1. O primeiro passo devemos importar a classe alert:

from selenium.webdriver.common.alert **import** Alert

2. O segundo passo é instanciar a classe alert.

alert = Alert(driver)

3. Atráves da instância podemos usar os métodos, como mostra a imagem:

A screenshot of a code editor with a red dashed border. It displays the Selenium Alert class and its methods. The first line is 'alert = Alert(driver)'. Below it, 'alert.' is followed by a list of methods: 'accept(self)', 'authenticate(self, username, password)', 'dismiss(self)', 'driver', 'send_keys(self, keysToSend)', 'text', '__class__', '__delattr__(self, args, kwargs)', '__dict__', '__dir__(self)', and '__doc__'. To the right of each method is its return type: 'Alert', 'Alert', 'Alert', 'Alert', 'Alert', 'Alert', 'object', 'object', 'object', and 'object'. At the bottom, a status bar shows 'Ctrl+Abaixo and Ctrl+Acima will move caret down and up in the editor' and a '>>' button.

```
alert = Alert(driver)
alert.
ale: m accept(self) Alert
# D m authenticate(self, username, password) Alert
val m dismiss(self) Alert
pri f driver Alert
# D m send_keys(self, keysToSend) Alert
ale p text Alert
# C f __class__ object
asse m __delattr__(self, args, kwargs) object
# S f __dict__ object
dri m __dir__(self) object
ale: >>
```

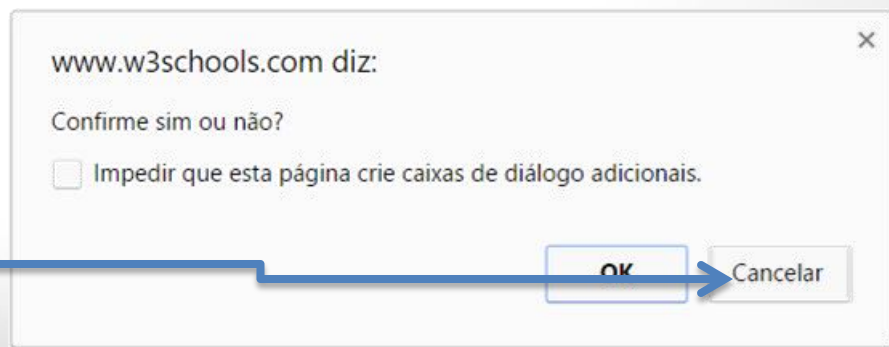
Selenium Métodos

Na classe Alert temos o método dismiss.

```
def dismiss(self):  
    """  
    Dismisses the alert available.  
    """  
    self.driver.execute(Command.DISMISS_ALERT)
```

1) `dismiss()` // O método `dismiss` vai fazer o clique no botão Cancelar/Cancel.

`alert.dismiss()`



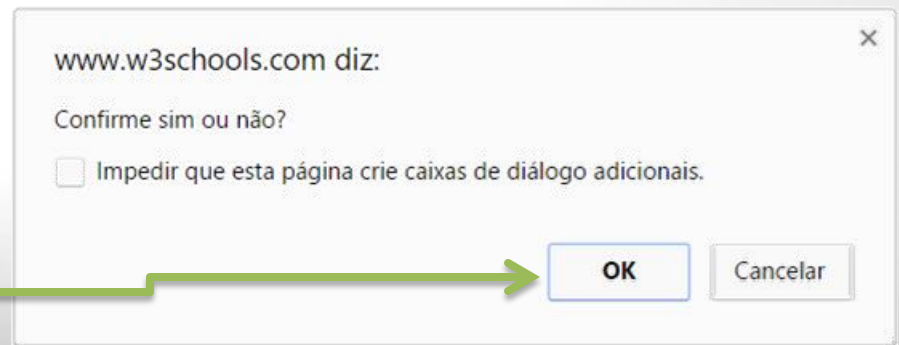
Selenium Métodos

Na classe Alert temos o método accept vai fazer o clique no botão Ok/Yes, para confirmar uma determinada ação.

```
def accept(self):  
    """  
    Accepts the alert available.  
  
    Usage::  
    Alert(driver).accept() # Confirm a alert dialog.  
    """  
    self.driver.execute(Command.ACCEPT_ALERT)
```

2) **accept() // O método accept.**

alert.accept()



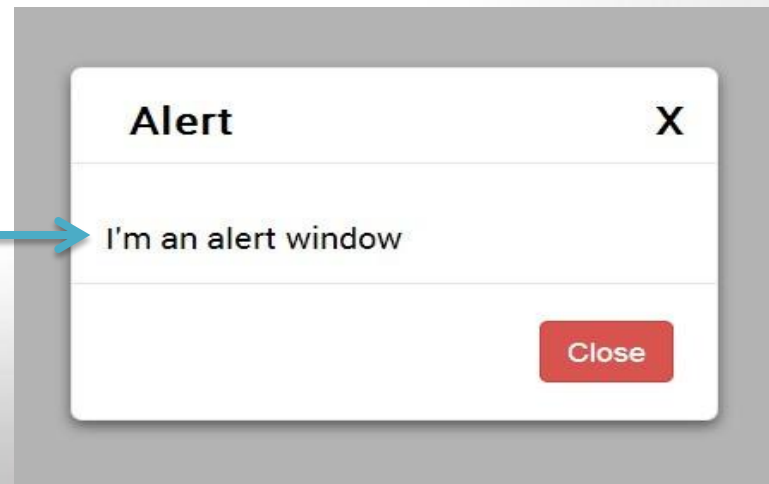
Selenium Métodos

Na classe Alert temos o método text.

```
@property
def text(self):
    """
    Gets the text of the Alert.
    """
    return self.driver.execute(Command.GET_ALERT_TEXT)["value"]
```

3) **text** // O método realiza a função de pegar o texto da caixa de alerta.

`Alert(driver).text`



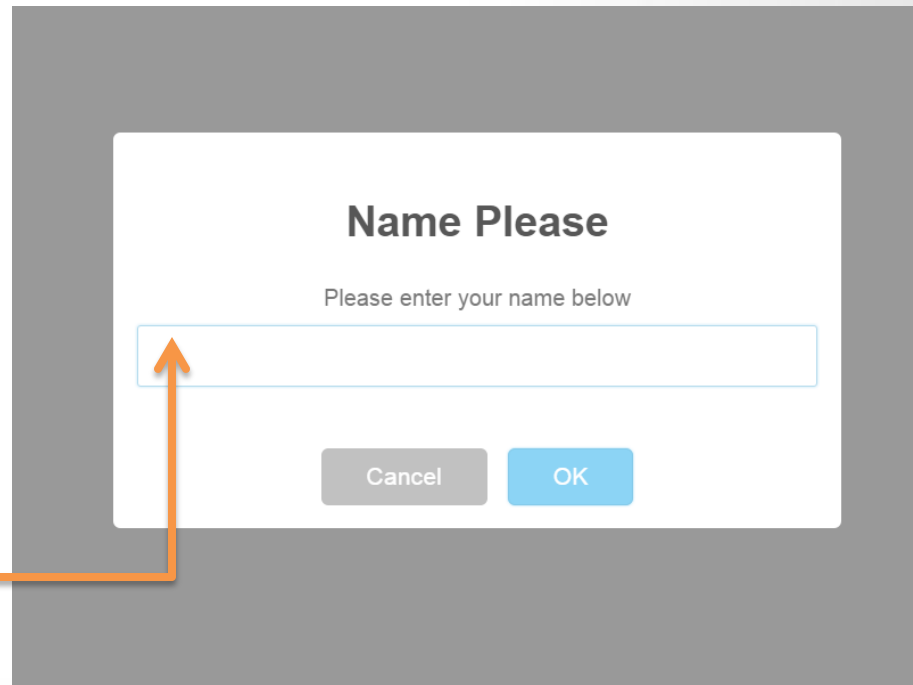
Selenium Métodos

Na classe Alert temos o método send_keys.

```
def send_keys(self, keysToSend):  
    """  
    Send Keys to the Alert.  
  
    :Args:  
    - keysToSend: The text to be sent to Alert.
```

4) **send_keys(String)** // Envia um texto para o alert box.

```
name_prompt = Alert(driver)  
name_prompt.send_keys("Reinaldo  
Mateus")  
name_prompt.accept()
```



Selenium Métodos

Na classe Alert temos o método authenticate.

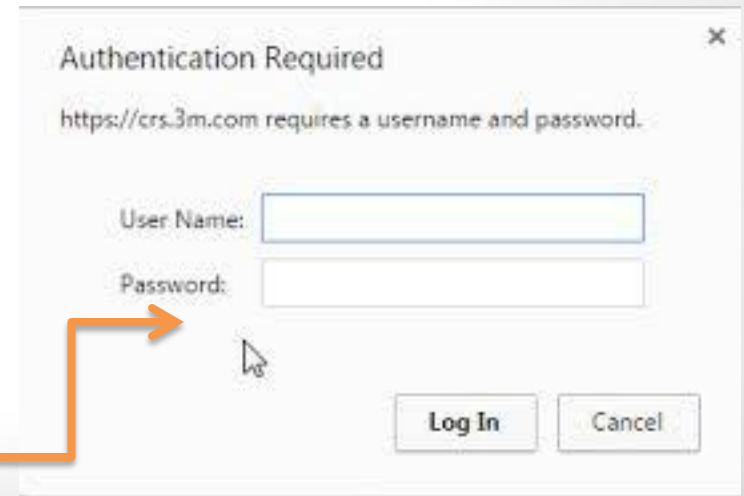
```
def authenticate(self, username, password):
```

```
    """
```

```
    Send the username / password to an Authenticated dialog (like with Basic HTTP Auth).  
    Implicitly 'clicks ok'
```

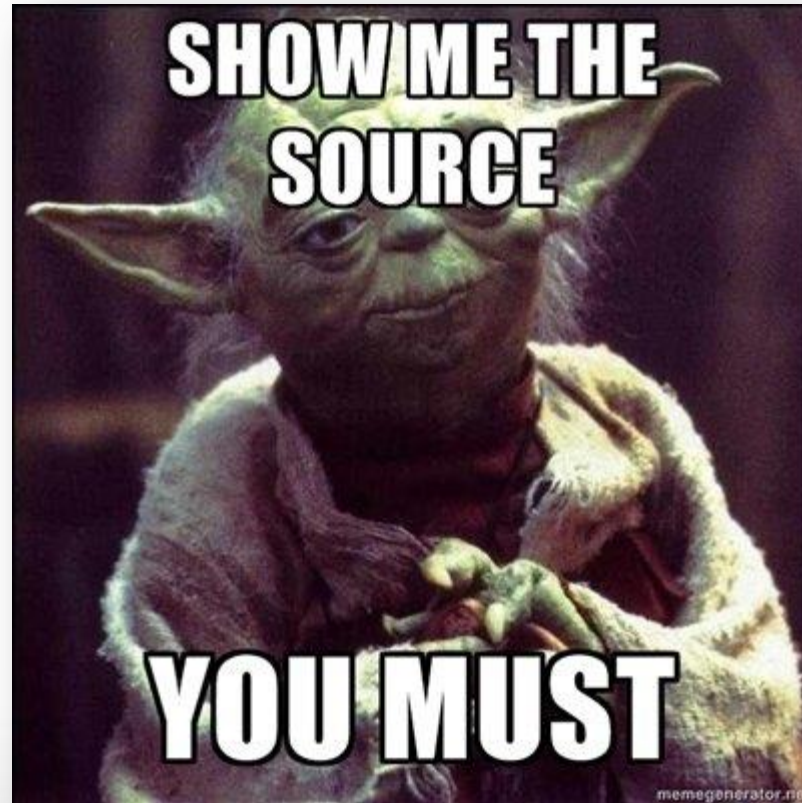
4) **authenticate**(user_name, password)

// Faz a autenticação em um pop-up.



```
driver.switch_to.alert.authenticate('reinaldo',  
'test123')  
Alert(driver).accept()
```

Exemplos



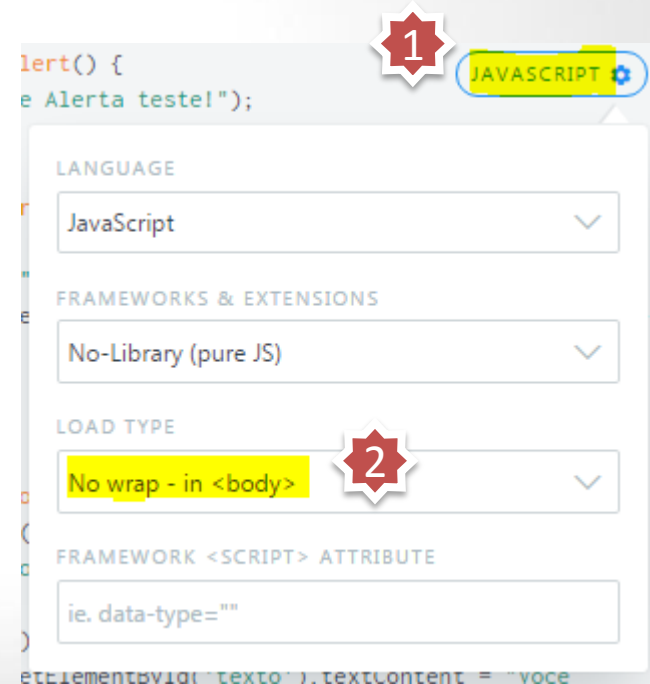
Prática 01



Vamos praticar um pouco, no link abaixo deixei o código html e javascript para realizar nossos testes com as caixa de diálogo.

Não esqueça de ir em JavaScript(1), e depois selecionar o campo load type (2) com “No wrap – in <body>” e aberte no botão “RUN”, no topo da tela.

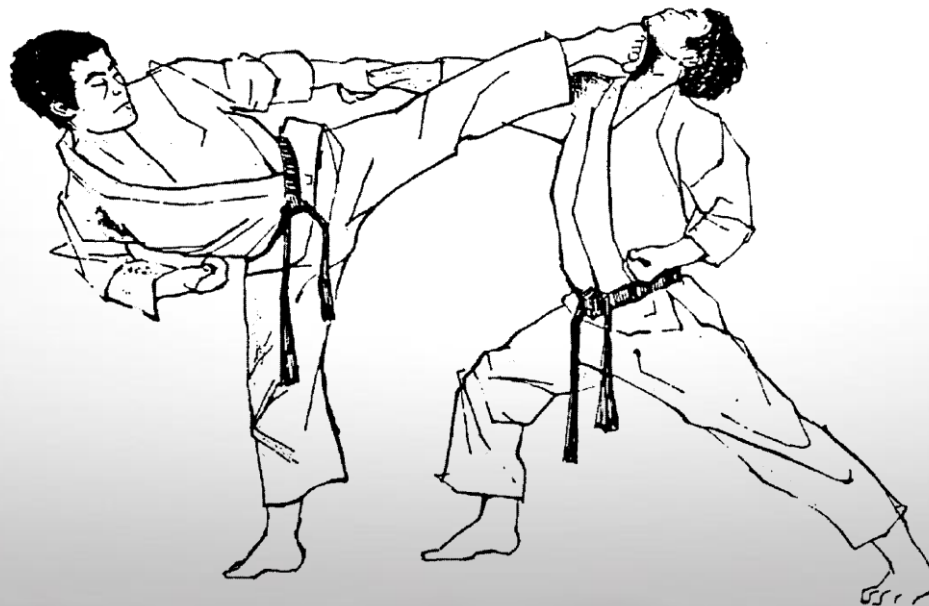
<https://jsfiddle.net/k9oo01sy/19/>



Dificuldades



- A maior dificuldade das pessoas é passar o foco atual para o foco no alert, além disso precisamos esperar a caixa de diálogo aparecer, a demora na espera vai gerar error.



Referências:



- [0] <https://developer.mozilla.org/pt-BR/docs/Web/API/Window/confirm>
- [1] https://www.w3schools.com/jsref/obj_window.asp
- [2] https://www.w3schools.com/jsref/met_win_alert.asp
- [3] https://www.w3schools.com/jsref/met_win_confirm.asp
- [4] https://www.w3schools.com/jsref/met_win_prompt.asp

Autor: Reinaldo M. R. Junior