San Jose State University
Department of Electrical Engineering
EE104, Fall 2023, Pham

# Automated Test Development with Selenium IDE

## References

- Selenium webpage: https://www.selenium.dev/selenium-ide/
- Selenium document: https://www.selenium.dev/selenium-ide/docs/en/introduction/getting-started
- Selenium download: https://www.selenium.dev/downloads/
- Tutorials:
    - https://www.tutorialspoint.com/selenium/
    - https://expleoacademy.com/int/6-steps-to-create-and-export-selenium-ide-tests/
    - https://www.youtube.com/watch?v=m4KpTvEz3vg&ab_channel=Simplilearn

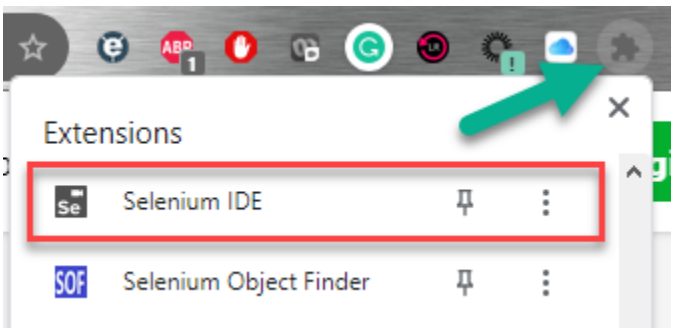
## Software Download

### Download browser plugin

From https://www.selenium.dev/downloads/, select download for Chrome. You can also download the plugins for Firefox or Edge or the latest Zip file and install it yourself. In this tutorial, we will just focus on Chrome for the purpose of learning. Other browser implementation may have some variations but with not much deviations from Chrome.

- Download Selenium IDE
- Optional: Download Selenium Object Finder

Now click on the Chrome Extension icon on the right top corner of the browser, you should see the installed extension(s):



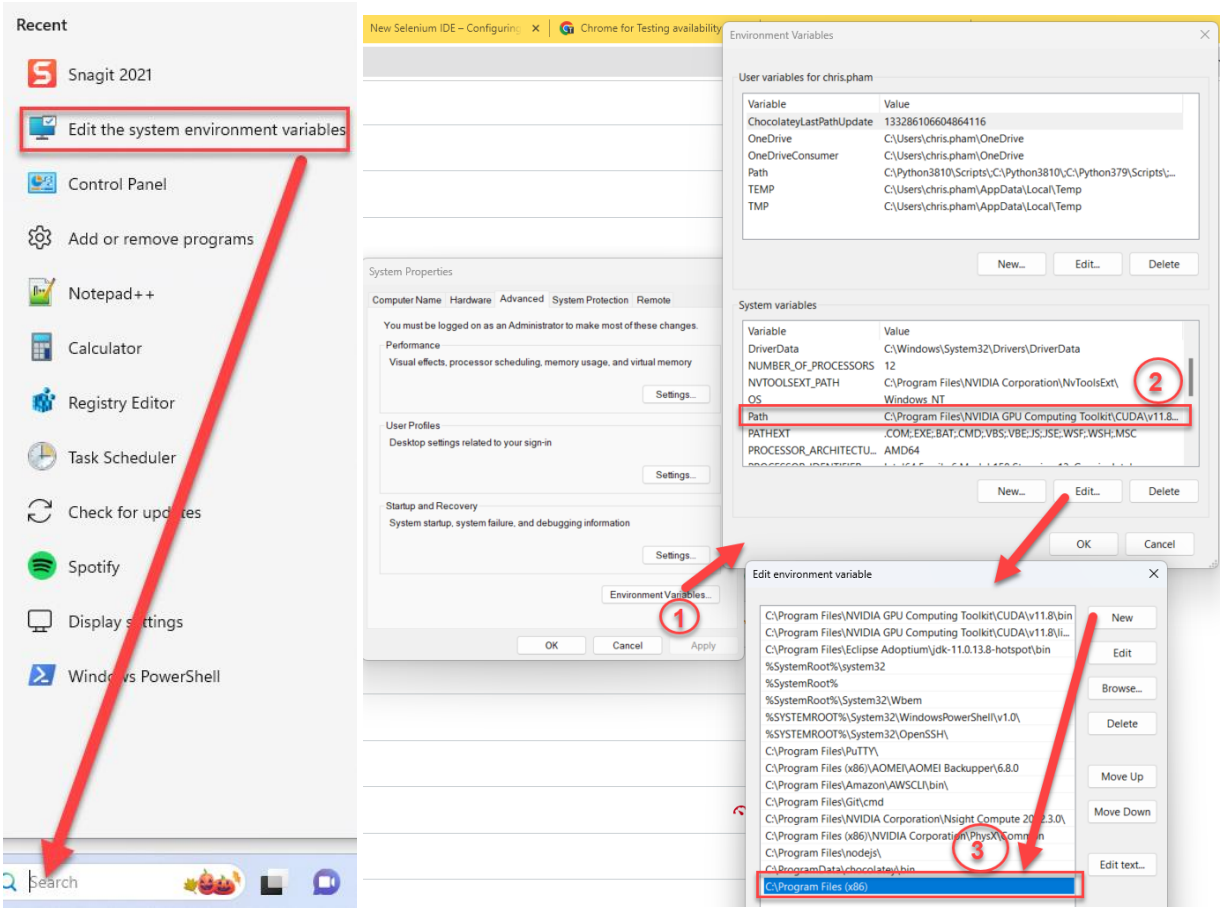
## Download the latest WebDriver for your browser

Chrome: https://chromedriver.chromium.org/downloads

or Chrome for Testing availability (googlechromelabs.github.io) or https://googlechromelabs.github.io/chrome-for-testing/#stable if you cannot find your Chrome driver on the previous link.

Save the Chrome driver chromedriver.exe to **C:\Program Files (x86)**

Add **C:\Program Files (x86)** to Path




## Install Python

- Install Python and add Python path (from python.org)
  Install Python 3.7.9 from https://www.python.org/downloads/release/python-379/
  You can use the web-based installer
  Let's do Customize Installation and install it to C:\Python379 for simplicity
- python.exe -m pip install --upgrade pip
- pip install selenium, pytest, python-dotenv

# Supporting File Creation

## Create .env file

This file is to keep all passwords, license codes or anything else you don't want to show in your python source code.

For example:
TEST_PASSWORD = "Password123"
LICENSE_CODE = "ABC123"

### Create a Readme.txt File

This is a text file to capture your instructions and other information about your scripts to give users a perspective of what to expect from your scripts.

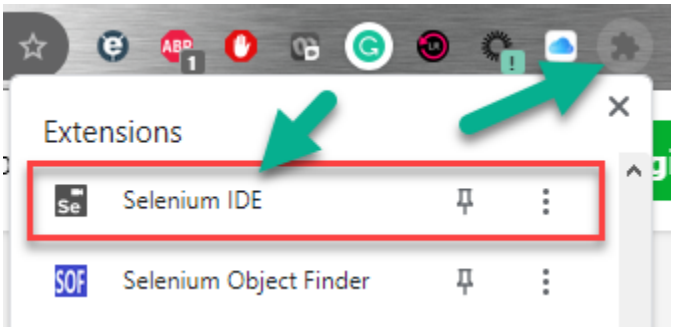
### Create a Test Plan File

This file explains the followings:

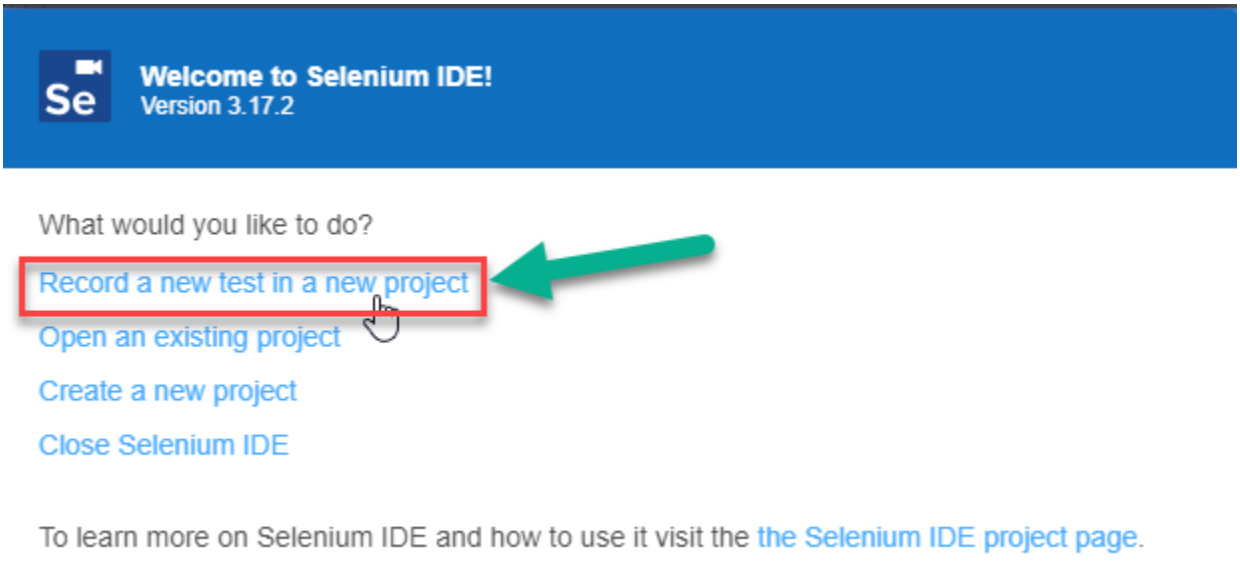
- Prerequisites
- A list of tests
- Test details with expected pass/fail conditions and side effects after running that test

## Record a Test

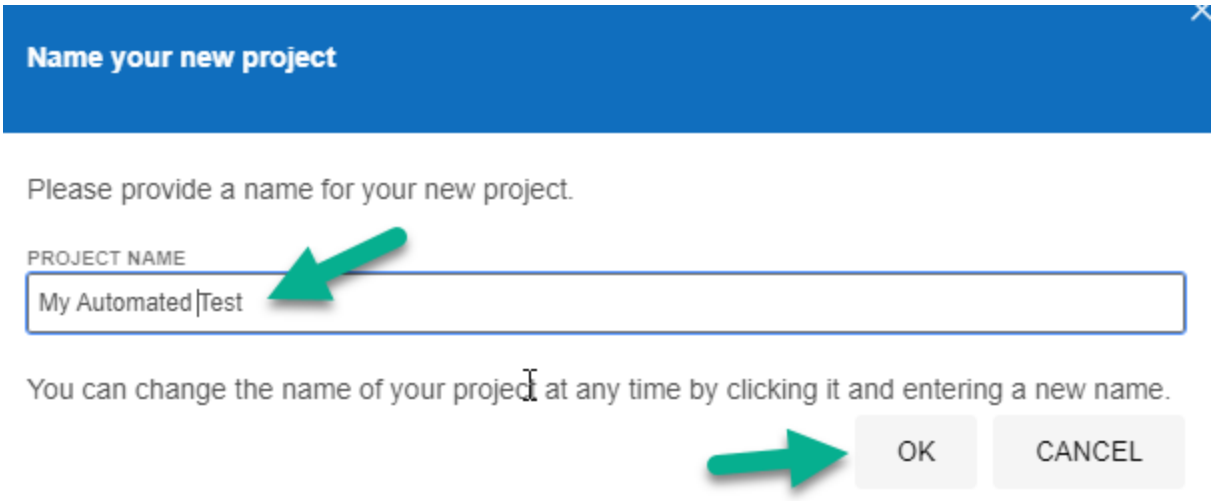
Click on the Chrome Extension icon to show the list of the installed browser extensions, then click on Selenium IDE to invoke the IDE.



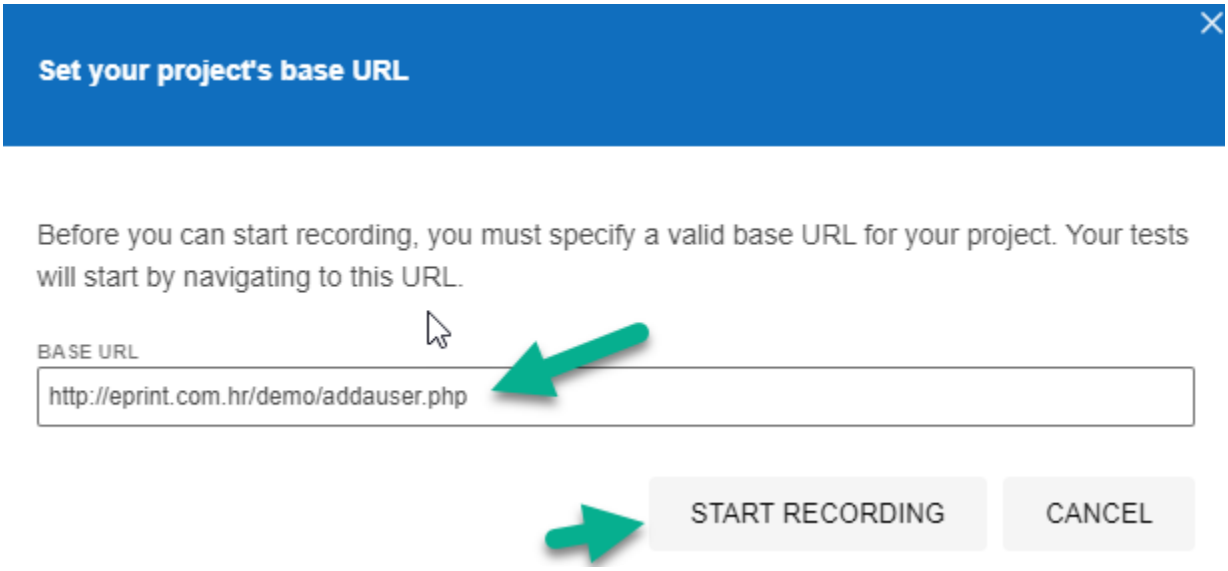

You will see the following popup:




Provide the Project Name and click OK.




On the next popup, provide the following website http://eprint.com.hr/demo/addauser.php and click START RECORDING to proceed:



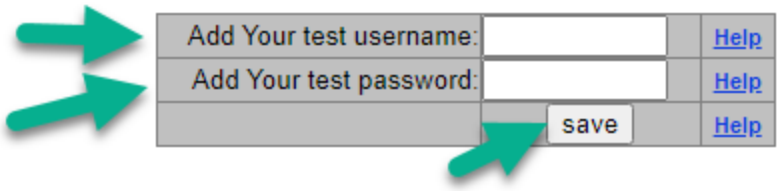

On the website, you will want to add the followings, but you will let the automation script does that for you:

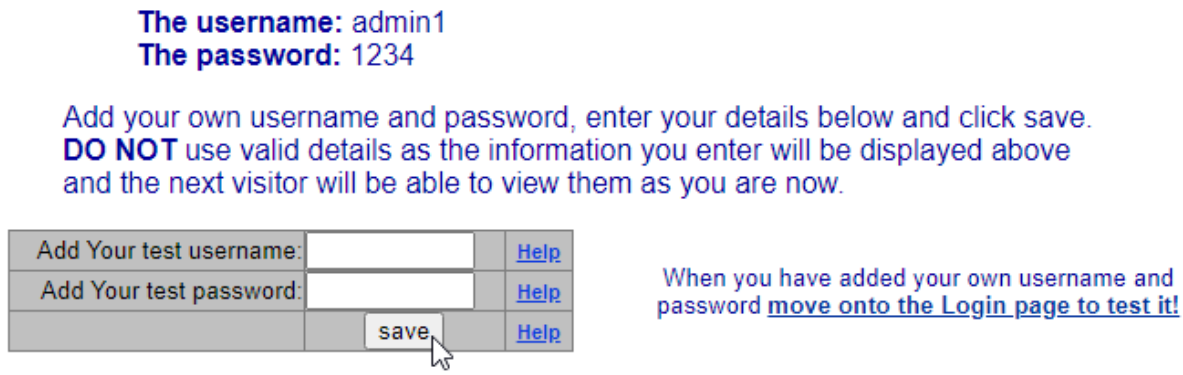
username: your first name
Password: your own password
Click “save”

Below is the current single record within the database:

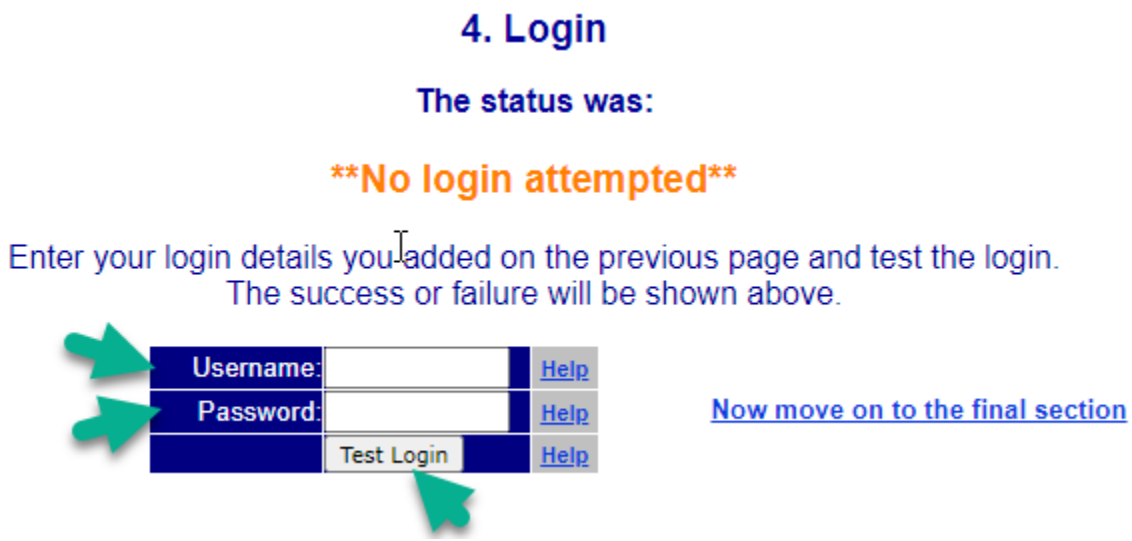
**The username:** admin
**The password:** 123456

Add your own username and password, enter your details below and click sav
them as you are now.

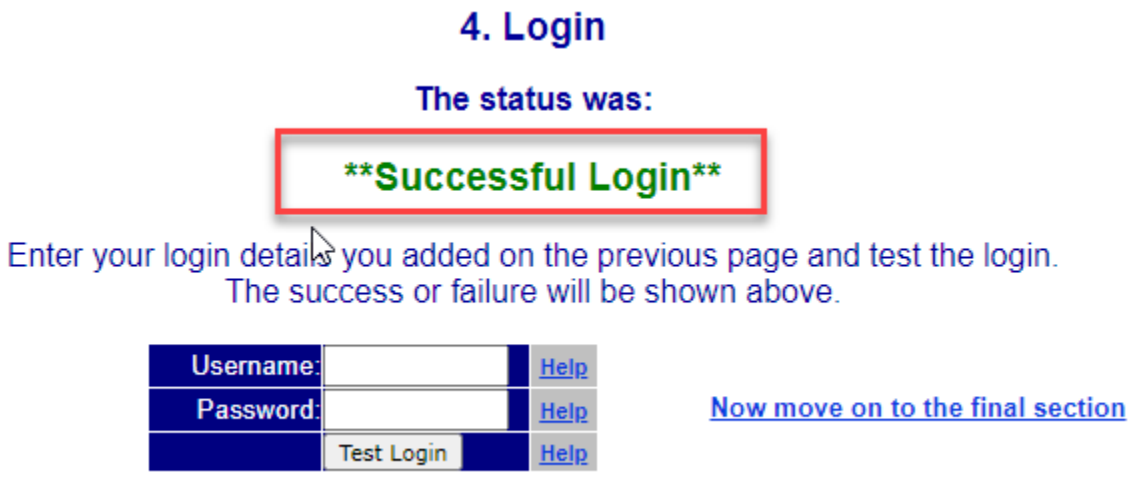




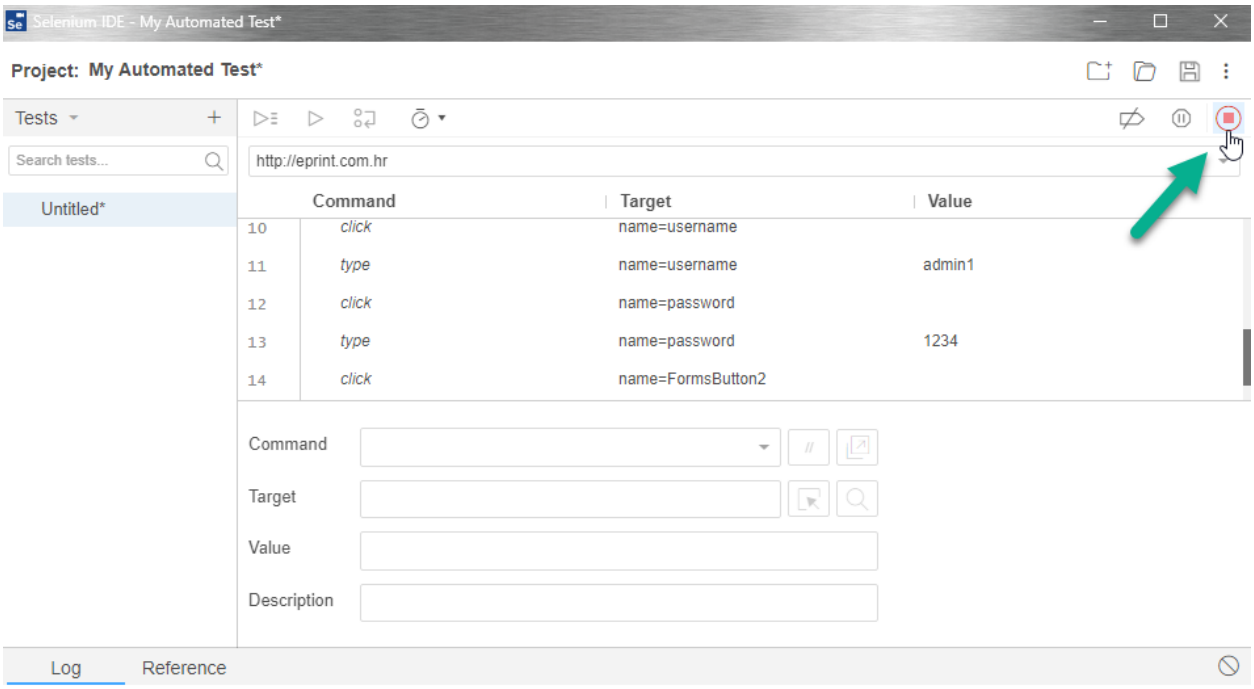

Then click on the link “move onto the Login page to test it!”




Enter your own username and password and press Test Login. If you typed the correct credentials, the Successful Login message will show:

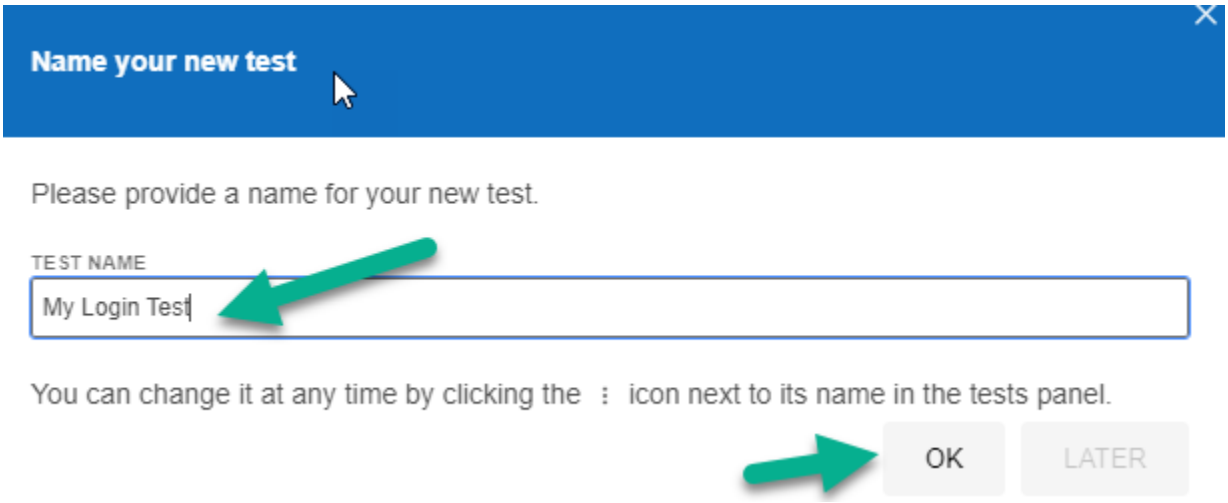



We can stop the recording now:

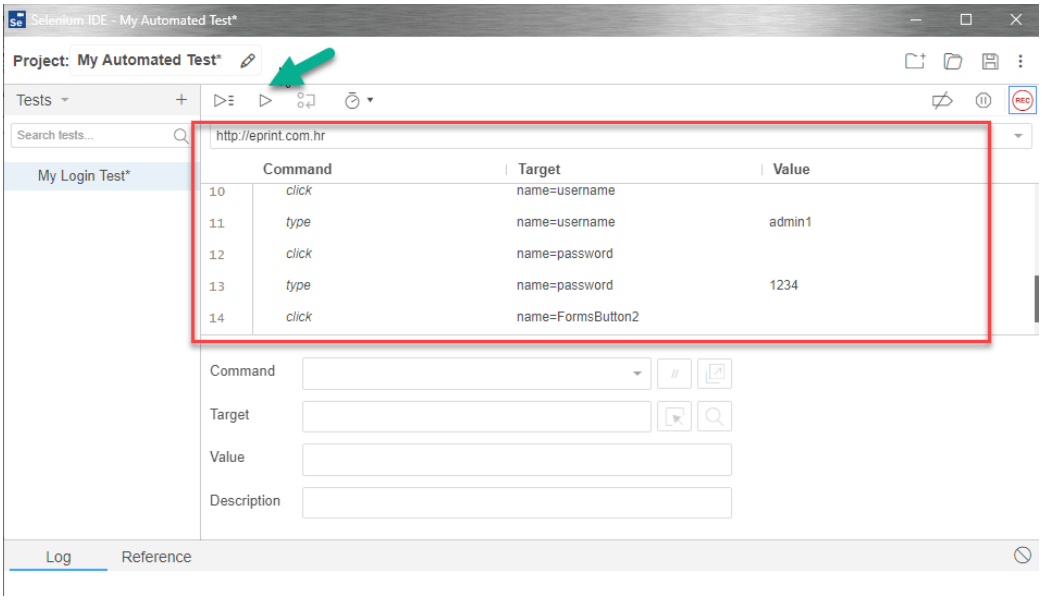



Note: The above window can be behind other windows. You should search for it or click on this icon in the bottom tray to bring it to the foreground: 

Provide a name for your test and click OK:

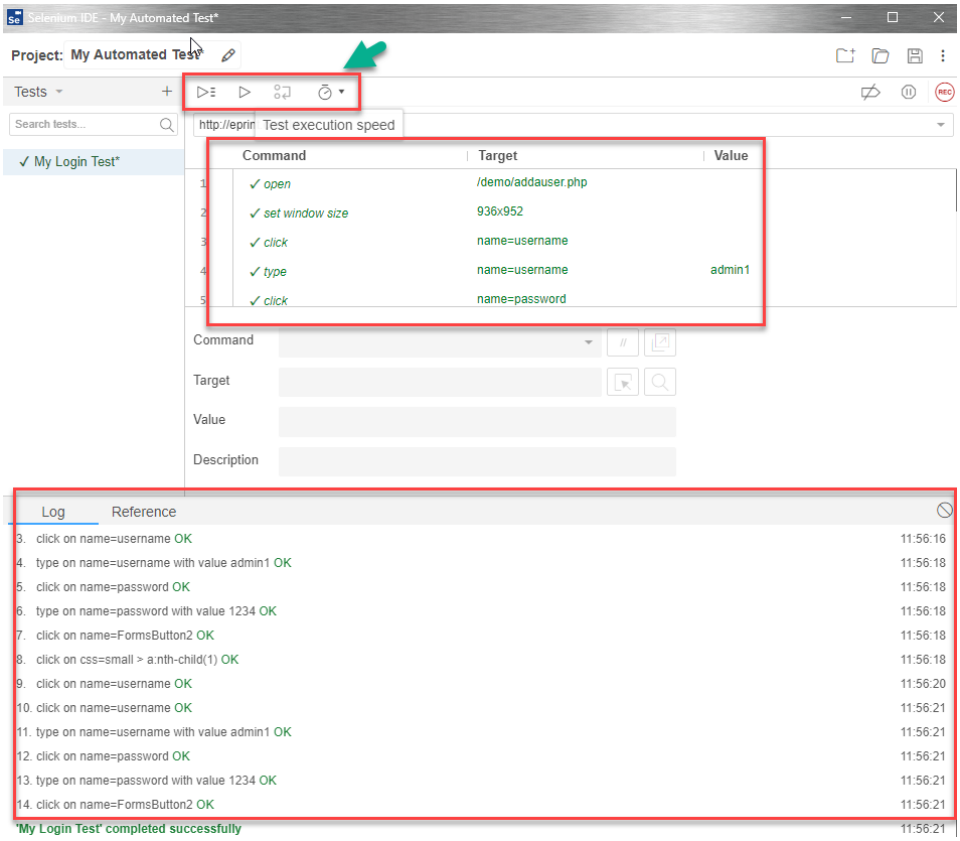


You will see the recorded steps in the Selenium IDE:




You can press the PLAY button to re-play the sequence automatically.

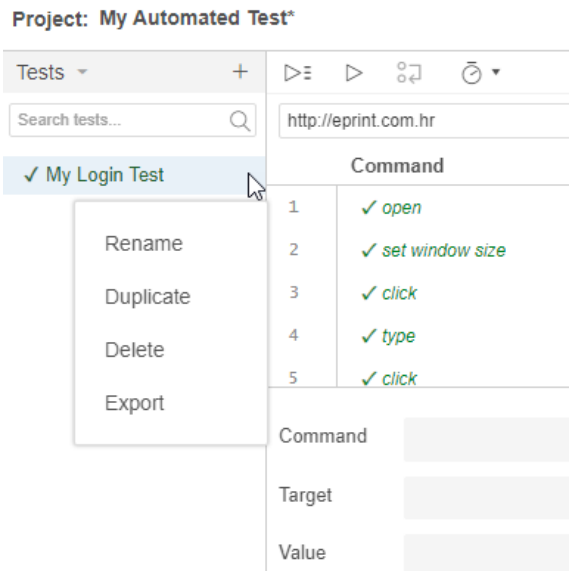
Feel free to change the execution speed and other features of the Selenium IDE to learn more about it.

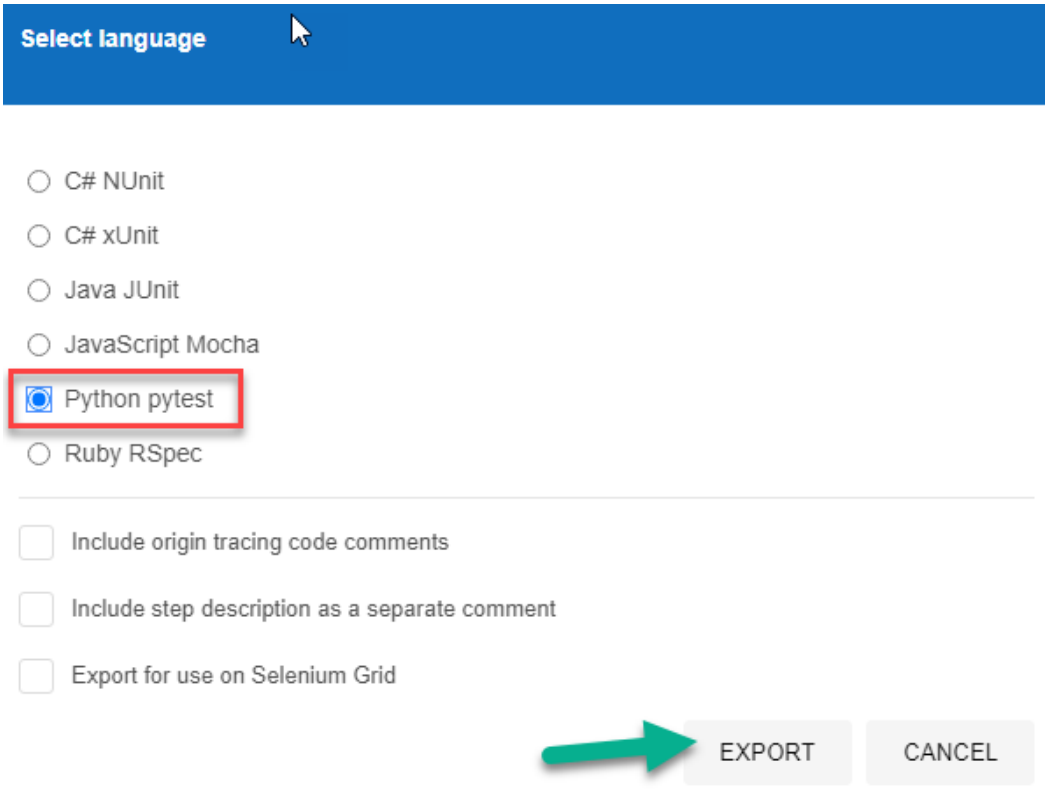



Save the project as “My Automated Test.side” where SIDE = Selenium IDE.

## Export the Test to Python

Click on the  name of the test, then click Export to export your test. Your test will be saved as **test_myLoginTest.py**.




## Additional Selenium IDE utilities to improve your script

Read by yourself this website to learn more details to improve you're your scripts: https://expleoacademy.com/int/6-steps-to-create-and-export-selenium-ide-tests/

## Modify your exported Python script to include the Chrome driver path

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities


class TestMyLoginTest():
  def setup_method(self, method):
    self.driver = webdriver.Chrome(executable_path='C:\Program Files (x86)\chromedriver.exe')
    self.vars = {}

  def teardown_method(self, method):
    self.driver.quit()

  def test_myLoginTest(self):
    self.driver.get("http://eprint.com.hr/demo/addauser.php")
    self.driver.set_window_size(936, 952)
    self.driver.find_element(By.NAME, "username").click()
    self.driver.find_element(By.NAME, "username").send_keys("admin1")
    self.driver.find_element(By.NAME, "password").click()
    self.driver.find_element(By.NAME, "password").send_keys("1234")
    self.driver.find_element(By.NAME, "FormsButton2").click()
    self.driver.find_element(By.CSS_SELECTOR, "small > a:nth-child(1)").click()
    self.driver.find_element(By.NAME, "username").click()
    self.driver.find_element(By.NAME, "username").click()
    self.driver.find_element(By.NAME, "username").send_keys("admin1")
    self.driver.find_element(By.NAME, "password").click()
    self.driver.find_element(By.NAME, "password").send_keys("1234")
    self.driver.find_element(By.NAME, "FormsButton2").click()
```

The open a Powershell windows as Administrator into the directory where you have your Python script, then run the following command:

**PS C:\test> pip install selenium**

**PS C:\test> pip install pytest**

```
PS C:\test>npm install -g selenium-side-runner
PS C:\test>selenium-side-runner <ProjectName>.side
```