

## Лабораторная работа № 3

по дисциплине

### «Математичні методи та технології тестування та верифікації програмного забезпечення»

**Тема:** Изучение технологии автоматизированного тестирования записи и воспроизведения (Record and playback) на примере использования интегрированной среды разработки Selenium IDE.

**Цель работы:** Знакомство с технологией автоматизированного тестирования записи и воспроизведения (Record and playback) на примере использования интегрированной среды разработки Selenium IDE. Получение практических навыков для составления тест-сюитов для тестирования веб-приложений.

#### Содержание отчета

1. Название и цель работы.
2. Выполнение индивидуального задания, которое находится в конце работы.
3. Выводы.
4. Ответы на контрольные вопросы.

#### Методические указания

Автоматизация тестирования (test automotion) – набор техник, подходов и инструментальных средств, которые позволяют исключить человека из выполнения некоторых задач в процессе тестирования.

Инструментальное средство автоматизированного тестирования (test automation tool)– программа или набор программ, которая позволяет создавать, редактировать, отлаживать и выполнять автоматизированные тесты, а также собирать статистику их выполнения.

Список технологий автоматизированного тестирования приведен в таблице 1.

Самая простая технология в использовании – это технология записи и воспроизведения. Использование этой технологии сводится к шагам:

1. Тестировщик вручную выполняет тест-кейс, а средство автоматизации его записывает.
2. Результаты записи представляются в виде кода на высокоуровневом языке программирования (в некоторых случаях специально разработанном).
3. Тестировщик редактирует полученный код.
4. Готовый код автоматизированного теста выполняется для проведения тестирования в автоматизированном режиме.

Selenium IDE (integrated development environment) - это инструмент, используемый для разработки тестовых сценариев. Он представляет собой простое в использовании дополнение к браузеру Firefox.

Для установки Selenium IDE пройдите по ссылке:

<https://addons.mozilla.org/en-US/firefox/addon/selenium-ide/> и нажмите кнопку добавить (рисунок 1).

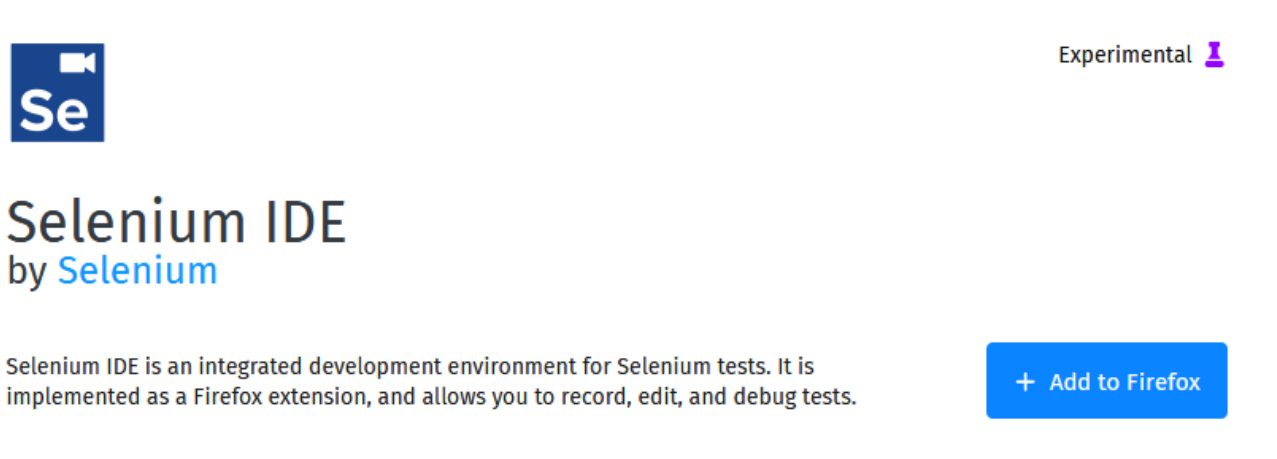


Рисунок 1 – Добавление Selenium IDE в браузер Firefox

После установки Selenium IDE доступен в браузере Firefox рисунок 3.

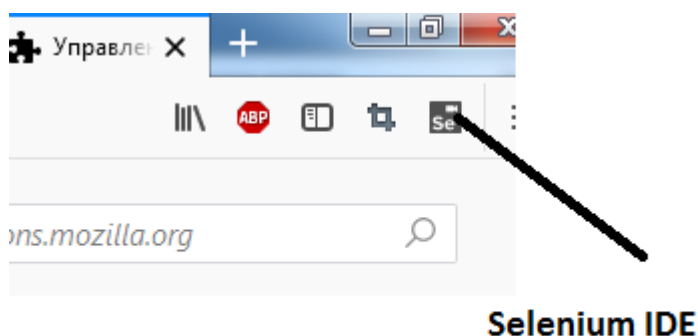


Рисунок 2 – Добавление Selenium IDE в браузер Firefox.

Приложение имеет вид, представленный на рисунке 3.

Подробное описание работы с приложением можно найти, пройдя по ссылке <https://www.seleniumhq.org/docs/>, русскоязычный вариант selenium2.ru.



- после нажатия на кнопку приложение находится в режиме записи, тестирующий выполняет все действия, которые есть в тест-кейсе. Запись необходимо приостановить, если все действия в тест-кейсе завершились.

Но при тестировании проверку нельзя заменить действием, проверки необходимо дописывать вручную.

Тест-кейсы можно сохранить в html-формате (рисунок 4).

Здесь выполняется такая последовательность шагов:

1. Открыть страницу <https://www.olx.ua/>.
2. Проверить, что на странице находится элемент «Мой профиль».
3. Проверить, что на странице находится элемент «Популярные запросы».

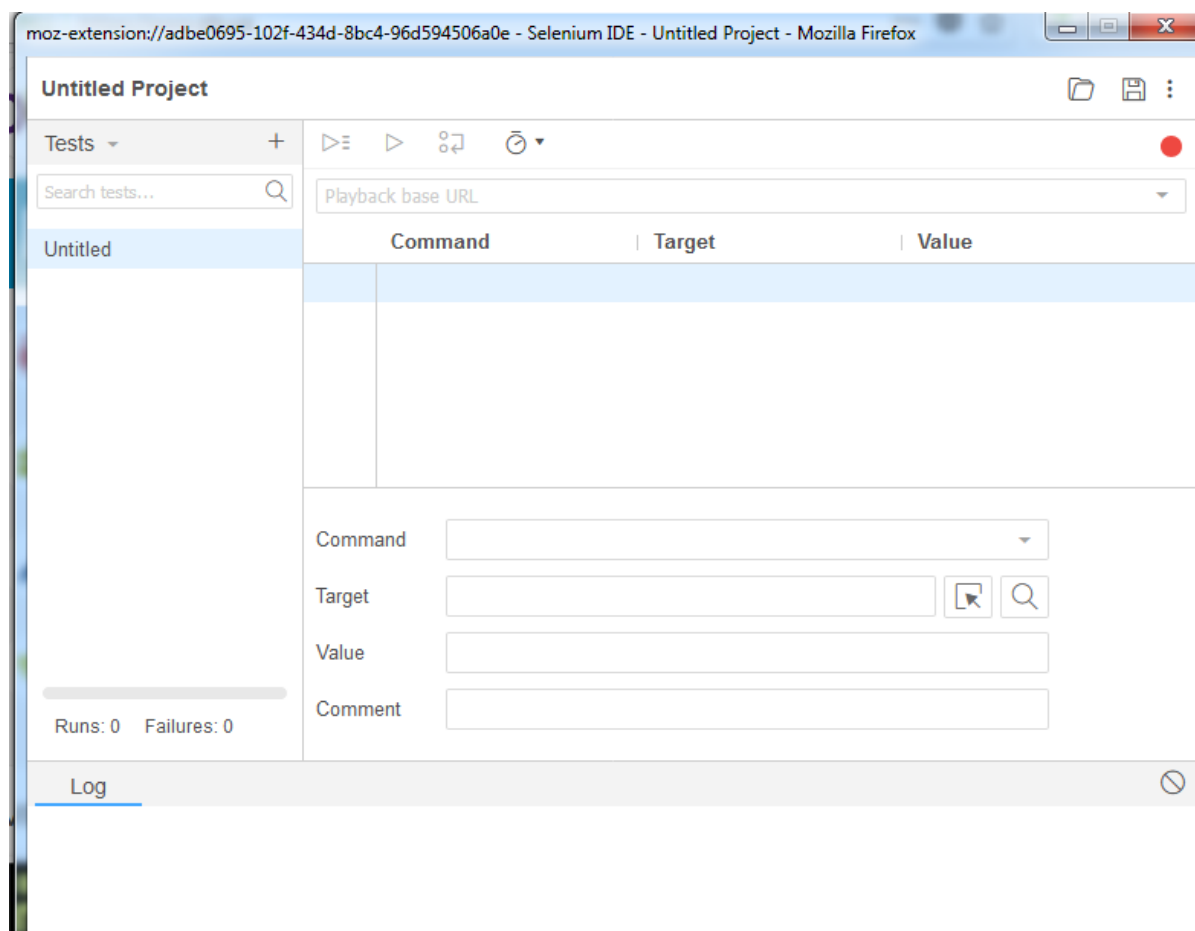


Рисунок 3 – Вид приложения Selenium IDE в браузере Firefox.

1_01		
open	https://www.olx.ua/uk/changelang/?lang=ru&l=https%3A%2F%2Fwww.olx.ua	
verifyText	css=span.link.inlblk > strong	Мой профиль
verifyText	//div[@id='lastwrapper']/div/div[2]/div/div[2]/ul/li[5]/a/span	Популярные запросы

Рисунок 4 – тест-кейс, сохраненный в html-формате.

**Таблица 1 - Технологии автоматизированного тестирования**

№	Подход	Суть	Преимущества	Недостатки
1	Частные решения	Для решения каждой отдельной задачи пишется отдельная программа	Быстро, просто	Нет системности, много времени уходит на поддержку. Почти невозможно повторное решение.
2	Тестирование под управлением данными (DDT)	Из тест-кейса выносятся входные данные и ожидаемые результаты	Один и тот же тест-кейс можно повторять многократно с разными данными	Логика тест-кейса по-прежнему строго определяется внутри, а потому для ее изменения
3	Тестирование под управлением ключевыми словами (KDT)	Из тест-кейса во вне выносятся описание его поведения	Концентрация на высокоуровневых действиях. Данные и особенности поведения хранятся вовне и могут быть изменены без изменения кода тест-кейса.	Сложность выполнения низкоуровневых операций.
4	Использование фреймворков	Конструктор, позволяющий использовать остальные подходы	Мощность и гибкость	Относительная сложность (особенно в создании фреймворка)
5	Запись и воспроизведение (Record & Playback)	Средство автоматизации записывает действия тестировщика и может воспроизвести их, управляя тестируемым приложением	Простота, высокая скорость создания тест-кейсов	Крайне низкое качество, линейность, неподдерживаемость тест-кейсов. Требуется серьезная доработка полученного кода
6	Тестирование под управлением поведением (BBT)	Развитие идей тестирования под управлением данными и ключевыми словами. Отличие – в концепции на бизнес сценариях без выполнения мелких проверок.	Высокое удобство проверки высокоуровневых пользовательских сценариев	Такие тест-кейсы пропускают большое количество функциональных и нефункциональных дефектов, а потому должны быть дополнены классическими низкоуровневыми тест-кейсами

На рисунке 5 видно, что в тест-сьюте с названием TAL\_SeleniumIDE\_Melkozerova находится 7 тест-кейсов (1\_01, 1\_02...2\_02), из которых 7 тестов были запущены, 6 пройдено успешно, 1 тест-кейс не прошел.

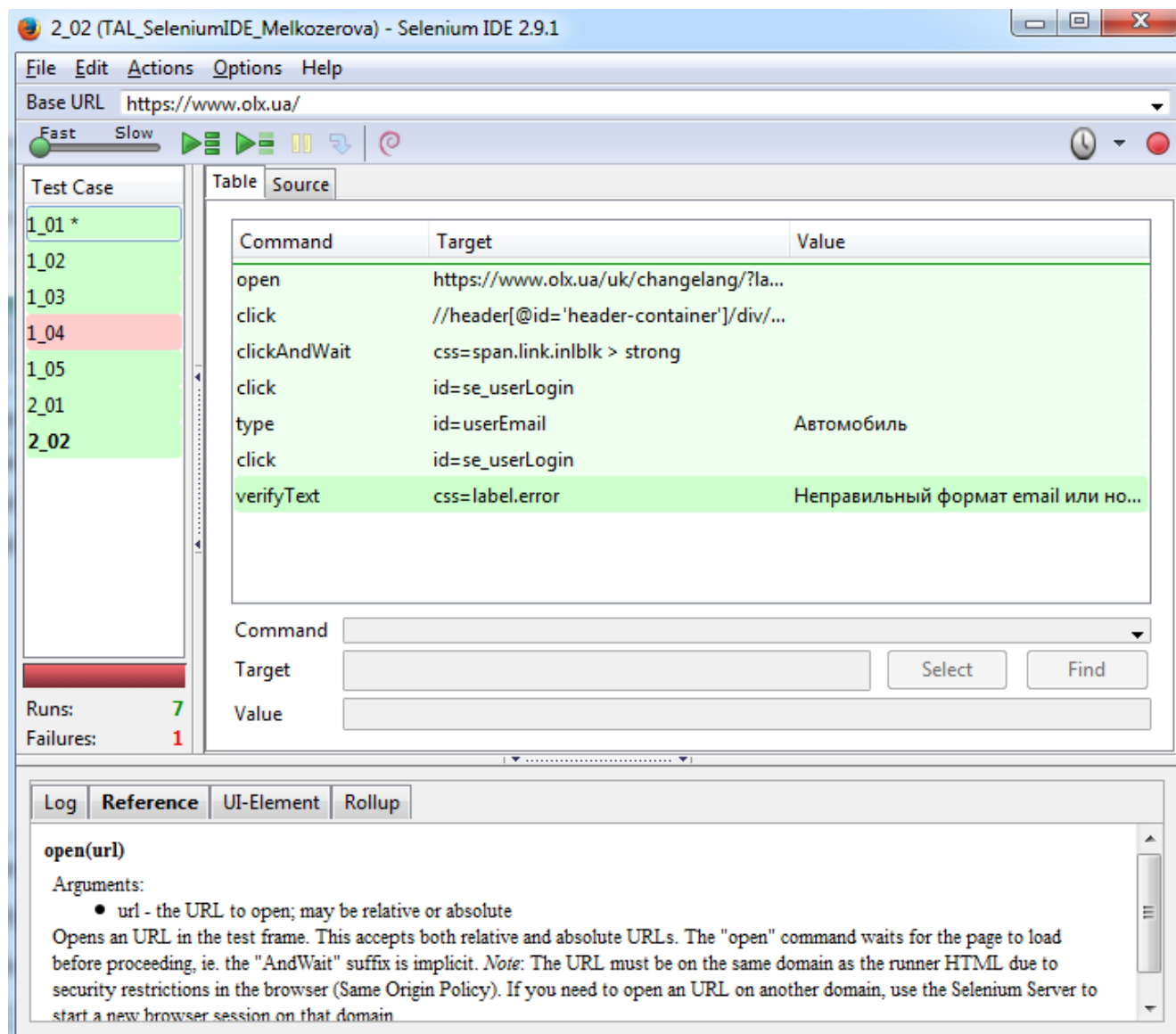


Рисунок 5 – Вид приложения Selenium IDE

## Язык команд Selenium IDE

Язык Selenium – это набор команд, которые составляют ваши тесты. Последовательность команд называется тестовым сценарием.

Команды Selenium бывают трех видов: действия (actions), считыватели (accessors) и проверки (assertions).

- Действия – это команды, которые обычно управляют состоянием приложения. Они совершают действия вроде «щелкнуть по той ссылке» или «выбрать эту операцию». Если действие не может быть выполнено, либо выполняется с ошибкой, то текущий тест прерывается. К большей части действий можно добавить «AndWait» («подождать»), «ClickAndWait». Этот суффикс

сообщает Selenium, что действие принудит браузер совершить запрос к серверу и что Selenium должен дожидаться загрузки новой страницы.

- Считыватели анализируют состояние приложения и сохраняют результаты в переменные, к примеру, команда «storeTitle». «Считыватели» также используют для автоматической генерации проверок.

- Проверки похожи на «Считыватели», но они проверяют соответствие состояния приложения ожидаемому. Например, удостоверится, что заголовок страницы определенный и проверить, что вон тот чекбокс отмечен.

Команды, которые выполняют проверку:

- assert (строгая проверка) – при ошибке выполнения такой тест прерывается;
- verify (нестрогая проверка) – при ошибке выполнения такой проверки фиксируется факт возникновения проблемы и тест продолжается;

- waitfor (ожидание) - команды ожидания сообщают Selenium о том, что необходимо дождаться выполнения определенных условий. Если условие выполняется, то тест без остановки продолжает работу. А если не выполняется, то тест будет остановлен до тех пор, пока оно не выполнится. А если такое условие так и не выполнится в течение установленного там-аута, то тест будет считаться проваленным.

Команды состоят из команды и параметров. Параметры не всегда необходимы, это зависит от самой команды.

Справка по командам описывает требования к параметрам каждой команды.

Параметры делятся на:

- локаторы для идентификации элементов пользовательского интерфейса на странице;

- текстовые шаблоны, для проверки с помощью команд verify и assert ожидаемого содержимого на странице;

- текстовые шаблоны или переменные языка для ввода текста в поля ввода или для выбора элемента из выпадающего списка.

## Локаторы в Selenium IDE.

Target – поле, в котором указывается с каким элементом будет выполняться операция.

Типы локаторов:

- identifier = ID или name;

- id = ID;

- DOM (document object model) – такие локаторы позволяют определить элемент на основе JavaScript, который оперирует объектной моделью документа;

- link – такие локаторы позволяют определить ссылку на основе ее части;

- CSS - позволяют определить элемент на основе выражения каскадной таблицы стилей (cascading style sheets);

- XPath (XML path language) – язык запросов к элементам XML или HTML документа. Строка XPath - это фактически путь к элементу в дереве, где каждый уровень разделяется косой чертой «/».

Часто используемые команды представлены в таблице 2.

Таблица 2 – Наиболее часто используемые команды в Selenium IDE, примеры использования

open	открывает страницу по заданному URL (Uniform Resource Locator)
<div> <div>Command</div> <div>open</div> </div> <div> <div>Target</div> <div>https://www.olx.ua/uk/changelang/?lang=ru&amp;l=https%3A%3F</div> <div>Select</div> <div>Find</div> </div> <div> <div>Value</div> <div></div> </div>	
click	Производит нажатие на ссылку, кнопку, чек-бокс, радио-кнопку.
<div> <div>Command</div> <div>click</div> </div> <div> <div>Target</div> <div>id=submit-searchmain</div> <div>Select</div> <div>Find</div> </div> <div> <div>Value</div> <div></div> </div>	
type	запись текста в текстовое поле
<div> <div>Command</div> <div>type</div> </div> <div> <div>Target</div> <div>id=headerSearch</div> <div>Select</div> <div>Find</div> </div> <div> <div>Value</div> <div>Игрушки</div> </div>	
verifyText	используется, если необходимо проверить как текст, так и соответствующий ему элемент интерфейса пользователя
<div> <div>Command</div> <div>verifyText</div> </div> <div> <div>Target</div> <div>//div[@id='lastwrapper']/div/div[2]/div/div[2]/ul/li[5]/a/spa</div> <div>Select</div> <div>Find</div> </div> <div> <div>Value</div> <div>Популярные запросы</div> </div>	
verifyTextPresent	используется для проверки определенного текста в любом месте страницы
<div> <div>Command</div> <div>verifyTextPresent</div> </div> <div> <div>Target</div> <div>Популярные запросы</div> <div>Select</div> <div>Find</div> </div> <div> <div>Value</div> <div></div> </div>	
storeText	используется для сохранения текста в переменную, в качестве переменной можно использовать только буквы и цифры (a1 – это произвольная переменная).
<div> <div>Command</div> <div>storeText</div> </div> <div> <div>Target</div> <div>//section[@id='body-container']/div[3]/div/div/table/tbody</div> <div>Select</div> <div>Find</div> </div> <div> <div>Value</div> <div>a1</div> </div>	
<div> <div>Command</div> <div>verifyNotText</div> </div> <div> <div>Target</div> <div>//section[@id='body-container']/div[3]/div/div/table/tbody</div> <div>Select</div> <div>Find</div> </div> <div> <div>Value</div> <div>\${a1}</div> </div>	

waitForPageToLoad	ожидание, пока загрузится страница. Автоматически вызывается при использовании команды clickAndWait.
-------------------	--

Есть возможность переконвертировать тест в различные языки программирования.

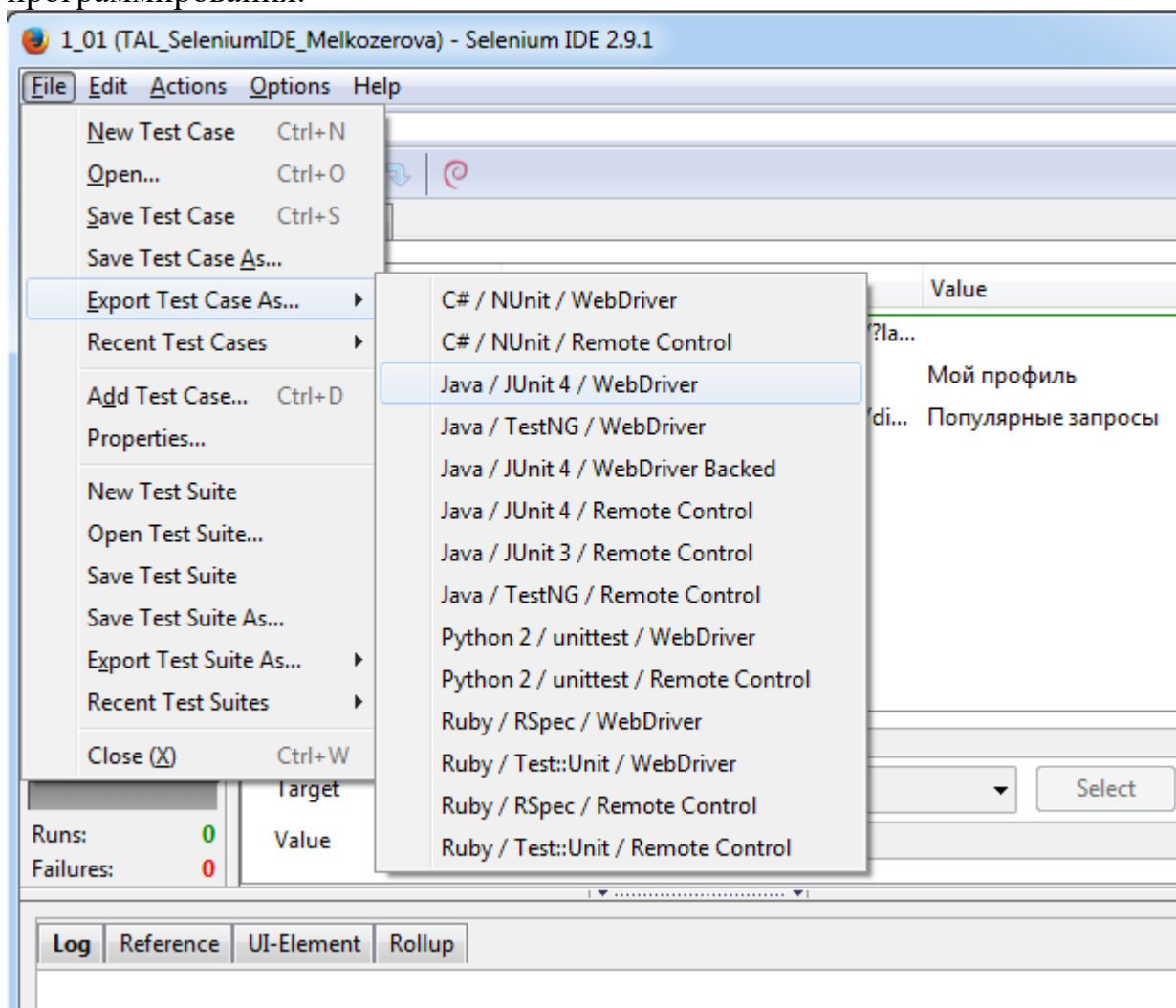


Рисунок 6

@Test

```
private void testTestThird() throws Exception{
    WebElement el=driver.findElement(By.name("login"));
    el.sendKeys("Melkozerova");
    WebElement el1=driver.findElement(By.name("pass"));
    el1.sendKeys("qwer123qwer123");
    WebElement
    el3=driver.findElement(By.xpath("html/body/div[3]/div[3]/div[3]/div[2]/div[1]/div[
    3]/form/p/input"));
    el3.click();
    WebElement
    el4=driver.findElement(By.xpath("html/body/div[1]/div[4]/ul/li[2]/a"));
    el4.click();
    WebElement el5=driver.findElement(By.id("to"));
    el5.sendKeys("Melkozerova@i.ua");
    WebElement el6=driver.findElement(By.name("subject"));
    el6.sendKeys("Subject");

    WebElement el7=driver.findElement(By.xpath("//*[id='text']"));
    el7.sendKeys("Body");
}
```



```

WebElement el8=driver.findElement(By.name("send"));
el8.click();
WebElement
el9=driver.findElement(By.xpath("html/body/div[1]/div[5]/div[2]/div/div/div[2]/div
[2]/div[3]/ul/li[1]/a"));
el9.click();
WebElement
el10=driver.findElement(By.xpath(".*[@id='mesgList']/form/div[1]/a/span[2]"));
System.out.println(el10.getText());
String Correct = el10.getText();
Assert.assertEquals(Correct, "Ольга Мелкозерова");
}

```

### Задание

Используя ссылку для своего варианта (таблица 1) выполнить в отдельных тест кейсах проверки.

1. Страница содержит строки «Фильтры», «Сортировка».
2. Перейти на украинскую версию и проверить, что вы находитесь на ней.
3. Убедиться, что вы на первой странице.
4. Вернуться на главную страницу.
5. Выполнить на главной странице поиск (слово в таблице 1). Убедиться, что вы выполнили правильный поиск.
6. Выполнить другой поиск, любой другой поиск. Убедиться, что первые три результата не совпадают с прошлой категорией.

Отчет предоставить в виде тест-сьюта, который содержит тест-кейсы с файлами 1\_01.html, 1\_02.html,..., 1\_06. html.

Таблица 1 – Страницы для выполнения индивидуального задания

Вариант	Ссылка на страницу	Слово для поиска, пункт 5 задания
1	<a href="https://pn.com.ua/ct/1003/">https://pn.com.ua/ct/1003/</a>	мышка
2	<a href="https://pn.com.ua/ct/1004/">https://pn.com.ua/ct/1004/</a>	ноутбук
3	<a href="https://pn.com.ua/ct/1132/">https://pn.com.ua/ct/1132/</a>	машина
4	<a href="https://pn.com.ua/ct/1047/">https://pn.com.ua/ct/1047/</a>	холодильник
5	<a href="https://pn.com.ua/ct/1044/">https://pn.com.ua/ct/1044/</a>	клавиатура
6	<a href="https://pn.com.ua/ct/1041/">https://pn.com.ua/ct/1041/</a>	игрушки
7	<a href="https://pn.com.ua/ct/1042/">https://pn.com.ua/ct/1042/</a>	подушки
8	<a href="https://pn.com.ua/ct/1043/">https://pn.com.ua/ct/1043/</a>	одеяла
9	<a href="https://pn.com.ua/ct/1062/">https://pn.com.ua/ct/1062/</a>	колонки
10	<a href="https://pn.com.ua/ct/1022/">https://pn.com.ua/ct/1022/</a>	видеокамеры
11	<a href="https://pn.com.ua/ct/1065/">https://pn.com.ua/ct/1065/</a>	велосипед
12	<a href="https://pn.com.ua/ct/1232/">https://pn.com.ua/ct/1232/</a>	самокат
13	<a href="https://pn.com.ua/ct/2097/">https://pn.com.ua/ct/2097/</a>	кубик
14	<a href="https://pn.com.ua/ct/1238/">https://pn.com.ua/ct/1238/</a>	диван

15	<a href="https://pn.com.ua/ct/2120/">https://pn.com.ua/ct/2120/</a>	стул
16	<a href="https://pn.com.ua/ct/1201/">https://pn.com.ua/ct/1201/</a>	стол
17	<a href="https://pn.com.ua/ct/1221/">https://pn.com.ua/ct/1221/</a>	комод
18	<a href="https://pn.com.ua/ct/1236/">https://pn.com.ua/ct/1236/</a>	вешалка
19	<a href="https://pn.com.ua/ct/2121/">https://pn.com.ua/ct/2121/</a>	двери
20	<a href="https://pn.com.ua/ct/1063/">https://pn.com.ua/ct/1063/</a>	окна
21	<a href="https://pn.com.ua/ct/1225/">https://pn.com.ua/ct/1225/</a>	МОТОЦИКЛЫ
22	<a href="https://pn.com.ua/ct/2074/">https://pn.com.ua/ct/2074/</a>	шины
23	<a href="https://pn.com.ua/ct/1067/">https://pn.com.ua/ct/1067/</a>	диски
24	<a href="https://pn.com.ua/ct/2089/">https://pn.com.ua/ct/2089/</a>	футболки
25	<a href="https://pn.com.ua/ct/2193/">https://pn.com.ua/ct/2193/</a>	носки
26	<a href="https://pn.com.ua/ct/2163/">https://pn.com.ua/ct/2163/</a>	чемодан