

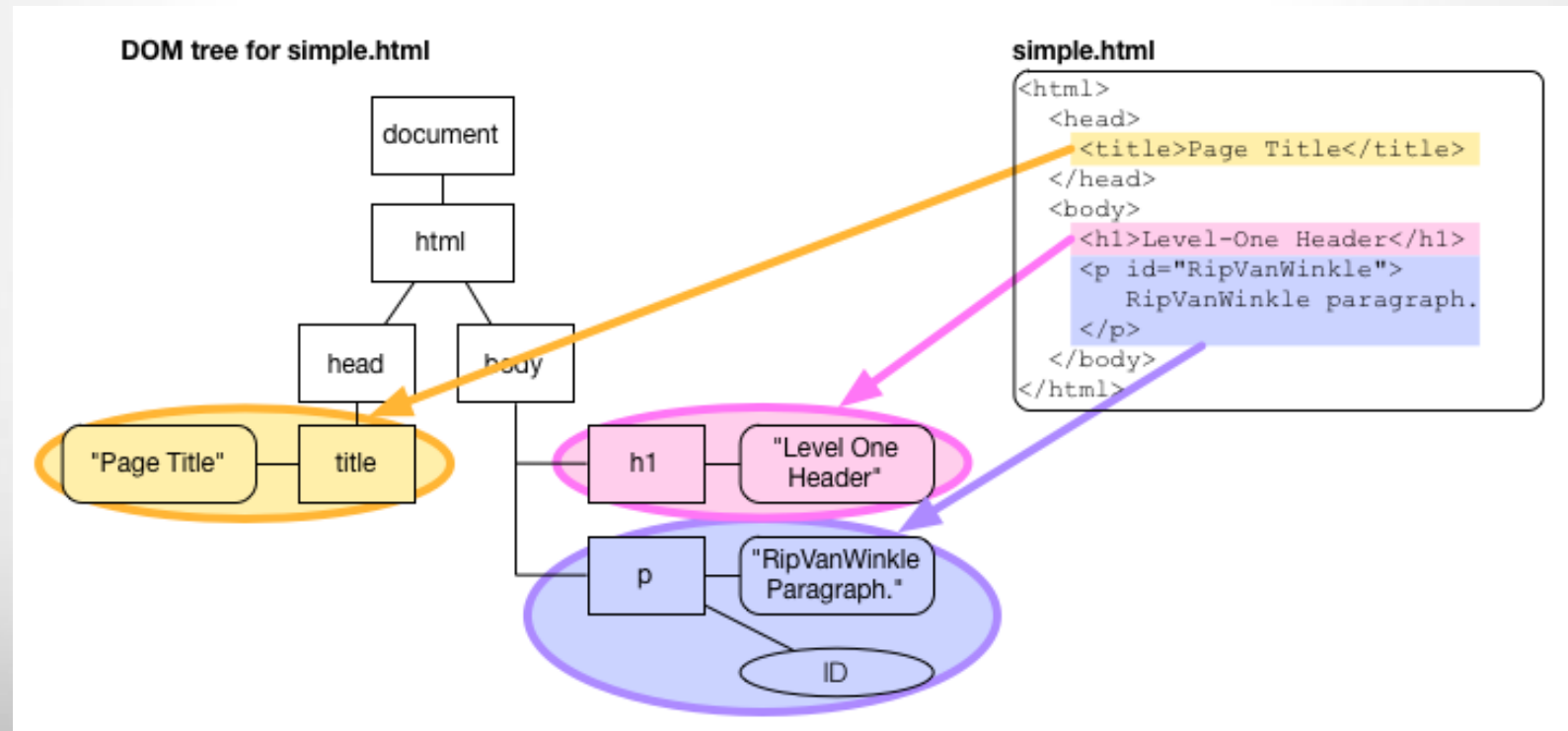


Selenium Básico

class WebDriver

Autor: Reinaldo M. R. Júnior

HTML Page:



Usando a classe **WebElement** e **WebDriver**.

A classe **WebElement** e **WebDriver** são as principais classes do Selenium, elas contêm as funções principais, você não sabe mais a classe **WebDriver** usa o **Remote WebDriver**, como vemos abaixo:

```
def __init__(self, command_executor='http://127.0.0.1:4444/wd/hub',  
             desired_capabilities=None, browser_profile=None, proxy=None,  
             keep_alive=False, file_detector=None):
```

Usando a classe WebDriver.

```
from selenium import webdriver
```

Quando importo a classe webdriver consigo usar todas as funções da classe, ela chama o `__ini__.py` do projeto, a primeira coisa que fazemos é instanciar o Browser.

```
from .firefox.webdriver import WebDriver as Firefox # noqa
from .firefox.firefox_profile import FirefoxProfile # noqa
from .chrome.webdriver import WebDriver as Chrome # noqa
from .chrome.options import Options as ChromeOptions # noqa
from .ie.webdriver import WebDriver as Ie # noqa
from .edge.webdriver import WebDriver as Edge # noqa
from .opera.webdriver import WebDriver as Opera # noqa
from .safari.webdriver import WebDriver as Safari # noqa
from .blackberry.webdriver import WebDriver as BlackBerry # noqa
from .phantomjs.webdriver import WebDriver as PhantomJS # noqa
from .android.webdriver import WebDriver as Android # noqa
from .remote.webdriver import WebDriver as Remote # noqa
from .common.desired_capabilities import DesiredCapabilities # noqa
from .common.action_chains import ActionChains # noqa
from .common.touch_actions import TouchActions # noqa
from .common.proxy import Proxy # noqa
__version__ = '3.0.1'
```

```
webdriver.|_
  ActionChains selenium.webdriver
  Android selenium.webdriver.android.webdr...
  BlackBerry selenium.webdriver.blackberry...
  Chrome selenium.webdriver.chrome.webdriv...
  ChromeOptions selenium.webdriver.chrome....
  DesiredCapabilities selenium.webdriver
  Edge selenium.webdriver.edge.webdriver
  Firefox selenium.webdriver.firefox.webdr...
  FirefoxProfile selenium.webdriver
  Ie selenium.webdriver.ie.webdriver
  Opera selenium.webdriver.opera.webdriver
Press Ctrl+Period to choose the selected (or first) suggestion and insert a dot afterwards >>>
```

Teste Básico – O que devemos fazer

1. Importar a biblioteca do Selenium;
2. Instanciar o Driver que vai usar;
3. Abrir a url/site;
4. Procurar o elemento que queremos manipular;
5. Aplicar a ação sobre o elemento;
6. Validar se a ação foi realizada.

Opções Básicas de Procura do Elemento.

- Funções de Pesquisa de Elemento:

- **find_element_by_id** ('id_name') # Procura um elemento pelo valor do atributo ID.

- Exemplo:

- ```
driver.find_element_by_id("pass")
```

- **find\_elements\_by\_id** # Procurando vários elementos pelo valor do atributo ID.

- Exemplo:

- ```
driver.find_elements_by_id("pass")
```

- **find_element_by_xpath** # Procura um elemento via Xpath.

- Exemplo:

- ```
driver.find_element_by_xpath ("//*[@id=email]")
```

- **find\_elements\_by\_xpath** # Procura vários elementos via Xpath.

- Exemplo:

- ```
driver.find_elements_by_xpath ("//*[@id=email]")
```

- **find_element_by_name** # Procura um elemento pelo atributo nome.

- Exemplo:

- ```
driver.find_element_by_name ("email")
```

- **find\_elements\_by\_name** # Procura vários elementos pelo atributo nome.

- Exemplo:

- ```
driver.find_elements_by_name ("email")
```

Opções Básicas de Procura do Elemento.

- Funções de Pesquisa de Elemento no Selenium:
 - **find_element_by_css_selector** # Procura um elemento via CSS Selector.
 - Exemplo:
driver. **find_element_by_css_selector** ("#pass")
 - **find_elements_by_css_selector** # Procura vários elementos via CSS selector.
 - Exemplo:
driver. **find_element****s_by_css_selector** ("#pass")

O # (Cerquilha) representa o atributo ID do elemento Web. Vamos ter uma aula somente sobre CSS Selector.

A diferença entre as duas funções é somente o “s” no final de element => elements.

Atenção!

- Quando procuramos vários elementos ele traz um **lista** de elementos, pra você acessar um elemento na lista, precisa indicar qual elemento quer acessar.
- Uma lista em python é representada pelo comando abaixo, os valores devem estar entre os **Colchetes** ([]):

```
lista = [1, 2, 3]
```

- Exemplo:

```
driver.find_elements_by_id("pass")[0]
```

- [0] - > Indica que eu quero acessar o primeiro elemento da lista, sempre começa por Zero, o segundo elemento seria o [1] .

Opções Básicas.

- Mais Opções:

- **find_element_by_partial_link_text** # Finds an element by a partial match of its link text.
- **find_elements_by_partial_link_text** # Finds elements by a partial match of their link text.
- **find_element_by_link_text** # Finds an element by link text.
- **find_elements_by_link_text** # Finds elements by link text.
- **find_element_by_class_name** # Finds an element by class name.
- **find_elements_by_class_name** # Finds elements by class name.
- **find_element_by_css_selector** # Finds an element by css selector.
- **find_elements_by_css_selector** # Finds elements by css selector.
- **find_element_by_tag_name** # Finds an element by tag name.
- **find_elements_by_tag_name** # Finds elements by tag name.
- **current_url** # Gets the URL of the current page.
- **page_source** # Gets the source of the current page.
- **get** # Loads a web page in the current browser session.
- **title** # Returns the title of the current page.
- **set_page_load_timeout** # Set the amount of time to wait for a page load to complete before throwing an error.
- **get_attribute** – Pega o valor do atributo de um determinado elemento.

Praticando um pouco!

```
driver.set_page_load_timeout(10)
driver.find_element_by_id()
driver.find_element_by_class_name()
driver.find_element_by_css_selector()
driver.find_element_by_xpath()
driver.find_element_by_name()
driver.find_element_by_link_text()
driver.find_element_by_partial_link_text()
print(driver.current_url)
print(driver.page_source)
print(driver.title)
```

<https://css-tricks.com/attribute-selectors/>

[title=one][rel ^=external]

Multiple Attribute Matches

Vital to note is that you can use **multiple attribute selectors** in the same selector, which requires all of them to match for the selector itself to match.

HTML

```
<h1 rel="handsome" title="Important note">Multiple Attributes</h1>
```

CSS

```
h1[rel="handsome"][title^="Important"] { color: red; }
```

Exemplo usando Dois Atributos via CSS Selector.

```
driver.find_element_by_css_selector ("input[name=email][id=email]")
```

*** Nesse exemplo usamos os dois atributos via ID e Name da tag input.**

Estratégia – Duas Tag/Atributo

Melhor ordem:

- ID;
- Class;
- CSS Selector;
- Xpath.

Pior Caso:

- Relacionar com Texto e Link.

Locator Strategies

- Class
- *CSS selectors*
- ID
- Link Text
- Partial Link Text
- Tag Name
- *XPath*

Good locators are:

- unique
- descriptive
- unlikely to change

That rules a few of these out

Start with IDs and Classes

Use CSS or XPath (with care)

Pratique no Facebook a seguinte Estratégia

Primeiro via ID e depois via CSS Selector, não esqueça de seguir os passos básicos de um teste.

```
driver.find_element_by_id("pass").clear() # limpa o campo
driver.find_element_by_id("email").send_keys("teste") # envia o texto
driver.find_element_by_id("pass").clear() # limpa o campo
driver.find_element_by_id("pass").send_keys("teste") # envia o texto
driver.find_element_by_css_selector("label#loginbutton
input#u_0_l").click() # pesquisa e dar o click
```

URL: <https://www.facebook.com/>

Dificuldades

- Atributos iguais podem atrapalhar os testes, você tem que pegar tag e atributos únicos, aqueles que são difíceis de mudar. Podemos verificar pelo firepath se existe mais de uma referência na página para aquele atributo ou tag.



Tarefa pra Casa

- Na página do facebook faça o cadastramento de uma nova conta.
- Qualquer problema deixe um comentário, que irei responder.

Referências:

- [0] <http://selenium-python.readthedocs.io/api.html>
- [1] http://www.w3schools.com/tags/att_a_rel.asp
- [2] http://www.w3schools.com/cssref/css_selectors.asp
- [3] [\[Webinar\] Mastering Test Automation: How To Use Selenium Successfully](#)