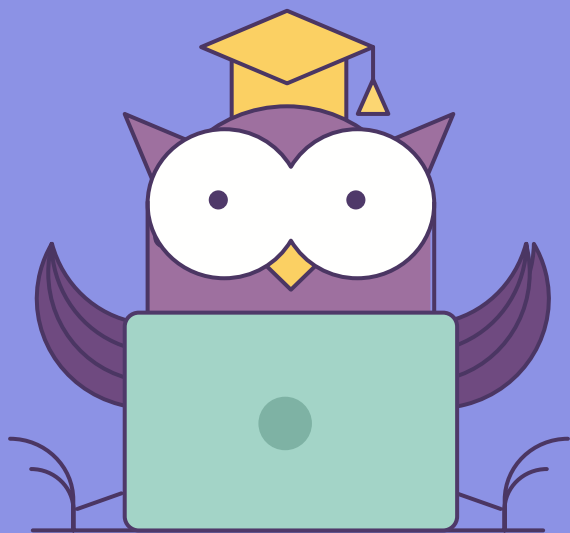




ОНЛАЙН-ОБРАЗОВАНИЕ

Меня хорошо слышно && видно?



Напишите в чат, если есть проблемы!

Ставьте  если все хорошо

Не забыть включить запись!



Михаил Чирков

QA Automation Engineer в ivi.ru (Web, Mobile)

Поиск элементов

Python QA Engineer



Правила вебинара



Буду делать паузы между блоками для просмотра вопросов. **Если что кричите!**



Чтобы выделить свой вопрос в чате пишем так “Вопрос: <текст вопроса>”



Off-topic или то что не успеем обсуждаем в Slack #канал группы или #general

Цели вебинара

- Основы ООП в Python
- Фронтенда для QA (DOM, Selectors, DevTools)
- Локаторы в Selenium
- Скандалы, интриги, расследования
- Какие селекторы лучше использовать
- Примеры спец. селекторов (ozon, ivi, avito)

00

ООП в python

+ обработка исключений

Демонстрация

https://github.com/konflic/pyhton_qa_element_search



Ссылки

- https://www.w3schools.com/python/python_classes.asp
- https://www.w3schools.com/python/python_inheritance.asp
- <https://realpython.com/python3-object-oriented-programming/>
- <https://docs.python.org/3/tutorial/errors.html>

Вопросы



01

Фронтенд для QA

Демонстрация

- DOM (Document Object Model)
- CSS Selectors
- DevTools
 - browser assistance selectors
 - document
 - querySelector
 - \$, \$x

Вопросы



02

Локаторы Selenium

- `find_element_by_id`
- `find_element_by_name`
- `find_element_by_xpath`
- `find_element_by_link_text`
- `find_element_by_partial_link_text`
- `find_element_by_tag_name`
- `find_element_by_class_name`
- `find_element_by_css_selector`

- `find_element_by_id` -> **id**
- `find_element_by_name` -> **name**
- `find_element_by_xpath` -> **xpath**
- `find_element_by_link_text` -> **Текст ссылки**
- `find_element_by_partial_link_text` -> **Часть текста ссылки**
- `find_element_by_tag_name` -> **Тэг**
- `find_element_by_class_name` -> **Значение атрибута class**
- `find_element_by_css_selector` -> **CSS селектор**

```
<form id="loginForm">...</form>
```

driver. ???

```
<form id="loginForm">...</form>
```

```
driver.find_element_by_id("loginForm")
```

```
<form id="loginForm">...</form>
```

```
driver.find_element_by_id("loginForm")
```

```
driver.find_element_by_tag_name("form") ???
```

- ~~find_element_by_id~~
- find_element~~s~~_by_name
- find_element~~s~~_by_xpath
- find_element~~s~~_by_link_text
- find_element~~s~~_by_partial_link_text
- find_element~~s~~_by_tag_name
- find_element~~s~~_by_class_name
- find_element~~s~~_by_css_selector

```
<div id="catalog">  
  <div class="product">...</div>  
  <div class="product">..</div>  
  ...  
</div>
```

```
driver.find_elements_by_class_name("product")
```

- Методы поиска единичных элементов если элемент не найден, бросают исключения **NoSuchElementException**.
- Методы поиска множественных элементов исключений не бросают, а просто возвращают пустой массив.

Демонстрация



Вопросы



03

**Скандалы, интриги,
расследования**

- `find_element(self, by=By.ID, value=None)`
- `find_elements(self, by=By.ID, value=None)`

```
class By(object):  
    """  
    Set of supported locator strategies.  
    """  
  
    ID = "id"  
    XPATH = "xpath"  
    LINK_TEXT = "link text"  
    PARTIAL_LINK_TEXT = "partial link text"  
    NAME = "name"  
    TAG_NAME = "tag name"  
    CLASS_NAME = "class name"  
    CSS_SELECTOR = "css selector"
```

```
<form id="loginForm">...</form>
```

driver. ???

```
<form id="loginForm">...</form>
```

```
driver.findElement(By.ID, "loginForm")
```

```
def find_element_by_id(self, id_):  
    """Finds an element by id.  
    :Args: - id\_ - The id of the element to be found.  
    :Returns: - WebElement - the element if it was found  
    :Raises: - NoSuchElementException - if the element wasn't found  
    :Usage: element = driver.find_element_by_id('foo')  
    """  
    return self.find_element(by=By.ID, value=id_)
```

```
def find_elements(self, by=By.ID, value=None):
    """
    Find elements given a By strategy and locator. Prefer the find_elements_by_* methods when
    possible.
    :Usage: elements = driver.find_elements(By.CLASS_NAME, 'foo')
    :rtype: list of WebElement
    """
    if self.w3c:
        if by == By.ID:
            by = By.CSS_SELECTOR
            value = '[id="%s"]' % value
        elif by == By.TAG_NAME:
            by = By.CSS_SELECTOR
        elif by == By.CLASS_NAME:
            by = By.CSS_SELECTOR
            value = ".%s" % value
        elif by == By.NAME:
            by = By.CSS_SELECTOR
            value = '[name="%s"]' % value
    # Return empty list if driver returns null
    # See https://github.com/SeleniumHQ/selenium/issues/4555
    return self.execute(Command.FIND_ELEMENTS, { 'using': by, 'value': value })['value'] or []
```

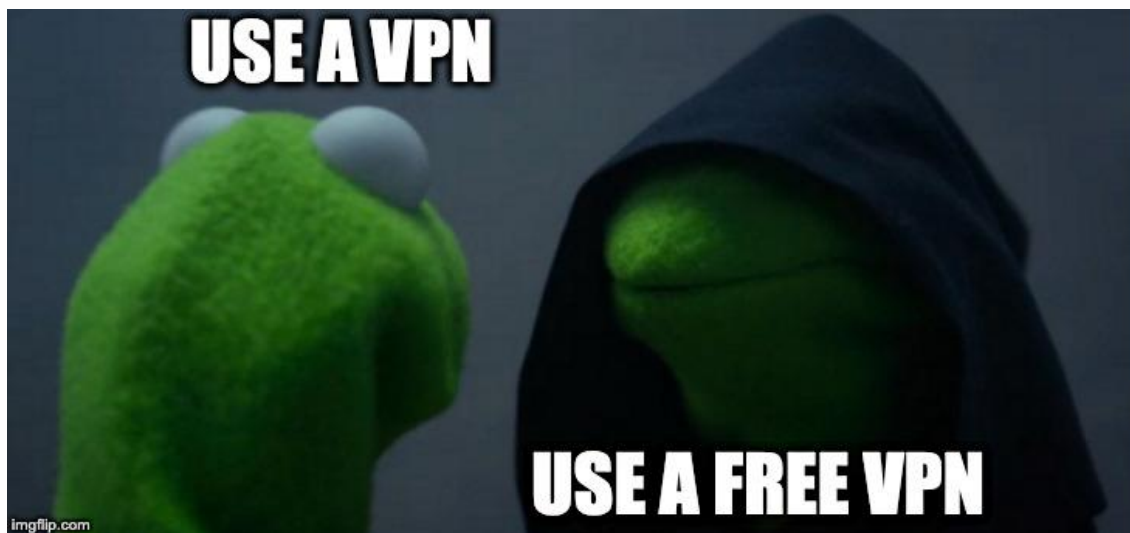

Вопросы



04

**Какой тип локаторов
лучше использовать?**

<https://bit.ly/2IV65ub>



When Front end developers need to locate elements on a web page, they use CSS selectors! Why not for selenium webdriver automation?

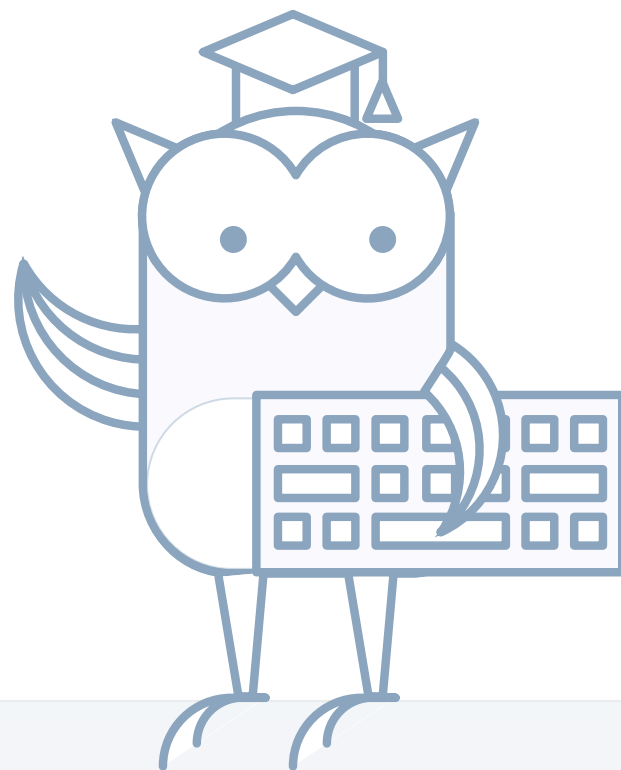


Why Choose CSS Selectors for Selenium Automation?

Other Selectors	⇒ By.cssSelector
-----	⇒ -----
By.className("register")	⇒ ".register"
By.tagName("table")	⇒ "table"
By.id("unique_id")	⇒ "#unique_id"
By.name("login")	⇒ "[name=login]"
By.xpath("//body/nav")	⇒ "body > nav"
By.xpath("//body//nav")	⇒ "body nav"

image created via carbon.now.sh

Вопросы



05

Оптимизация фронтенда

Ссылки

- <https://www.ivy.ru>
- <https://www.ozon.ru>
- <https://avito.ru>

Вопросы



Опрос

<https://otus.ru/polls/4266/>



**Спасибо
за внимание!**

