

## Практична робота: Selenium WebDriver + Selenide + JUnit + Maven

У цій практичній роботі ви дізнаєтеся, як використовувати Selenium WebDriver для написання автоматизованих тестів веб-додатків. Ви дізнаєтеся про основні функції WebDriver, такі як знаходження елементів на сторінці, взаємодію з формами та кнопками, навігацію між сторінками та перевірку результатів тестів.

Ця практична робота містить низку вправ, що допоможуть вам навчитися автоматизувати тестування веб-додатків з використанням Selenium WebDriver. На завершення ви отримаєте достатньо практичного досвіду, щоб самостійно створювати автоматизовані тести для веб-додатків.

### Для Windows:

1. Щоб забезпечити ефективний процес розробки тестів, нам потрібна IDE (інтегроване середовище розробки), яка може працювати з Java, включаючи компіляції проектів, перевірку синтаксису, підсвічування та інше. В даній практичній вам знадобиться IntelliJ IDEA IDE як IDE.

Завантажте безкоштовну версію (Community):

<https://www.jetbrains.com/idea/download/>

2. При роботі з середовищем розробки за допомогою декількох фреймворків вам потрібно налаштувати додаткові файли та бібліотеки та керувати ними. Вам потрібно використати будь-який інструмент автоматизації збірки, який допоможе вам виконувати всі ці дії. Ми будемо використовувати Maven.

Встановіть Maven:

Перейдіть за посиланням і завантажте архів програми

<https://maven.apache.org/download.cgi>

розпакуйте його.

3. Якщо у вас немає JAVA на комп'ютері (а після вивчення Jmeter мала б бути), вам необхідно встановити Java Development Kit(JDK).

Ось посилання на Oracle - виберіть останню версію Java SE JDK (дивітьщо щоб підходила під вашу ОС)

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

SE – означає Standard Edition. Виберіть файл інсталяції для вашого ПК скачайте його та встановіть.

Перезавантажте комп'ютер (не пропускайте цей пункт)

Щоб перевірити чи встановлена Java зробіть наступне:

- Відкрийте Command Line для Windows (Win + R) > enter cmd

Напишіть java -version або java -v на натисніть Enter

Якщо покаже версію - супер!

ЯКЩО НІ- робимо наступні кроки:

- 1) Відкриваємо System Settings (Win + Pause keys) -> Advanced System Set -> Environment Variables > Path (in System variables block) > EDIT > додаємо шлях до папки bin для JDK.
- 2) Створіть нову або відредагуйте вже існуючу змінну (variable): JAVA\_HOME. Вона має включати шлях (path) до root JDK папки.

Відкрийте термінал та перевірте версію джави знову.

4. Ви можете створити свій проект самостійно або використати проект-приклад який містить всі необхідні конфігурації та dependencies фреймворків.

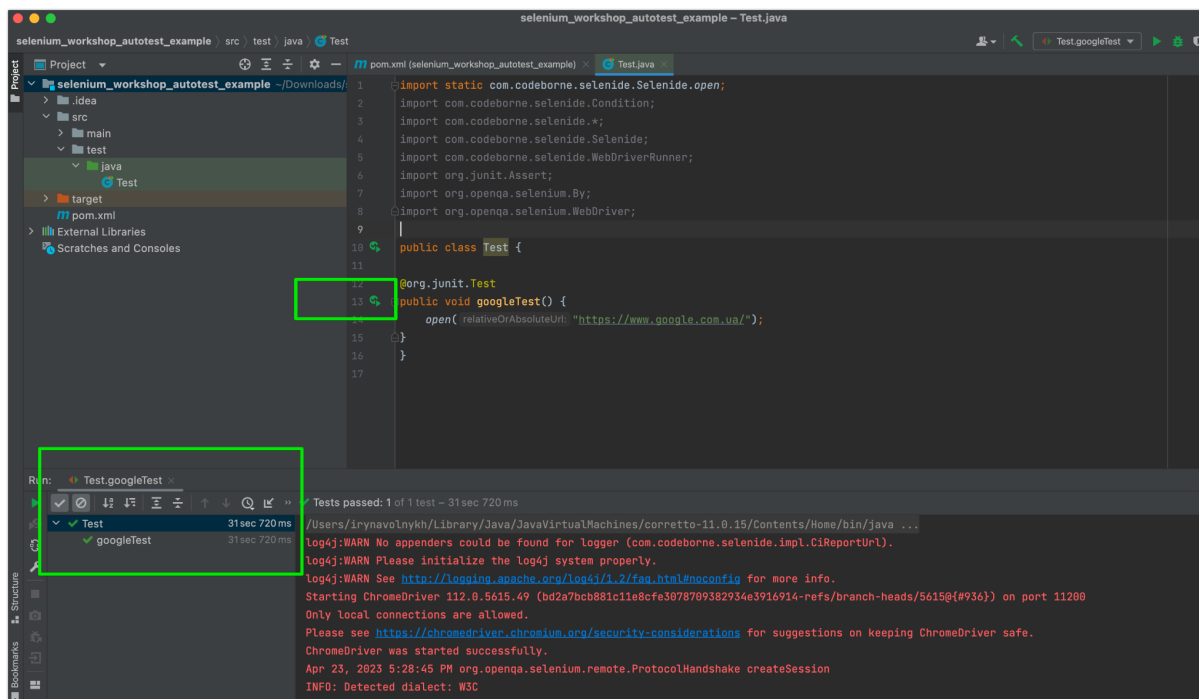
Для цього:

Відкрийте IntelliJ Idea> open new project - вам треба при відкритті вибрати **ВСЮ** папку (selenium\_workshop\_autotest\_example) проекту. Тільки розархівуйте його спочатку

### [проект]

[https://drive.google.com/file/d/1XgrxLJZIGtJRyDoiCIWJVCDBBkZYUzNM/view?usp=share\\_link](https://drive.google.com/file/d/1XgrxLJZIGtJRyDoiCIWJVCDBBkZYUzNM/view?usp=share_link)

5. Після того як ви відкрили проект - спробуйте через декілька хвилин запустити тест: Він повинен бути Passed.



The screenshot shows the IntelliJ IDEA IDE interface. The main editor displays a Java file named 'Test.java' with the following code:

```
1 import static com.codeborne.selenide.Selenide.open;
2 import com.codeborne.selenide.Condition;
3 import com.codeborne.selenide.*;
4 import com.codeborne.selenide.Selenide;
5 import com.codeborne.selenide.WebDriverRunner;
6 import org.junit.Assert;
7 import org.openqa.selenium.By;
8 import org.openqa.selenium.WebDriver;
9
10 public class Test {
11
12     @org.junit.Test
13     public void googleTest() {
14         open(relativeOrAbsolutePath: "https://www.google.com.ua/");
15     }
16 }
17
```

The 'Test.java' file is highlighted with a green box. Below the editor, the 'Run' tool window shows the execution of the 'Test.googleTest' test. The test passed successfully, with a duration of 31sec 720ms. The console output includes the following messages:

```
Tests passed: 1 of 1 test - 31sec 720 ms
/Users/irynavolnykh/Library/Java/JavaVirtualMachines/corretto-11.0.15/Contents/Home/bin/java ...
log4j:WARN No appenders could be found for logger (com.codeborne.selenide.impl.CiReportUrl).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
Starting ChromeDriver 112.0.5615.49 (bd2a7ccb881c11a8cfe3078709382934e3916914-refs/branch-heads/5615@{#936}) on port 11280
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Apr 23, 2023 5:28:45 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
```

Для тих у кого MAC OS

- IDE (IntelliJ IDEA) - треба теж поставити
- Maven - теж треба (але вам треба src.zip)
- Робимо кроки 4 та 5 - якщо не працює - ставимо JDK - <https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html> але версія для MacOS.

Зверніть увагу! Вам не треба нічого прописувати (шляхи змінні) - тільки ставити - то все тільки для Windows необхідно.

- Робимо кроки 4 та 5

Практика:

1. В тестовому проєкті на прикладі написаного тесту створити новий тест з анотацією JUnit @Test (зверніть увагу! Що ми працюємо всередині даного класу):

```

1  import com.codeborne.selenide.Condition;
2  import com.codeborne.selenide.Selenide;
3  import com.codeborne.selenide.WebDriverRunner;
4  import org.junit.Assert;
5  import org.openqa.selenium.By;
6  import org.openqa.selenium.WebDriver;
7
8  import static com.codeborne.selenide.Selenide.$;
9  import static com.codeborne.selenide.Selenide.open;
10
11
12  public class Test {
13
14
15      @org.junit.Test
16      public void googleTest() {
17          open(relativeOrAbsoluteUrl: "https://www.google.com.ua/");
18      }
19
20      @org.junit.Test
21      public void newTest() {
22      }
23  }
24
25

```

2. Створити 3 простих тести для перевірки наступного функціоналу:

2.1. Пошук у Google com:

Кроки:

1. open ("https://google.com")
2. \$("ТУТ CSS селектор полю пошуку").click();
3. \$("ТУТ CSS селектор будь-якого елемента на сторінці з результатами").shouldBe(visible);

2.2 Написати ще 2 тести до будь-якого вебсайту застосовуючи наступні методи:

click()  
hover()  
shouldBe()  
shouldHave()  
size(int)

Всю необхідну інформацію по використанню даних методів шукайте тут:  
<https://selenide.org/documentation.html>

Приклад:

```
@org.junit.Test  
public void logoTest() {  
    open("https://pethouse.ua/ua/shop/rybam/dekor-dlya-akvariuma/");  
    $("#tpl-logo").click();  
    assert WebDriverRunner.url().equals("https://pethouse.ua/ua/");  
}
```