



Selenium-IDE

Обзор инструмента

План тренинга

- ✓ Что такое Selenium, что в него входит
- ✓ Что собой представляет Selenium IDE
- ✓ Создание, отладка и запуск тестов
- ✓ Обзор важных команд Selenium IDE
- ✓ Преимущества и недостатки Selenium IDE
- ✓ DEMO

Что такое Selenium

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

Что входит в набор инструментов Selenium



Selenium IDE

– Расширение браузера Firefox, которое позволяет записывать и воспроизводить действия пользователя в браузере.



Selenium WebDriver

– Набор библиотек для различных языков программирования, позволяющих управлять браузером из программы, написанной на этом языке программирования.



Selenium RC

– Предшественник Selenium WebDriver, который в настоящее время имеет статус "замороженного" и в дальнейшем развиваться не будет.



Selenium Server

– HTTP-сервер, написанный на java, который принимает команды с удалённой машины, где работает сценарий автоматизации, и исполняет их в браузере. Несколько серверов Selenium могут образовывать распределённую сеть, которая называется ***Selenium Grid***.

С чем работает набор инструментов Selenium

Поддерживаемые платформы

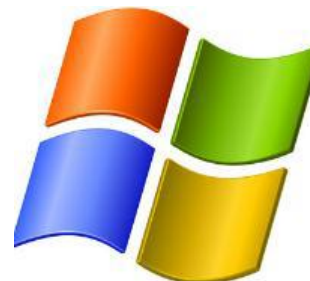
- Браузеры

- ✓ Firefox
- ✓ Internet Explorer
- ✓ Safari
- ✓ Opera и другие



- Операционные системы

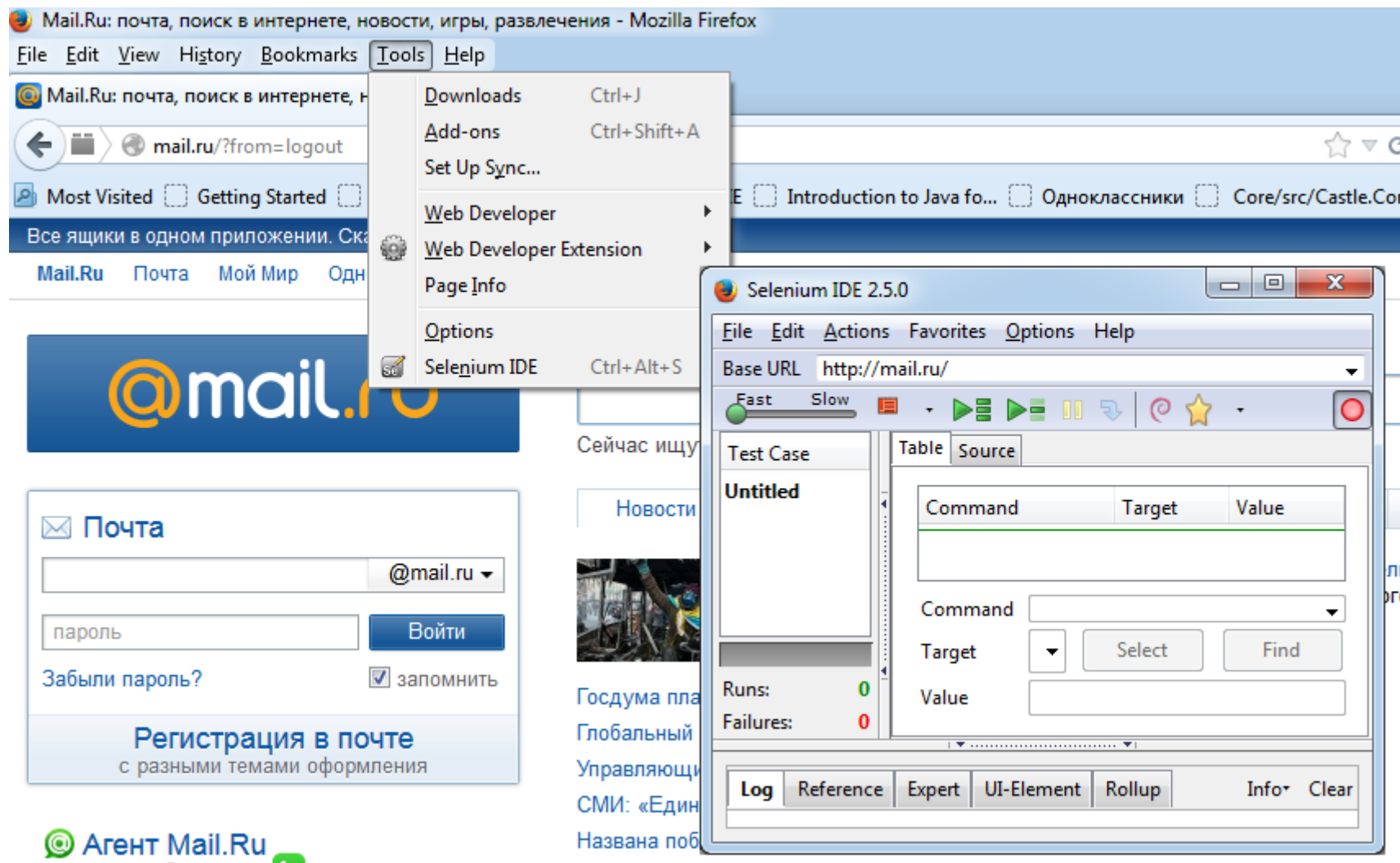
- ✓ Windows
- ✓ OS X
- ✓ Linux
- ✓ Solaris и другие



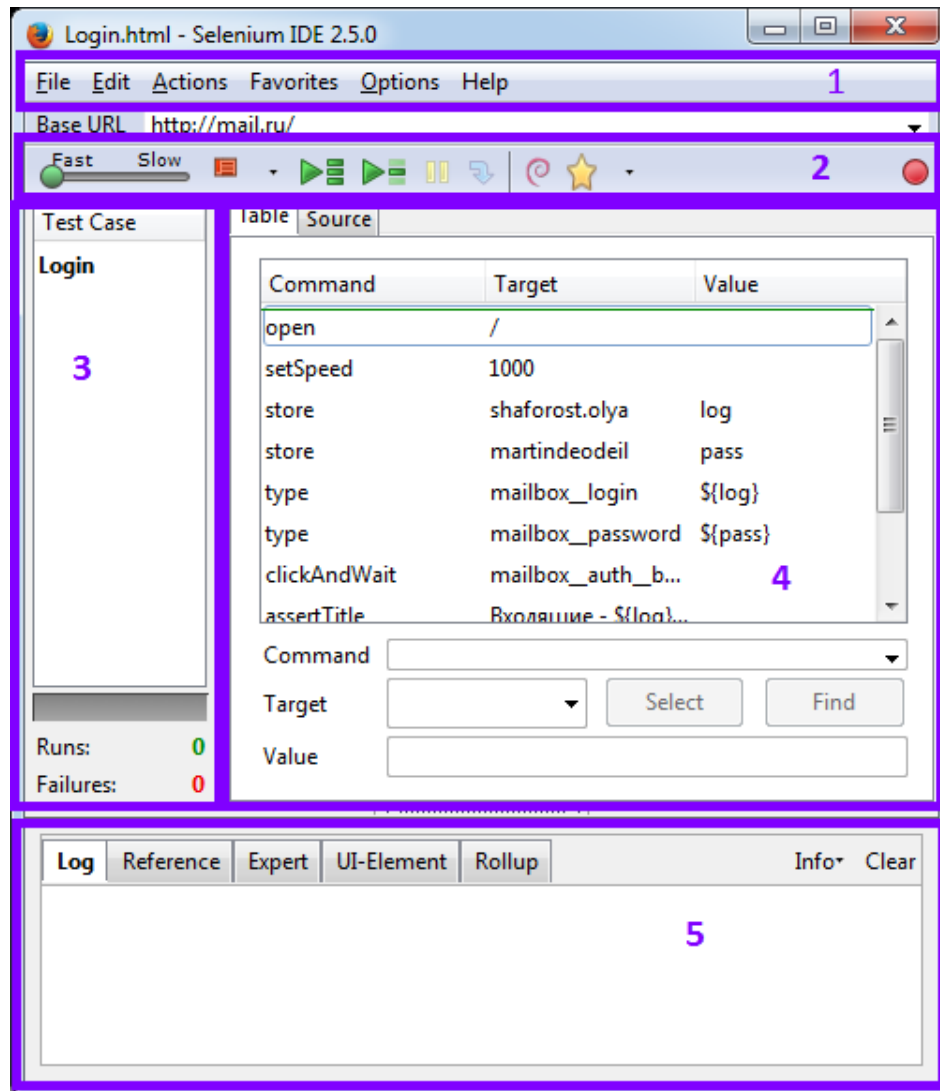
Что собой представляет Selenium IDE

- ✓ Представляет из себя плагин для Mozilla Firefox, который записывает и воспроизводит действия пользователя в браузере.
- ✓ В IDE могут быть автоматически записаны такие действия как нажатия на элементы, открытие страниц, заполнение полей ввода и т.п..
- ✓ Проверки добавляются в сценарий с помощью выбора доступных проверок из контекстного меню на элементе, который нужно проверить.
- ✓ Selenium IDE не требует от тестировщика глубоких знаний программирования.
- ✓ С помощью IDE можно быстро ознакомиться с языком команд и понять принцип работы Selenium.
- ✓ Реализовать в сценариях IDE сложную логику или проверку просто невозможно.

Описание интерфейса Selenium IDE



Описание интерфейса Selenium IDE



1. Панель меню
2. Панель инструментов
3. Панель списка тестовых сценариев
4. Панель тестового сценария
5. Информационная панель

Описание интерфейса Selenium IDE

Панель инструментов:



1. “Управление скоростью”:

Позволяет контролировать скорость выполнения тестового сценария.

2. “Запустить все”:

Запускает весь набор тестов, в случае, если он содержит несколько тестовых сценариев.

3. “Запустить”:

Запускает выбранный в данный момент тест.

4. “Пауза”/”Возобновить”:

Останавливают и возобновляют исполнение текущего тестового сценария.

5. “Шаг”:

Позволяет выполнять тест “по шагам”, то есть по одной команде за раз. Предназначено для отладки тестовых сценариев.

6. “Применить правила группировки”:

Эта продвинутая функция дает возможность группировки повторяющихся последовательностей команд Selenium в одно действие. Более подробную информацию о правилах группирования можно найти в “Документации по расширению UI-Element” в разделе “Помощь”.

7. “Запись”:

Записывает действия пользователя в браузере.

Описание интерфейса Selenium IDE

Панель тестового сценария

Table Source		
Command	Target	Value
open	/	
setSpeed	1000	
store	shaforost.olya	log
store	martindeodeil	pass
type	mailbox_login	\${log}
type	mailbox_password	\${pass}
clickAndWait	mailbox_auth_button	
assertTitle	Входящие - \${log}@m...	
assertText	id=PH_user-email	\${log}@mail.ru

Command	type	
Target	mailbox_login	Select Find
Value	\${log}	

- ✓ **Command**
Говорит что Selenium делать
- ✓ **Target**
Идентификатор UI элемента на странице
- ✓ **Value**
Поле для дополнительных параметров

Описание интерфейса Selenium IDE

Информационная панель

Log	Reference	UI-Element	Rollup	Info	Clear
[info] Executing: setSpeed 1000					
[info] Executing: open /					
[info] Executing: waitForPageToLoad					
[info] Executing: store @mail.ru domain					
[info] Executing: store apsavin32517 log					
[info] Executing: store 32517apsavin pass					
[info] Executing: type mailbox__login \${log}					
[info] Executing: type mailbox__password \${pass}					
[info] Executing: click mailbox__auth__button					
[info] Executing: waitForPageToLoad					
[info] Executing: assertText PH_user-email \${log}\${domain}					
[info] Changed test case					

✓ Лог

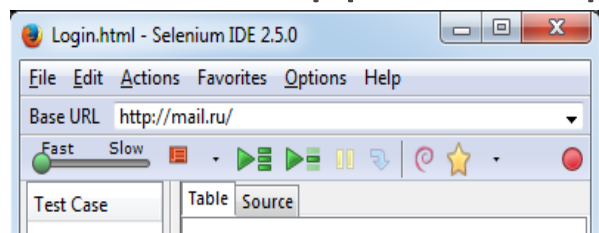
✓ Справка

✓ UI-element

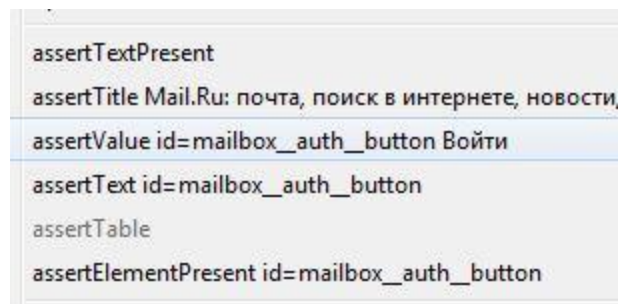
✓ Группировка

Создание, отладка и запуск тестов

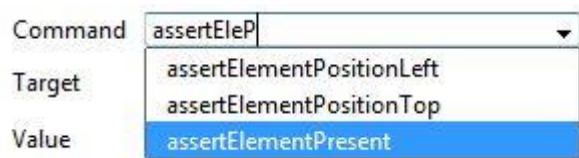
Способы создания сценариев



✓ Запись



✓ Контекстное меню



✓ Редактирование команд

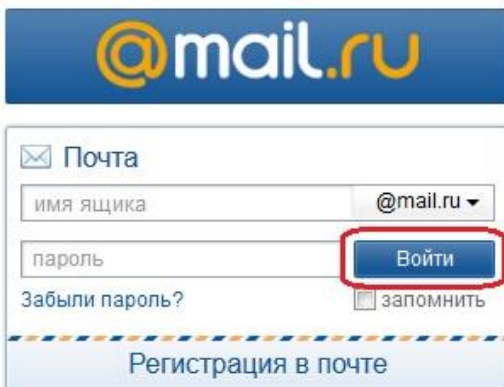
Создание, отладка и запуск тестов

Что такое локатор?

- ✓ Для определения элемента на странице браузера используется локатор.
- ✓ Локатор указывается в поле «target» панели тестового сценария.
- ✓ Самый распространённый способ создания локаторов – использование запросов **XPath**.
- ✓ В Selenium IDE нет такого понятия как объект.

Создание, отладка и запуск тестов

Локаатор с помощью XPath



```
<div class="mailbox_auth_layout_row mailbox_auth_layout_row_last">
  <div class="mailbox_auth_layout_row_col mailbox_auth_layout_row_col_right">
    <span class="w-mailbox_auth_button w-mailbox_auth_button_main">
      <span class="w-mailbox_auth_button w-mailbox_auth_button_outer">
        <span class="w-mailbox_auth_button w-mailbox_auth_button_inner">
          <input id="mailbox_auth_button" class="mailbox_auth_button" type="submit"
            tabindex="7" value="Войти" style="background-color: transparent;">
        </span>
      </span>
    </span>
  </div>
</div>
```

xpath=//input[@id='mailbox__auth__button']

xpath=//*[@id='mailbox__auth__button']

//input[@class='mailbox__auth__button']

//input[@value='Войти']

//input[@type='submit']

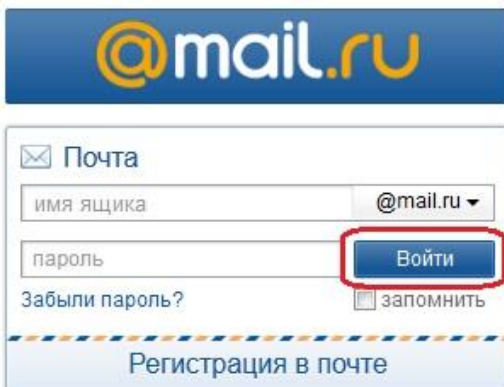
//input[contains(@class, 'button')]

//div[contains(@class, 'row__col_right')]/input

//span[contains(@class, 'auth__button')]/input[contains(@class, 'auth__button')]

Создание, отладка и запуск тестов

Локаатор с помощью XPath



Почта

Имя ящика @mail.ru

Пароль

Войти

Забыли пароль?

запомнить

Регистрация в почте

```
<div class="mailbox_auth_layout_row mailbox_auth_layout_row_last">
  <div class="mailbox_auth_layout_row_col mailbox_auth_layout_row_col_right">
    <span class="w-mailbox_auth_button w-mailbox_auth_button_main">
      <span class="w-mailbox_auth_button w-mailbox_auth_button_outer">
        <span class="w-mailbox_auth_button w-mailbox_auth_button_inner">
          <input id="mailbox_auth_button" class="mailbox_auth_button" type="submit"
            tabindex="7" value="Войти" style="background-color: transparent;">
        </span>
      </span>
    </span>
  </div>
</div>
```

- ✓ `text()` – функция, которая возвращает набор текста указанного элемента
- ✓ `contains()` – функция проверяющая содержится подстрока в строке или нет
- ✓ `//input[@id="mailbox__auth__button"]/..` - локаатор, ссылающийся на span, внутри которого находится указанный input (т.е. «/..» – уровень выше)
- ✓ чем проще запрос – тем лучше!

Обзор важных команд Selenium IDE

✓ Действие

Действия команд, которые изменяют состояние приложения
(click, type, select)

✓ Доступ

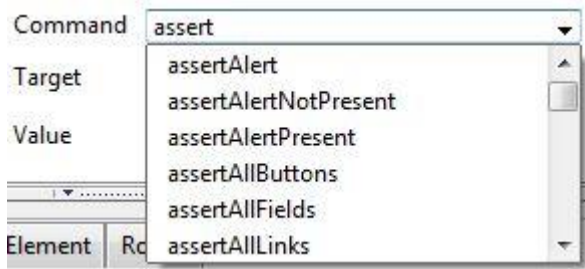
Проверяют состояния приложения и хранят значения переменных (storeText, storeXPathCount)

✓ Утверждение

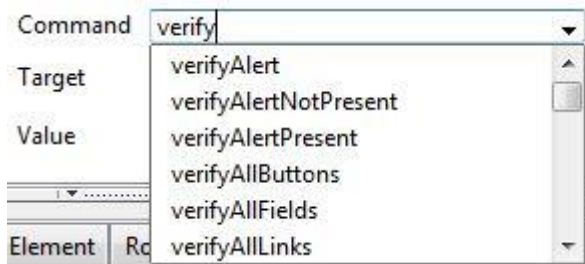
Проверяют состояния приложения, но возвращают логическое значение, либо основываются на сравнении этого значения (assert, verify, waitFor)

Обзор важных команд Selenium IDE

Проверки. Делятся на два типа



- ✓ Assert
в случае неудачи полностью
останавливается выполнение текущего
теста



- ✓ Verify
в случае неудачи фиксирует ошибку и
переходит к выполнению следующей
команды, если таковая есть

Обзор особенностей Selenium IDE

user-extensions.js

- ✓ JavaScript код, который позволяет расширять функциональность Selenium
- ✓ Задаётся
 - В настройках Selenium IDE
 - Параметром при запуске selenium-server.jar
(java -jar selenium-server.jar -userExtensions <file>)
 - В selenium-server.jar: обязательно одним файлом

Обзор особенностей Selenium IDE

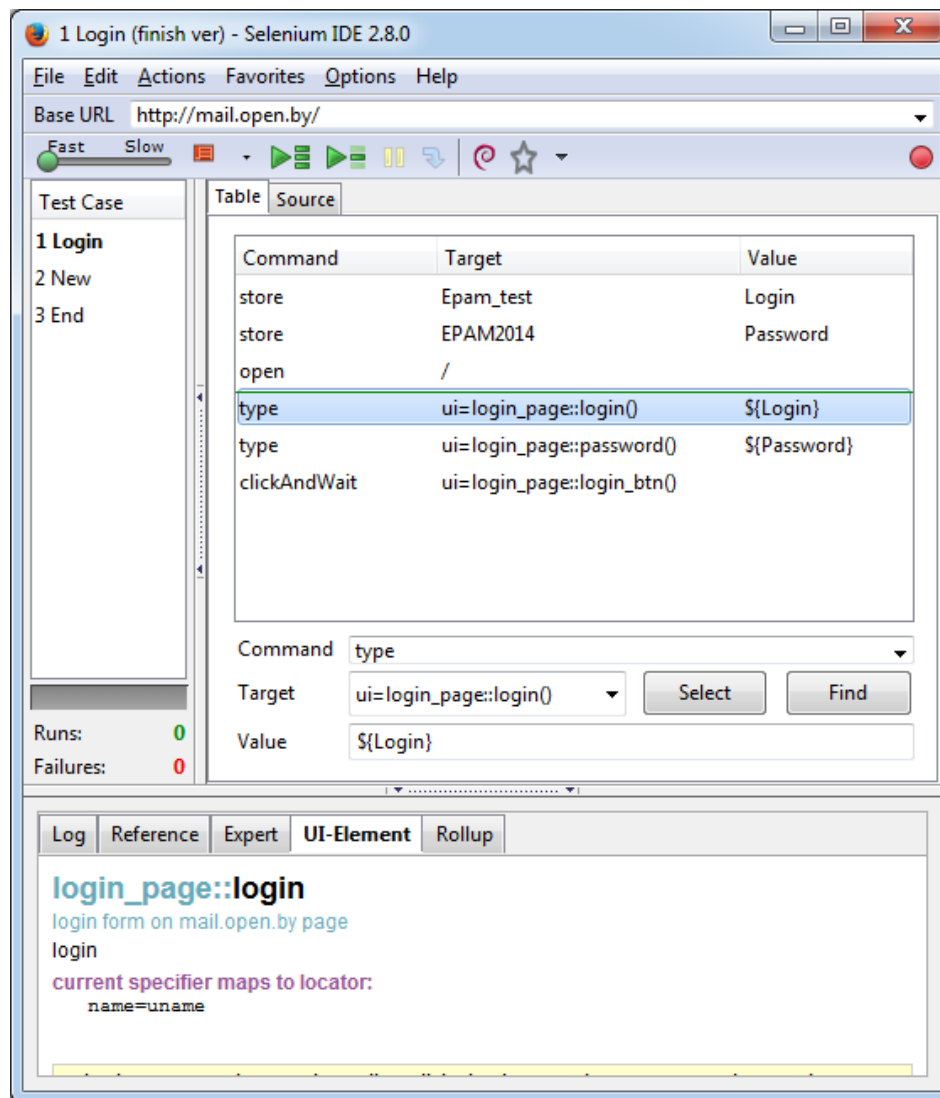
UI-map.

- ✓ Позволяет хранить описания элементов на UI в одном месте (файле).
- ✓ В случае изменения локаторов править придётся всего лишь 1 файл, не изменяя сами тест кейсы.
- ✓ Для создания требуются минимальные навыки программирования.

Обзор особенностей Selenium IDE

UI-map.

```
31 pathRegexp: 'some regexp'
32 });
33
34 /* Elements for login page*/
35 myMap.addElement('login_page', {
36   name: 'login',
37   description: 'login',
38   locator: 'name=uname'
39 });
40
41 myMap.addElement('login_page', {
42   name: 'password',
43   description: 'password',
44   locator: 'name=pass'
45 });
46
47 myMap.addElement('login_page', {
48   name: 'login_btn',
49   description: 'login button',
50   locator: 'name=login'
51 });
52
```



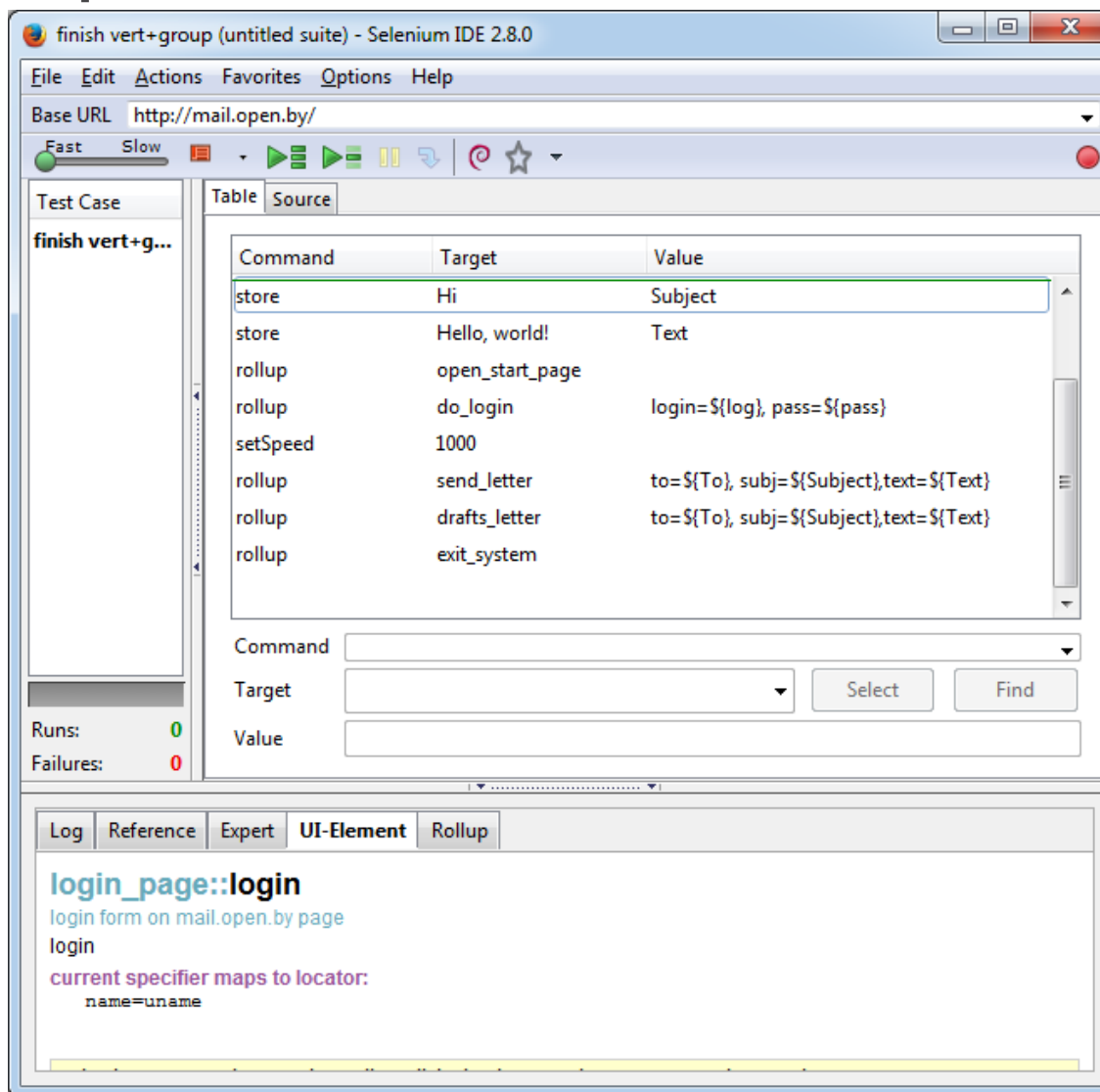
Обзор особенностей Selenium IDE

Rollup.

- ✓ Обеспечивает бизнес уровень.
- ✓ Позволяет составлять кейсы из существующих шагов, не вникая в детализацию проверок.
- ✓ Для создания требуются минимальные навыки программирования.

Обзор особенностей Selenium IDE

Rollup.



Запуск тестов написанных на Selenium IDE

- Selenium IDE

- Selenium Server

```
– java -jar selenium-server.jar -htmlSuite          *firefox  
http://www.google.com      ./MySuite.html Report.html  
– java -jar selenium-server.jar -htmlSuite          *iexplore  
http://www.google.com      ./MySuite.html Report.html
```

Report html-file

Test suite results

result: failed
totalTime: 67
numTestTotal: 4
numTestPasses: 2
numTestFailures: 2
numCommandPasses: 3
numCommandFailures: 0
numCommandErrors: 2
Selenium Version: 2.18
Selenium Revision: .0

Test Suite

Login
Send Letter To Me
Check Inbox
Logout

Login.html

Login		
open	/	
setSpeed	1000	
store	apsavin32517	log
store	32517apsavin	pass
type	mailbox__login	\${log}
type	mailbox__password	\${pass}
click	mailbox__auth__button	
pause	5000	
assertElementPresent	portal-menu__user-email	false

Send%20Letter%20To%20Me.html

Send Test Letter		
store	Test letter	subj
store	apsavin32517	log
store	Hello! Something send to me. Have a nice day.	body
assertElementPresent	HeaderBtnSentMsg	
click	HeaderBtnSentMsg	
waitForElementPresent	sentmsgab_compose_to	
waitForPageToLoad		
pause	5000	
type	sentmsgab_compose_to	\${log}@mail.ru
pause	5000	
type	sentmsgab_compose_subj	\${subj}
pause	5000	
type	sentmsgcomposeEditor	\${body}
click	//a[@class='button-a js-send' and @role='submit']	
waitForPageToLoad		
assertElementPresent	//div[contains (text(),'отправлено')]	

Преимущества и недостатки Selenium IDE

Преимущества

- ✓ Простота создания теста
- ✓ Не требует знаний языка программирования

Недостатки

- ✓ Нет возможности использовать базы данных
- ✓ Нет возможности использовать файловые хранилища
- ✓ Стандартный отчет без скриншотов

Home Work

Тестирование любого почтового веб-сервиса.