

## Лабораторна робота №5

**Тема.** Тестування web-ресурсу за допомогою засобу автоматизованого тестування Selenium.

**Мета.** Навчитися використовувати утиліту Selenium для тестування web-ресурсів.

### Теоретичні відомості

**Selenium** — це інструмент для автоматизованого управління браузерами. Найбільш популярною областю застосування Selenium є автоматизація тестування веб-додатків. Проте за допомогою Selenium можна автоматизувати будь-які інші однотипні дії, що виконуються через браузер.

Розробка Selenium підтримується виробниками популярних браузерів. Вони адаптують браузери для більш тісної інтеграції з Selenium, а іноді навіть реалізують вбудовану підтримку Selenium в браузері. Selenium є центральним компонентом багатьох інших інструментів і фреймворків автоматизації.

Selenium підтримує десктопні та мобільні браузери. Selenium дозволяє розробляти сценарії автоматизації практично на будь-якій мові програмування. За допомогою Selenium можна організовувати розподілені стенди, які складаються з сотень машин з різними операційними системами і браузерами, і виконувати хмарні обчислення.

**Selenium** – це проект, в рамках якого розробляється серія програмних продуктів з відкритим вихідним кодом (open source):

- Selenium WebDriver,
- Selenium RC,
- Selenium Server,
- Selenium Grid,
- Selenium IDE.

**Selenium WebDriver** – це програмна бібліотека для управління браузерами. Часто вживають також більш коротку назву – WebDriver.

Іноді кажуть, що це «драйвер браузера», але насправді це ціле сімейство драйверів для різних браузерів, а також набір клієнтських бібліотек на різних мовах, що дозволяють працювати з цими драйверами.

Це основний продукт, що розробляється в рамках проекту Selenium. Selenium WebDriver називається також Selenium 2.0, причина цього буде пояснена далі.

Як вже було сказано, WebDriver являє собою сімейство драйверів для різних браузерів плюс набір клієнтських бібліотек для цих драйверів на різних мовах програмування:

В рамках проекту Selenium розробляються драйвери для браузерів Firefox, Internet Explorer і Safari, а також драйвери для мобільних браузерів Android та iOS. Драйвер для браузера Google Chrome розробляється в рамках проекту Chromium, а драйвер для браузера Opera (включаючи мобільні версії) розробляється компанією Opera Software. Тому вони формально не є частиною проекту Selenium, поширюються і підтримуються незалежно. Але логічно, звичайно, можна вважати їх частиною сімейства продуктів Selenium.

Аналогічна ситуація і з клієнтськими бібліотеками – в рамках проекту Selenium розробляються бібліотеки для мов Java, .Net (C #), Python, Ruby, JavaScript. Всі інші реалізації не мають відношення до проекту Selenium, хоча, можливо, в майбутньому, якісь з них можуть влитися в цей проект.

**Selenium RC** – це попередня версія бібліотеки для управління браузерами. Абревіатура RC в назві цього продукту розшифровується як Remote Control, тобто це засіб для «віддаленого» управління браузером.

Ця версія з функціональної точки зору значно поступається WebDriver. Зараз вона знаходиться в законсервованому стані, не розвивається і навіть відомі баги не виправляють. А всім, хто стикається з обмеженнями Selenium RC, пропонується переходити на використання WebDriver.

Іноді Selenium RC називається також Selenium 1.0, тоді як WebDriver називається Selenium 2.0. Хоча насправді дистрибутив версії 2.0 включає в себе одночасно обидві реалізації – і Selenium RC, і WebDriver. А от коли вийде версія 3.0 – в ній залишиться тільки WebDriver.

З технічної точки зору WebDriver не є результатом еволюційного розвитку Selenium RC, вони базуються на абсолютно різних принципах і у них практично немає загального коду. Об'єднує їх лише цей факт, що обидві реалізації були зроблені в рамках проекту Selenium. Ну, або якщо бути зовсім точним, WebDriver спочатку був самостійним проектом, але в 2008 році відбулося злиття і зараз WebDriver являє собою основний вектор розвитку проекту Selenium.

**Selenium Server** – це сервер, який дозволяє керувати браузером з віддаленого комп'ютера через мережу. Спочатку на тому комп'ютері, де повинен працювати браузер, встановлюється і запускається сервер. Потім, на іншому, (технічно можна і на цьому ж комп'ютері) запускається програма, яка, використовуючи спеціальний драйвер RemoteWebDriver, з'єднується з сервером і відправляє йому команди. Він у свою чергу запускає браузер і виконує в ньому ці команди, використовуючи драйвер, відповідний цьому браузеру:

Selenium Server підтримує одночасно два набори команд – для нової (WebDriver) і старої версії(Selenium RC).

**Selenium Grid** – це кластер, що складається з декількох Selenium-серверів. Він призначений для організації розподіленої мережі, що дозволяє паралельно запускати багато браузерів на великій кількості машин.

Selenium Grid має топологію «зірка», тобто в його складі є виділений сервер, який називається «хаб» або «комутатор», а інші сервера називаються «ноди» або «вузли». Мережа може бути гетерогенною, тобто комутатор і вузли можуть працювати під управлінням різних операційних систем, на них можуть бути встановлені різні браузери. Одне із завдань Selenium Grid полягає в тому, щоб «підбирати» відповідний вузол, коли під час старту браузера вказуються вимоги для нього – тип браузера, версія, операційна система, архітектура процесора та ін. атрибути.

Раніше Selenium Grid був самостійним продуктом. Зараз фізично продукт один – Selenium Server, але у нього є кілька режимів запуску: він може працювати як самостійний сервер, як комутатор кластера, або як вузол кластера, це визначається параметрами запуску.

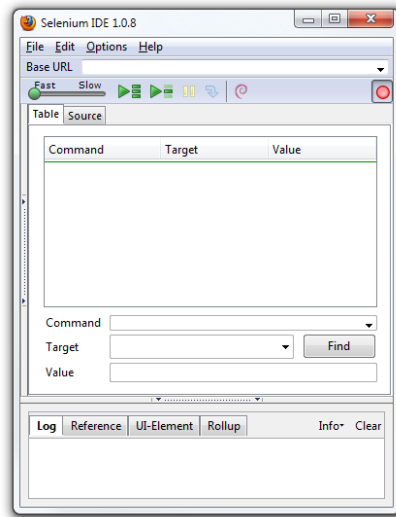
**Selenium IDE** – плагін до браузера Firefox, який може записувати дії користувача, відтворювати їх, а також генерувати код для WebDriver або Selenium RC, в якому виконуються ті ж самі дії. Загалом, це т. зв. «Selenium-рекордер».

Тестивальники, які не вміють (або не хочуть) програмувати, використовують Selenium IDE як самостійний продукт, без перетворення записаних сценаріїв в програмний код. Це, звичайно, не дозволяє розробляти досить складні тестові набори, але деяким вистачає і простих лінійних сценаріїв.

**Selenium IDE** ( Integrated Development Environment , інтегрована середовище розробки ) – це інструмент, який використовується для розробки тестових сценаріїв . Він являє собою простий у використанні додаток до браузера Firefox і, в цілому є найбільш ефективним способом розробки тестових сценаріїв. Додаток серед іншого містить контекстне меню, яке дозволяє користувачеві спочатку вибрати будь-який елемент інтерфейсу на веб-сторінці, а потім вибрати команду зі списку команд Selenium з параметрами , встановленими у відповідності з обраним елементом.

## Запуск IDE

Щоб запустити Selenium IDE, виберіть його з меню "Веб-розробка" браузера Firefox. Доповнення відкриє порожнє вікно, призначене для редагування тестових сценаріїв, а також меню для їх завантаження або збереження.



## Можливості IDE

### Панель меню

Меню "File" ("Файл") має розділи "Test Case" ("Тестовий сценарій") і "Test Suite" ("Набір тестів"). Ці розділи дозволяють створити, відкрити, зберегти чи експортувати тестовий сценарій в код на будь-якій бажаній мові програмування. Також є можливість відкрити нещодавно використаний тестовий сценарій. Аналогічні дії можна зробити і з набором тестів.

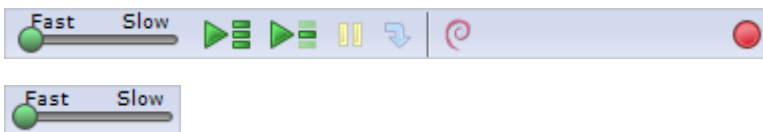
Меню "Редагування" ("Edit") дозволяє виконувати операції копіювання, вставки, видалення, скасування змін і вибору всього набору команд вашого тестового сценарію.

Меню "Опції" ("Options") дозволяє змінити настройки доповнення. У цьому меню ви можете встановити тайм-аут для певних команд, додати користувацькі розширення основного набору команд Selenium, а також налаштувати формат (мова), який буде використовуватися для збереження тестових сценаріїв.

Меню "Допомога" – це стандартне меню довідки за Firefox, тільки опція "UI-Element Documentation" ("Документація UI-Element") відноситься до Selenium IDE.

### Панель інструментів (Toolbar)

На панелі інструментів знаходяться кнопки, за допомогою яких можна управляти виконанням тестових сценаріїв, в тому числі покроковим виконанням для налагодження. Крайня права кнопка, на якій зображена червона точка - це кнопка запису.



"Управління швидкістю": дозволяє контролювати швидкість виконання тестового сценарію.



"Запустити все": Запускає весь набір тестів, у разі, якщо він містить декілька тестових сценаріїв.



"Запустити": Запускає вибраний в даний момент тест. У разі, коли завантажений лише один тест, кнопки "Запустити" і "Запустити всі" працюють ідентично.



"Пауза" / "Відновити": Зупиняють і відновлюють виконання поточного тестового сценарію.



"Крок": Дозволяє виконувати тест "по кроках", тобто по одній команді за раз. Призначено для налагодження тестових сценаріїв.



Режим "TestRunner": Ця команда дозволяє запустити тестовий сценарій в браузері, використовуючи Selenium Core TestRunner. TestRunner практично не використовується в даний час, і його підтримка, швидше за все, буде припинена. Дана кнопка потрібна для оцінки тестових сценаріїв на зворотну сумісність з TestRunner. Більшості користувачів, швидше за все, вона не знадобиться.



"Застосувати правила угруповання": Ця просунута функція дає можливість угруповання повторюваних послідовностей команд Selenium в одну дію. Більш детальну інформацію про правила групування можна знайти в "Документації з розширення UI-Element" в розділі "Допомога".



"Запис": Записує дії користувача в браузері.

## Панель тестового сценарію

У цій панелі відображається набір команд Selenium, з яких складається тестовий сценарій. На ній розташовані дві вкладки, перша з яких, "Table" ("Таблиця"), відображає команди і їх параметри.

Command	Target	Value
open	/	
waitForPageToLoad		
clickAndWait	xpath=id('menu_download')/a	
assertTitle	Downloads	
verifyText	xpath=id('mainContent')/h2	Downloads

Друга, "Source" ("Вихідний код"), відображає тестовий сценарій у форматі, який використовується для збереження файлу. За замовчанням це HTML, проте він може бути змінений на мову програмування, таку як Java або C #, або ж на скриптову мову, таку як Python. Більш детальну інформацію ви знайдете в меню "Options" ("Опції"). Вкладка "Source" ("Вихідний код") дозволяє користувачеві редагувати тестовий сценарій в текстовій формі, в тому числі використовуючи операції копіювання, вирізання та вставки.

Поля введення даних "Command" ("Команда"), "Target" ("Мета") і "Value" ("Значення") відображають обрану в даний момент команду, а також її параметри. За допомогою цих полів можна модифікувати обрану команду. Значення першого параметра, описаного у вкладці "Reference" ("Довідка") нижній панелі, вказується в полі "Мета". Якщо в "Довідці" описаний також другий параметр, то він завжди вказується в полі "Значення".

Command	clickAndWait	
Target	xpath=id('menu_download')/a	Find
Value		

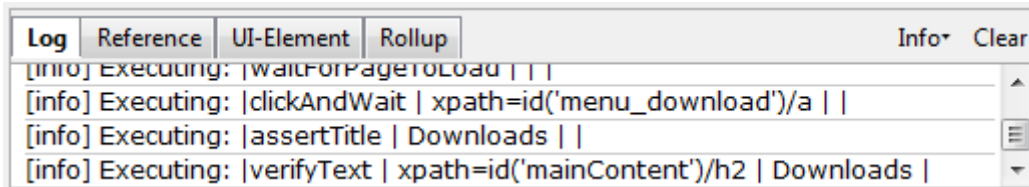
Коли ви починаєте вводити текст в поле "Команда", з'являється випадаюче меню, що містить команди, які починаються з вже введених символів і ви можете вибрати з цього списку необхідну.

## Вкладки “Log”, “Reference”, “UI-Element”, “Rollup” (“Лог”, “Довідка”, “UI-Element”, “Групування”)

Нижня панель використовується для чотирьох різних функцій : логу , довідки , документація UI - Element і групування - залежно від того , яка вкладка вибрана.

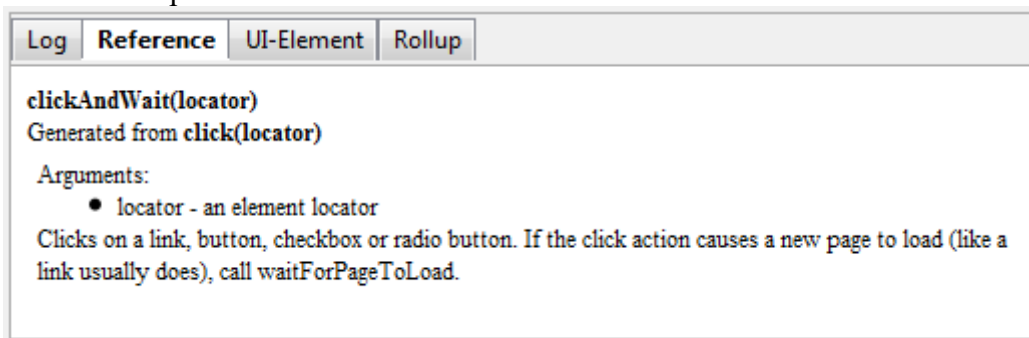
### Лог

Коли користувач запускає тестовий сценарій , в панелі даної вкладки автоматично з'являються інформаційні повідомлення і повідомлення про помилки , навіть якщо користувач не вибрав вкладку " Лог" до цього. Ці повідомлення часто корисні при налагодженні тестового сценарію . Зверніть увагу на кнопку " Очистити " ( " Clear " ) , яка дозволяє очистити лог . Також зверніть увагу на те , що кнопка "Інформація " ( "Info " ) є випадаючим меню , в якому можна вибрати різний ступінь деталізації записів в лог.



### Довідка

Вкладка " Довідка " вибирається за замовчуванням кожен раз , коли користувач вводить або модифікує команди і параметри в табличному режимі. У цьому режимі панель " Довідка " відображає інформацію про поточну команду. При введенні і зміні команд в табличному режимі або режимі правки вихідного коду вкрай важливо впевнитися , що параметри , зазначені в полях " Мета " і " Значення " , відповідають зазначеним у списку параметрів на панелі " Довідка " . Кількість, порядок і типи зазначених параметрів повинні відповідати кількості , порядку і типам параметрів , які є допустимими для даної команди . У разі невідповідності будь-якого з трьох пунктів команда не буде виконана коректно



### Практичне завдання:

Створити 10 автоматизованих тестів, використовуючи плагін Selenium IDE для браузера Firefox для тестування сайту згідно варіанту.

1	<a href="https://www.udacity.com/">https://www.udacity.com/</a>
2	<a href="https://www.udemy.com/">https://www.udemy.com/</a>
3	<a href="https://www.edx.org/">https://www.edx.org/</a>
4	<a href="https://www.utest.com/">https://www.utest.com/</a>
5	<a href="http://prometheus.org.ua/">http://prometheus.org.ua/</a>
6	<a href="https://coursera.org/">https://coursera.org/</a>
7	<a href="https://www.codecademy.com">https://www.codecademy.com</a>
8	<a href="http://www.futurelearn.com">www.futurelearn.com</a>
9	<a href="https://www.open2study.com/">https://www.open2study.com/</a>
10	<a href="https://www.datacamp.com/">https://www.datacamp.com/</a>