**Assisted Practice: 1.5 Working with External Elements**

This section will guide you to:

* How to handle External elements using Selenium.

**Development Environment**

* Eclipse IDE for Enterprise Java Developers Version Oxygen.3a Release (4.7.3a)
* JavaDevelopment Kit Version 8
* Selenium standalone server Version 3.141.59

This lab has mainly three subsections, namely:

1.5.1 Handling External pop ups.

1.5.2 Handling new Tabs and new Windows.

1.5.3 Pushing the code to your GitHub Repository.

**Step 1.5.1:** Handling External pop ups.

* WebDriver does ability to interact with multiple windows, which includes alerts using the method switchTo. This method allows to switch the control to pop-up while keeping the browser open in the back ground.
* Open Eclipse
* Syntax for handling the various pop ups
* To click on ‘OK’ button in pop up

Syntax: WebDrive driver = new chromeDriver();

driver.switchTo().alert().accept();

* To click on ‘Cancel’ button in pop up

Syntax: WebDrive driver = new chromeDriver();

driver.switchTo().alert().dismiss();

* To Capure the alert message

Syntax: WebDrive driver = new chromeDriver()

driver.switchTo().alert().getText();

* To enter the information

Syntax: WebDrive driver = new chromeDriver()

driver.switchTo().alert().sendKeys(“text”);

* To exit from the popup

Syntax: WebDrive driver = new chromeDriver();

driver.switchTo().alert().close();

**Step 1.5.2:** Handling new Tabs and new Window.

* Opening new tab

Syntax: WebDrive driver = new chromeDriver();  
 driver.findElement(By.id(“xyz”)).sendKeys(Keys.CONTROL + “t”);

* Opening new Window

Syntax: WebDriver driver = new chromeDriver();

driver.findElements(By.id(“xyz”).sendKeys(Keys.CONTROL + “w”);