

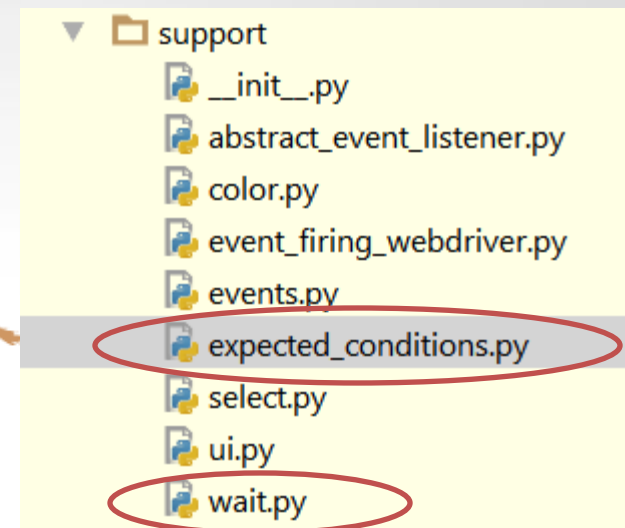
Selenium Básico

Esperas Explícitas e Implícitas.



Autor: Reinaldo M. R. Junior

Selenium WebDriver



No Selenium temos dois tipos de esperas implícita e explícita, a função ***implicitly_wait(time_to_wait)***, está dentro da classe WebDriver, podendo ser chamada direta.

A função Explícita está dentro do pacote de support em ***expected_conditions.py*** e ***wait.py***, você usa a classe ***WebDriverWait*** e a função ***until*** ou ***until_not*** passando a condição que está em ***expected_conditions***.

Implícita

Uma espera implícita é dizer WebDriver para consultar o DOM por um determinado período de tempo, enquanto todos os elementos não estão carregados. A configuração padrão é 0.

Depois de definir, a espera implícita está definido o tempo máximo de vida da instância do objeto WebDriver.

Explícita

A espera explícita diz ao WebDriver esperar uma determinada condição e um tempo máximo, para que ele aguarde antes de prosseguir com o teste, encontrado o elemento/condição antes do tempo máximo ele simplesmente segue o teste. O que torna esse tipo de espera a melhor opção.

Exemplos

```
from selenium.webdriver.support.ui import WebDriverWait  
from selenium.webdriver.support import expected_conditions  
as EC
```

```
element = WebDriverWait(driver, 10).until(lambda x:  
    x.find_element_by_id("someId"))
```

```
is_disappeared = WebDriverWait(driver, 30, 1, (ElementNotVisibleException)).\  
    until_not(lambda x: x.find_element_by_id("someId").is_displayed())
```

```
element = WebDriverWait(driver, timeout, poll_frequency=0.1).until(  
    EC.element_to_be_clickable((By.XPATH, elem)))
```

FluentWait

Em aplicações que usam muito AJAX, com campos muito dinâmicos, o WebDriver pode acabar "se perdendo" e lançando exceções como ***NoSuchElementException*** ou ***StaleElementReferenceException***.

O grande problema é que esses erros podem ser intermitentes, gerando os famosos falsos negativos.

Testes com essa característica são chamados de flaky tests.

FluentWait - ignored_exceptions

```
def fluentwait(driver, elem, timeout=60):
```

```
    # self.FindFrame(driver)
```

```
    try:
```

```
        element = WebDriverWait(driver, timeout, poll_frequency=1,  
ignored_exceptions=[ElementNotVisibleException,
```

```
ElementNotSelectableException]).until(EC.element_to_be_clickable((By.XPATH, elem)))
```

```
        return element
```

```
    except NoSuchElementException:
```

```
        raise Exception("Except Element not found!")
```

Resumindo

- Na implícita não é claro o elemento que ele esta esperando, na Explicita é bem claro que condição e elemento ele esta esperando:
`expected_conditions.element_to_be_clickable((By.XPATH, ".//*[@id='lst-ib']"))`
- Não é recomendado pelos especialistas as esperas fixas da própria linguagem, por poder acrescentar lentidão ao projeto.

Referências:

- [0] http://docs.seleniumhq.org/docs/04_webdriver_advanced.jsp
- [1] http://www.w3schools.com/js/js_htmlDOM.asp
- [2] <http://stackoverflow.com/questions/22656615/what-is-difference-between-implicit-wait-vs-explicit-wait-in-selenium-webdriver>
- [3] <http://stefanteixeira.com.br/2014/04/29/entendendo-os-tipos-de-esperas-no-selenium-webdriver/>