|  |
| --- |
| 1. **What are limitations in selenium webdriver** |
| Ex:  1. It only supports web based applications and does not support windows based application.  2. Its an open source tool so in case of any technical issue we need to rely on the selenium community forums to get issue resolved.  3. Need to know at least one of the supported language very well in order to automate the application successfully.  4. No inbuilt reporting capability so you need plugins like JUnit and TestNG for test reports.  5. Lot of challenges with IE browser.  6. It needs very much expertise resources. The resource should also be very well versed in framework architecture.  7. It is difficult to test Image based application.  8. It does not support built in add-ins support.  9. It does not provide any built in IDE for script generation and it need other IDE like Eclipse for writing scripts.  10. Its script creation time is bit high.  11. It does not support file upload facility.  12. Selenium partially supports for Dialog boxes. |
|  |
| 1. **Installing/configure selenium** |
| Ex:  1. Download and install latest version of Eclipse.  2. Install java.  3. Verify Java installation.  4. Download WebDriver Jar Files.  5. Start Eclipse and configure it with selenium  6. Select WorkSpace on eclipse start up  7. Create new java project from File > New > Project > Java Project and give your project name  8. Create new package  9. Create new class  10. Add external jar file to java build path  11. Right click on project 'testproject' > Select Properties > Select Java build path > Navigate to Libraries tab  Click on add external JARs button > select both .jar files from D:\selenium-2.33.0.  Click on add external JARs button > select all .jar files from D:\selenium-2.33.0\libs |
|  |
| 1. **What are different ways of locating elements in selenium** |
| Ex: There are different ways in selenium to locate elements:  **1. Using Id**  <div id=“name”>  driver.findElement(By.id(“name”));  **2. Using Class Name**  <span class=“even”>  driver.findElement(By.className(“even”));  **3. Using Tag Name**  <img src=“./logo.png”>  driver.findElement(By.tagName(“img”));  **4. Using Name**  <input name=“color” value=”blue”>  driver.findElement(By.name(“color”));  **5. Using Link Text**  <a href=“http://xyz/“>Software Testing</a>  driver.findElement(By.linkText(“Software Testing”));  **6. Using Partial Link Text**  <a href=”http://www.xyz.com/”>Software Testing</a>  driver.findElement(By.partialLinkText(“Testing”)); |
|  |
|  |
| 1. **Which is fastest way to identify elements in web page?** |
| Ex: The Best and Fastest Way to Find Elements Using Selenium WebDriver:  1. ID selectors (By.ID – Matches by @id attribute)  2. CSS and Name selectors (Matches by CSS selector or @name attribute)  3. XPath locators (Matches with arbitrary XPath expression) |
|  |
| 1. **What is absolute path and relative path in xpath** |
| Ex:  Absolute path: html/body/div[1]/section/div[1]/div/div/div/div[1]/div/div/div/div/div[3]/div[1]/div/h4[1]/b  Relative xpath: //\*[@class='featured-box']//\*[text()='Testing'] |
|  |
| 1. **Different types of waits or synchronization in selenium webdriver** |
| Ex:  Explicit Waits:  This can be achieved in 2 ways.  1. Thread.sleep:  Thread.sleep waits the specified time irrespective of the object state.  Ex: Thread.sleep(30000);  2. WebDriverWait:  WebDriverWait wait = new WebDriverWait(driver, 30);  WebElement o\_element = wait.until(ExpectedConditions.elementToBeClickable(By.id("Object Id")));  Implicit Waits:  WebDriver driver = new FirefoxDriver();  driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);  driver.get("Your URL");  WebElement o\_Element = driver.findElement(By.id("Object Id")); |
|  |
| 1. **How to save screen shots using selenium webdriver** |
| Ex: File screenshot = ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE); |
|  |
| 1. **How to handle multiple windows in selenium webdriver** |
| Ex:  Selenium provides 2 methods getWindowHandle() and getWindowHandles() to deal with the multiple browser windows.  getWindowHandle() – This method returns the current browser window handle id that the web-driver is currently holding.  getWindowHandles() – This method returns set of browser window handle ids that are invoked/opened by the web-driver.  And we use switchTo().window(window\_handle\_id) to switch to different browser windows of the web-driver. |
|  |
| 1. **How to launch webpage using chrome driver** |
| Ex:  System.setProperty("webdriver.chrome.driver","C://chromedriver.exe");  WebDriver driver=new ChromeDriver(); // this line launch chrome browser  driver.get("http://www.google.co.in"); |
|  |
| 1. **What is desired capabilities in selenium webdriver** |
| Ex:  The desired capability is a series of key/value pairs that stores the browser properties like browsername, browser version, the path of the browser driver in the system, etc. to determine the behaviour of the browser at run time.  Desired capability can also be used to configure the driver instance of Selenium WebDriver.  We can configure driver instance like FirefoxDriver, ChromeDriver, InternetExplorerDriver by using desired capabilities.  Desired Capabilities are more useful in cases like:  In mobile application automation, where the browser properties and the device properties can be set.  In Selenium grid when we want to run the test cases on a different browser with different operating systems and versions.  importorg.openqa.selenium.WebDriver;  importorg.openqa.selenium.ie.InternetExplorerDriver;  importorg.openqa.selenium.remote.DesiredCapabilities;  public class IEtestforDesiredCapabilities  {  public static void main(String[] args)  {  //it is used to define IE capability  DesiredCapabilities capabilities = DesiredCapabilities.internetExplorer();  capabilities.setCapability(CapabilityType.BROWSER\_NAME, "IE");  capabilities.setCapability(InternetExplorerDriver.  INTRODUCE\_FLAKINESS\_BY\_IGNORING\_SECURITY\_DOMAINS,true);  System.setProperty("webdriver.ie.driver", "C:\\IEDriverServer.exe");  //it is used to initialize the IE driver  WebDriver driver = newInternetExplorerDriver(capabilities);  driver.manage().window().maximize();  driver.get("http://gmail.com");  driver.quit();  }  } |
|  |
| 1. **How to set language while opening website** |
| Ex:  **Using Firefox Browser :**  FirefoxProfile profile = new FirefoxProfile();  //setting the locale french : ‘fr’  profile.setPreference(“intl.accept\_languages”,”fr”);  driver = new FirefoxDriver(profile);  driver.get(“http://google.co.in&#8221;);  **Using Chrome Browser :**  System.setProperty(“webdriver.chrome.driver”,”D:/DollarArchive/chromedriver.exe”);  ChromeOptions options = new ChromeOptions();  options.addArguments(“–lang= sl”);  ChromeDriver driver = new ChromeDriver(options);  driver.get(“http://google.co.in&#8221;);  Unfortunately it won’t work for IE browser, We need to change it manually. |
|  |
| 1. **How to handle windows based popups (upload and dropdown)** |
| Ex:  String parentWindowHandler = driver.getWindowHandle(); // Store your parent window  String subWindowHandler = null;  Set<String> handles = driver.getWindowHandles(); // get all window handles  Iterator<String> iterator = handles.iterator();  while (iterator.hasNext()){  subWindowHandler = iterator.next();  }  driver.switchTo().window(subWindowHandler); // switch to popup window  // perform operations on popup  driver.switchTo().window(parentWindowHandler); // switch back to parent window |
|  |
| 1. **Write code to verify any application login page is working or not** |
| (u should write code to use textbox, button click events) |
| Ex:  public class LoginPageTest extends IntegrationTest {  private HtmlUnitDriver driver;  @Before  public void setup() throws MalformedURLException, UnknownHostException{  driver = new HtmlUnitDriver(true);  driver.get(System.getProperty("login.url"));  }  @Test  public void testAuthenticationFailureWhenProvidingBadCredentials(){  driver.findElement(By.id("username")).sendKeys("fakeuser");  driver.findElement(By.id("password")).sendKeys("fakepassword");  driver.findElement(By.id("login")).click();  assertTrue(driver.getCurrentUrl().endsWith("failed"));  }  @Test  public void testAuthenticationSuccessWhenProvidingCorrectCredentials(){  driver.findElement(By.id("username")).sendKeys("validuser");  driver.findElement(By.id("password")).sendKeys("validpassword");  driver.findElement(By.id("login")).click();  assertTrue(driver.getCurrentUrl().endsWith("/<name\_of\_webapp>/"));  }  } |
|  |
| 1. **How to select items from dropdown/select box** |
| Ex:  *# filename: dropdown.rb*  ...  run do  @driver.get 'http://the-internet.herokuapp.com/dropdown'  dropdown = @driver.find\_element(id: 'dropdown')  select\_list = Selenium::WebDriver::Support::Select.new(dropdown)  select\_list.select\_by(:text, 'Option 1')  selected\_option = select\_list.selected\_options[0].text  expect(selected\_option).to eql 'Option 1'  end |
|  |
| 1. **How to know if checkbox is checked or not in webpage** |
| Ex:  isChecked = driver.findElement(By.xpath("//input[@type='checkbox']"));  if (isChecked = false){  check the box;  }else}  do nothing;  } |
|  |
| 1. **Tell me code to pass values from parent window to child window** |
| Ex:  Passing Data between Parent and Child Window. You can get the Parent Window Value in Child Window and set different Value to Parent Window,  Also you can Disable Parent Element (Button) from Child Window by using pure JavaScript function. Following are the code snippet shows how you can achive the same. 1. Create parent.jsp as mentioned below. <html>     <head>         <title>Parent Window</title>     </head>     <body>     <form NAME="form1" METHOD="POST">     <input type="text" id="data" value="23444" />     <input type="submit" id="submit1" name="submit1" value="Open Child Popup window" onclick="javascript:openChildWindow()" />     </form>     </body>     <script>     function openChildWindow() {         window.open('http://localhost:8080/jsp-multipleform/child.jsp','childWindow','width=400,height=400');     }     </script> </html>  2. Create child.jsp as mentioned below. <html>     <head>         <title>Child Window</title>     </head>     <body>     <form NAME="form2" METHOD="POST">         <div id="mainDiv">         <!--  <input type="text" id="data2"/> -->         <input type="submit" id="submit2" name="submit2" value="Set Parent Element" onclick="javascript:initializeMainDiv()" />         </div>     </form>     </body>     <script>     function initializeMainDiv() {         document.getElementById("mainDiv").innerHTML = "Parent window data field value is: " +         window.opener.document.getElementById("data").value;         window.opener.document.getElementById("data").value="99999";         window.opener.document.getElementById('submit1').disabled = true;     }     </script> </html> |
|  |
| 1. **write code to find out if all links are working or not** |
| Ex:  package Selenium.Locators;  import java.util.List;  import java.net.MalformedURLException;  import java.net.URL;  import java.util.ArrayList;  import java.util.HashSet;  import java.util.Iterator;  import java.util.NoSuchElementException;  import java.util.Set;  import java.util.concurrent.TimeUnit;  import org.openqa.selenium.By;  import org.openqa.selenium.JavascriptExecutor;  import org.openqa.selenium.Keys;  import org.openqa.selenium.WebDriver;  import org.openqa.selenium.WebDriver.Navigation;  import org.openqa.selenium.WebDriver.Options;  import org.openqa.selenium.WebElement;  import org.openqa.selenium.firefox.FirefoxDriver;  import org.openqa.selenium.firefox.FirefoxProfile;  import org.openqa.selenium.support.ui.\*;  import com.google.common.base.Function;  import com.google.common.base.Predicate;  import Selenium.utils.\*;  //import org.openqa.selenium.\*;  //import org.openqa.selenium.firefox.\*;  public class program {    public static List findAllLinks(WebDriver driver)    {    List elementList = new ArrayList();    elementList = driver.findElements(By.tagName("a"));    elementList.addAll(driver.findElements(By.tagName("img")));    List finalList = new ArrayList();    for (WebElement element : elementList)    {    if(element.getAttribute("href") != null)    {    finalList.add(element);      }    }    return finalList;    }  public static String isLinkBroken(URL url) throws Exception    {    //url = new URL("http://yahoo.com");  String response = "";  HttpURLConnection connection = (HttpURLConnection) url.openConnection();  try  {      connection.connect();       response = connection.getResponseMessage();        connection.disconnect();      return response;  }  catch(Exception exp)  {  return exp.getMessage();    }    }    public static void main(String[] args) throws Exception {  // TODO Auto-generated method stub  FirefoxDriver ff = new FirefoxDriver();  ff.get("http://toolsqa.wpengine.com/automation-practice-switch-windows/");    //ff.get("http://www.yahoo.com/");        List allImages = findAllLinks(ff);      System.out.println("Total number of elements found " + allImages.size());        for( WebElement element : allImages){       try         {          System.out.println("URL: " + element.getAttribute("href")+ " returned " + isLinkBroken(new URL(element.getAttribute("href"))));         //System.out.println("URL: " + element.getAttribute("outerhtml")+ " returned " + isLinkBroken(new URL(element.getAttribute("href"))));         }         catch(Exception exp)         {         System.out.println("At " + element.getAttribute("innerHTML") + " Exception occured -&gt; " + exp.getMessage());       }        }      }    } |
|  |
| 1. **Write code on how to use javascriptexecutor?** |
| Ex: |
| CalculatorExampleTest:   |  |  | | --- | --- | |  | package Javascript;  import org.junit.AfterClass;  import org.junit.BeforeClass;  import org.junit.Test;  import org.openqa.selenium.By;  import org.openqa.selenium.JavascriptExecutor;  import org.openqa.selenium.WebDriver;  import org.openqa.selenium.WebElement;  import org.openqa.selenium.firefox.FirefoxDriver;  import static org.hamcrest.MatcherAssert.assertThat;  import static org.hamcrest.Matchers.is;   public class CalculatorExampleTest {      static WebDriver driver;      private static String url = "http://www.anaesthetist.com/mnm/javascript/calc.htm";      //Setup Driver      @BeforeClass      public static void setupTest() {          driver = new FirefoxDriver();          driver.navigate().to(url);          driver.manage().window().maximize();      }      @Test      public void calculatorJavaScriptTest() {          //1-) Click "9"          driver.findElement(By.name("nine")).click();         //2-) Click "+"          driver.findElement(By.name("add")).click();           //3-) Click "3"          driver.findElement(By.name("three")).click();           //4-) Declare JavaScriptExecutor and call "Calculate()" function          JavascriptExecutor js =(JavascriptExecutor)driver;          js.executeScript("Calculate();");           //5-) Assert that result is 12          WebElement result = driver.findElement(By.name("Display"));          assertThat(result.getAttribute("value"), is("12"));      }       //Close Driver      @AfterClass      public static void quitDriver() {          driver.quit();      }  } | |
| 1. **Difference between assert and verify?** |
| Ex:  **Assert command in selenium:**   * When an "assert" command fails then test execution will be aborted. * Assert is best used when the check value has to pass for the test to be able to continue to run. Like a log in. **Verify command in selenium:** * When a "verify" command fails then test will continue executing and logging the failure. * Verify is best used to check non critical things. Like the presence of a headline element.   package testWordpress;    import org.testng.Assert;  import org.testng.annotations.Test;  public class AssertionsTest {   @Test   public void testMultiply() {   System.out.println(&quot;Before Error &quot;);   try{   Assert.assertEquals(21, multiply(10, 5));   }catch(Throwable t){   // recovered   // java code to fail the test   System.out.println(&quot;After Error &quot;);   }   }    public int multiply(int x, int y) {   return x / y;   }  } |
|  |
| 1. **Difference between driver.close and driver.quit methods?** |
| Ex: |
| **Driver.close()**  Selenium webdriver providing above method i.e. close() which is useful in closing current browser/window which is currently open by Selenium WebDriver, Let’s see below example to understand clearly.  **public** **class** BrowserClose(){  *//Declare Selenium WebDriver variables*  **public** **static** WebDriver driver;  **public** **static** String baseUrl;   @BeforeTest  **public** **void** setup(){   driver = **new** FirefoxDriver();  baseUrl="http://www.wikishown.com";  driver.manage().window().maximize();  }   @Test  **public** **void** wikiShown(){  *//Navigate to above website urk*  driver.Navigate().to(baseUrl);  driver.manage().timeouts().implecitlyWait(30,TimeUnit.SSECONDS);   }  @AfterTest  **public** **void** tearDown(){  driver.close();  }  }  **Driver.quit()**  It will close all the browsers/window which is open by selenium Webdriver, for this best example is cross browser testing of selenium test means running test scripts in different browsers at a time.  **public** **class** BrowserClose(){  *//Declare Selenium WebDriver variables*  **public** **static** WebDriver driver;  **public** **static** String baseUrl;   @BeforeTest  **public** **void** setup(){   driver = **new** FirefoxDriver();  baseUrl="http://www.wikishown.com";  driver.manage().window().maximize();  }  @Test  **public** **void** wikiShown(){  *//Navigate to above website urk*  driver.Navigate().to(baseUrl);  driver.manage().timeouts().implecitlyWait(30,TimeUnit.SSECONDS);   }  @AfterTest  **public** **void** tearDown(){  driver.quit();  }  } |
| 1. **Common exceptions in selenium?** |
| Ex: |
|  |
| how to handle Ajax calls in selenium? |
| Ex: write code |
|  |
| we have webtable, need to click on second row from table. |
| when we click on child will be populated. first column in primary column for each row. |
| tell me steps to verify child form has proper data or not |
| Ex: |
|  |
| How to assign the value to textbox other than sendkeys method? |
|  |
| Selenium grid, how to execute scripts on multiple browser |
|  |
| Write code for drag/drop in selenium |
|  |
| Write code for right click in seelnium |
|  |
| Write code fro scroll to specific element |
|  |