**SELENIUM CONCEPTS**

1. **Common Exceptions in Selenium**

No such element

No such Frame

Firefox not connected

JavaScript Error

1. **Difference between assert and verify**

Both **Assert and Verify** commands are used to find whether a given input is present or not on the webpage.

When an “**assert**” command fails, the test execution will be aborted.

Where if a “**verify**” fails, the test will continue executing and logging the failure.

1. **Difference between driver.close and driver.quit methods**

webDriver.Close() - Close the browser window that the driver has focus of

webDriver.Quit() - Calls Dispose()

webDriver.Dispose() Closes all browser windows and safely ends the session

1. **Different types of waits or synchronization in selenium webdriver**

**Conditional Synchronization:**

We specify a condition along with timeout value, so that tool waits to check for the condition and then come out if nothing happens.

**Unconditional Synchronization:**  
In this we just specify timeout value only. We will make the tool to wait until certain amount of time and then proceed further.

**Implicit Wait:**

An implicit wait is to tell Web Driver to poll the DOM for a certain amount of time when trying to find an element or elements if they are not immediately available.

**Explicit Wait:**

We need to define a wait statement for certain condition to be satisfied until the specified timeout period. If the Webdriver finds the element within the timeout period the code will get executed.

1. **How to handle multiple windows in selenium webdriver**

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class Multi {

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

System.setProperty("webdriver.chrome.driver","C:\\Selenium\\chromedriver.exe");

WebDriver driver = new ChromeDriver();

driver.get("http://seleniumhq.org/");

driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

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

driver.findElement(By.linkText("Documentation")).click();

System.out.println(driver.getCurrentUrl());

driver.navigate().back();

System.out.println(driver.getCurrentUrl());

Thread.sleep(30000);

driver.navigate().forward();

System.out.println("Forward");

Thread.sleep(30000);

driver.navigate().refresh();

}

}

**How to save screen shots using selenium webdriver**

import java.io.IOException;

import org.apache.commons.io.FileUtils;

import org.openqa.selenium.OutputType;

import org.openqa.selenium.TakesScreenshot;

import java.io.File;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.testng.annotations.Test;

public class Screenshot {

@Test

public void TestJavaS1()

{

System.setProperty("webdriver.chrome.driver","C:\\Selenium\\chromedriver.exe");

WebDriver driver = new ChromeDriver();

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

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

File src= ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);

try {

FileUtils.copyFile(src, new File("C:/Selenium/error.png"));

}

catch (IOException e)

{

System.out.println(e.getMessage());

}

}

}