Window Handling

How do we identify parent window and child windows?

When a user hits a URL, a webpage opens. This main page is the ***parent window*** i.e the main window on which the user has currently landed and will perform any operation. This is the same webpage that will open when our *Selenium* automation script will execute. All the windows which will open inside your main window will be termed as ***child windows.***

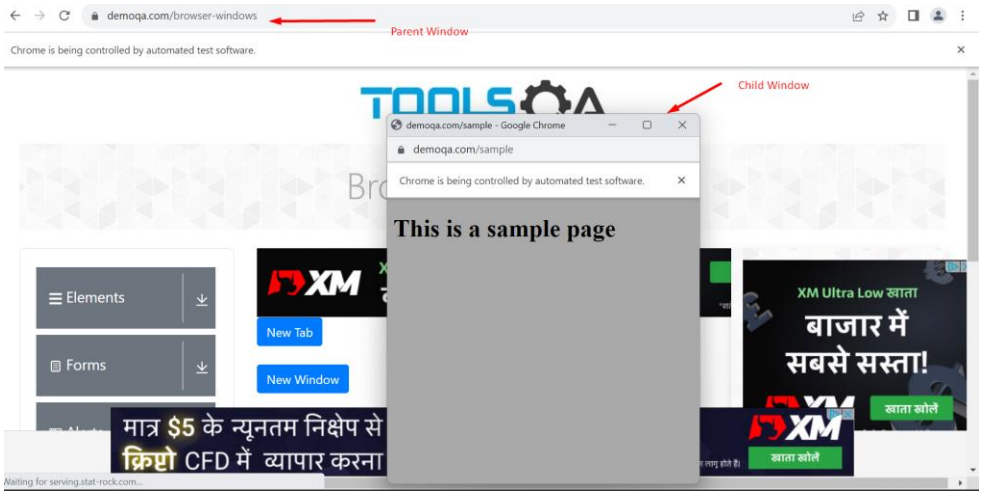
What are the different methods used for window handling in Selenium?

- **getWindowHandle( ):** When a website opens, we need to handle the main window i.e the parent window using driver.getWindowHandle( ); method. With this method, we get a unique ID of the current window which will identify it within this driver instance. This method will return the value of the **String type.**
- **getWindowHandles( ):** To handle all opened windows which are the **child windows** by web driver, we use driver.getWindowHandles( ); method. The windows store in a **Set** of String type and here we can see the transition from one window to another window in a web application. Its return type is **Set <String>.**
- **switchto():** Using this method we perform switch operation within windows.

```
package practice;

import java.util.Iterator;
import java.util.Set;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;

import io.github.bonigarcia.wdm.WebDriverManager;

public class WindowHandling {
	public static void main(String[] args) throws InterruptedException {

    WebDriverManager.chromedriver().setup();
    WebDriver driver = new ChromeDriver();
    driver.manage().window().maximize();
    driver.get("https://demoqa.com/browser-windows");

     driver.findElement(By.id("windowButton")).click();
      String mainwindow = driver.getWindowHandle();
      System.out.println("Main window handle is " + mainwindow);

      Set<String> s1 = driver.getWindowHandles();
```

```
        System.out.println("Child window handle is " + s1);
        Iterator<String> i1 = s1.iterator();

        while (i1.hasNext()) {
           String ChildWindow = i1.next();
              if (!mainwindow.equalsIgnoreCase(ChildWindow)) {
              driver.switchTo().window(ChildWindow);
              WebElement text = driver.findElement(By.id("sampleHeading"));
              Thread.sleep(5000);
              System.out.println("Heading of child window is " + text.getText());
              driver.close();
              System.out.println("Child window closed");
           }
        }
        //  Switch back to the main window which is the parent window.
        driver.switchTo().window(mainwindow);
            }
}
```

Output




```
Console ×
<terminated> WindowHandling [Java Application] C:\Users\Amol\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.4.v20221004-1257\jre\bin\javaw.exe  (13-Aug-2023, 1:40:24 pm – 1:40:39 pm) [pid:
SLF4J: No SLF4J providers were found.
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See https://www.slf4j.org/codes.html#noProviders for further details.
Main window handle is 3B60AE0DE4B1594A6B0174CD2356E231
Child window handle is [3B60AE0DE4B1594A6B0174CD2356E231, 2ED2ED605F6BFA93DF23C12783F5BC43]
Heading of child window is This is a sample page
Child window closed
```

- Launch the website "https://demoqa.com/browser-windows" and click on the windows - "windowbutton".
- String mainwindow = driver.getWindowHandle(): It stores parent window value in a unique identifier of string type.

- Set<String> s = driver.getWindowHandles(): All child windows are stored in a set of strings.
- Iterator<String> i = s.iterator() : Here we will iterate through all child windows.
- if (!MainWindow.equalsIgnoreCase(ChildWindow)) : Now check them by comparing the main window with the child windows.
- driver.switchTo().window (ChildWindow): Switch to the child window and read the heading.
- WebElement text = driver.findElement(By.id("sampleHeading")): Find the element and store in a web element through which we will get the text of heading using gettext() method.