

<p>__LOCATORS</p> <p>ID:WE item = driver.findElement(By.id("user"));</p> <p>Name:WE locator = driver.findElement(By.name("admin"));</p> <p>Link Text:WE item = driver.findElement(By.linkText("How to use locators"));</p> <p>Partial Link Text:WE item=driver.findElement(By.partialinkText("How to use locators"));</p> <p>Tag Name:String[] linkTexts = new String[linkElements.size()];</p> <p>Class Name:WE element=driver.findElement(By.className("sample"));</p> <p>CSS Selector:WE CheckElements = driver.findElements(By.cssSelector("input[id=email]"));</p> <p>XPath:</p> <p>1-What is absolute XPath: HTML/head/body/table/tr/td</p> <p>Single slash '/' anywhere in xpath signifies to look for the element immediately inside the parent element.</p> <p>2-What is relative XPath: //tag[@attribute='value']</p> <p>Double slash '/' signifies to look for any child or nested-child element inside the parent element.</p> <p>3-CSS vs XPath?</p> <p>Css is html expression XPath java Script expression.</p> <p>Css selectors are often easier to read than XPath.</p> <p>CSS selectors work faster than XPath.</p> <p>Xpath we can go back and forward it is benfit They have completely different syntax.</p> <p>4-How do we use xpath?</p> <p>Firefox has Firebug and Fire path tool.It provides default Xpath and scc element locator very easy to use and modify.</p> <p>5-Css base form?</p> <p>tag[attribute="value"]</p> <p># => Id . => Class</p> <p>6 Using wildcards in CSS Selectors:</p> <p>* => Represents the starting text</p> <p>\$ => Represents the ending text</p> <p>*\$ => Represents the text contained</p> <p>Syntax: tag[attribute~special character~="value"]</p>	<p>CSS XPTH TIPS</p> <p>1-XPath:Syntax: //tag[@attribute='value']</p> <p>Relative xpath using single '/' for Login link</p> <p>//div[@id='navbar']/div/div/div/ul/li[2]/a</p> <p>Relative xpath using double '/' for Login link.</p> <p>//div[@id='navbar']/ul/li[2]/a</p> <p>Don't use "*", always use the tag name.</p> <p>Using Text of the element to build xpath</p> <p>Finding Login link:</p> <p>//div[@class='homepage-hero']/a[text()='Enroll now']</p> <p>Using Contains to find the elements:</p> <p>//tag[contains(attribute, 'value')]</p> <p>Finding Login link:</p> <p>//div[@id='navbar']/a[contains(text(),'Login')] //div[@id='navbar']/a[contains(@class,'navbar-link') and contains(@href,'sign_in')]</p> <p>Using Starts-With to find the elements:</p> <p>//tag[starts-with(attribute, 'value')]</p> <p>Finding Login link:</p> <p>//div[@id='navbar']/a[starts-with(@class,'navbar-link')]</p> <p>Parent Syntax: xpath-to-some-element/parent::<tag></p> <p>Preceding Sibling (coming before in position)</p> <p>Syntax: xpath-to-some-element/preceding-sibling::<tag></p> <p>Following Sibling</p> <p>Syntax: xpath-to-some-element/following-sibling::<tag></p> <p>2-Using wildcards in CSS Selectors Example:</p> <p>input[class='inputs'] -> Only 1 matching node</p> <p>input[class^='inputs'] -> Two matching nodes</p> <p>input[class='displayed-class'] -> No matching nodes input</p> <p>[class\$='class'] -> One matching node input</p> <p>[class*='displayed-class'] -> One matching node</p>	<p>NAVIGATE ACTIONS- RADIO BUTTONS</p> <p>driver.get(baseUrl + "");//</p> <p>Thread.sleep(3000);</p> <p>driver.quit();</p> <p>1-Navigate method</p> <p>driver.navigate().to(urlToNavigate);</p> <p>driver.navigate().back();</p> <p>driver.navigate().forward();</p> <p>driver.navigate().refresh();</p> <p>2-How to handle Radio Buttons</p> <p>1.First we verify default value (isSelected)</p> <p>2.We select and verify if it is selected</p> <p>3.We select different option and verify previous option if not</p> <p>For Listing Radio Buttons</p> <p>Store in boolean: boolean isChecked = false;</p> <p>Find element list: List<WebElement> radioButtons=driver.find.....</p> <p>Take size :int size = radioButtons.size();</p> <p>Do for each loop: for (int i=0; i<size; i++) {</p> <p>isChecked = radioButtons.get(i).isSelected();</p> <p>if (!isChecked) {</p> <p>radioButtons.get(i).click();</p> <p>3.Element Displayed</p> <p>Textbox. —HideButton — Dropdown</p> <p>Example :</p> <p>WebElement textBox = driver.findElement(By.id("displayed-text"));</p> <p>System.out.println("Text Box Displayed: " + textBox.isDisplayed());</p>	<p>DROP DONW HANDLING</p> <p>1-How to handle drop down box</p> <p>Select / dropdown</p> <p>1- inspect it</p> <p>2- If tag is Select -> locate that webElement</p> <p>3- create Select class and pass webElement</p> <p>3 options to select:</p> <p>1.selectByIndex(). :sel.selectByIndex(2);</p> <p>2.selectByValue(). :sel.selectByValue("benz");</p> <p>3.selectByVisibleText() :sel.selectByVisibleText("BMW");</p> <p>→ To get all options : getOptions()</p> <p>→ To find out multiple selectable dropdown select.isMultiple() -> returns boolean</p> <p>2-How to handle drop down box (second explanation *****</p> <p>We can use the select class to test the dropdown menus.</p> <p>Select class provides method for selecting options, getting all the options, getting the selected option. But the select has to be passed a web element as constructor and that web element should have the html tag select</p> <p>If the dropdown list does not have select tag:</p> <p>1. We have to locate the dropdown</p> <p>2. We have to click on that dropdown</p> <p>3. We have to locate the option that we want to select separately</p> <p>4. We click on the option</p> <p>3-Drop down multiple select (do for each loop)</p> <p>List<WebElement> selectedOptions = sel.getAllSelectedOptions();</p> <p>for (WebElement option : selectedOptions) {</p> <p>System.out.println(option.getText()); }</p>
<p>SWITCH WINDOW</p> <p>1-How to switch tabs/windows using selenium?</p> <p>For selenium there is no difference between window and a tab.</p> <p>To switch between them we need to pass the window handles to the driver.switchTo.Window().</p> <p>We can get the window handles by driver.getWindowHandles() method.</p> <p>To verify a certain tab, we need to first switch to the tab. Selenium can only control/see the current tab</p> <p>@Tes. public void test() throws InterruptedException {</p> <p>// Get the handle</p> <p>String parentHandle = driver.getWindowHandle();</p> <p>System.out.println("Parent Handle: " + parentHandle);</p> <p>// Find Open Window button</p> <p>WebElement openWindow=driver.findElement(By.id</p> <p>// Get all handles. ("openwindow"));</p> <p>openWindow.click();</p> <p>Set<String> handles = driver.getWindowHandles();</p> <p>// Switching between handles</p> <p>for (String handle: handles) {</p> <p>System.out.println(handle); }</p> <p>// Switch back to the parent window. }</p> <p>2-Switch between handles ?</p> <p>for (String handle: handles) {</p> <p>System.out.println(handle);</p> <p>if (handle.equals(parentHandle)) {</p> <p>driver.switchTo().window(handle);</p> <p>WebElement searchBox = driver.findElement(By.id("search-courses"));</p> <p>searchBox.sendKeys("python");</p> <p>driver.close();</p> <p>break;</p> <p>} // Switch back to the parent window</p>	<p>SWITCH IFRAME -ALERT POP UP</p> <p>1-How to switch frames using selenium?</p> <p>We can switch to frames using three methods</p> <p>1-By passing the frame web element to the switchToFrame() method</p> <p>2-By passing the id or name of the frame element to the SwitchTo method</p> <p>3- By passing index to the driver.switchToFrame method</p> <p>When yo work the framework don't forgot Maximize the browser's window</p> <p>@Test public void test() throws InterruptedException {</p> <p>// Switch to frame by Id</p> <p>driver.switchTo().frame("courses-iframe");</p> <p>WebElement searchBox = driver.findElement(By.id("abcs"));</p> <p>searchBox.sendKeys("python");</p> <p>//driver.switchTo().defaultContent();</p> <p>driver.findElement(By.id("name")).sendKeys("Test Successful");</p> <p>2-How to handle SwitchAlert? Two kind of alert?</p> <p>Alert pop up: Only pop up say something just you should say ok</p> <p>Confirm pop up: Confirm usually ask ok or cancel button</p> <p>- We can not locate the popup because it is not element, it is java script function we can not find inside the html code, we need different class.</p> <p>-You can find in Script tag, like alert function, displayed confirm</p> <p>Steps:Locate related button-crest alert class, it is going to give option</p> <p>@Test. public void test() throws InterruptedException {</p> <p>driver.findElement(By.id("name")).sendKeys("Anil");</p> <p>driver.findElement(By.id("alertbtn")).click();</p> <p>Alert alert = driver.switchTo().alert();</p> <p>alert.accept(); dismiss/sendkey/ }</p> <p>3-Java script execute get the current iframe</p> <p>JavaScriptExecutor jsExecutor = (JavaScriptExecutor)driver;</p> <p>String currentFrame =jsExecutor.executeScript("return self.name");</p> <p>Tips: Firefox fire path tools if there is airfare shows the iframe left side on the search box easy to see and locate</p>	<p>ACTIONS CLASS</p> <p>1-How to use Action class what is common action ?</p> <p>Syntax: Actions action = new Actions(driver);</p> <p>→ Move to element -> moveToElement()</p> <p>→ right click -> contextClick()</p> <p>→ drag and drop -> dragAndDrop(src, target);</p> <p>→ double click-> doubleClick()</p> <p>→ click and hold -> clickAndHold()</p> <p>→ release() -> let go mouse off.</p> <p>build() -> builds all the steps in Actions</p> <p>perform() -> Executes those steps as ONE statement</p> <p>action.contextClick(); -> right click</p> <p>A- Example drag and drop actions</p> <p>driver.switchTo().frame(0);</p> <p>WebElement fromElement = driver.findElement(By.id("draggable"));</p> <p>WebElement toElement = driver.findElement(By.id("droppable"));</p> <p>Actions action = new Actions(driver);</p> <p>action.clickAndHold(fromElement);</p> <p>action.moveToElement(toElement)</p> <p>action.release().build().perform();</p> <p>B-Mouse hover</p> <p>driver.get(baseUrl);</p> <p>js.executeScript("window.scrollBy(0, 600)");</p> <p>WebElement mainElement = driver.findElement(By.id("mouseover"));</p> <p>Actions action = new Actions(driver);</p> <p>action.moveToElement(mainElement).perform();</p> <p>action.moveToElement(subElement).click().perform();</p> <p>C-Slider:</p> <p>Actions action = new Actions(driver);</p> <p>action.dragAndDropBy(element, 100, 0).perform();</p>	<p>SELENIUM ADVANCE</p> <p>1- Finding all link</p> <p>Aseptically during the regression testing finding all link buttons with those method its very useful this method can e use It help the Broke link , link moved different location problem</p> <p>@Test 3= Than find the all link and test it</p> <p>public void testFindLinks() {</p> <p>driver.get(baseUrl);</p> <p>SearchPage navigateToFlightsTab(driver);</p> <p>List<WebElement> linksList = clickableLinks(driver);</p> <p>for (WebElement link : linksList) {</p> <p>String href = link.getAttribute("href");</p> <p>try {</p> <p>System.out.println("URL " + href + " returned " + linkStatus(new URL(href)));</p> <p>} catch (Exception e) {</p> <p>System.out.println(e.getMessage()); }</p> <p>1-Find all clickable link /</p> <p>public static List<WebElement> clickableLinks(WebDriver driver) {</p> <p>List<WebElement> linksToClick = new ArrayList<WebElement>();</p> <p>List<WebElement> elements = driver.findElements(By.tagName("a"));</p> <p>elements.addAll(driver.findElements(By.tagName("img"));</p> <p>for (WebElement e : elements) {</p> <p>if (e.getAttribute("href") != null) {</p> <p>linksToClick.add(e); }</p> <p>return linksToClick; }</p> <p>2-Find the link status</p> <p>public static String linkStatus(URL url) {</p> <p>try {</p> <p>URLConnection http = (URLConnection)</p> <p>url.openConnection();</p> <p>http.connect();</p> <p>String responseMessage = http.getResponseMessage();</p> <p>http.disconnect();</p> <p>return responseMessage; }</p> <p>catch (Exception e) {</p> <p>return e.getMessage(); }</p>
<p>HANDLING AND SYNCHRONIZATION IN SELENIUM</p> <p>1- Maximize the browser's window?</p> <p>driver.manage().window().maximize();</p> <p>In Selenium two type of wait</p> <p>2-Implicit Wait?If elements are not immediately available, an implicit wait tells Web Driver to poll the DOM for a certain amount of time.</p> <p>The default setting is 0. **Once set, the implicit wait is set for the duration of the Web Driver object.</p> <p>driver.manage().timeouts().implicitlyWait(3, TimeUnit.SECONDS);</p> <p>3-Explicit Wait? An explicit waits is what we can define to wait for a certain condition to occur before proceeding further in the code.</p> <p>WebDriverWait wait = new WebDriverWait(driver, 3);</p> <p>WebElement emailField=wait.until</p> <p>(ExpectedConditions.elementToBeSelected (By.id("user_email")))</p> <p>Tips:1 Some common use</p> <p>- elementToBeClickable</p> <p>-presenceOfElementLocated</p> <p>-elementToBeSelected</p> <p>-visibilityOfElementLocated</p> <p>Tips 2:Wait selenium method, Tread.Sleep is a java method,</p> <p>SAME DVANCE Use</p> <p>Calender selection:Locate send or click</p> <p>Auto suggest/complete: Locate box, create option list, find the list size</p> <p>Put your select inside the if condition and take inside the for each loop.</p> <p>And do iterate with for loop</p> <p>Windows size :use java script</p> <p>long height = (Long) js.executeScript("return window.innerHeight;");</p> <p>long width = (Long) js.executeScript("return window.innerWidth;");</p> <p>Scrolling element into view:</p> <p>js.executeScript("window.scrollBy(0, 1900);");</p> <p>js.executeScript("window.scrollBy(0, -1900);");</p> <p>Screen shot: We use under the @After annotation</p> <p>@After public void tearDown() throws Exception {</p> <p>String fileName = getRandomString(10) + ".png";</p> <p>String directory = "/Users/anil.tomar/Desktop/";</p> <p>File sourceFile = ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);</p> <p>FileUtils.copyFile(sourceFile, new File(directory + fileName));</p>	<p>COMMUN QUESTIONS</p> <p>close() -> closes current page and does not kill obj</p> <p>quite() -> closes all pages and kills the driver ob</p> <p>getText() -> gets the text which is outside of diamond brackets</p> <p>getAttribute()-> gets the attribute value which is inside of brackets</p> <p>Assert -> if test fails execution fail. use for critical things.Like hard assert</p> <p>Verify -> When it fails continues execution .We use for non_critical things.</p> <p>Like soft Assertion in testNG</p> <p>Handling multiple tabs: Use for each loop</p> <p>1-getWindowHandle() : returns ID for current page</p> <p>2-getWindowHandles() returns IDs for all open windows and return as Set<String></p> <p>Upload in Selenium -> using sendKeys</p> <p>- locate upload bo Then using webElement and sendKeys and pass path of the file.</p> <p>How can we download using Selenium?</p> <p>You can NOT!</p> <p>WEB TABLE HANDLING</p> <p>Approaching to web tables using selenium:</p> <p>1.Locate the table</p> <p>2.Inspect it if thead or tbody under the table</p> <p>3. If there are <thead> or <tbody> then locate them too</p> <p>close() -> closes current page and does not kill obj</p> <p>quite() -> closes all pages and kills the driver object</p> <p>1. Finding rows in webTable?</p> <p>→ Locate the table ,Then using tagName() pass "tr"</p> <p>2) Finding data in WebTables ?</p> <p>→ Locate the table, Then using tagName() pass "td"</p> <p>Do you write javaScript code ? (JSExecutor):</p> <p>→ I google it and pass it.</p>	<p>TYPE OF FRAMEWORK</p> <p>1- Data-Driven Testing Framework.:</p> <p>Test Scripts, the Test Data is separated and kept outside</p> <p>Test Data is read from the external files (Excel Files,Scenario Outline-Data Pr</p> <p>++Changes to the Test Scripts do not affect the Test Data</p> <p>++Test Cases can be executed with multiple Sets of Data</p> <p>++A Variety of Test Scenarios can be executed by just varying the Test</p> <p>++Data in the External Data File</p> <p>→More time is needed to plan and prepare Test Scripts and Test Data</p> <p>3-Behavior Driven Automation Framework.</p> <p>Is an Agile software development technique that encourages collaboration between developers QA and non-technical or business participant software development process based on test-driven development based on TDD with ideas from domain-driven design and object-oriented analysis and design to provide software development and management</p> <p>Popular Tools:Cucumber (Cucumber Company)-JBehave(github project)</p> <p>SpecFlow(github- techtalk project)</p> <p>3-Keyword-Driven /Table-Driven Testing Framework.</p> <p>Type of Functional Automation Testing Framework which is also known as or Action Word based testing .Basically divide the Test Case in 4 format</p> <p>Test Step: It is a very small description of the Test Step</p> <p>Test Object: Web Page object/element, name like Username & Password.</p> <p>Action: Name of the action, which is going to perform click, open browser,</p> <p>Test Data: Data can be any value which is needed by the Object to perform any action, like Username value for Username field.</p> <p>++Keywords are re-usable.</p> <p>The Hybrid Test Automation Framework.</p> <p>framework is the combination of one or more frameworks</p>	<p>GIT</p> <p>1-What is GIT?</p> <p>Git is a version control system.Git allows a team of people to work together, all using the same files. And it helps the team cope with the confusion that tends to happen when multiple people are editing the same files.</p> <p>2-How to share a project on Git using Eclipse?</p> <p>Create a local repository first</p> <p>Team -> Share project</p> <p>Select the your project</p> <p>Click on Create repository</p> <p>3-On the Git perspective, we will use two views?</p> <p>Git Repositories -> shows what repos we have</p> <p>Git Staging -> shows whats staged and what is not, allows us to stage, commit and push changes'</p> <p>Untagged changes -> this is the working directory, files we changes will show up here first.</p> <p>to ignore file : right click -> ignore (do it in the unstanged changes)</p> <p>4- Git Basic Work Flow for Testers</p> <p>Pull:A pull any changes from the Git repository and merges them into local repository</p> <p>Commit: move from staged to local repo-Committing is the process which records changes in the repository.</p> <p>Push: move from local repo to remote -Pushing sends the recent commit history from your local repository up toGi</p> <p>5-Code Review over git?</p> <p>1-Developer makes changes.</p> <p>2-Developer packs changes into some archive file</p> <p>3-Developer sends file to development team</p> <p>4-Random member of development team, opens change-file automatically in diffing utility, makes notes</p> <p>5-If code is good, random member says so, otherwise replies with requested notes</p>
<p>REPORTING</p> <p>Selenium+Cucumber+Jenkin:</p> <p>1-Basic pretty html: html:target/site/cucumber-pretty</p> <p>Write the code in future pile inside option .It create basic report in target file</p> <p>2-Html plugin (jenkins): Json:target/cucumber.json.</p> <p>We put this code in future file ,also add pom plugin, -> it is going to create pie chat report report inside the target, and also jerkin is going to generate report. By using this plugin <testFailureIgnore=true><testFailureIgnore></p> <p>If any test fail it is going to continue the run and generate report and email.</p> <p>Selenium+JUnit+Jenkins report: in pom write the report type "xml"</p> <p>And configure jenkins it is going to give a report</p> <p>Selenium_Testing Report generating:</p> <p>Create test -run with testing-go project box-refresh the project and there is</p> <p>1-Default suit (if you did not give different name) report generated in site name is "Default Test"</p> <p>It creates basic report , test case name result time "</p> <p>2-Emailable report this is second one this is also in project folder it has link and little bit detail report</p> <p>3-Index report In same folder you ll see" index" file. This is the third one basic index. style test.</p> <p>https://www.youtube.com/watch?v=tcYdgFDXuw</p>	<p>REPORTING + SCREENSHOT</p> <p>How to take SCREENSHOTS using selenium?</p> <p>You can use the TakeScreenshot interface to take the screenshots.</p> <p>In our framework, the logic for taking screen shot is in after hook method if any scenario fails screenshot will be captured and it will be include in the html report.</p> <p>If scenario fails it going to take screen shoot.There is a possibility can take picture for all feature steps but we don't want to full then memort=y for each details.. In Hook Class.</p> <p>@After public void tearDown(Scenario scenario) {</p> <p>if (scenario.isFailed()) {</p> <p>final byte[] screenshot = ((TakesScreenshot)</p> <p>Driver.getInstance()).getScreenshotAs(OutputType.BYTES);</p> <p>scenario.embed(screenshot, "image/png"); }</p> <p>***** Scenario.embed(takescreenshot)driver.getscreenshot method take the screens shot</p>	<p>RERUN THE FAIL TEST</p> <p>1-How to RERUN failed tests in Cucumber?</p> <p>Step 1:We can use the rerun:target/rerun.txt option to rerun failed tests.This will create a text file with the list of failed tests.</p> <p>Step 2:Then we can create another runner to run feature files from that list</p> <p>Like login runner,Order runner,Seach runner..</p> <p>-If we add both of the runner to the pom file, this project will always execute the all tests, then it will re run the failed tests once more</p> <p>2-How to RERUN failed tests in Junit or TestNG?</p> <p>1. Jenkins has an option to rerun failed tests.</p> <p>But it applies to junit or Testing tests. This feature does not work with cucumber because all cucumber is on bir junit test. This only applies to Junit or Testing tests.</p> <p><testFailureIgnore=true><testFailureIgnore> -> this option will make maven continue to run the tests even if sth fails so that we can generate a report at the end .</p>	<p>JENKIN</p> <p>1-What is Jenkins?</p> <p>Jenkins is an open source tool with plugin built for continuous integration purpose.</p> <p>2-Continuous integration?</p> <p>In software development, when multiple developers or teams are working on different segments of same web application for integration porpuse</p> <p>3-What are the two components Jenkins is mainly integrated with?</p> <p>Version Control system like GIT, (necessary)</p> <p>And build tools like Apache Maven. (necessary)</p> <p>4-Jenkins server configuration:</p> <p>Jenkins runs the tests on some server. And that server needs to have everything needed to run the tests: browsers, java, maven, git.</p> <p>As a tester, it is not out job to configure that server. All installations are carried out by team responsible for server maintenance team.</p> <p>5- Configuration of the Jenkins (How do we configurate Jenkins)</p> <p>-First we need to set the java path and maven path</p> <p>-Jenkins use our maven the built our project because we want to do that:)</p> <p>- We are gonna tell the Jenkins where is my java where is my maven</p> <p>- Go to ->manage system ->-configure system-></p> <p>- School down on terminal find the java and maven path and pass the related area</p> <p>-You can add the "E-mail notification" if you want to send email (usually company has email and their setup before)</p>
<p>AUTOMATION + FUTURE FILE</p> <p>1-How do you automate one test case? all steps involved?</p> <p>-I go through the scenario. We first have to manually execute the test to make sure it hows its work to functionality (because it is a new We are not going to decide is it can be automated or not because we already did that in print planning</p> <p>-write the feature file. In the feature I try to reuse the steps from.other. scenario/feature files. This makes writing tests very easy in my framework.</p> <p>-Then I implement the new steps (which is underline yellow it is new because) in the feature file. .Then do a dry run. Then get the new step defs from the console and copy into the test definitions java file.</p> <p>Then we write the necessary logic to implement the step.</p> <p>-We can call our page objects, call our utilities.</p> <p>-Once the test-passes, we check in the code to git, create a review.</p> <p>Once it is reviewed by a team member, I can close the jira story and move on to another one.</p> <p>Scenario: Login with valid credential</p> <p>Given I am on the home page</p> <p>When I login using valid credentials</p> <p>Then I should be able to verify I am logged in</p>	<p>CUCUMBER</p> <p>Feature: Describe the current test script which has to be executed.</p> <p>Scenario: Describes the steps and expected outcome for a particular test case.</p> <p>Scenario Outline+Example: Same scenario can be executed for multiple sets of data using scenario outline.</p> <p>Background: Step needs to be perform before in each scenario like</p> <p>Given user logged in as databases administrator</p> <p>And all the junk values are cleared</p> <p>@Tags: (@Smoke, @Login, @Regression</p> <p>Cucumber by default runs all scenarios in all the feature files</p> <p>In real time projects there could be hundreds of feature file which are not required to run at all times.</p> <p>Runner ***FeatureFile ***StepDefinition</p> <p>@RunWith(Cucumber.class)</p> <p>@Cucumber.Options (</p> <p>format={"SimpleHtmlReport:report/smokeTest.html"},tags={"@smokeTest"},)</p> <p>glue->Step dean path</p> <p>dryRun:True/false</p> <p>Public class JUnitRunner { }&Junit</p> <p>Public class Runner extends AbstractTestNGCucumberTest { } ->& TestNg</p>	<p>JUNIT (junit version 5) September 2017</p> <p>1- Annotations for Junit testing</p> <p>@BeforeClass method will be invoked only once, before starting all the tests.</p> <p>@Before that method will be invoked before each test.</p> <p>@Test that method is the test method.</p> <p>@After that method will be invoked after each test.</p> <p>@AfterClass that method will be invoked only once, after all the tests.</p> <p>2-The common methods of Assert class are as follows:</p> <p>1-void assertEquals(boolean expected, boolean actual):</p> <p>checks that two primitives/objects are equal. It is overloaded.</p> <p>2-void assertTrue(boolean condition): checks that a condition is true.</p> <p>3-void assertFalse(boolean condition): checks that a condition is false.</p> <p>3-Required jar files</p> <p>You need to load junit4.jar and hamcrest-core.jar files.</p> <p>4-JUNIT VS TESTING</p> <p>Testng : More annotations, has built-it parallel running, supports dependencies etc..</p> <p>JUnit: Supported by many other tools , does not support group test</p> <p>TESTNG ANNOTATION:Before &After (Suit-Class-TEST-GroupsMethod-</p> <p>@DataProvider->It is alone -></p>	<p>TESTNG. (version 6.11 February 2017)</p> <p>1- In TestNG how can you disable a test?</p> <p>To disable the test case you don't want, you can use annotations @Test(enabled = false)</p> <p>2-Mention what does the "suite test" does in TestNG?</p> <p>"Suite Test" is done when you have to run few unit test together, " Suite Test" bundle this unit test together. XML file is used to run the suite test.</p> <p>3-Explain what is parametric testing?</p> <p>Parameterized testing allows developers to execute the same test over and over again using different values. In two different ways TestNG allows you to pass parameters directly to your test methods.</p> <ul style="list-style-type: none"> With testing.xml. With Data Providers <p>4-Explain what is Group Test in TestNG?</p> <p>Groups are determined in your testing.xml file using the group test. Explain what it means when you have to pass parameters using data-providers in TestNG?</p> <p>//This method will provide data to any test method that declares that its Data Provider is named "provider1!". Support data Provide &parameterized testing</p> <p>@DataProvider(name = "provider1")</p> <p>public Object[][] createData() {</p> <p>return new Object[][] {</p> <p>{ "Cedric", new Integer(36) },</p> <p>{ "Anne", new Integer(37) }</p>