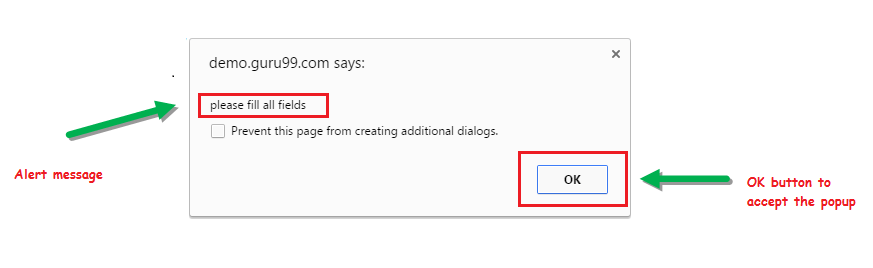
**Alert**

Alert is a small message box which displays on-screen notification to give the user some kind of information or ask for permission to perform certain kind of operation. It may be also used for warning purpose.

Here are few alert types:

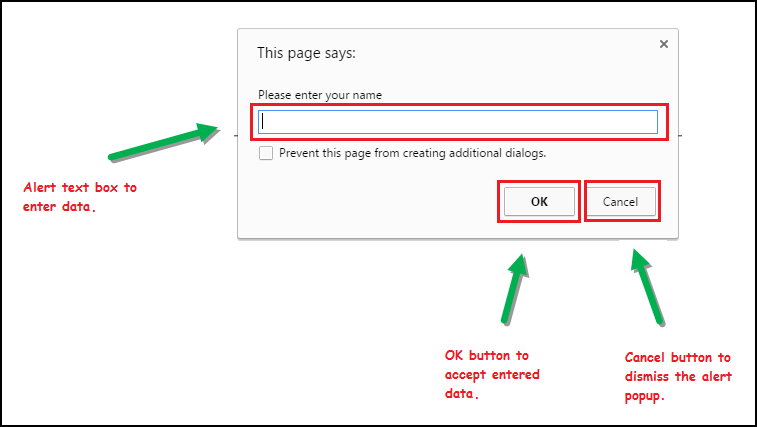
**1) Simple Alert**

This simple alert displays some information or warning on the screen.

[](https://www.guru99.com/images/3-2016/032216_1314_AlertPopuph1.png)

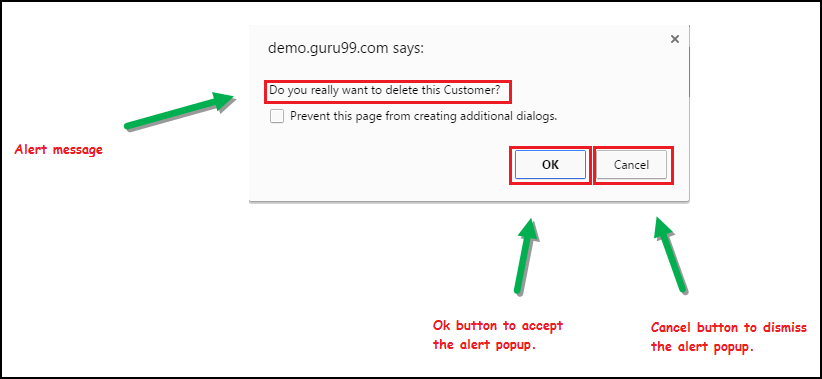
**2) Prompt Alert.**

This Prompt Alert asks some input from the user and selenium webdriver can enter the text using sendkeys(" input…. ").

[](https://www.guru99.com/images/3-2016/032216_1314_AlertPopuph2.png)

**3) Confirmation Alert.**

This confirmation alert asks permission to do some type of operation.

[](https://www.guru99.com/images/3-2016/032216_1314_AlertPopuph3.png)

**How to handle Alert in Selenium WebDriver**

Alert interface provides the below few methods which are widely used in Selenium Webdriver.

1) void dismiss() **// To click on the 'Cancel' button of the alert.**

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

2) void accept() **// To click on the 'OK' button of the alert.**

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

3) String getText**() // To capture the alert message.**

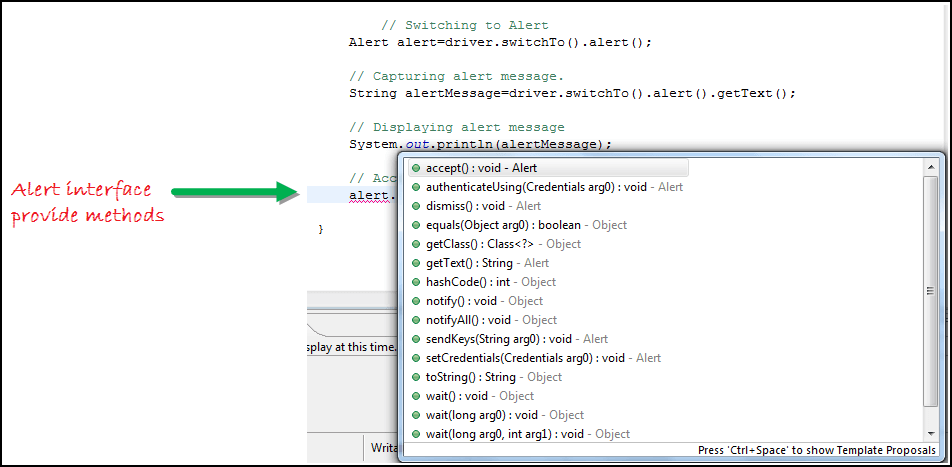
driver.switchTo().alert().getText();

4) void sendKeys(String stringToSend) **// To send some data to alert box.**

driver.switchTo().alert().sendKeys("Text");

You can see a number of Alert methods are displayed as shown in below screen suggested by Eclipse.

We can easily switch to alert from the main window by using Selenium's **.switchTo()** method.

[](https://www.guru99.com/images/3-2016/032216_1314_AlertPopuph4.png)

**How to handle Selenium Pop-up window using Webdriver**

In automation, when we have multiple windows in any web application, the activity may need to switch control among several windows from one to other in order to complete the operation. After completion of the operation, it has to return to the main window i.e. parent window. We will see this further in the article with an example.

In selenium web driver there are methods through which we can handle multiple windows.

**Driver.getWindowHandles();**

To handle all opened windows by web driver, we can use "Driver.getWindowHandles()" and then we can switch window from one window to another in a web application. Its return type is Iterator<String>.

**Driver.getWindowHandle();**

When the site opens, we need to handle the main window by **driver.getWindowHandle()**. This will handle the current window that uniquely identifies it within this driver instance. Its return type is String.

To handle multiple windows in Selenium WebDriver, We follow the following steps.

Now, we will automate the given below scenario to see how to handle multiple windows using Selenium Webdriver.

In this scenario, we will use "Guru99" demo site to illustrate window handling.