

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. Н. КАРАЗІНА
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК
КАФЕДРА БЕЗПЕКИ ІНФОРМАЦІЙНИХ СИСТЕМ І ТЕХНОЛОГІЙ

Лабораторна робота №6
з навчальної дисципліни
**«Математичні методи та технології тестування та верифікації
програмного забезпечення»**

Виконала:

студентка групи КС-23

Рузудженк С.Р

Перевірив:

доцент

Нарєжній О. П.

Харків – 2019

Лабораторна робота №6
з навчальної дисципліни
«Математичні методи та технології тестування та верифікації
програмного забезпечення»

Тема: Selenium WebDriver

Мета: Selenium WebDriver.

Хід роботи

Запускаємо *IntelliJIDEA*, створюємо проект *Maven*. Додаємо відповідні залежності у файл *pom.xml*.

```
<dependency>
  <groupId>org.seleniumhq.selenium</groupId>
  <artifactId>selenium-java</artifactId>
  <version>3.14.0</version>
</dependency>
```

Створюємо клас *WebTesting*, а в ньому - методи та поля, необхідні нам для подальшого використання основного класу. При цьому ми робимо позначки типу *@Before*, *@After*, які визначають час проведення цих тестів. Таким чином, методи з анотацією *@Before* будуть виконуватися перед кожною тестовою функцією з анотацією *@Test*. Аналогічно з *@After*.

```

import org.junit.After;
import org.junit.Before;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;

import java.util.ArrayList;
import java.util.List;

public class WebDriverSetting {

    static WebDriver driver;
    List<WebElement> webElementList;
    ArrayList<String> wblmt;
    ArrayList<String> wblmt1;
    List<WebElement> webElementList1;

    @Before
    public void setUp(){
        System.setProperty("webdriver.chrome.driver",
            "C:\\\\Users\\Anna\\Desktop\\it\\chromedriver_win32\\chromedriver.exe");
        this.driver = new ChromeDriver();
        System.out.println("Test started");
    }

    @After
    public void quit(){
        driver.quit();
        System.out.println("Test ended");
    }
}

```

У першому тесті ми заходимо на сторінку <https://pn.com.ua/>, обираємо категорію «Диваны», потім першого популярного виробника і перевіряємо, що всі відфільтровані товари відносяться до нього. Так, у нижче приведеному коді ми шукаємо за допомогою *xpath* необхідні елементи на сторінці та взаємодіємо з ними.

```

@Test
public void FirstTest(){
    System.out.println("\t\tThe 1-st test");
    driver.get("https://pn.com.ua/");
    System.out.println("Processing of test");
    driver.findElement
        (By.xpath(".*[@id='column-center']/section/div[5]/ul/li[1]/a")).click();
    driver.findElement
        (By.xpath(".*[@id='producers-filter-block']/div[3]/span[3]/a")).click();
    wblmt = new ArrayList<String>();
    webElementList = driver.findElements
        (By.xpath(".*[@id='column-center']" +
            "/section/div[3]/li/*div[2]/div[1]/*[contains(text(), 'AMF ')]"));
    for(int i=0; i<webElementList.size(); i++){
        webElementList.get(i).getText();
        System.out.println(webElementList.get(i).getText());
        wblmt.add(webElementList.get(i).getText());
    }
    System.out.println("\nCOMPARING\n");
    boolean condition = true;
    for (int i = 0; i < webElementList.size(); i++) {
        System.out.println(wblmt.get(i));
        condition = (wblmt.get(i).equals(webElementList.get(i).getText()));
    }
    Assert.assertTrue( message: "Something have gone wrong!", condition);
}

```

Першим популярним виробником є компанія AMF, тому і пошук елементів проводиться командою `contains(text(), 'AMF ')`. За результатами, виведеними у консолі, ми можемо впевнитись, що елементи на сторінці дійсно належать виробнику AMF.

Test started	COMPARING
The 1-st test	
Processing of test	AMF Ньюс
AMF Ньюс	AMF Тедди (Teddy)
AMF Тедди (Teddy)	AMF Алекс
AMF Алекс	AMF SMS (СМС) 1.2
AMF SMS (СМС) 1.2	AMF Верона угловой
AMF Верона угловой	AMF Элиза
AMF Элиза	AMF Флекс
AMF Флекс	AMF Арго (ППУ)
AMF Арго (ППУ)	AMF Гера
AMF Гера	AMF Одиссей (ППУ)
AMF Одиссей (ППУ)	AMF Герман
AMF Герман	AMF Одиссей
AMF Одиссей	AMF Аякс (ППУ)
AMF Аякс (ППУ)	AMF Виктор угловой
AMF Виктор угловой	AMF Юниор
AMF Юниор	AMF Эстер
AMF Эстер	AMF Стелла
AMF Стелла	AMF Скай
AMF Скай	AMF SMS (СМС) 0.8
AMF SMS (СМС) 0.8	AMF SMS (СМС) 1.4
AMF SMS (СМС) 1.4	AMF Арабика (Arabika)
AMF Арабика (Arabika)	AMF Арго
AMF Арго	AMF Токио
AMF Токио	AMF Гермес (ППУ)
AMF Гермес (ППУ)	Test ended

Популярные производители | все

AMF 41	BRW 125	IKEA 890	Matroluxe 106
Tia-sport 8	Зенит 83	ЛИВС 350	Маяк 26

У другому тесті ми заходимо на сторінку <https://pn.com.ua/>, обираємо категорію «Диваны», потім першого популярного виробника і перевіряємо, що кількість відфільтрованих товарів відповідає числу, зазначеному біля назви бренду до підключення фільтра.

AMF 41

```

@Test
public void SecondTest() {
    System.out.println("\t\tThe 2-nd test");
    driver.get("https://pn.com.ua/");
    System.out.println("Processing of test");
    driver.findElement(By.xpath(".*[@id='column-center']/section/div[5]/ul/li[1]/a")).click();
    int sum=0;
    int expectedSum = Integer.parseInt(driver.findElement
        (By.xpath(".*[@id='producers-filter-block']/div[3]/span[3]/small")).getText());
    driver.findElement(By.xpath(".*[@id='producers-filter-block']/div[3]/span[3]/a")).click();
    wblmt = new ArrayList<String>();
    wblmt1 = new ArrayList<String>();
    webElementList = driver.findElements
        (By.xpath(".*[@id='column-center']/section/div[3]/*li*/div[2]/div[1]" +
            "/*[contains(text(), 'AMF ')]"));
    for(int i=0; i<webElementList.size(); i++){
        webElementList.get(i).getText();
        System.out.println((i+1)+" " + webElementList.get(i).getText());
        wblmt.add(webElementList.get(i).getText());
        sum++;
    }
    driver.findElement
        (By.xpath(".*[@id='column-center']/section/div[1]/div[2]/ul/li[4]/a/*")).click();
    webElementList1 = driver.findElements
        (By.xpath(".*[@id='column-center']/section/div[3]/*li*/div[2]/div[1]" +
            "/*[contains(text(), 'AMF ')]"));
    System.out.println("\t\tGO TO THE NEXT PAGE");
    for(int j=0; j<webElementList1.size(); j++) {
        webElementList1.get(j).getText();
        System.out.println((sum+1) + " " + webElementList1.get(j).getText());
        wblmt1.add(webElementList1.get(j).getText());
        sum++;
    }
    System.out.println("THE AMOUNT OF ELEMENTS IS: "+sum);
    boolean condition = true;
    if(expectedSum!=sum)
        condition = false;
    System.out.println("Expected sum of elements is: "+expectedSum);
    Assert.assertTrue( message: "Something have gone wrong", condition);
}

```

За результатами тесту ми маємо отримати 41 товар бренду AMF.

AMF в Харькове (найден 41 товар)

У даному випадку ми прописуємо перехід на наступну сторінку, а також додаємо змінну, яка буде рахувати кількість наших елементів. Результат можемо спостерігати на консолі.

```

Test started
    The 2-nd test
Processing of test
1 AMF Ньюс
2 AMF Тедди (Teddy)
3 AMF Алекс
4 AMF SMS (СМС) 1.2
5 AMF Верона угловой
6 AMF Элиза
7 AMF Флекс
8 AMF Арго (ППУ)
9 AMF Гера
10 AMF Одиссей (ППУ)
11 AMF Герман
12 AMF Одиссей
13 AMF Аякс (ППУ)
14 AMF Виктор угловой
15 AMF Юниор
16 AMF Эстер
17 AMF Стелла
18 AMF Скай
19 AMF SMS (СМС) 0.8
20 AMF SMS (СМС) 1.4
21 AMF Арабика (Arabika)
22 AMF Арго
23 AMF Токио
24 AMF Гермес (ППУ)
25 AMF Джейн
26 AMF SMS (СМС) 1.0
27 AMF Хеппи 1.3
28 AMF Хеппи 1.5
29 AMF Ролан угловой
30 AMF Аякс
31 AMF Людвиг
32 AMF Роберт
33 AMF Тони
34 AMF Барселона
35 AMF Гермес
36 AMF Эрик
37 AMF Свити 1.3
38 AMF Хеппи 1.7
39 AMF Свити 1.7
40 AMF Вена Бонд (Vienna)
41 AMF Свити 1.5
    GO TO THE NEXT PAGE
    THE AMOUNT OF ELEMENTS IS: 41
    Expected sum of elements is: 41
    Test ended


```

У третьому тесті ми заходимо на сторінку <https://pn.com.ua/>, обираємо категорію «Холодильники», додаємо у зрівняння перші два холодильники, переходимо на сторінку зрівняння товарів. Перевіряємо, що на сторінці відображено два товари.

```

@Test
public void ThirdTest(){
    System.out.println("\t\tThe 3-rd test");
    driver.get("https://pn.com.ua/");
    System.out.println("Processing of test");
    driver.findElement(By.xpath(".*[@id='column-center']/section/div[3]/ul/li[3]/a")).click();
    driver.findElement
        (By.xpath(".*[@id='column-center']/section/div[3]/article[1]" +
            "/div[2]/div[2]/span[2]/a")).click();
    driver.findElement
        (By.xpath(".*[@id='column-center']/section/div[3]/article[2]" +
            "/div[2]/div[2]/span[2]/a")).click();
    driver.findElement
        (By.xpath(".*[@id='column-center']/section/div[1]/div[1]/a")).click();
    System.out.println("The elements on the page 'Сравнения':");
    int sum=0;
    wblmt = new ArrayList<String>();
    webElementList = driver.findElements(By.xpath(".*[@id='spec-tbl']/tbody/tr[3]/td/*"));
    for (int i=0; i<webElementList.size(); i++) {
        webElementList.get(i).getText();
        System.out.println((i+1)+" "+webElementList.get(i).getText());
        wblmt.add(webElementList.get(i).getText());
        sum++;
    }
    int expectedResult=2;
    Assert.assertEquals( message: "Something have gone wrong", expectedResult, sum);
}

```

 **добавить в сравнение**



Сравнение (2 модели)

Таким чином результатом прогону третього тесту буде вивід на консоль усіх елементів, які знаходяться на сторінці порівняння товарів.

```
Test started
    The 3-rd test
Processing of test
The elements on the page 'Сравнения':
1 Холодильник, 311 л, 2 двери, 2 камеры
2 Холодильник, 384 л, 2 двери, 2 камеры
Test ended
```

У четвертому тесті ми заходимо на сторінку <https://pn.com.ua/>, обираємо категорію «Холодильники», додаємо у зрівняння перші чотири холодильники. Переходимо на сторінку порівняння товарів та переконуємося, що тут знаходяться чотири елемента. Видаляємо один товар і переконуємося, що на сторінці залишилося три елемента.

```
@Test
public void FourthTest() {
    System.out.println("\t\tThe 4-th test");
    driver.get("https://pn.com.ua/");
    driver.findElement(By.xpath(".*[@id='column-center']/section/div[3]/ul/li[3]")).click();
    driver.findElement
        (By.xpath(".*[@id='column-center']/section/div[3]/article[1]/div[2]" +
            "/div[2]/span[2]/a")).click();
    driver.findElement
        (By.xpath(".*[@id='column-center']/section/div[3]/article[2]/div[2]" +
            "/div[2]/span[2]/a")).click();
    driver.findElement
        (By.xpath(".*[@id='column-center']/section/div[3]/article[3]/div[2]" +
            "/div[2]/span[2]/a")).click();
    driver.findElement
        (By.xpath(".*[@id='column-center']/section/div[3]/article[4]/div[2]" +
            "/div[2]/span[2]/a")).click();
    driver.findElement(By.xpath(".*[@id='column-center']/section/div[1]/div[1]")).click();
    boolean condition = true;
    wblmt = new ArrayList<String>();
    System.out.println("The elements on the page 'Сравнения:");
    webElementList = driver.findElements(By.xpath(".*[@id='spec-tbl']/tbody/tr[3]/td/p"));
    System.out.println("1 " + webElementList.get(0));
    System.out.println("2 " + webElementList.get(1));
    System.out.println("3 " + webElementList.get(2));
    System.out.println("4 " + webElementList.get(3));
    if (webElementList.size() != 4)
        condition = false;
    driver.findElement(By.xpath(".*[@id='mCSB_1_container']/div[1]/div/div[2]/a")).click();
    System.out.println("Elements on the page 'Сравнения' after deleting one element:");
    webElementList = driver.findElements(By.xpath(".*[@id='spec-tbl']/tbody/tr[3]/td/*"));
    if (webElementList.size() != 3)
        condition = false;
    for (int i = 0; i < webElementList.size(); i++) {
        System.out.println(webElementList.get(i).getText());
        wblmt.add(webElementList.get(i).getText());
    }
    Assert.assertTrue("Something have gone wrong", condition);
}
```

Результати даного тесту виводяться на консоль у наступному вигляді:

```
Test started
    The 4-th test
The elements on the page 'Сравнения':

1 .//*[ @id='spec-tbl' ]/tbody/tr[3]/td/p]
2 .//*[ @id='spec-tbl' ]/tbody/tr[3]/td/p]
3 .//*[ @id='spec-tbl' ]/tbody/tr[3]/td/p]
4 .//*[ @id='spec-tbl' ]/tbody/tr[3]/td/p]

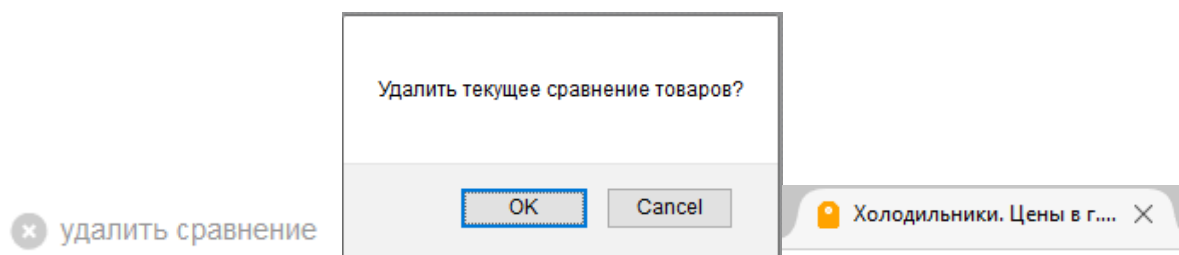
Elements on the page 'Сравнения' after deleting one element:
Холодильник, 311 л, 2 двери, 2 камеры
Холодильник, 138 л, 2 двери, 2 камеры
Холодильник, 310 л, 2 двери, 2 камеры
Test ended
```

У п'ятому тесті ми заходимо на сторінку <https://pn.com.ua/>, обираємо категорію «Холодильники», додаємо у зрівняння перші чотири холодильники. Переходимо на сторінку порівняння товарів та натискаємо клавішу «Удалить сравнение». У вікні, що при цьому з'являється, ставимо «ОК». Переконаємося, що виконується переадресація на сторінку з категорією «Холодильники».

```
@Test
public void FifthTest(){
    System.out.println("\t\tThe 5-th test");
    driver.get("https://pn.com.ua/");
    System.out.println("Processing of test");
    driver.findElement(By.xpath("//*[ @id='column-center' ]/section/div[3]/ul/li[3]/a")).click();
    driver.findElement
        (By.xpath("//*[ @id='column-center' ]/section/div[3]/article[1]/div[2]" +
            "/div[2]/span[2]/a")).click();
    driver.findElement
        (By.xpath("//*[ @id='column-center' ]/section/div[3]/article[2]/div[2]" +
            "/div[2]/span[2]/a")).click();
    driver.findElement
        (By.xpath("//*[ @id='column-center' ]/section/div[3]/article[3]/div[2]" +
            "/div[2]/span[2]/a")).click();
    driver.findElement
        (By.xpath("//*[ @id='column-center' ]/section/div[3]/article[4]/div[2]" +
            "/div[2]/span[2]/a")).click();
    driver.findElement
        (By.xpath("//*[ @id='column-center' ]/section/div[1]/div[1]/a")).click();
    System.out.println("The elements on the page 'Сравнения:");
    wblmt = new ArrayList<String>();
    webElementList = driver.findElements(By.xpath("//*[ @id='spec-tbl' ]/tbody/tr[3]/td/*"));
    for (int i=0; i<webElementList.size(); i++) {
        webElementList.get(i).getText();
        System.out.println((i+1)+" "+webElementList.get(i).getText());
        wblmt.add(webElementList.get(i).getText());
    }
    driver.findElement
        (By.xpath("html/body/div[1]/div[2]/div/div/div[2]/section/div[2]/div/div[1]" +
            "/div[1]/a/*")).click();
    driver.switchTo().alert().accept();
    System.out.println("We were redirected to the page: "+ driver.getCurrentUrl());
    String title = driver.getTitle();
    System.out.println("We are at the page: "+ title);
    Assert.assertEquals( message: "Something have gone wrong",
        expected: "Холодильники. Цены в г. Харьков. Сравнить цены в Прайс Навигатор. Купить", title);
}
```


Вивід на консоль показує, що переадресація виконалась, і ми дійсно знаходимося на сторінці категорії «Холодильники».

```
Test started
    The 5-th test
Processing of test
The elements on the page 'Сравнения':
1 Холодильник, 311 л, 2 двери, 2 камеры
2 Холодильник, 384 л, 2 двери, 2 камеры
3 Холодильник, 138 л, 2 двери, 2 камеры
4 Холодильник, 310 л, 2 двери, 2 камеры
We were redirected to the page: https://pn.com.ua/ct/2163/
We are at the page: Холодильники. Цены в г. Харьков. Сравнить цены в Прайс Навигатор. Купить
Test ended
```



У шостому тесті ми заходимо на сторінку <https://pn.com.ua/>, у пошуку задаємо «Samsung», натискаємо кнопку пошуку та переконуємося, що назви всіх знайдених товарів починаються на «Samsung».

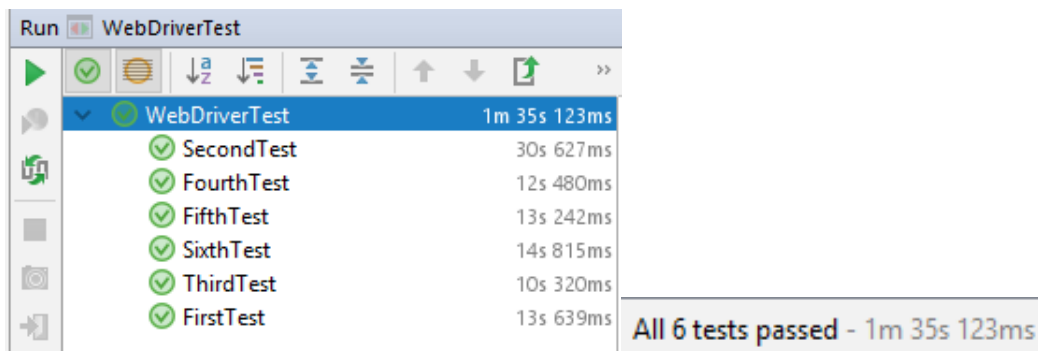
```
@Test
public void SixthTest(){
    System.out.println("\t\tThe 6-th test");
    driver.get("https://pn.com.ua/");
    System.out.println("Processing of test");
    driver.findElement(By.xpath(".*[@id='column-center']/section/div[2]" +
        "/ul/li[1]/a")).click();
    WebElement element = driver.findElement
        (By.xpath("html/body/div[1]/header/div/div[2]/div[4]/form/div[1]/div[1]/*"));
    element.sendKeys(...charSequences: "Samsung");
    element.submit();
    String search = "Samsung";
    String string = "";
    wblmt = new ArrayList<String>();
    webElementList = driver.findElements(By.xpath(".*strong[contains(text(),' грн')]"));
    System.out.println(webElementList.size());
    driver.get("https://pn.com.ua/ct/1232/?fn=samsung&view=table");
    for(int i=1; i<webElementList.size()-1; i++){
        String url = ".*[@class='catalog-list']/li["+i+"]/article/div[2]/div[1]/a";
        string = driver.findElement
            (By.xpath(url)).getText().substring(beginIndex: 0, endIndex: 7);
        System.out.println(string);
    }
    for (int i = 0; i < wblmt.size() ; i++) {
        System.out.println(wblmt.get(i));
    }
    Assert.assertEquals(message: "Something have gone wrong", expected: "Samsung", search);
}
```



Таким чином, у консолі ми бачимо, що всі елементи починаються на «Samsung».

```
Test started
    The 6-th test
Processing of test
Amount of elements: 24
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Samsung
Test ended
```

Усі тести пройшли успішно.



Висновки

Отже, SeleniumWebDriver представляє собою набір бібліотек для різних мов програмування, він дозволяє керувати браузером з програми, написаної даною мовою програмування. Це надійний фреймворк автоматизації, який здатен працювати з будь-яким браузером. З його допомогою можна розробляти великі тестові набори, в тому числі й тести з досить складною логікою поведінки та перевірок.

Таким чином, у ході виконання лабораторної роботи було вивчено фреймворк *SeleniumWebDriver*. А також розроблені шість тестів, за допомогою яких виконується перевірка коректності роботи сайту. Усі тести пройшли успішно, а результати тестів виведені на консоль.