Selenium ChromeOptions

✅ ChromeOptions in Selenium – Complete Guide with Examples

`ChromeOptions` is a powerful class in Selenium WebDriver used to customize and configure the behavior of Google Chrome at runtime. It helps when you need to:

- Run tests in **headless** mode
- Disable extensions
- Handle **SSL certificates**
- Load Chrome with **specific profiles**
- Set browser **capabilities**, etc.

🔹 1. Basic Syntax

```
java

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.chrome.ChromeOptions;

ChromeOptions options = new ChromeOptions();
WebDriver driver = new ChromeDriver(options);
```

🔹 2. Commonly Used ChromeOptions

✅ a) Headless Mode

Run Chrome in the background without GUI.

```
java

options.addArguments("--headless");
```

✅ b) Disable Notifications

Avoid "Allow/Deny" popups.

```
java

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

✅ c) Start Maximized

Launch Chrome in full screen.

```java
options.addArguments("start-maximized");
```

✅ d) Incognito Mode

Run in private mode.

```java
options.addArguments("--incognito");
```

✅ e) Disable Extensions

Ignore installed Chrome extensions.

```java
options.addArguments("--disable-extensions");
```

✅ f) Disable Infobars

Removes “Chrome is being controlled by automated software” message.

```java
options.setExperimentalOption("excludeSwitches", Arrays.asList("enable-automation"));
```

✅ g) Add Chrome Profile

Load specific user profile.

```java
options.addArguments("user-data-dir=C:/Users/YourUser/AppData/Local/Google/Chrome/User Data");
```

✅ h) Set Custom Download Directory

Auto download files to a custom folder.

```java
Map<String, Object> prefs = new HashMap<String, Object>();
prefs.put("download.default_directory", "D:\\Downloads");
options.setExperimentalOption("prefs", prefs);
```

🔹 3. Combining Multiple Options

java

```java
ChromeOptions options = new ChromeOptions();
options.addArguments("--headless", "--incognito", "--disable-popup-blocking");
```

🔹 4. DesiredCapabilities (Older Use)

`DesiredCapabilities` is now merged with **`ChromeOptions`**, but for older versions:

java

```java
DesiredCapabilities capabilities = DesiredCapabilities.chrome();
capabilities.setCapability(ChromeOptions.CAPABILITY, options);
WebDriver driver = new ChromeDriver(capabilities);
```

> ⚠️ In recent Selenium versions (4+), use **`ChromeOptions`** directly.

🔹 5. Experimental Options

Used to set browser preferences.

java

```java
Map<String, Object> prefs = new HashMap<>();
prefs.put("profile.default_content_setting_values.notifications", 2); // Disable notifications
options.setExperimentalOption("prefs", prefs);
```

🔹 6. Using Options with WebDriverManager

java

```java
WebDriverManager.chromedriver().setup();
ChromeOptions options = new ChromeOptions();
options.addArguments("--start-maximized", "--disable-infobars");
WebDriver driver = new ChromeDriver(options);
```

🔹 7. Headless Mode with Screenshots Example

java

```java
options.addArguments("--headless");
WebDriver driver = new ChromeDriver(options);
driver.get("https://example.com");

File screenshot = ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);
FileUtils.copyFile(screenshot, new File("example.png"));
```

🧪 Hands-on Assignment Ideas

🔸 **Assignment 1: Open Chrome in incognito + headless mode and take a screenshot of `https://www.google.com`.**

🔸 **Assignment 2: Launch Chrome with disabled notifications and verify no notification appears on a site like `https://www.redbus.in`.**

🔸 **Assignment 3: Set a custom download directory and automatically download a sample file from the internet.**

📌 Summary Table

Option	Description
`--headless`	Run Chrome without UI
`--disable-notifications`	Disable browser notifications
`--start-maximized`	Start in full screen
`--incognito`	Start in private mode
`--disable-extensions`	Ignore all extensions
`--disable-infobars`	Remove “controlled by automation” warning
`user-data-dir=...`	Load specific user profile
`prefs`	Set experimental preferences (like downloads)

Would you like me to set up a small project folder with Maven + Selenium + ChromeOptions usage for hands-on practice?

Absolutely! Let's go **step-by-step** through the **most common use cases** of `ChromeOptions` in Selenium, with **Java code + detailed explanations** for each.

✅ 1. Basic Setup Using ChromeOptions

✅ Code:

```
java

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.chrome.ChromeOptions;

public class ChromeOptionsExample {
    public static void main(String[] args) {
        // Create ChromeOptions object
        ChromeOptions options = new ChromeOptions();

        // Launch Chrome with options
        WebDriver driver = new ChromeDriver(options);

        driver.get("https://www.google.com");
        System.out.println("Page Title: " + driver.getTitle());

        driver.quit();
    }
}
```

🔍 Explanation:

- `ChromeOptions` is used to modify browser properties at runtime.
- `ChromeDriver` accepts these options and launches Chrome accordingly.

✅ 2. Launch Chrome in Headless Mode

✅ Code:

```
java

ChromeOptions options = new ChromeOptions();
options.addArguments("--headless", "--window-size=1920,1080");
WebDriver driver = new ChromeDriver(options);
```

🔍 Explanation:

- `--headless`: Runs Chrome without a UI.
- `--window-size=1920,1080`: Helps render pages as if in a visible window (important for screenshots).

✅ 3. Disable Browser Notifications

✅ Code:

```java
ChromeOptions options = new ChromeOptions();
options.addArguments("--disable-notifications");
WebDriver driver = new ChromeDriver(options);
```

🔍 Explanation:

Prevents popup alerts like *"Do you want to allow notifications?"* when visiting certain websites.

✅ 4. Open Chrome in Incognito Mode

✅ Code:

```java
ChromeOptions options = new ChromeOptions();
options.addArguments("--incognito");
WebDriver driver = new ChromeDriver(options);
```

🔍 Explanation:

Opens Chrome in **private browsing** mode, where cookies/history won't be saved.

✅ 5. Start Chrome Maximized

```java
ChromeOptions options = new ChromeOptions();
options.addArguments("start-maximized");
WebDriver driver = new ChromeDriver(options);
```

✅ 6. Disable Infobar "Chrome is being controlled..."

```java
ChromeOptions options = new ChromeOptions();
options.setExperimentalOption("excludeSwitches", Arrays.asList("enable-automation"));
WebDriver driver = new ChromeDriver(options);
```

✅ 7. Set Custom Download Directory

✅ Code:

```java
import java.util.HashMap;
import java.util.Map;
```

```
Map<String, Object> prefs = new HashMap<>();
prefs.put("download.default_directory", "D:\\Downloads");  // Your desired path

ChromeOptions options = new ChromeOptions();
options.setExperimentalOption("prefs", prefs);

WebDriver driver = new ChromeDriver(options);
```

🔍 Explanation:

- Automatically saves downloaded files to the specified folder.
- Useful in automated download test cases.

✅ 8. Full Working Example: Headless + Screenshot

```java
import org.openqa.selenium.*;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.chrome.ChromeOptions;
import java.io.File;
import org.apache.commons.io.FileUtils;

public class HeadlessScreenshot {
    public static void main(String[] args) throws Exception {
        ChromeOptions options = new ChromeOptions();
        options.addArguments("--headless", "--window-size=1920,1080");

        WebDriver driver = new ChromeDriver(options);
        driver.get("https://www.example.com");

        // Take screenshot
        File screenshot = ((TakesScreenshot) driver).getScreenshotAs(OutputType.FILE);
        FileUtils.copyFile(screenshot, new File("example.png"));

        System.out.println("Screenshot taken.");
        driver.quit();
    }
}
```

> 🔧 Make sure to add **`commons-io`** dependency in your **`pom.xml`** if you're using Maven.

✅ 9. Maven Dependency for Selenium + IO (if needed)

```xml
<dependencies>
    <dependency>
        <groupId>org.seleniumhq.selenium</groupId>
        <artifactId>selenium-java</artifactId>
        <version>4.18.1</version>
    </dependency>
    <dependency>
        <groupId>commons-io</groupId>
        <artifactId>commons-io</artifactId>
        <version>2.11.0</version>
    </dependency>
</dependencies>
```

🧪 Assignments (for practice):

1. **Headless Screenshot:**
   - Visit `https://www.wikipedia.org`, take a screenshot in headless mode, and save it.
2. **Incognito Download Test:**
   - Open Chrome in incognito.
   - Download a file and verify it lands in your specified directory.
3. **Handle Popups:**
   - Use `--disable-notifications` to test websites with frequent popups (like `redbus.in`, `makemytrip.com`).

Would you like a complete **Maven project folder** structure or a GitHub repo with all these examples in one place?