

Selenium IDE

автоматизация и тестирование

О себе

Александр Юрченко

- 3 года в автоматизированном тестировании
- TestComplete, Selenium, QTP (UFT)
- aliaksandr_yurchanka@epam.com
- Skype: aliaksandr.yurchanka

План лекций

- Selenium IDE
- Locators
- WebDriver



План текущей лекции

- Обзор инструментов тестирования
- Обзор компонентов Selenium
- Selenium IDE. Как это работает.



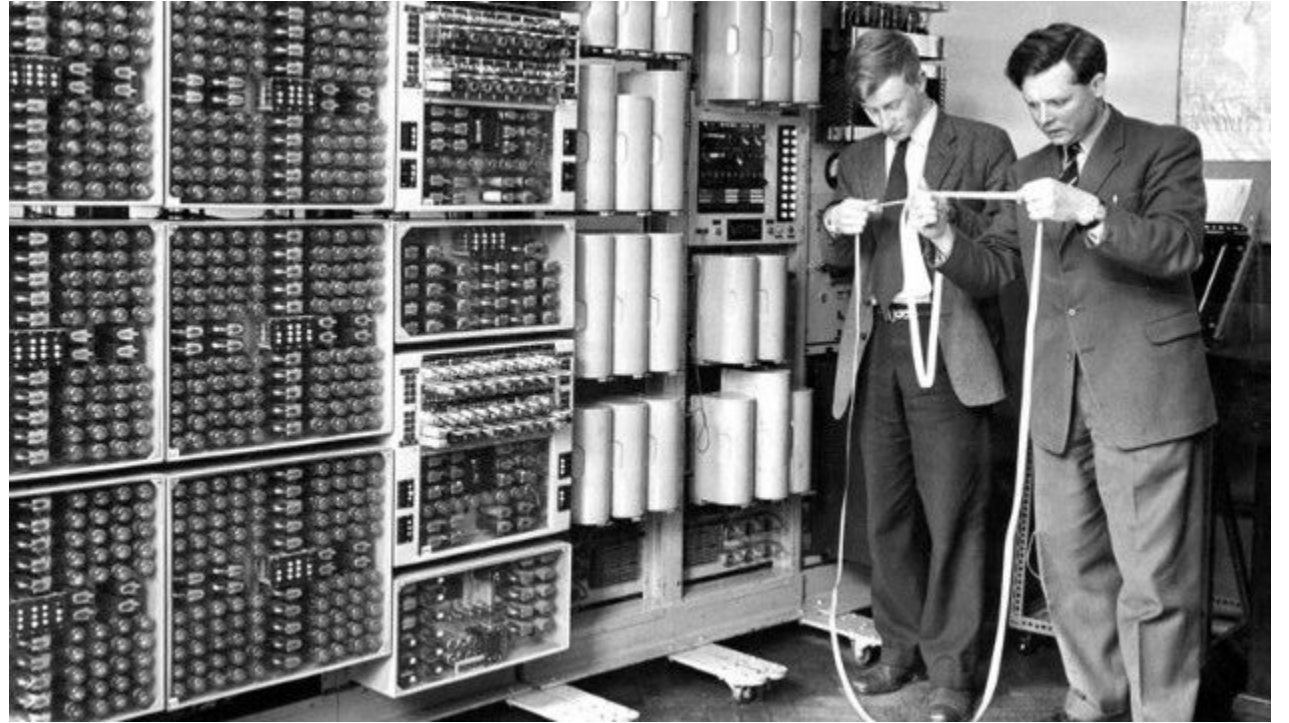
Цели



- Запись, редактирование тестового скрипта
- Запуск тестов
- Анализ результатов

Automation testing History

- Record/Playback
- Script-Based
- Data-driven testing
- Keyword-Driven



Что такое Selenium

- **Selenium** – это набор инструментов для автоматизации веб-приложений на различных платформах
- Бесплатное решение для автоматизации тестирования с открытым кодом
- <http://seleniumhq.org/> - официальный сайт



Сравнение с другими инструментами



Достоинства

- Открытый проект. Бесплатный
- Можно исправлять проблемы прямо в исходном коде
- Возможность выбора языка программирования для написания тестов

Сравнение с другими инструментами



Недостатки

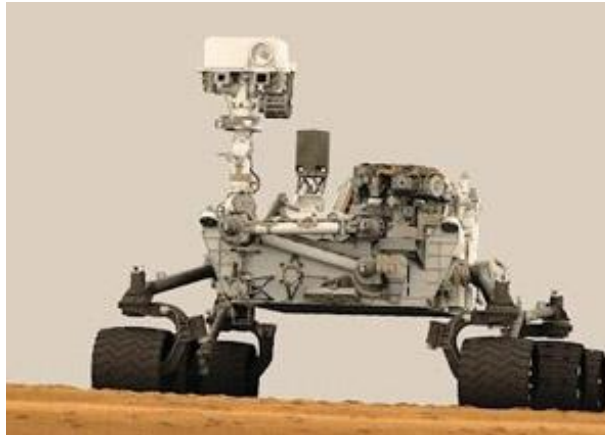
- Нет поддержки производителя
- Только веб приложения

Обзор компонентов

IDE



WebDriver



Grid



- **Selenium IDE** – record and play инструмент
- **WebDriver** – позволяет создавать полноценные проекты, содержащие тесты для вашего приложения
- **Grid** — распределение тестов по нескольким машинам

Поддерживаемые платформы



С чего начать



Selenium IDE

- небольшой сценарий для быстрого воспроизведения бага

WebDriver

- тесты со сложной логикой поведения и проверок

Grid

- запускать тесты удаленно с разными операционными системами, с разными браузерами на разных машинах

Установка Selenium IDE



Mozilla Firefox

docs.seleniumhq.org/download



Selenium IDE плагин для
Firefox

**SeleniumHQ**
Browser Automation

[edit this page](#) search selenium:

[Projects](#) [Download](#) [Documentation](#) [Support](#) [About](#)

[Selenium Downloads](#)

[Latest Releases](#)

[Previous Releases](#)

[Source Code](#)

[Maven Information](#)

Donate to Selenium

with Google Checkout

Downloads

Below is where you can find the latest releases of all the Selenium components. You can also find a list of [previous releases](#), [source code](#), and additional information for [Maven users](#) (Maven is a popular Java build tool).

Selenium IDE

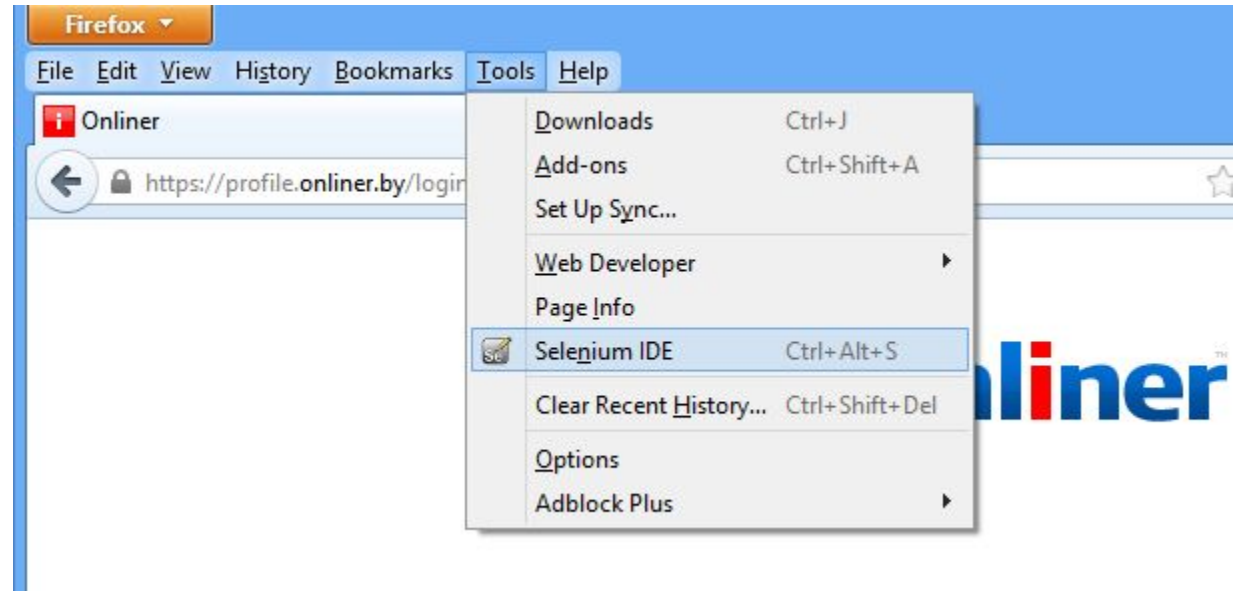
Selenium IDE is a Firefox plugin that does record-and-playback of interactions with the browser. Use this to either create simple scripts, assist in exploratory testing. It can also export Remote Control or WebDriver scripts, though they tend to be somewhat brittle and should be overhauled into some sort of Page Object-y structure for any kind of resiliency.

Download latest released version **2.0.0** released on 26/April/2013 or view the [Release Notes](#)

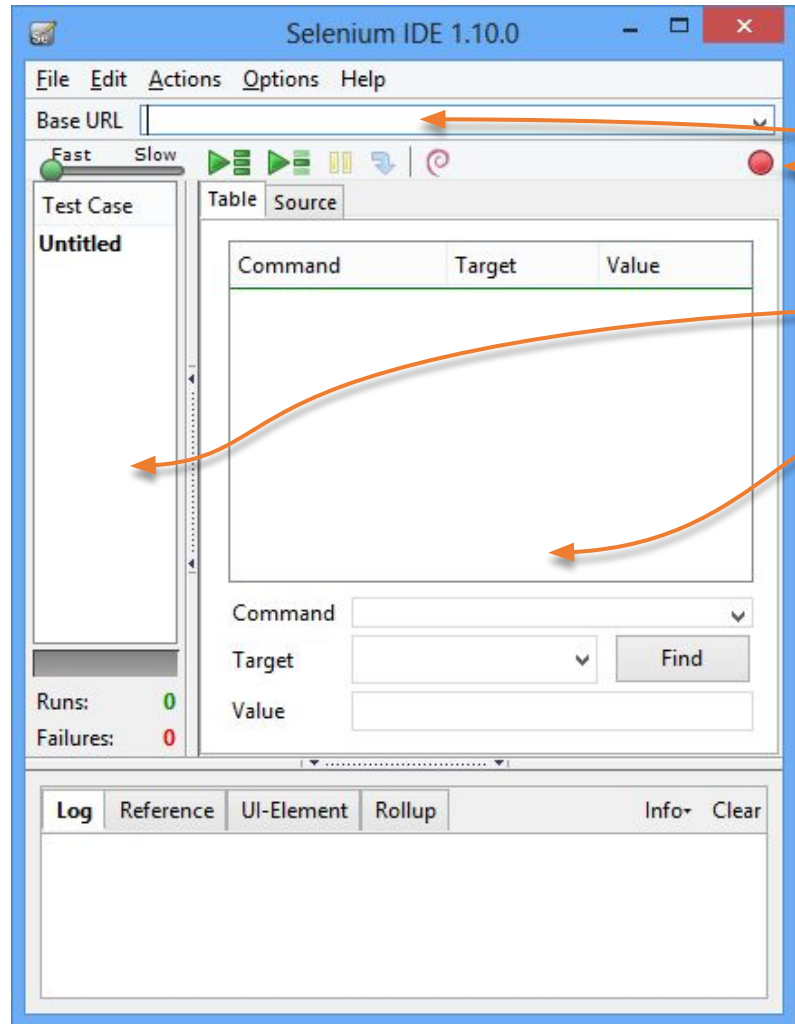
Запуск Selenium IDE

Selenium IDE можно запустить следующим образом:

- Через меню - Tools -> Selenium IDE
- Горячие клавиши - Ctrl+Alt+S
- Закрепить кнопку на панели инструментов



Обзор интерфейса

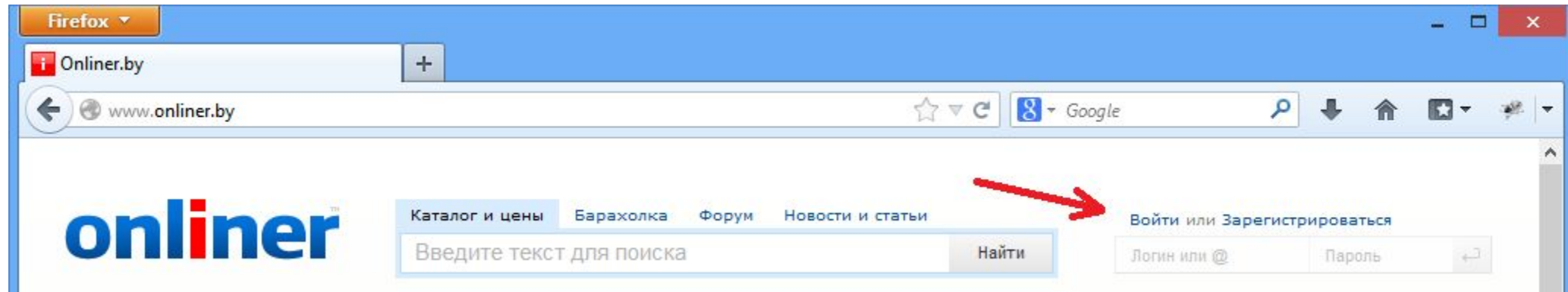


Основные элементы

- Стартовый URL
- Кнопка записи действий
- Список тест кейсов (Test Suite)
- Список команд (скрипт)

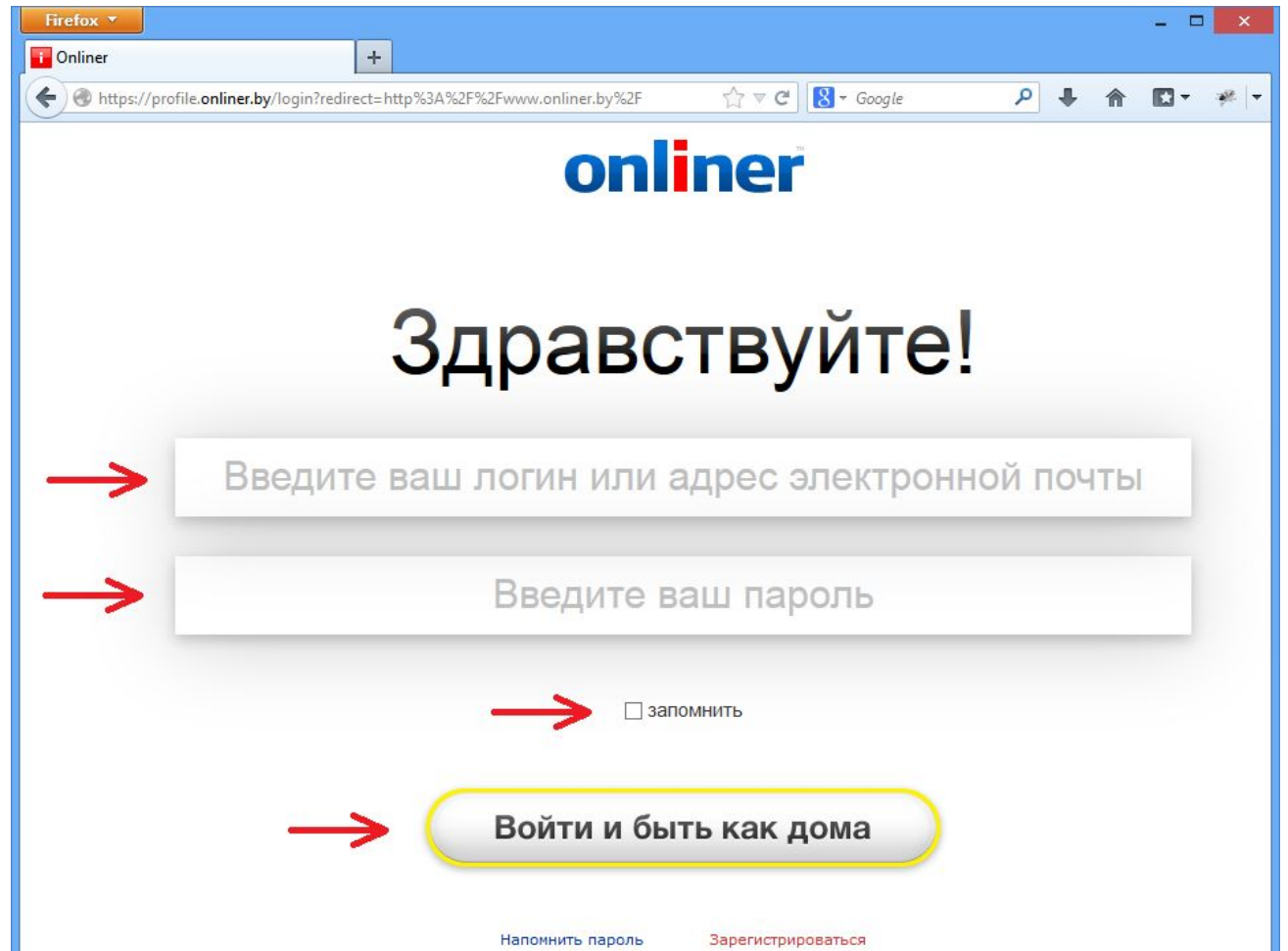
Сценарий для записи действий

1. Открыть <http://www.onliner.by/>
2. Кликнуть по ссылке “Войти”



Сценарий для записи действий

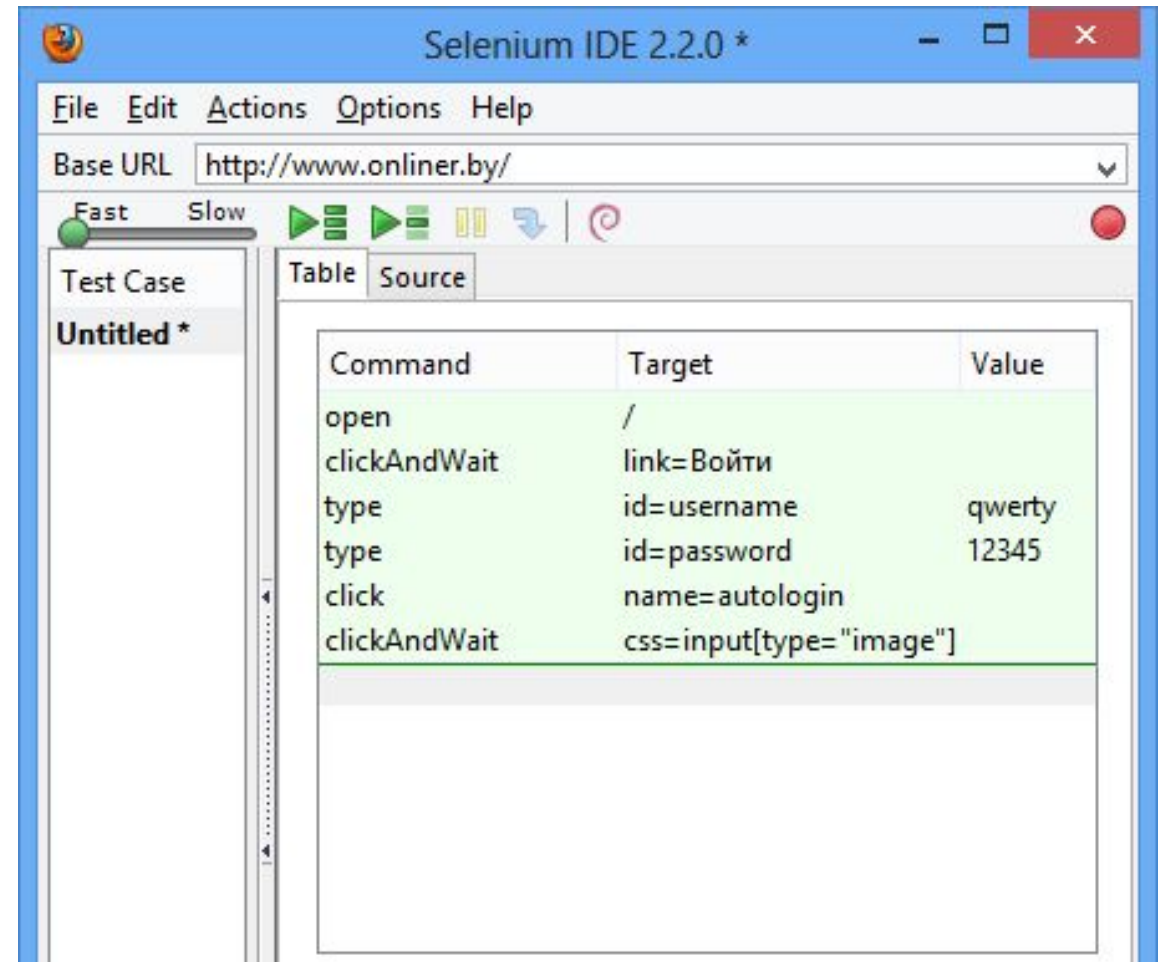
3. Заполнить поля
4. Кликнуть чекбокс “запомнить”
5. Нажать кнопку “Войти”



Записанный скрипт

Записаны следующие команды:

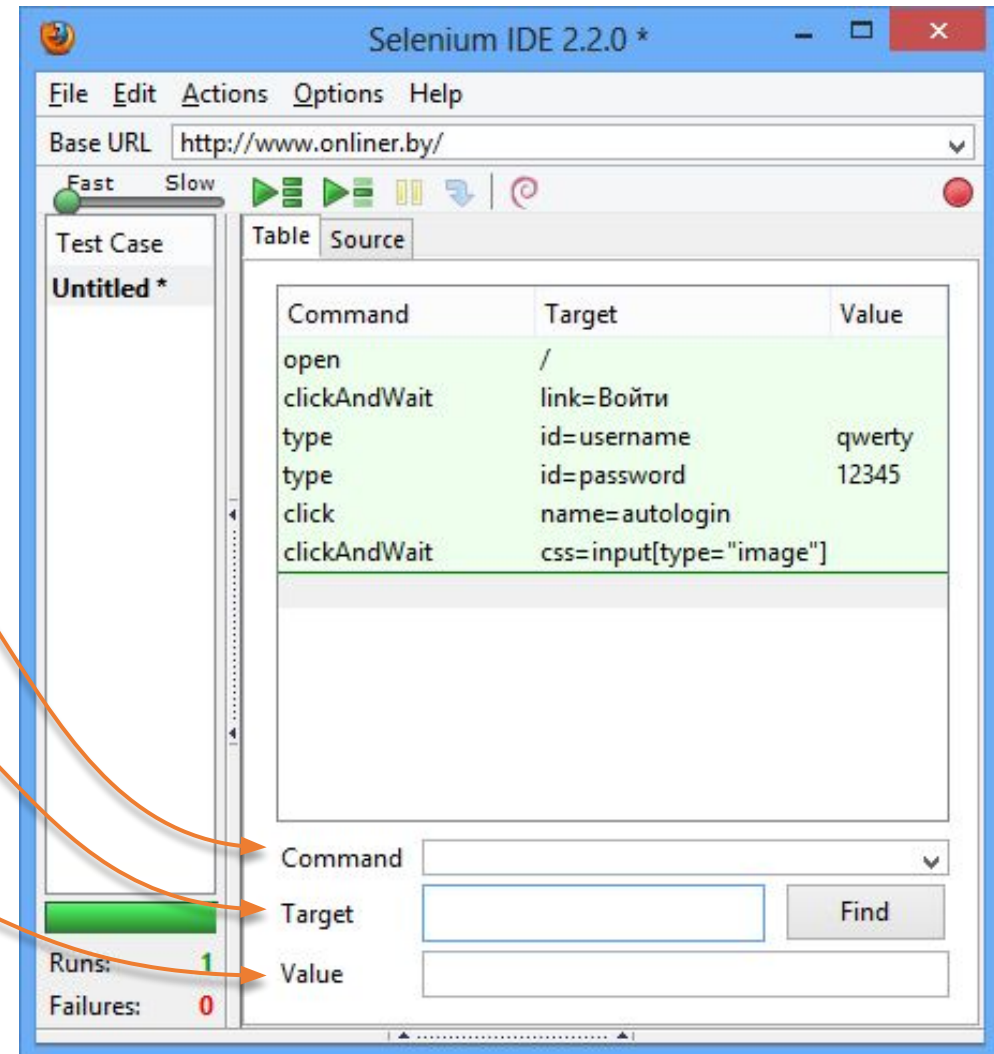
- **open** – URL который нужно открыть
- **clickAndWait** – ждет загрузки страницы
- **type** – ввод текста
- **click** – клик по чекбоксу



Структура команды Selenium IDE

Команда состоит из трех частей:

- Какое действие необходимо сделать
- Над каким объектом необходимо произвести это действие
- Нужны ли для этого дополнительные данные



Синхронизация

Имеются команды ожидания двух типов:

*AndWait

- ожидание загрузки страницы после действия
- Работает только когда страница перегружается
- Не подходит для AJAX-сайтов

waitFor*

- команды ожидания элемента на UI



Типы проверок

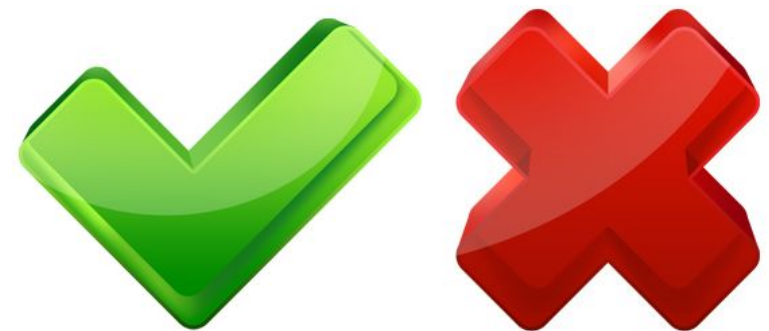
Если Fail

Assert

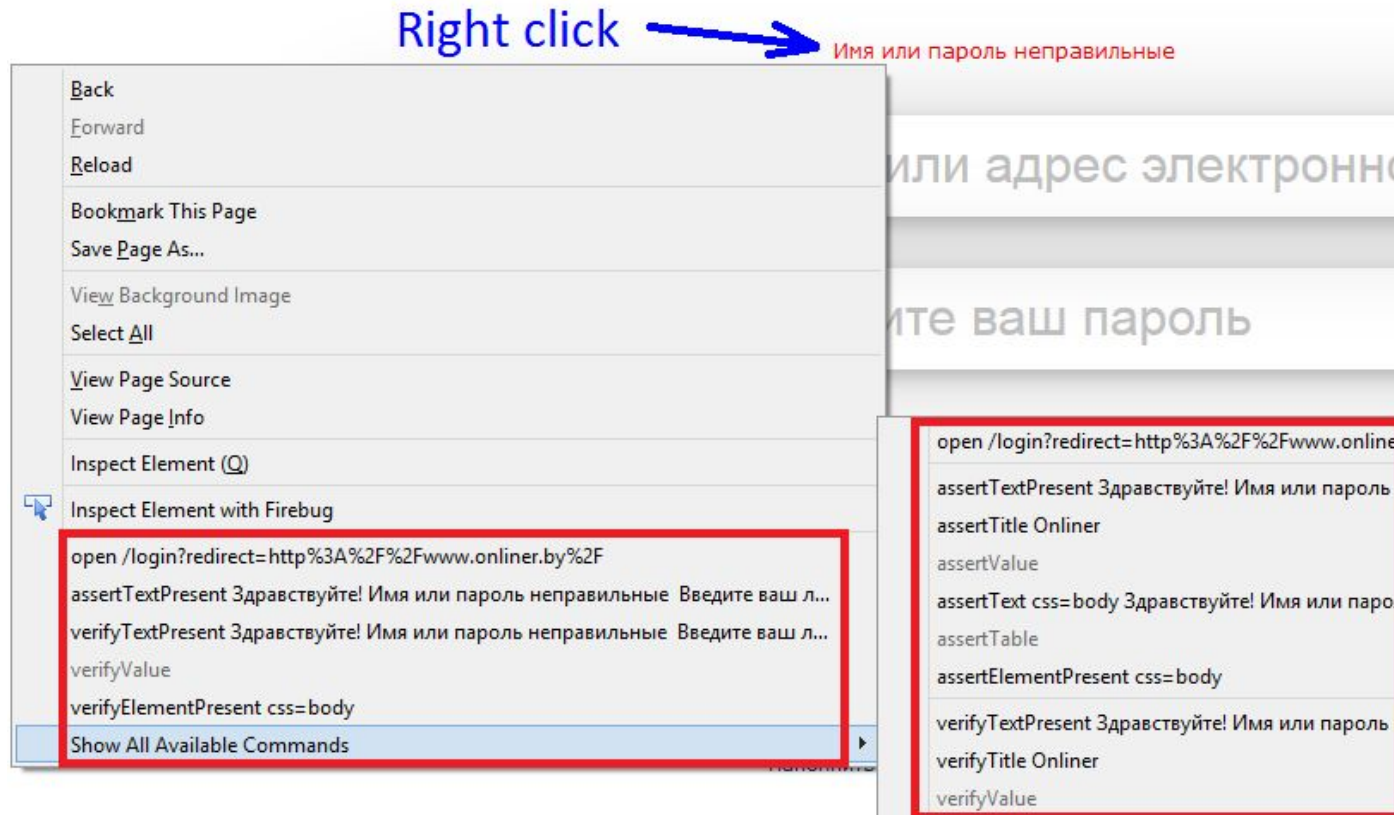
- Записывает в отчет сообщение об ошибке
- Останавливает тест

Verify

- Записывает в отчет сообщение об ошибке
- Продолжает тест



Добавление проверки



Проверку можно добавить двумя способами:

- Правый клик по элементу на странице
- В Selenium IDE. Правый клик по списку записанных степов -> Insert New Command

Добавление проверки

Добавлены были следующие проверки:

- Проверка наличия текста на странице
- Проверка чекбокс включен
- Проверка отсутствия текста на странице

Login.html - Selenium IDE 2.2.0

File Edit Actions Options Help

Base URL

Fast Slow

Test Case

Login

Command	Target	Value
open	/	
clickAndWait	link= Войти	
verifyText	css=h1	Здравствуйте!
type	id=username	qwerty
type	id=password	12345
click	name=autologin	
verifyChecked	name=autologin	
clickAndWait	css=input[type="image"]	
verifyTextNotPresent	Имя или пароль неправильные	

Command

Target Find

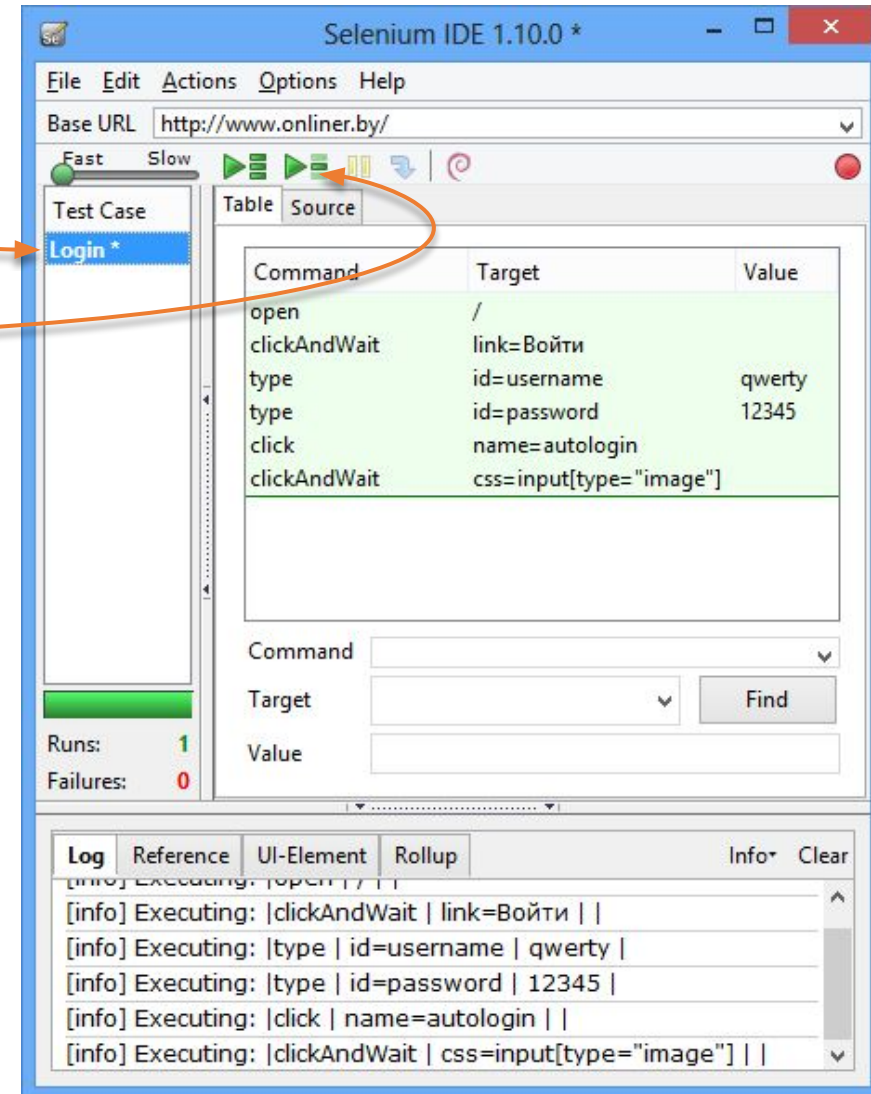
Value

Runs: 1

Failures: 1

Воспроизведение скрипта

1. Выбрать нужный скрипт из списка
2. Нажать кнопку “Play current test case”



Анализ результатов теста

Login.html - Selenium IDE 1.10.0

File Edit Actions Options Help

Base URL http://www.onliner.by/

Test Case Login

Command	Target	Value
open	/	
clickAndWait	link=Войти	
verifyTextPresent	Здравствуй!	
type	id=username	qwerty
type	id=password	12345
click	name=autologin	
verifyChecked	name=autologin	
clickAndWait	css=input[type="image"]	
verifyTextNotPresent	Имя или пароль неправильные	

Runs: 1
Failures: 1

Log Reference UI-Element Rollup Info+ Clear

[info] Executing: | open | / | |
[info] Executing: | clickAndWait | link=Войти | |
[info] Executing: | verifyTextPresent | Здравствуй! | |
[error] false
[info] Executing: | type | id=username | qwerty |
[info] Executing: | type | id=password | 12345 |
[info] Executing: | click | name=autologin | |
[info] Executing: | verifyChecked | name=autologin | |
[info] Executing: | clickAndWait | css=input[type="image"] | |
[info] Executing: | verifyTextNotPresent | Имя или пароль неправильные | |
[error] true

- verifyTextPresent
- verifyChecked
- verifyTextNotPresent

Здравствуйте!

Имя или пароль неправильные

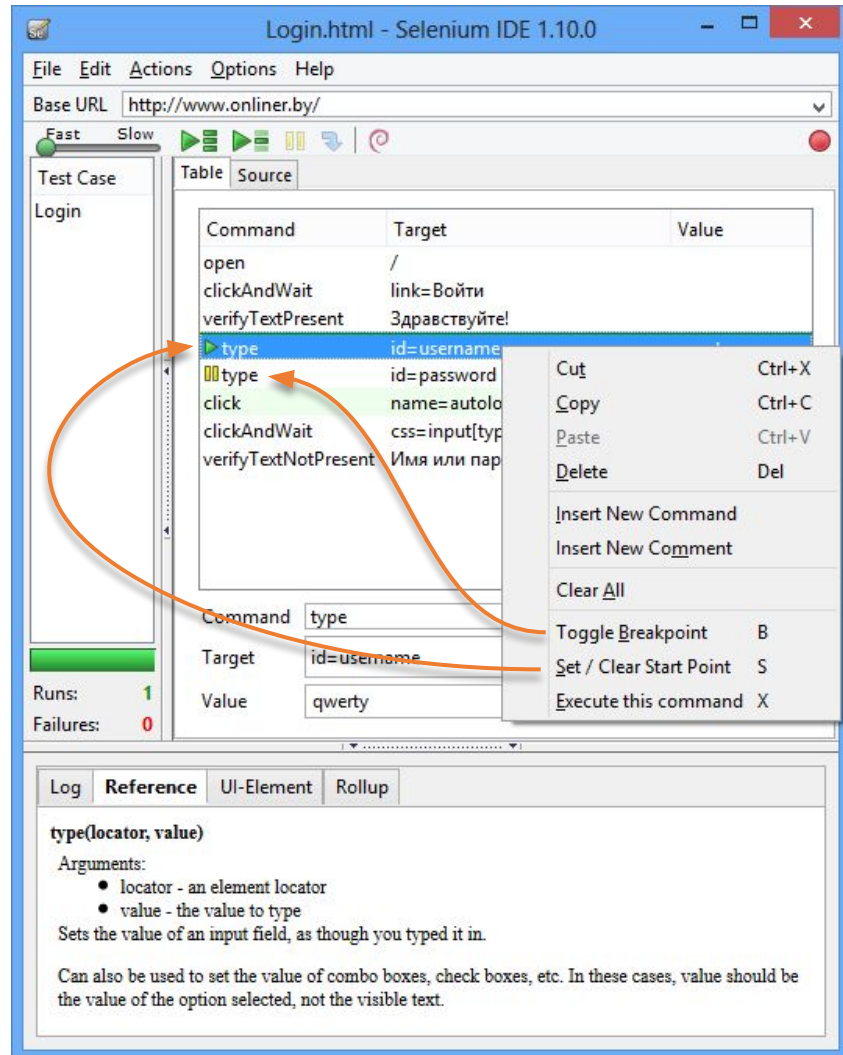
ваш логин или адрес электронной

Введите ваш пароль

☐ запомнить

Войти и быть как дома

Отладка

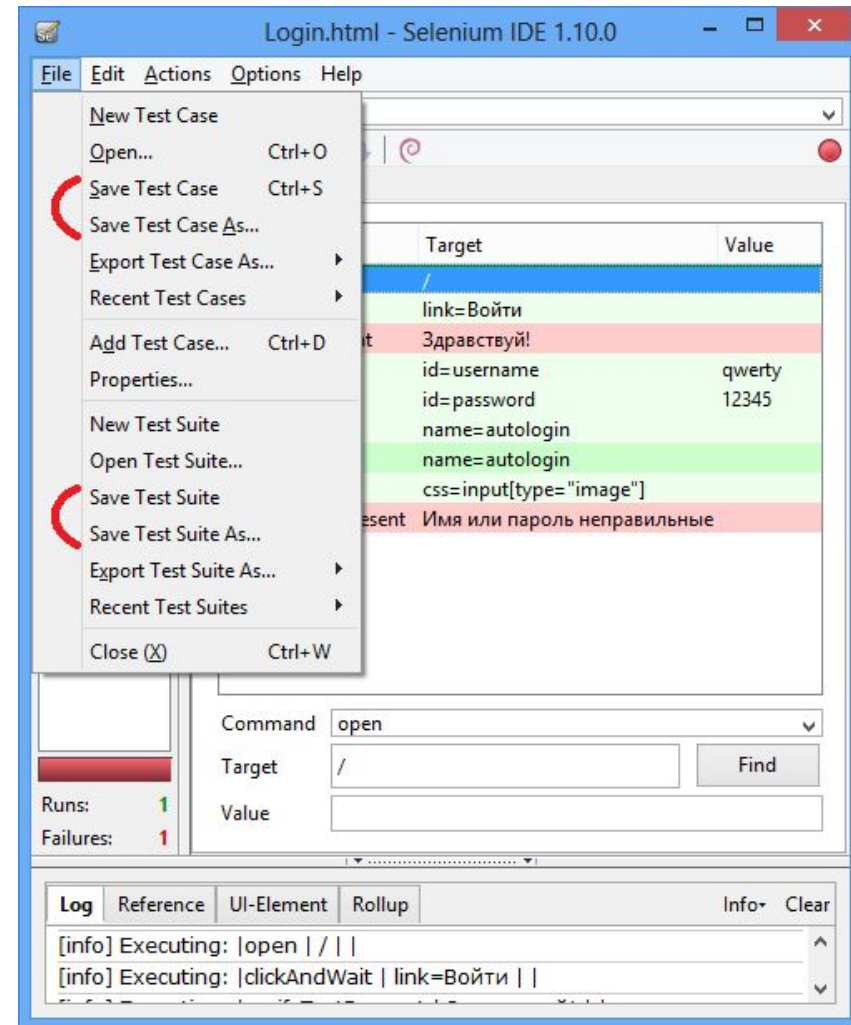


Режим отладки позволяет:

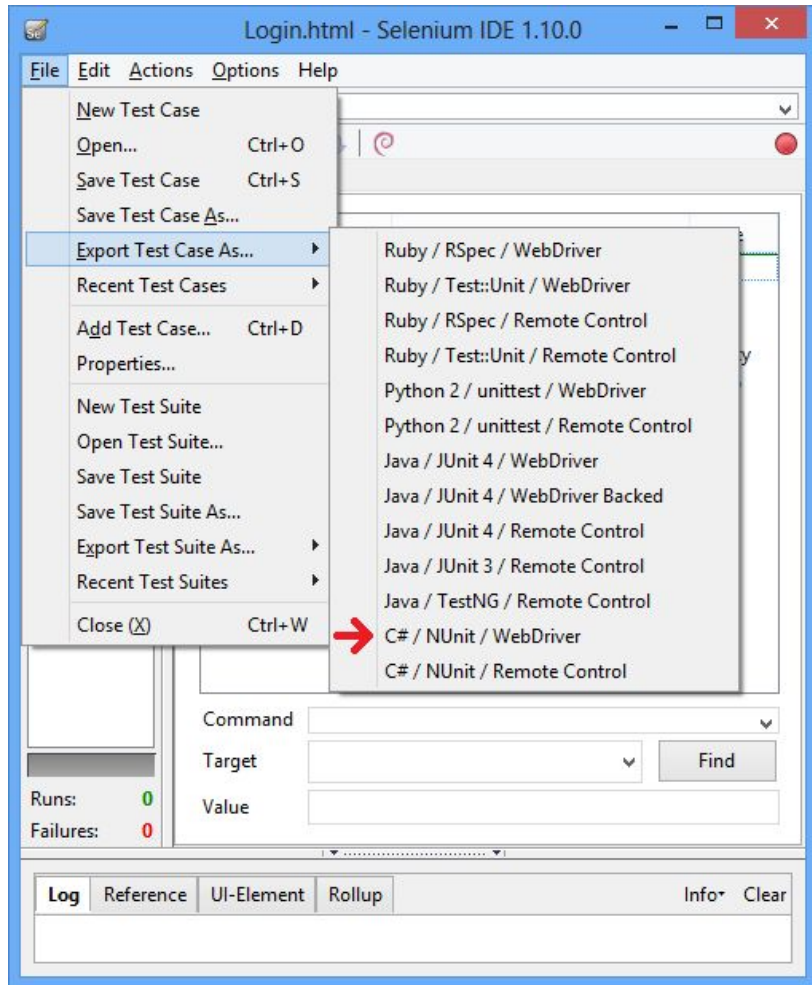
- Запустить скрипт с нужного места (S)
- Рассмотреть что находится на странице (B)
- Изменить что либо на странице вручную во время выполнения скрипта (B)
- Выполнить одну команду из скрипта (X)

Сохранение Test Case, Test Suite

- Тесты можно сохранить как **Test Case** и как **Test Suite**
- В **Test Suite** (набор тестов) можно добавлять и удалять тесты
- Файл Test Suite должен находиться в одной папке с файлами Test Case
- Ctrl+D – добавить Test Case в Test Suite с диска
- New Test Case – создать **новый** Test Case в открытом Test Suite

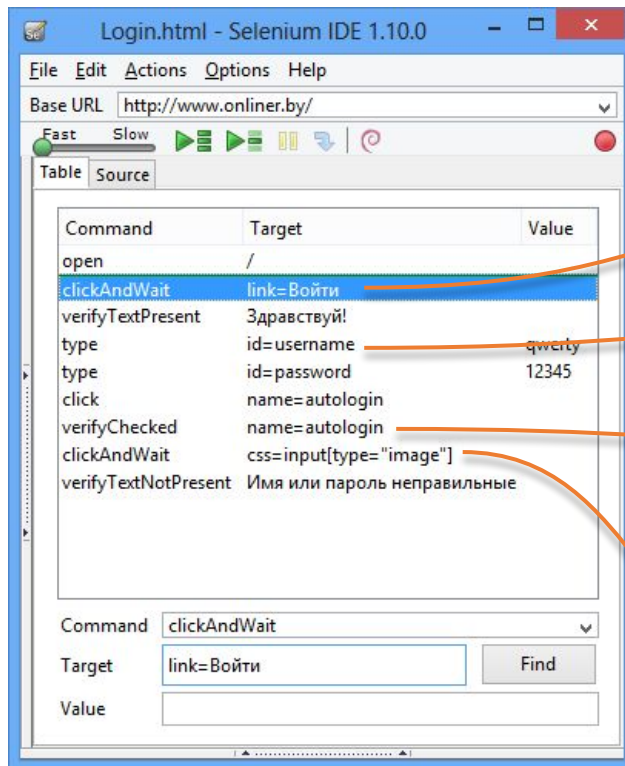


Экспорт скрипта



Локаторы

Локатор - это строка, уникально идентифицирующая UI-элемент.



По тексту
ссылки

```
<div class="b-top-profile">  
  <div class="b-top-profile__log-in">  
    <p>  
      <a href="https://profile.onliner.by/login?redirect=http%3A%2F%2Fwww.onliner.by%2F">Войти</a>  
    </p>  
  </div>  
</div>
```

По
"id"

```
<form autocomplete="off" method="post" action="">  
  <div class="inputs">  
    <div class="inputs-tip">Введите ваш логин или адрес электронной почты</div>  
    <input id="username" class="i-p bdc trans" type="text" title="" value="" name="username">  
  </div>  
</form>
```

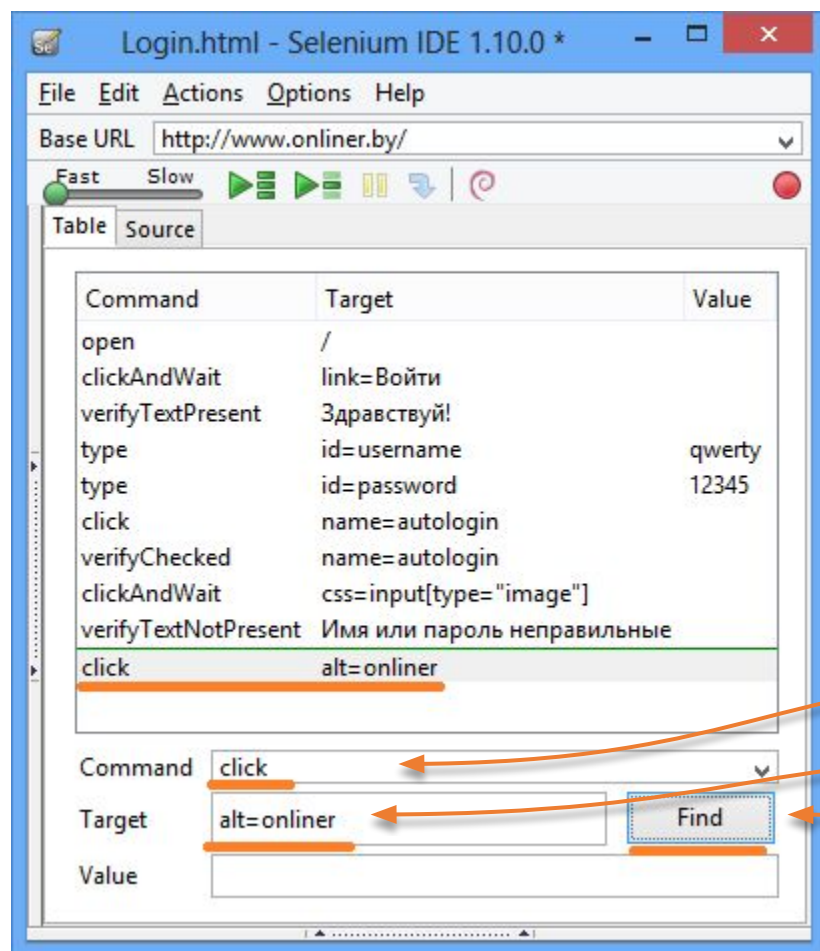
По
"name"

```
<div class="save-me">  
  <label>  
    <input type="checkbox" name="autologin">  
    запомнить  
  </label>  
</div>
```

По
css

```
<div class="b-btn">  
  <div class="btn" style="width: 402px;">  
    <input type="image" value="" name="" alt="Войти и быть как дома" src="/static/img/btn-enter.png">  
  </div>  
</div>
```

Добавление команды вручную



```
<div id="minWidth">  
  <div class="logo">  
    <a href="http://www.onliner.by">  
        
    </a>  
  </div>  
</div>
```

onliner

- Выбор команды
- Локатор элемента
- Подсветить элемент на странице

Demo



Преимущества и недостатки Selenium IDE

Преимущества

- Простота создания теста
- Не требуется знаний языка программирования
- Keyword-driven подход

Недостатки

Нет возможности использования:

- Баз данных
- Файловых хранилищ
- Стандартный отчёт



Автоматизация - быть или не быть?

Автоматизировать лучше

- Тесты, которые выполняются чаще всего
- Манипуляции с большим количеством данных, вычисления
- Тесты которые нужно проходить на разных конфигурациях

Не использовать автоматизацию

- UI сильно изменится в ближайшем будущем
- Тест кейс не имеет предсказуемого результата
- Краткосрочный проект



Вспомогательные инструменты

- **FireBug** — инспектор, редактор HTML. Возможность получить Xpath, CSS элемента
- **FireFinder** — расширение для FireBug. Генерирует Xpath, CSS локаторы, позволяет создавать свои и сразу же проверять
- **XPath Checker** – интерактивный редактор XPath выражений
- **Implicit Wait** — Плагин для Selenium IDE. Устраняет необходимость использовать `waiForElementPresent` перед Click, Type, Select.



Ссылки

- Документация по Selenium на русском
<http://selenium2.ru/docs.html>

Домашняя работа

1. Открыть <http://ru.wikipedia.org/>
2. Перейти к созданию учетной записи. Ссылка “Создать учётную запись”
3. Заполнить поля
 - Имя учетной записи: “Alex”
 - Заполнить поля Пароль и Подтверждение пароля
 - Email любой
4. Нажать кнопку “Создать учетную запись”
5. Проверить что на странице отображается сообщение “Ошибка создания учетной записи”

Можно похожее задание на примере любого другого сайта.

Детали домашнего задания будут в письме после лекции.

Вопросы?

