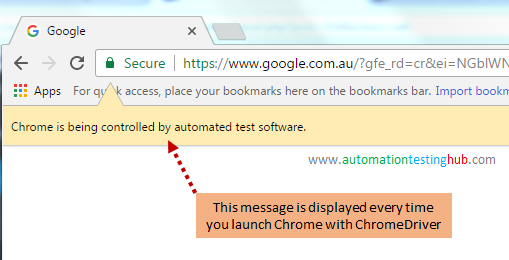
Did you observe that there is a message – **‘Chrome is being controlled by automated test software’** which comes up when you launch browser with ChromeDriver?



This message would come up every time you launch Chrome. To remove this message, you would need to use **disable-infobars argument from ChromeOptions** class in your code. The code snippet for this is given below –



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | public class ChromeTest {  @Test  public void LaunchChrome\_Method1() {  System.setProperty("webdriver.chrome.driver","D:\\Drivers\\chromedriver.exe");  ChromeOptions options = new ChromeOptions();  options.addArguments("disable-infobars");  WebDriver driver = new ChromeDriver(options);  driver.get("<http://www.google.com>");  } |

Try it out and check if you are able to launch Chrome browser without the message. Let us now jump over to the next method of setting up ChromeDriver.

**Click Allow on Show Notifications popup using Selenium Webdriver**

Top of Form

Bottom of Form

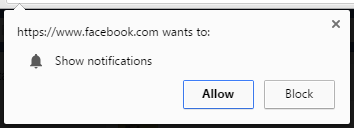
Top of Form

0 votes

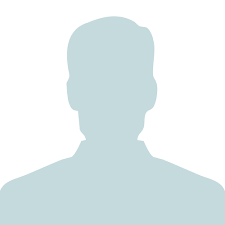
Bottom of Form

Top of Form

I'm trying to login to Facebook. After a successful login, I get a browser popup:

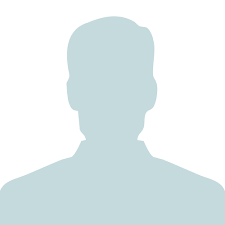
Can I click Allow and proceed forward using selenium webdriver?

* [selenium-webdriver](https://www.edureka.co/community/tag/selenium-webdriver)
* [selenium](https://www.edureka.co/community/tag/selenium)
* [selenium-testing](https://www.edureka.co/community/tag/selenium-testing)

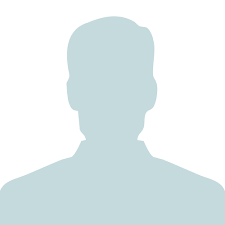
Jun 20, 2018 in [Selenium](https://www.edureka.co/community/selenium) by [Martin](https://www.edureka.co/community/user/376192) • 4,310 points • 8,286 views

Bottom of Form

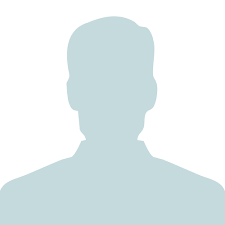
sorry no allow this code because i am using

[commented](https://www.edureka.co/community/7202/click-allow-notifications-popup-using-selenium-webdriver?show=21984#c21984) Sep 24, 2018 by anonymous

Can you please be little more specific? Can't really understand what you're trying to say.

[commented](https://www.edureka.co/community/7202/click-allow-notifications-popup-using-selenium-webdriver?show=22284#c22284) Sep 25, 2018 by [Vardhan](https://www.edureka.co/community/user/119311)

i am facing this issue on windows using python in selenium

[commented](https://www.edureka.co/community/7202/click-allow-notifications-popup-using-selenium-webdriver?show=57130#c57130) Sep 17, 2019 by anonymous

Try this:

alert\_obj = driver.switch\_to.alert

alert\_obj.accept()

https://d1jnx9ba8s6j9r.cloudfront.net/imgver.1539773373/img/userimages/1354066249.jpg[commented](https://www.edureka.co/community/7202/click-allow-notifications-popup-using-selenium-webdriver?show=57131#c57131) Sep 17, 2019 by [Omkar](https://www.edureka.co/community/user/400553)

Top of Form

Bottom of Form

**1 answer to this question.**

Top of Form

Bottom of Form

Top of Form

+1 vote

Bottom of Form

Top of Form

Create a instance of ChromeOptions class

ChromeOptions options = new ChromeOptions();

Then Add chrome switch to disable notification - "**--disable-notifications**"

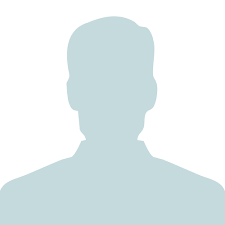
options.addArguments("--disable-notifications");

After that set path for driver exe

System.setProperty("webdriver.chrome.driver","path/to/driver/exe");

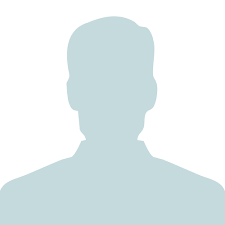
**and then p**ass ChromeOptions instance to ChromeDriver Constructor

WebDriver driver =new ChromeDriver(options);

[answered](https://www.edureka.co/community/7202/click-allow-notifications-popup-using-selenium-webdriver?show=7204#a7204) Jun 20, 2018 by [Samarpit](https://www.edureka.co/community/user/248322) • 5,830 points

Bottom of Form

But the request is not to disable notifications. The request is to click "Allow".

[commented](https://www.edureka.co/community/7202/click-allow-notifications-popup-using-selenium-webdriver?show=24041#c24041) Oct 5, 2018 by pinkninjatester

@pinkninjatester, unfortunately there is no option to do that. Because the above subject is not treated as a browser popup, but infact it is treated as a browser notification.  
  
Disabling this notification should hopefully solve your problem as the execution will continues without any hindrance. Let me know if anything otherwise happens :)

## Automation Script for Selenium

To access newly created Firefox profile in Selenium Webdriver software test, we need to use webdrivers inbuilt class 'profilesIni' and it's method getProfile as shown below.

**Selenium code for the profile**

This is a code to implement a profile, which can be embedded in the selenium code.

ProfilesIni profile = new ProfilesIni();

// this will create an object for the Firefox profile

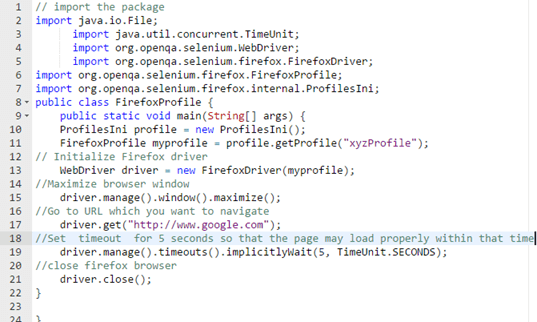
FirefoxProfile myprofile = profile.getProfile("xyzProfile");

// this will Initialize the Firefox driver

WebDriver driver = new FirefoxDriver(myprofile)

Let see the implementation of this code in following examples.

### Firefox Profile Example 1

[](https://www.guru99.com/images/c-sharp-net/052716_0853_FirefoxProf6.png)

// import the package

import java.io.File;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.openqa.selenium.firefox.FirefoxProfile;

import org.openqa.selenium.firefox.internal.ProfilesIni;

public class FirefoxProfile {

public static void main(String[] args) {

ProfilesIni profile = new ProfilesIni();

FirefoxProfile myprofile = profile.getProfile("xyzProfile");

// Initialize Firefox driver

WebDriver driver = new FirefoxDriver(myprofile);

//Maximize browser window

driver.manage().window().maximize();

//Go to URL which you want to navigate

driver.get("http://www.google.com");

//Set timeout for 5 seconds so that the page may load properly within that time

driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);

//close firefox browser

driver.close();

}

}

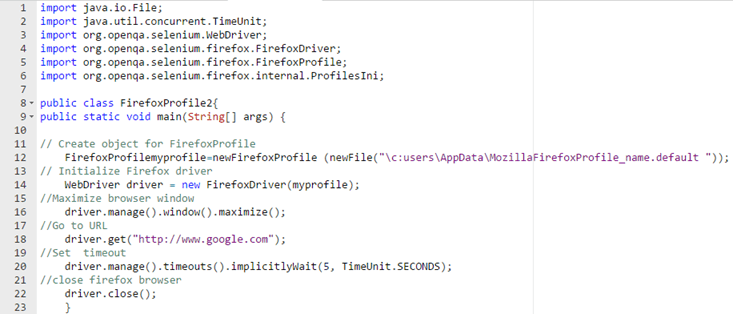
**EXPLANATION FOR THE CODE:**

Below is the explanation of code line by line.

* **Code line 2-7**: First of all we need to import the package required to run the selenium code.
* **Code line 8**: Make a public class "FirefoxProfile."
* **Code line 9**: Make an object (you need to have basic knowledge of oops concepts).
* **Code line 10-11**: We need to initialize Firefox profile with the object of myprofile .
* **Code line 13**: Create object for Firefox
* **Code line 15**: Maximize window.
* **Code line 17**:Driver.get use to navigate to given URL .
* **Code line 19**: Set timeout is used to wait for some time so that browser may load the page before proceeding to next page.
* **Code line 21**:Close Firefox.

Let's see one more example.

### Firefox Profile Example 2

[](https://www.guru99.com/images/c-sharp-net/052716_0853_FirefoxProf7.png)

import java.io.File;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.openqa.selenium.firefox.FirefoxProfile;

import org.openqa.selenium.firefox.internal.ProfilesIni;

public class FirefoxProfile2{

public static void main(String[] args) {

// Create object for FirefoxProfile

FirefoxProfilemyprofile=newFirefoxProfile (newFile("\c:users\AppData\MozillaFirefoxProfile\_name.default "));

// Initialize Firefox driver

WebDriver driver = new FirefoxDriver(myprofile);

//Maximize browser window

driver.manage().window().maximize();

//Go to URL

driver.get("http://www.google.com");

//Set timeout

driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);

//close firefox browser

driver.close();

}

**Explanation for the code**:

Below is the explanation of code line by line.

* **Code line 1-6:** First of all we need to import the package required to run the selenium code.
* **Code line 8**: Make a public class FirefoxProfile 2 .
* **Code line 12**: Make the object of myprofile by referring to the exact path .
* **Code line 14**: Create object for firefox
* **Code line 16**: Maximize window.
* **Code line 18**: Driver.get use to navigate to given URL .
* **Code line 20**: Set timeout is used to wait for some time so that browser may load the page before proceeding to next page.
* **Code line 22**: Close Firefox.

**Summary**:

* Automating Firefox profile makes a lot of sense as such they handles SSL certificates settings.
* Firefox profile can be customized to suit your Selenium automation requirement.
* Firefox profile should be such that it should be easy to load and have some user-specific proxy settings to run a good test.
* To access newly created Firefox profile in Selenium Webdriver software test, we need to use webdrivers inbuilt class 'profilesIni' and its method getProfile.