
ДОМАШНЯЯ РАБОТА №4

Задание

На уроке мы разворачивали и собирали систему для автоматизированного тестирования на python, а также смотрели тестовый вариант сценария для тестов (собран по сайту <https://habr.com/ru/post/262929/>). Тестовый сценарий не проходил.

В качестве домашнего задания нужно обновить тестовый сценарий и сделать так, чтобы он проходил.

Напомню условия задачи:

Задача:

1. Исправить тестовый сценарий таким образом, чтобы он проходил.

Ограничения:

1. Изменения в сценарии должны быть минимальными.

Подсказки, найденные на паре:

1. Сценарий не проходит, потому как на странице нет кнопки "Найти" на которую ему нужно нажать.
2. В качестве решения задачи можно выбрать другой поисковик для теста.
3. В качестве решения задачи, можно попытаться что-нибудь ввести в строку поиска, чтобы кнопка поиск появилась.

Что будет оцениваться:

1. Работа тестового сценария
2. Именование переменных
3. Наличие функций и их именование.
4. Структура проекта (насколько легко разобраться в том что и как работает)

Мне удалось добиться успешного прохождения всех трёх шагов теста. Для этого пришлось редактировать аргументы вызываемых функций, поскольку в примере был код для старой версии selenium ([ссылка](#)). Пример на скриншоте ниже.

Old API	New API
<code>find_element_by_id('id')</code>	<code>find_element(By.ID, 'id')</code>
<code>find_element_by_name('name')</code>	<code>find_element(By.NAME, 'name')</code>
<code>find_element_by_xpath('xpath')</code>	<code>find_element(By.XPATH, 'xpath')</code>

Скрипт открывает Яндекс в Firefox, находит поле ввода запроса, вводит «пробел», имитирует нажатие «ENTER», затем проверяет что открылась страница «Задан пустой поисковый запрос» и выводит отчёт.

```

1. from behave import *
2. from selenium import webdriver # необходим в этом скрипте, отсутствовал в примере
3. from selenium.webdriver.common.keys import Keys
4. from selenium.webdriver.common.by import By # необходим для новой версии Selenium
5. from selenium.webdriver.support.ui import WebDriverWait
6. from selenium.webdriver.support import expected_conditions as EC
7. import time
8.
9. #Откроем главную страницу. Передадим в качестве аргумента адрес страницы.
10. @given('website "{url}"')
11. def step(context, url):
12.     #Измените строку, для выполнения теста в другом браузере
13.     context.browser = webdriver.Firefox()
14.     context.browser.maximize_window()
15.     context.browser.get("http://ya.ru")
16.
17.     #Теперь нажмем на кнопку "Найти"
18. @then("push button with text '{text}'")
19. def step(context, text):
20.     WebDriverWait(context.browser, 120).until(
21.         EC.element_to_be_clickable((By.XPATH, '//button'))
22.     )
23.     context.browser.find_element(By.NAME, 'text').send_keys(' ')
24.     context.browser.find_element(By.NAME, 'text').send_keys(Keys.ENTER)
25.
26.
27.     #Проверим, что мы на странице с результатами поиска, есть некоторый искомый
    текст
28. @then("page include text '{text}'")
29. def step(context, text):
30.     WebDriverWait(context.browser, 120).until(
31.         EC.presence_of_element_located((By.XPATH, '//*[contains(text(),
    "%s")] ' % text))
32.     )
33.     assert context.browser.find_element(By.XPATH, '//*[contains(text(),
    "%s")] ' % text)
34.
35.     context.browser.quit()

```

```
first_test.py - C:\selenium\features\steps\first_test.py (3.11.2)
File Edit Format Run Options Window Help

from behave import *
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
import time

#Откроем главную страницу. Передадим в качестве аргумента адрес страницы.
@given('website "{url}"')
def step(context, url):
    #Измените строку, для выполнения теста в другом браузере
    context.browser = webdriver.Firefox()
    context.browser.maximize_window()
    context.browser.get("http://ya.ru")

    #Теперь нажмем на кнопку "Найти"
    @then("push button with text '{text}'")
    def step(context, text):
        #WebDriverWait(context.browser, 120).until(
        #    EC.element_to_be_clickable((By.XPATH, '//button'))
        #)
        context.browser.find_element(By.NAME, 'text').send_keys(' ')
        context.browser.find_element(By.NAME, 'text').send_keys(Keys.ENTER)

    #Проверим, что мы на странице с результатами поиска, есть некоторый искомый текст
    @then("page include text '{text}'")
    def step(context, text):
        WebDriverWait(context.browser, 120).until(
            EC.presence_of_element_located((By.XPATH, '//*[contains(text(), "%s")] ' % text))
        )
        assert context.browser.find_element(By.XPATH, '//*[contains(text(), "%s")] ' % text)
        context.browser.quit()
```

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.2604]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\selenium\features>behave -i 1.feature
Feature: Checking search # 1.feature:2

    Scenario: Check some text in search results # 1.feature:4
        Given website "ya.ru" # steps/first_test.py:10
        Then push button with text 'Найти' # steps/first_test.py:18
        Then page include text 'Задан пустой поисковый запрос' # steps/first_test.py:28

1 feature passed, 0 failed, 0 skipped
1 scenario passed, 0 failed, 0 skipped
3 steps passed, 0 failed, 0 skipped, 0 undefined
Took 0m10.584s

C:\selenium\features>
```