



Меня хорошо видно && слышно?

Ставьте  , если все хорошо
Напишите в чат, если есть проблемы

Правила вебинара



Активно участвуем



Задаем вопрос в чат или голосом



Off-topic обсуждаем в Slack #канал группы или #general



Вопросы вижу в чате, могу ответить не сразу



ОНЛАЙН-ОБРАЗОВАНИЕ

Онлайн-образование

Проверить, идет ли запись!



Архитектура тестов. Page object



Вяземский Семён Валерьевич

QA Automation Lead

Beeline

svvyazemsky@gmail.com

Маршрут вебинара

Page Object в действии



Архитектура



Фреймворк



Тестовая библиотека

Цели вебинара | После занятия вы сможете

1 Использовать Page Object паттерн

2 Разделять бизнес-логику и остальные методы

3 Узнаете разницу между фреймворком и библиотекой

The background of the slide is a high-angle, blue-tinted aerial photograph of a dense urban skyline, likely New York City. Overlaid on this image is a semi-transparent blue band that contains a white network diagram. This diagram consists of numerous small dots connected by thin white lines, creating a web-like structure that spans the width of the slide. Centered within this band is the text "Page Object" in a large, white, sans-serif font.

Page Object

Спагетти

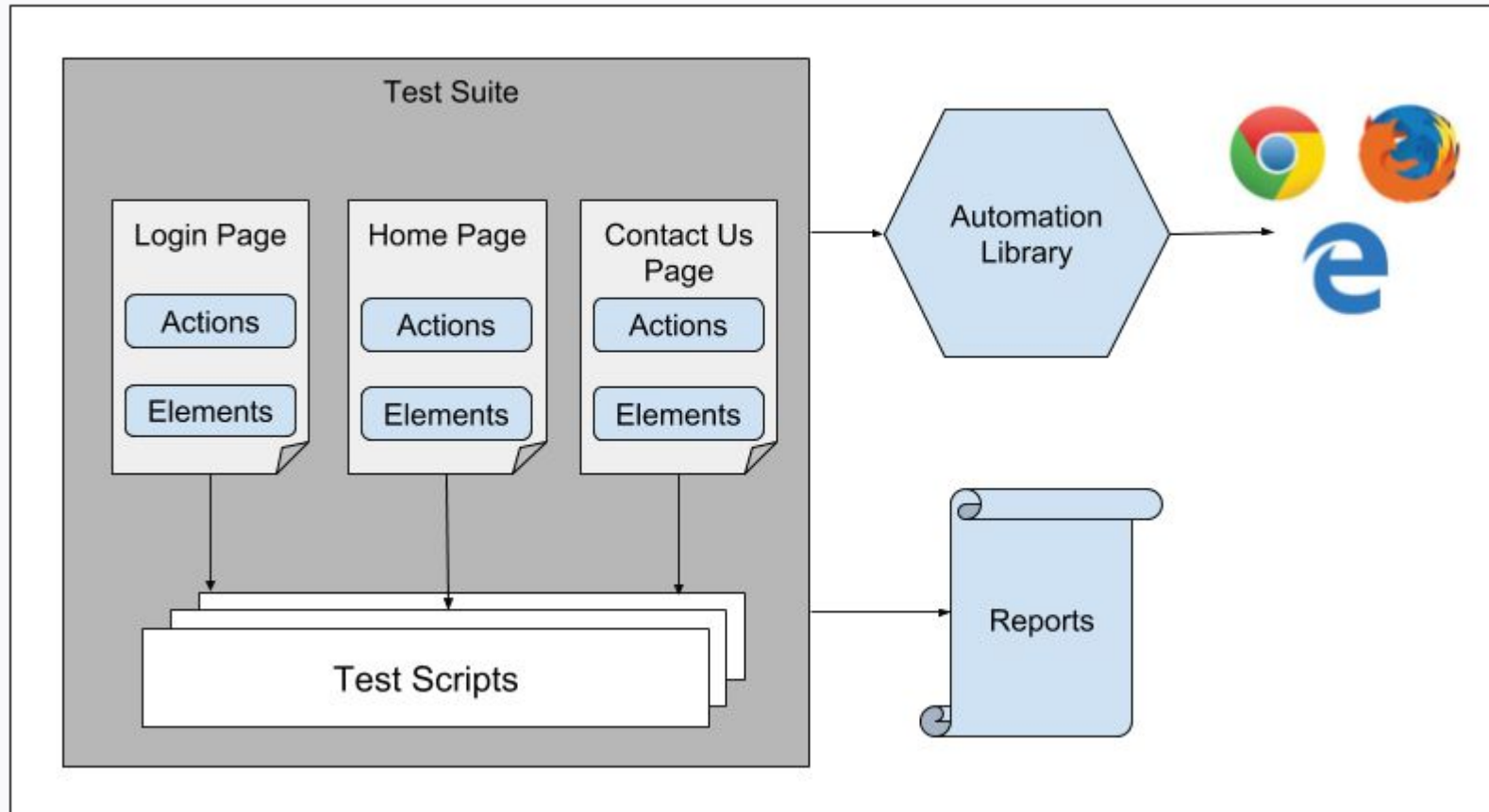
```
from selenium.webdriver import Chrome
from selenium.webdriver.common.by import By

def test_google_search():
    # Открыть браузер
    driver = Chrome()
    # Переход на страницу
    driver.get('https://google.com')
    search_input = driver.find_element_by_name('q')
    # Ввод текста
    search_input.send_keys('Selenium')
    search_input.submit()

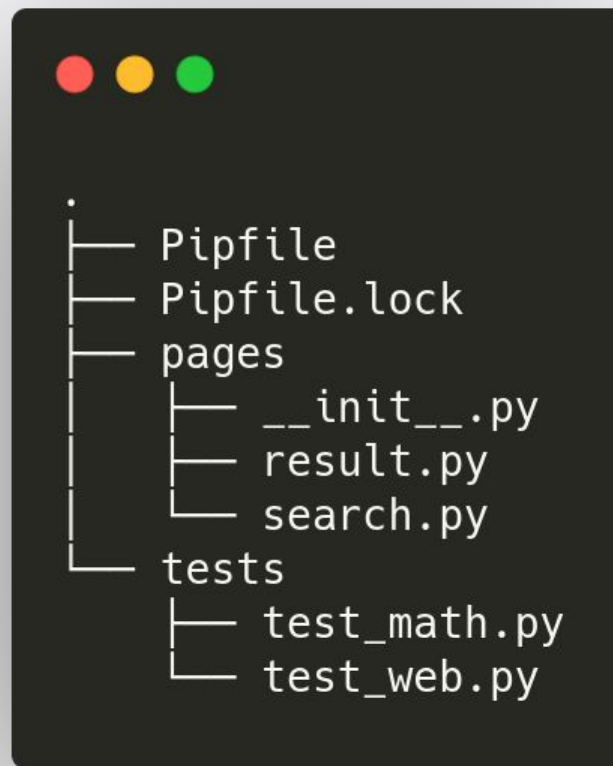
    # Проверка результатов поиска
    search_result = driver.find_elements(
        by=By.CSS_SELECTOR,
        value='#rso .srg div.g'
    )
    assert len(search_result) > 0
    for result in search_result:
        title_text = result.find_element_by_tag_name('h3').text
        assert 'Selenium' in title_text
```


Page object. Зачем?

- Убрать дублирование кода
- Улучшить визуально тесты
- Упростить написание новых тест-кейсов



Page object. Разделяй и властвуй



Page object. Base page

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

class BasePage(object):

    def __init__(self, driver):
        self.driver = driver
        self.base_url = "https://demo.opencart.com/admin/"

    def find_element(self, locator, time=10):
        return WebDriverWait(self.driver, time).until(EC.presence_of_element_located(locator),
                                                    message=f"Can't find element by locator {locator}")

    def find_elements(self, locator, time=10):
        return WebDriverWait(self.driver, time).until(EC.presence_of_all_elements_located(locator),
                                                    message=f"Can't find elements by locator {locator}")

    def go_to_site(self):
        return self.driver.get(self.base_url)
```

Page object. Pages

```
from selenium.webdriver.common.by import By
from .base import BasePage

class AdminLoginPage(BasePage):

    username = (By.ID, 'input-username')
    password = (By.ID, 'input-password')
    submit_button = (By.CSS_SELECTOR, 'button(type="submit")')

    def __init__(self, driver):
        super().__init__(driver)
        self.driver = driver

    def _set_username_(self, name):
        self.find_element(locator=self.username[1]).send_keys(name)

    def _set_password_(self, password):
        self.find_element(locator=self.password[1]).send_keys(password)

    def login(self):
        self._set_username_('demo')
        self._set_password_('demo')
        self.find_element(locator=self.submit_button).click()
```


Page object. Tests



```
import pytest
```

```
def test_login(admin_page):  
    admin_page.login('demo', 'demo')  
    assert admin_page.driver.title == 'Dashboard'
```

Page object



The image features a blue-toned aerial photograph of a dense urban skyline, likely New York City, with numerous skyscrapers. A semi-transparent blue band with a white geometric network pattern of dots and lines runs horizontally across the middle of the image. The word "Архитектура" is written in white, bold, sans-serif font within this band.

Архитектура

Архитектура. Что это?

Архитектура программного обеспечения — совокупность важнейших решений об организации программной системы.

Архитектура включает: выбор структурных элементов и их интерфейсов, с помощью которых составлена система, а также их поведения в рамках сотрудничества структурных элементов;

Задача Архитектуры - строить среду

The background of the slide is a blue-tinted aerial photograph of a dense city skyline, likely New York City. Overlaid on this is a semi-transparent network pattern consisting of numerous small dots connected by thin lines, creating a web-like structure. The word "Фреймворк" is centered in the middle of the slide in a large, white, sans-serif font.

Фреймворк

Фреймворки

- **Фреймворк** - это набор тулов и библиотек
- Фреймворк запускает тесты, задаёт единый формат тестов, репортит результаты, удобно определяет тестовые данные (константы, датасетовы и т.д.) и объединяет библиотеки специфичные для тестирования конкретного продукта.
- Тестовый фреймворк увеличивает скорость имплементация тестов, увеличивает переиспользование кода, упрощает поддержку тестов, делает тесты более читаемыми, унифицирует репортинг и упрощает запуск тестов на различных окружениях.

Слайд с домашним заданием

1 *Научиться реализовывать паттерн PageObject на практике.*

В имеющемся проекте автоматизации приложения OpenCart на данный момент имеются описания селекторов страниц и небольшой пул автотестов.

Необходимо перевести код на паттерн PageObject. Добавить 5 новых тестов написанных в этой парадигме.

Выбор функциональности или сценариев остается на ваше усмотрение.

Рефлексия



Отметьте 3 пункта, которые вам запомнились с вебинара



Что вы будете применять в работе из сегодняшнего вебинара?

Следующий вебинар

Тема: Работа с окнами



22.05.2020



Ссылка на вебинар будет в ЛК за 15 минут




Материалы к занятию
в ЛК — можно изучать



Обязательный
материал обозначен
красной лентой

Список материалов для изучения

<https://www.infoq.com/articles/layered-test-automatation/>

The background of the image is an aerial view of a city skyline, likely New York City, with numerous skyscrapers. The image is overlaid with a semi-transparent blue layer that features a white network pattern of dots and lines. The text is centered within this blue layer.

Заполните, пожалуйста,
опрос о занятии по ссылке в чате
<https://otus.ru/polls/10231/>

Спасибо за внимание!
Приходите на следующие вебинары



Вяземский Семён Валерьевич

QA Automation Lead

Beeline

svvyazemsky@gmail.com