

AUTOMATED SOFTWARE TESTING (502072)

Lab 6 – Mouse event Prepared by Nguyen Thanh Quan (MSc)

1. GOALS: This lab helps students to

- Practise automation testing of web applications using.
- Be able to write test cases with Selenium Library – Robot Framework.
- Understand various functionalities of Selenium Library.
- Be able to work with various web elements and mouse events.

2. OBJECTIVES

- Install automated testing environment with Selenium/Robot Framework.
- Install web drivers.
- Write and execute test cases.
- Work with mouse events such as click, hover etc.

3. CONTENT

3.1 PREREQUISITES

To complete this assignment, students must prepare to install the required libraries below which are also presented in Lab3. More importantly, students should have a basic understanding of testing concepts.

To complete this assignment, students must prepare to install the following libraries and packages. More importantly, students should have a basic understanding of testing concepts.

- Python
Go to python official site – <https://www.python.org/downloads/> and download the latest version or the prior version of python as per your operating system.
Remember to set PATH correctly to use Python after installation.
- Pip
PIP gets installed along with python. Run ``pip --version`` to check pip version
- Robot Framework
Use pip – python package manager to install the robot framework and the command for it is as follows

``pip install robotframework``
``robot --version`` to check robot framework version.
- wxPython for Ride IDE
wxPython is needed for Robot Framework Ride, which is an IDE for Robot Framework.

Windows:

<https://sourceforge.net/projects/wxpython/files/wxPython/2.8.12.1/>

Linux: Install wxPython with the package manager of OS.

- Selenium library (<https://github.com/robotframework/SeleniumLibrary/>)

```
`pip install --upgrade robotframework-seleniumlibrary`
```

- Selenium webdriver
(<https://www.selenium.dev/documentation/webdriver/>)
`pip install --upgrade robotframework-seleniumlibrary`

- Robot Framework Ride
Use pip command to install Ride IDE.

```
`pip install robotframework-ride`.
```

To open Ride IDE, run ``ride.py``

3.2 THEORY

Students must review the theory of Robot Framework and Selenium Library which are presented in Lab2 and Lab3.

3.3 PRACTICE

For testing, it becomes important to understand how to interact with the browser and locate the html elements. It is very easy to work with input fields with robot framework. In this lab, we will learn how to work with form fields namely textbox, radio button, dropdown, checkbox, text area etc using Selenium Library. To work with form field elements, we need the locator, which is the main unique identifier for that the fields and it can be id, name, class, etc.

Let create a new project and import Selenium Library properly.

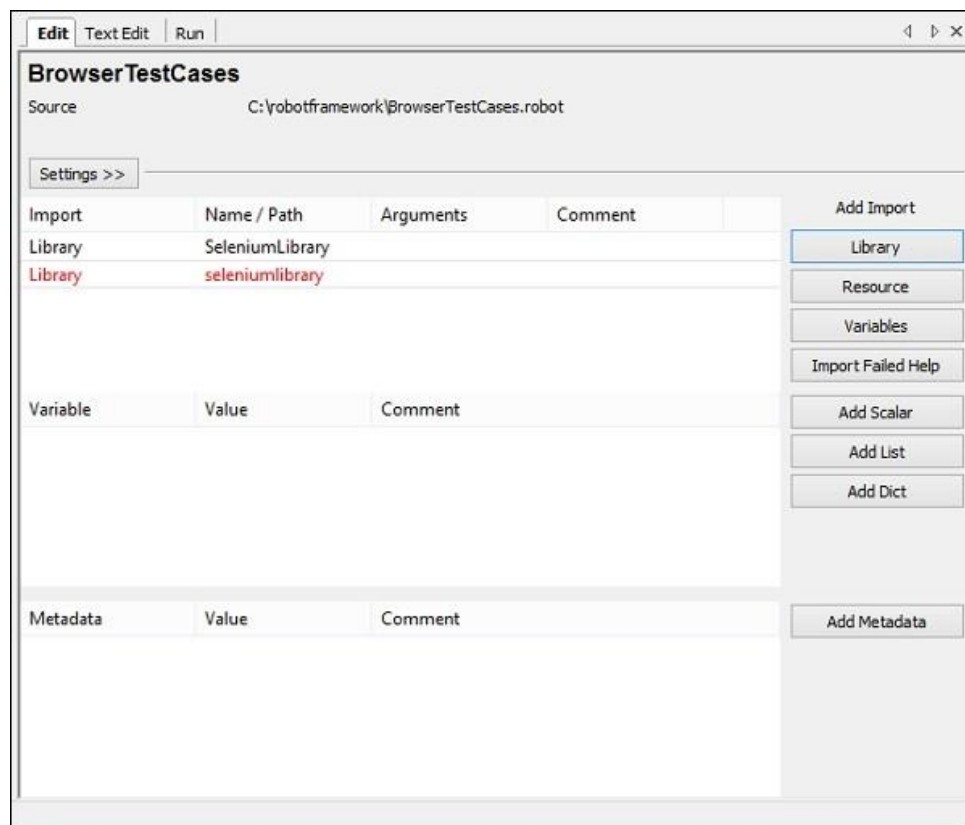


Figure 1 - Valid/invalid library import

The name given has to match with the name of the folder installed in site-packages. In case the names do not match, the library name will be in **red** as shown above. Library import in red is as good as the library does not exist inside python. Now, we have completed selenium library import.

The official document page of Robot Framework provides users with detailed explanation and demo, students should refer to the page during the time working with this lab.

<https://docs.robotframework.org/docs>

<https://robotframework.org/robotframework/latest/RobotFrameworkUserGuide.html#introduction>

<https://robotframework.org/SeleniumLibrary/SeleniumLibrary.html>

Locating elements:

<https://selenium-python.readthedocs.io/locating-elements.html>

Chrome/Edge extensions for locating Xpath:

<https://www.browserbear.com/blog/9-best-chrome-extensions-to-find-xpath-for-selenium-and-other-automation-tools/>

x/y coordinates:

https://robotframework.org/SeleniumLibrary/SeleniumLibrary.html?_ga=2.47015591.1688393203.1707139684-1841019508.1706956421#Click%20Element%20At%20Coordinates

Let review and research about locators for form fields element to complete the assignment.

Assignment 1: Open browser

- URL to open: <https://practice-automation.com/>

Assignment 2: Interact with web elements via mouse with Selenium library

Test case 1:

- Navigate to "hover" page by clicking "Hover" button displayed on the page just opened.
- Determine coordinates to "hover" on "Mouse over me" and verify text change.

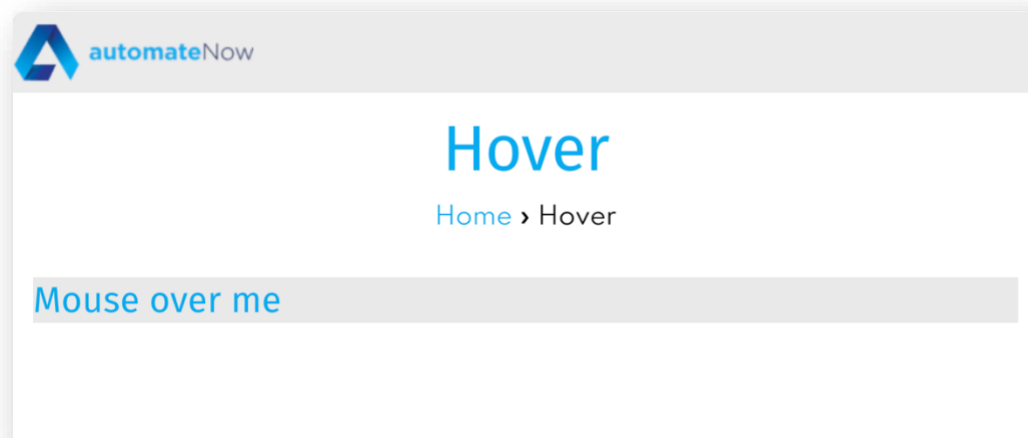


Figure 2 – Before hover

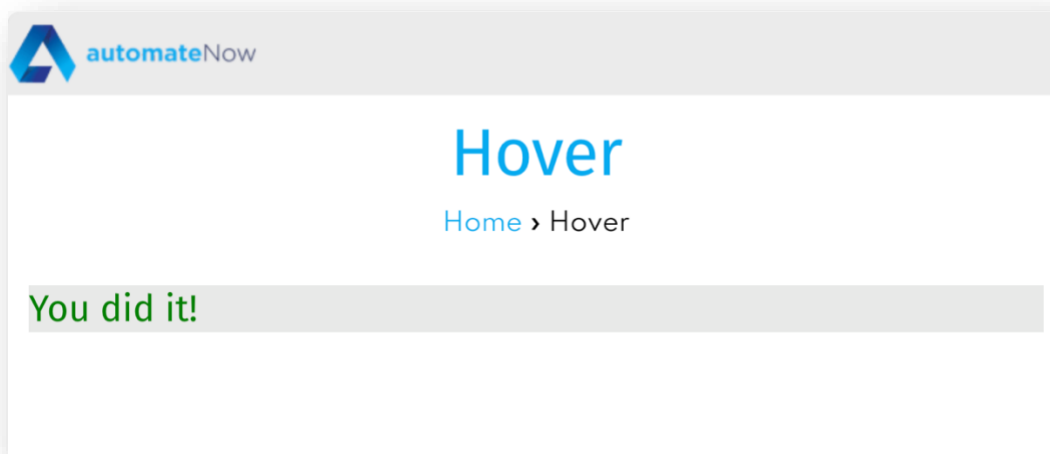


Figure 3 – After hover

Test case 2:

- Navigate to "Click events" page by clicking "Click events" button displayed on the page just opened.
- Determine coordinates to "click" the buttons and verify text change.

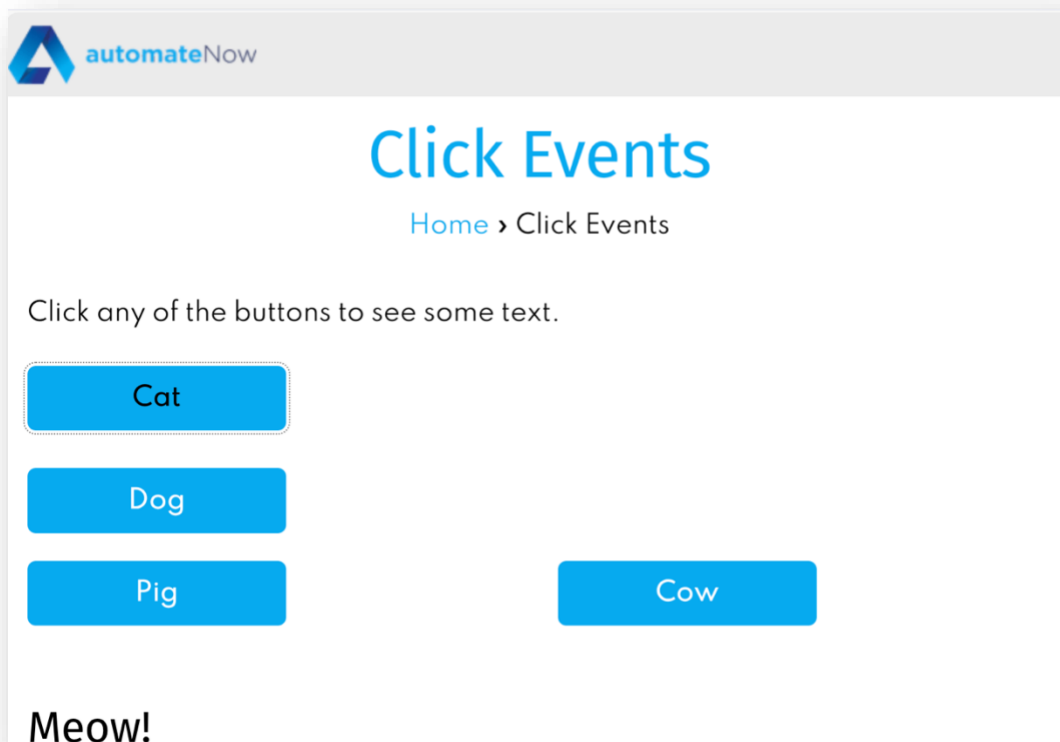


Figure 4 – Click events page

Test case 3:

- Navigate to "Gestures" page by clicking "Gestures" button displayed on the page just opened.
- Determine coordinates to "click", hold and drag the card below and verify if it is moved or not.

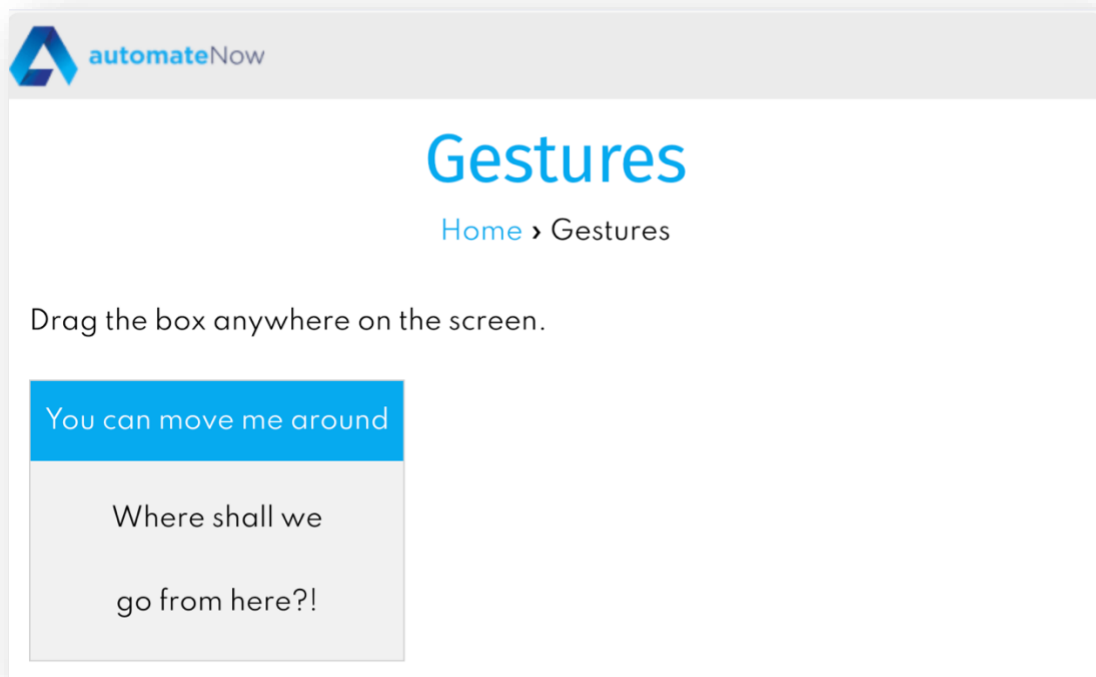


Figure 5 - Gestures

4. REFERENCES

- [1] Daich, G., Price, G., Ragland, B., Dawood, M. "Software Test Technologies Report." STSC, Hill Air Force Base, Utah, August 1994.
- [2] <https://robotframework.org/>
- [3] <https://practice-automation.com/>
- [4] <https://robotframework.org/SeleniumLibrary/SeleniumLibrary.html>

5. REVISION HISTORY

Revision	Date	Author(s)	Description
1.0	Dec 2023	Nguyen Thanh Quan (MSc)	Created