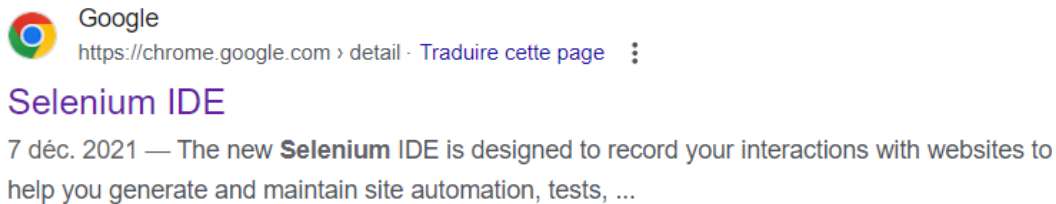


# Tester une application web avec Selenium

## Installer le plugin chrome Selenium

Ouvrir google.fr sous chrome  
chercher plugin selenium



en cliquant dessus on arrive à sa page :



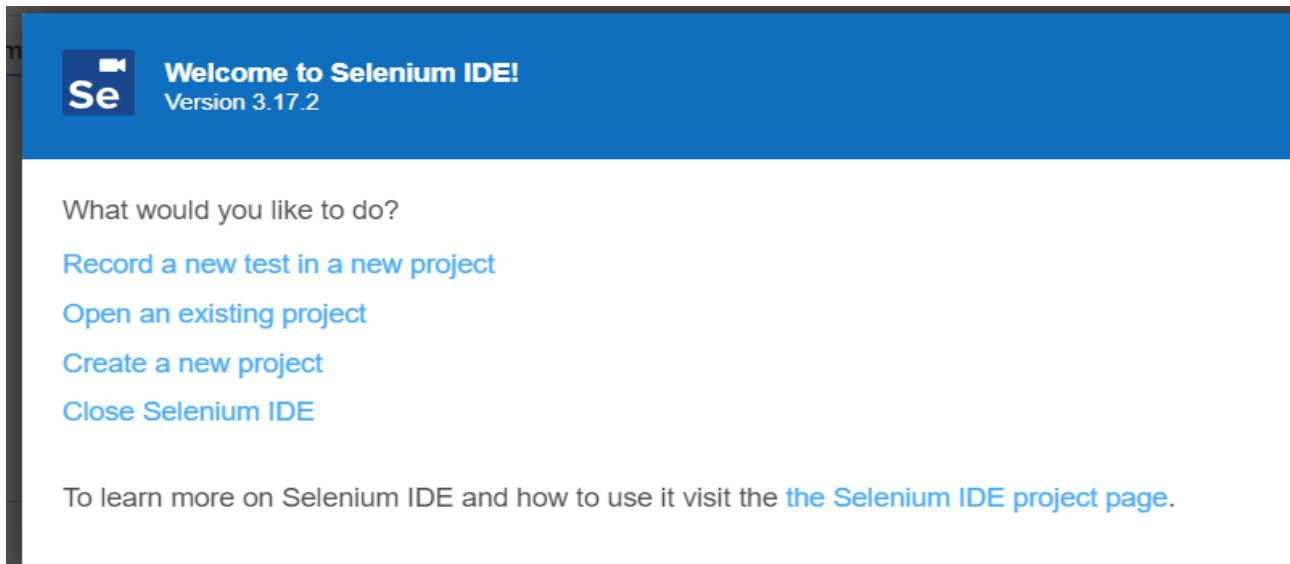
Cliquer sur Ajouter

## Enregistrer un test

Ouvrir en cliquant sur son bouton sur la palette en haut à droite :



On aura :



cliquer sur Record a new test in a new project

**Name your new project**

Please provide a name for your new project.

PROJECT NAME

You can change the name of your project at any time by clicking it and entering a new name.

OK CANCEL

mettez le nom du projet (ca peut etre l'url du site à tester)

cliquer sur OK

**Set your project's base URL**

Before you can start recording, you must specify a valid base URL for your project. Your tests will start by navigating to this URL.

BASE URL

START RECORDING CANCEL

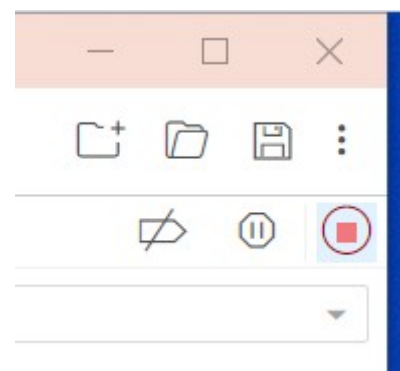
mettez l'url de votre site à tester

En cliquant sur start recording, le site s'ouvre dans une nouvelle fenetre chrome

Agrandir la fenetre

Faites les actions du tests

A la fin : fermer l'enregistrement : le bouton rouge en haut à droite du plugin :



Entrer le nom de votre test, vous aurez après votre test sous forme :

Project: [mzamouneisi.github.io/eisiesn360frtangular2\\*](https://mzamouneisi.github.io/eisiesn360frtangular2*)

Tests ▾ +

Search tests... 🔍

test\_login\*

	Command	Target	Value
15	type	id=password	Eisis2020
16	send keys	id=password	\${KEY_ENTER}
17	click	css= btn-outline-dark	
18	click	id=password	
19	click	id=password	
20	double click	id=password	
21	type	id=password	Eisi2020
22	send keys	id=password	\${KEY_ENTER}

Command  //

Target

Value

Description

Cliquer sur le bouton enregistrer en haut à droite



On peut rejouer le test en cliquant sur l'icône :



Lorsqu'on a plusieurs tests, on clique sur l'icône pour les rejouer tous.

## Exporter le test en Java, Python, JS, ...

Exportons notre test en java :

Clique droit sur le nom du test :

test\_login

- Rename
- Duplicate
- Delete
- Export

Choisir Export :

## Select language

- ☐ C# NUnit
- ☐ C# xUnit
- ☒ Java JUnit
- ☐ JavaScript Mocha
- ☐ Python pytest
- ☐ Ruby RSpec

- ☐ Include origin tracing code comments
- ☐ Include step description as a separate comment
- ☐ Export for use on Selenium Grid

EXPORT

Clique sur Export :  
Choisir le dossier dans lequel la classe de test va être enregistrée.

## Utiliser le test à partir de Java

Ajoutons un projet java avec cette commande maven :  
`mvn archetype:generate -DgroupId=$pkg.$appName -DartifactId=$appName -DarchetypeArtifactId=maven-archetype-quickstart -Dversion=0.1 -DinteractiveMode=false`

Remplacer \$pkg , \$appName par leur valeur. Ex : com.mz , test\_selenium.

Copier la classe de test générée par selenium vers `test_selenium/src/test/java/com/mz/test_selenium/`  
Ajouter le bon package à cette classe.

Ajouter ces deux dépendances dans le pom.xml :

```
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.5</version>
</dependency>
<dependency>
  <groupId>org.seleniumhq.selenium</groupId>
  <artifactId>selenium-java</artifactId>
  <version>4.18.1</version>
</dependency>
```

Ça devrait compiler.

Nettoyer le code :

\* remplacer : le type de code :

```
driver.manage().window().setSize(new Dimension(1099, 577));
```

par :

```
driver.manage().window().fullscreen();
```

Afin de ne pas avoir l'erreur : is not clickable at point (655, 430).

Exemple de code pour se connecter :

```
@Test
public void testlogin() {
    driver.get("https://mzamouneisi.github.io/eisiesn360frtangular2/");
    driver.manage().window().fullscreen();
    driver.findElement(By.id("username")).click();
    driver.findElement(By.id("username")).sendKeys("manager.demo1@eisi-consulting.fr");
    driver.findElement(By.id("password")).click();
    driver.findElement(By.id("password")).sendKeys("Eisi2020");
    driver.findElement(By.id("password")).sendKeys(Keys.ENTER);
}
```

lancer les tests par : mvn test