Selenium Notes

Selenium is a browser automation tool, commonly used for writing end-to-end tests of web applications.

|  |  |
| --- | --- |
| Key | Meaning |
| "file.separator" | Character that separates components of a file path. This is "/" on UNIX and "\" on Windows. |
| "java.class.path" | Path used to find directories and JAR archives containing class files. Elements of the class path are separated by a platform-specific character specified in the path.separator property. |
| "java.home" | Installation directory for Java Runtime Environment (JRE) |
| "java.vendor" | JRE vendor name |
| "java.vendor.url" | JRE vendor URL |
| "java.version" | JRE version number |
| "line.separator" | Sequence used by operating system to separate lines in text files |
| "os.arch" | Operating system architecture |
| "os.name" | Operating system name |
| "os.version" | Operating system version |
| "path.separator" | Path separator character used in java.class.path |
| "user.dir" | User working directory |
| "user.home" | User home directory |
| "user.name" | User account name |

RestAPI(Representational State Transfer Application interface)/ web servcies

Web application lifecycle- User interface to MiddleWire to Database to DevOps

1.User with browser- IE, FireFox, Chrome, Safari

As a user you browse to any website. As QA we can launch application and manually test the web application. Qa job is to create 100 of test cases and automate those test cases. Manual testing taking time and more resources. That’s why automation testing is very important to save time and money.

2.User Interface/Front End Application Server- Angular 2, React js, Angular Js, HTML, CSS, JavaScript

Rest API Call(most impt thing how to make restAPI call)

3.MiddlwWite/Back End/RestAPI Application Server- is the brain of the application, where we have all business logic. done by the back end developer.(Java,C#, Ruby, python). It use http methods to send request to database. GET, Post, put, Delete

Apache server, tom cat server, glassfish server, jboss

Then we have to know how to make Rest API call, don’t have to know about back end

application.

4. Database- Oracle, sql, mongoDB control by DBA’s.

5. Dev Ops who deploy the app into cloud environment. Aws.

Sh “ss.. @ubuntu you connect to server first.

What is RestAPI?

Api call you make from front end developer, your domain/path. Api is way to connect to backend or database. Front end don’t directly call to back end, he calls through rest API to back end and database. Routing make a service call to back end.

CURD operation- RestAPI support curd operation.

Create – Read – Update – Delete: MiddleWire communicate to database thru CRUD.

User Interface communicate thru RestAPI/Middlewire thru GET-Post-update-Delete

http methods-

Get - Call read/ get some data

Post - you are writing/ new information like when you creating a account.

Update - update existing data

Delete - to delete data from database

Postman – My Workspace has 3 sections.

you have all kind of method you can use like get, post, update, delete. There are lot more. Then you need your url and then you send request. There are 3 sections in postman.

* You select your method and enter url/parameter then send request.
* Request - you have Authorization, Headers, body, Pre-request and test.

Only get method doesn’t have body. Rest other method does.

* Response – you get back the data/result what you requested for.

[Www.google.com](http://Www.google.com) is your base uri, [www.google.com/asldkj342](http://www.google.com/asldkj342) your resources, [www.google.com/asldj342?user=’test’&pass=’abd123](http://www.google.com/asldj342?user='test'&pass='abd123)’ you parameters or you can add parameters by click on Params button and entering key and values

I tested manually, tested status code, tested request body, response body. I also tested header.

Selenium is a suite of tools to automate web browser.

Selenium IDE

Selenium RC

WebDriver or Selenium2 – we would be using WebDriver to automate web browser.

Selenium Grid

Most important thing about WebDriver, for example chrome or geicko, to make sure it support the current version of chrome browser and firefox browser we have installed on our machine.

[Www.seleniumHQ.org/download](http://Www.seleniumHQ.org/download)

Few things we need to download in order to run Selenium.

-Client driver: to talk to server, we have to use specific language. According to the language you would use you would download client driver. In my case Java.

-Server: where we have all our jar files, which need to be copied to my project.

-Also need geckodriver for firefox and chromedriver for Chrome.

Might have to do this for webDrivers:

Muhammads-MacBook-Air:~ bravo1516$ cd eclipse-workspace

Muhammads-MacBook-Air:eclipse-workspace bravo1516$ ls

First SeleniumProj1

Group\_Discussion SeleniumProject1

JavaOOPMarch2018 SeleniumProject1March2018

John\_Tutorial String\_Lesson

Mafi\_Batch Unit-Test

Practise\_ Weekeend

Sel-Project1 automation

Sele maven-framework

SeleniumPro practise

Muhammads-MacBook-Air:eclipse-workspace bravo1516$ cd Sel-Project1

Muhammads-MacBook-Air:Sel-Project1 bravo1516$ ls

bin driver lib src

Muhammads-MacBook-Air:Sel-Project1 bravo1516$ cd driver

Muhammads-MacBook-Air:driver bravo1516$ chmod 777 chromedriver

Muhammads-MacBook-Air:driver bravo1516$ chmod 777 geckodriver

Muhammads-MacBook-Air:driver bravo1516$ chmod 777 geckodriver

Muhammads-MacBook-Air:driver bravo1516$

WebDriver is interface, which control every browser. Implemented by one of the concrete class.

TestNg – is a testing framework inspired from Junit and Nunit but introducing some new functionalities that make it more powerful and easier to use, such as Annotaions. Annotatios give you what sequence you need execute your test case. Need to have atleast one test(annotation0)method to run your script. To control your test.

First you need to have beforeMethod, in which you set up your driver, url

Driver path, url.

Then go to test, where you have test cases, execute them then move to next step

Aftermethod - it is simply a clean-up process, where you close the browser.

What is locators?

We use locators to find specific webElement on your application. We have id, name, cssSelector, xpath, name, linktext, partiallinktext, tagename, class name.

Html docs start with <Html>

<head>

<title> Page title</title>

</head>

then move on to the body

<body>

<h1>This is a heading</h1>

<p>this is a paragraph</p>(--You can have multiple paragraph)

<p>This is another paragraph</p>

</body>

What is CSS? How all the elements would look like on your website.

Cascading style sheets. It basically describes how HTML elements are to be displayed on screen, paper or in other media.

CSS save lot of work. It can control the layout of multiple web pages at once.

External stylesheets are stored in CSS files.

CSS syntax-

Selector Declaration Declaration

H1 {color:blue; font-size:12px;}

Property - Color = property

Value - blue

Property = Font-size

Value = 12px =

Id Selector: with specific id. You have to put#

Class selector: elements with a specific class attribute. You have to put (dot .)

Grouping Selectors-

How to insert CSS?

There are three ways to insert CSS

-External style sheet

-Internal Style sheet

-Inline Style

What is stylesheet?

To manage your website at one place. You don’t want to type everything on your html doc, that’s why we have stylesheet to design out doc and give instruction there to design our doc.

<ul> un order list - will give you bullet point dot dot dot

<ol> order list – will give you numbering 1,2,3,4

org.pagache.maven.archetypes maven-archtype-quickstart 1.3

Maven is a build management tool. Your application has lifecyle, you design, you complie, you run, you install, you deploy. As build tool you have to support those lifecyles.

It also manages dependencies. Only support by Java not any other programming language.

POM is a heart of Maven.

Maven LifeCycle

* Compile
* Install
* Test
* Deploy

Maven- POM.xml where you store all the dependencies.

We also have Maven central Repo, which stores all the dependencies can be used in maven.

“Maven.apache.org/download.cgi” - To download Maven

For mac you can use brew, to install Maven. brew install maven

M2\_home and mvn\_home are maven version.

rm Project4

rm -rf Project4.   //remove directory

Mvn -version

Java –version

.m2 is local repo for maven

“Search.maven.org” - The Central Repository

maven has some templates, which known as archtype.

Group id: this is generally unique amongst an organization or a project. Alphabet is group of company, which has google, dropbox, googlemap, android, gmail. Alphabet could be group ID. Timewarners – cnn, hbo and so on.

Arifact- The artifactID is generally the name of the project is known by.

Since we start using maven project, we would interact we src folder a lot, which has 2 subfoler main and test.

In main java you have all business logic.

In Test java you have all unit tests.

After build it take place on target folder.

Framework – main purpose of framework to reuse our code multiple times. is a collection of API with class that simplifies development.  The framework provides the basis of test automation and simplifies the automation effort. The main advantage of a framework of assumptions, concepts and tools that provide support for automated software testing is the low cost for maintenance.

Purpose of xml to carry data from one place to another place. Jason can carry data from one point to another point.

We have now generic Module. It has its own pom.xml. Every Maven module has main and test directory under src. Generic will not deal test cases, it will prepare your test cases but not run it. Your application module will run test cases. You need API to run test cases, opening browser, getting connection to database and cloud.

Under one Maven Project you can have multiple module. Generic pom help you create connection between all module. Every application will depend on generic.

Each module has Heart(POM). Generic module is connected to root pom, and then all the application module can use generic pom to have access of all the dependencies.

Page-Factory or Page object Model.

Main purpose of Page object model or page-factory to get all heavy work done in Main folder all your business logic and methods and keep your test folder nice, clean and light.

Absolute xpath: starts with html, it uses complete path from the root element.

Relative xpath: you can simply start by referencing the element you want and go from there.

CSSS selector:

#Id

.classname

../Generic/driver/chromeDriver is to go one step back for relevant path.

Driver.get() you can’t do back and forth.

Driver.navigate().to()/forward()/back(). To navigate back and forward.

Synchronization – will only wait till item is clickable or visible.

Thread.sleep – will wait according to seconds you have entered to wait.

Instead of running local, we can use cloud.

Jira- Project management tool.

Confluence – Business analyst are stake holders, what to build, what product are we building, take all the notes, planning, product, architecture flow, all sort of documents you can store in confluence. Information about tools can be store in confluence. Team notes, meeting, client information. It’s a cloud.

Jira – we will be using daily basis. Basically how to connect different teams like front end, back end, dev ops, and QA working on application. People could be anywhere in the world, as long as you have internet connection you could connect to team in Agile environment. Team collaboration software. You can log a defect ticket.

Every time you assign a ticket, Lead will receive a ticket.

Basic Concepts in JIRA – Issues, Projects, Workflow and components.

Agile- Low risk, easy to adapt changes, more interaction with stakeholders. High visibility.

Waterfall – Poor visibility, can’t handle changes, High risk.

Agile is time boxed, iterative approach to software delivery that builds software incrementally from the start of the project. If you have project due in 6 months, you have 2 weeks sprint, 13 sprint. Each sprint you work on different part of the application.

Scrum is a framework used by teams to manage work. Scrum implements the principles of Agiles as a set of artifacts, practices and roles. Scrum give you discipline how your agile can be perform. Every company has their own way to work on scrum. But they all have common ground. Just give you guideline and you build on top of it.

Sprints- also known as an iteration, is a short(ideally two to four week) period in which the development team implements and delivers a discrete product increment, a working milestone version. If you have 2 weeks sprint and holiday coming up then, you would assign less story points.

Backlog you have all user stories, then in sprint you can drag all user stories accordingly. You have 100 of tickets in backlog, it would never be empty. You have prioritized which need to be go first. Backlog of have some tickets and current sprint would have some tickets. To do, in progress, done. Once developers done with their work, its testers job to test their codes.

Workflow – Open, in progress, re-opened, resolved, closed.

Made out of status and issue has lifecycles. They flow from one place to other place. Create issue/open issue, can be created by stakeholders. Start progress, we should call in progress state. You have option to close it, or change your mind, stop progress. Back to open state. Or resolve the issue, validate the particular user story. It could be reopen. At the end you have to close the issue.

Story Points – estimated task- how long will it take. You assign some point to each ticket, then you distribute that. We want to know how much we can get it done based on points system. Every company has their own points system.

Backlog – gather all the task that is ready for sprint.

User Story – application you are building its for user. As a user I want to login to amazon, I can select some items. And make payment.

Epic – Large user story that break into small stories.

Project has components(sub-section) and versions(milestones)

Components and Version – components are sub-section of your projects, which issues are group into smaller parts. Components can have multiple Epic, each epic could have multiple user story, each user story could have sub task.

Versions are points and timelines of your projects that helps to organize the releases the products.

Project Categories

Level 1

Projects

Level 2

Versions(milestones)

Components(sub-section)

Issues

Level 3

Issue types

Issue types

Sub- Task

Sub-Task

Sub-Task

Sub-Task

Level 4

Plugin use for multiple for purpose, compile, run, produce jar file. We can use plugin to run our test cases thru Maven.

To read data from MsS Excel- JXL/ApachePOI https://poi.apache.org/download.html

Googsheet – google sheet developer APIs

Database – Oracle, MySql, MongoDB

30 user story per sprint.

25 user story per sprint.

Sauce Labs – Browser Stacks

I am using cloud provider browserstack, inside my project I am using build tool maven, I have system properties or variables, like browser type, browser version, platformType and user will feed those values.

Cross browser Testing: IE, firefox, chrome, safari. Different versioin of chrome, firefox, different version of operating system. We can’t have everything in one machine same time. That’s why we have sauce lab or browser stacks which store everything in cloud one place. We can run through cloud and don’t have to run locally. Its very convenient. You build the framework, you can go away from your office.

RemoteWebDriver(Url, DesiredCapabilities) Desired capabilities prepare for you the browser and operating system.

User Interface MiddleWire Database

Html/css/javascript restAPI services oracle/mongoDb

httpMethods

(get,post,PUT,Delete) CRUD(create,read

Selenium postman/restAssured jdbc/queries

Crud is to create connection between database and restAPI.

Postman you have to do manually, rest assured you can automate.

We can also turn on testing listener. By Edit configuration and check mark listener. for test case report management.

ATU reporting tool - for test case report management.

Extent reporting tool – for test case report management.

For extent we need to upload dependencies in our root pom

<dependency>  
 <groupId>com.aventstack</groupId>  
 <artifactId>extentreports</artifactId>  
 <version>3.0.7</version>  
</dependency>  
<dependency>  
 <groupId>com.relevantcodes</groupId>  
 <artifactId>extentreports</artifactId>  
 <version>2.41.1</version>  
</dependency>

We need to add testLogger in our test cases to show each step in test report.

We json file to transfer data from one location to another location.

How you select multiple items from cart? Or anywhere on application?

-use find elements, using xpath, using xpath(contains).

Implicitly wait This means that we can tell Selenium that we would like it to wait for a certain amount of time before throwing an exception that it cannot find the element on the page.

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

Explicitly wait – An explicit wait makes WebDriver wait for a certain condition to occur before proceeding further with execution

WebDriverWait wait=new WebDriverWait(driver, 20);

Atlassian own JIRA, it provides cloud services. It’s a project management tool.

JIRA- Atlassian own JIRA and provide cloud service.

As front end developer – Angular framework use by front developer, they use type script or java script for that. Front end has to do with appearance of the website/application like HTML CSS and javascript while backend deals with all the functions and behind the scene stuff.

As bank-end/RestAPI developer – we have glassfish/jersey framework.

REST- Representational State transfer

You have to make call to restAPI then restAPI connect to database. Use http. Front end has to talk to RestAPI, by making a call and url need to be provided. Using http protocol to talk to backend. Frontend can’t talk direct to database, frontend has to send request to restAPI and restAPI talk to database. All the request sent by user/clients process by restApi then goes to database. Clients can’t directly communicate to database.

Get- to read data

Post – to submit some data

Put – to edit something

Delete – to delete something

Status code –

200 ok, success

300 Redirection

400 client errors

404 not found/bad request

500 server errors

unofficial codes

xml and json takes data from one place to another place. Html just displays data.

Postman – postman is use for restAPI testing. You can make connection of libraries and pick a time. You can automate it. You can make collection of libraries. Run api button it will run everything for you. Its better to use restAssured to automate restApi.

Ng serve is command to run local your angular framework

Rest Assured we use to Automate RestAPI.

Response response = given().when().get(apiHome, “AllEmployees”).then().statusCode(200).extract().response();

3 sets.

Given()= content-type, cookies, params

When() = http method type

Then()= status code results.

How would you open new tab in selenium?

String selectLinkOpeninNewTab = Keys.chord(Keys.CONTROL,Keys.RETURN);

driver.findElement(By.linkText("urlLink")).sendKeys(selectLinkOpeninNewTab);

Aws – big companies like Netflix, they don’t need their own data structure, they can use AWS. we use 3 things, EC2 your status, S3 bucket, route53 maintain your dns your domain.

Search.maven.org – The central repository for dependencies

Reason we want to have bigger size ram, every object take place in heap. The bigger size the more object you can store.

jmeter

Jmeter.Apache(Binaries)- For performance, Load and stress testing.

Bravo1516$ java -jar ApacheJmetere.jar

-http request default – set Url e.g [www.statefarm.com](http://www.statefarm.com) it’s a home page

-http request – set sub url/api e.g /AutoQuotePurchase-web/customer/welcome?conversationId=1594ced7-fe21-4722-82ef-cb03b74b5e50

Add a Threadgroup, number of thread, ramp-up and loop

Right click thread group, Add, Listners.

-view result tree – to check report

-view Results in table – to check report/results

-summary report.

-listeners is very important, you send a request it listens to you and perform actions according to your request.

Mobile Automation-

* Appium has to be running.

Android Studio- Go to tools, AVD manager, if you don’t have virtual device set up

-Phone – select Nexus 5x click next

- select the images – we need lollypop, the code and app we had was running with lollypop.

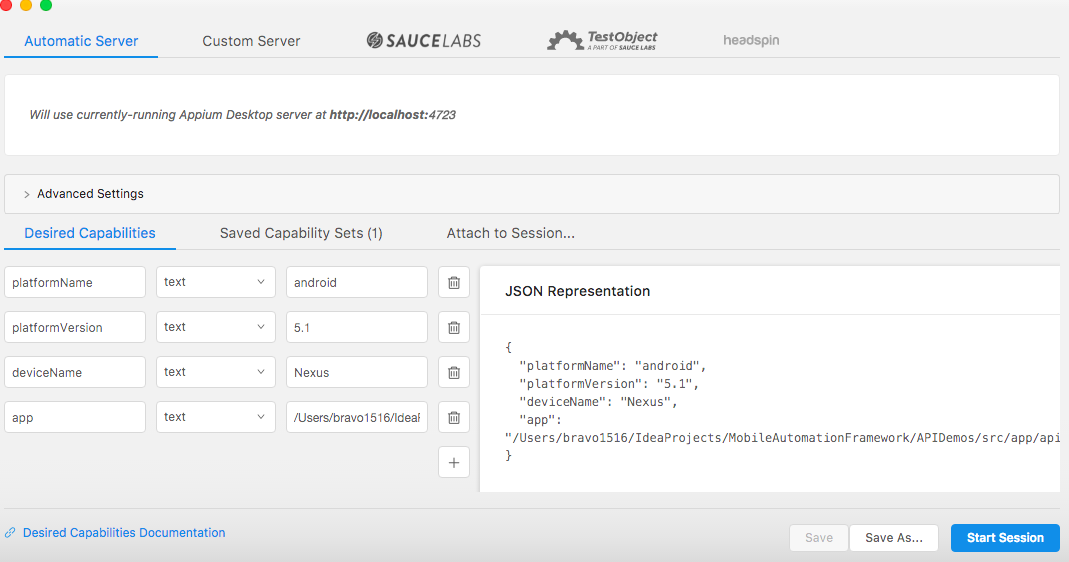
- go to x86 images, download Lollipop 22 x85\_64 Android 5.1(Google Apis) click next

- don’t need to change AVD name, click finish

- now you have device installed, click play button under Actions and your emulator will launch.

- Appium provide you inspector, to locate webelements or you can use UI Auotmator Viewer.

Under desired capabilities you set up all your virtual machine infor.



Test Plan include – Approach, Environment, what part of application need to be test by automation and manual.

Test scenarios – functionality that need to tested

Test cases – Expected results, Actual results, pre-conditions, post-conditions.

To start Jenkins – go to your terminal type Jenkins enter then go to your browser and type localhost:8080

Jenkins – you can use to run your test cases. You can set up specific time you want to run. You can build report for your test cases.

How to do configuration:

Source Code Mangement – Git repository url

Build triggers – to schedule a time run your project

Build periodically - 11 22 \* \* 5 min/hr/day/month/day-week

* Type in terminal to get local host ip ./ngrok http 8080

Build Environment – Delete workspace before build starts or use secret text for files

Build – Maven version – Maven, Goals test or –fn test

Post-Build Actions – you can add your email address and attachment

-set up your git url

-Build Triggers- how many projects you want to run, build periodically, schedule a time to run

[https://crontab.guru/#30\_12\_\*\_\*\_5](https://crontab.guru/#30_12_*_*_5)

or we could schedule it from webhook in github.

- Build environment –

- Build – Maven – test or –fn test

- Post build(Editable Email Notification)-

- send notification, just add your email address to Project Recipient List

-how to attach screenshot or report– statefarm/screenshot/\*.png or /screenshot/\*\*/\*.png or \*/Extent-Reports/\*.html

CI-CD pipeline.

Continuous Integration Conti deployment.(Jenkins)

You need to config Jenkins, you run test from Jenkins, you need to know how to setup task on jenkins.

Front end developer and back-end/RestAPI developers throw their work and commit their code, which goes to github. Jenkins can get code from github, by using github url. Jdk, Maven and git config on Jenkins first. Whenever you do commit on github, jekins immediately notify and build. First Jenkins clone the repo from github, then keep checking for updates. On Jenkins you can assign here is my repository and here is my url, this is the maven task. I need to deploy this. Deployed to QA environment/Staging environment/beta environment. Another task, Now it gives signal to QA environment. Qa task on Jenkins, its not manual, you do configure once then its automatic test cases. You can run against particular build. It could be run local or cloud environment browser stack/sauce labs. If anything fails, build fail, verify those, log a bug thru jira or alm. It won’t be deployed to prod environ. You report the dev team. Jenkins is simply task management. Run everyday, midnight, every weekend. Everytime you have new deployment you get signal. Front and back end will fix the issue and commit the code to github, they don’t touch Jenkins. Jenkins will fetch the task from git hub. Or login to Jenkins, and build task. Depends how you set up your Jenkins. Dev ops can write code using ruby scripting / shell scripting to deploy code to Prod environ. One organization git hub can have multiple repositories, one application have multiple repositories and merge as one application possible. That integration done in Jenkins. Continuous integration. Once you build it, you deploy it. We deploy multiples time. Continuous deployment. One organization can have multiple Jenkins node under one master Jenkins. One for WeBAutomation, mobile automation, connected devices and then different components of web, mobile and connected devices. Jenkins make things very easy for developers, tester and entire team.

Application Features – provided by stakeholders to Developers, could be BA, policy maker, PM. Dev Team build application according to those features.

During sprint, while developers are working on code, tester make skeleton of framework, of the story, when its ready to deploy to QA or staging then you get locators. Before that you work on test cases, prepare test scenarios. You have to work while developers writing codes.

If you have information about the application, you can build your framework, without even getting locators. By creating a page factory, methods, logic, and framework. Once you get the code, all the you need do is plug in the locators. Your framework is already set up. We can automate as it implemented. If in current sprint there are 5 features need to be tested. Developers task is complete and your task is also complete few hours behind.

There are 2 scenarios how you automate your test cases while in scrum environment. One is neck to neck, same sprint development done and you have to automate your test same sprint. N-1, if you are in sprint 15, it is possible automation will be done in sprint 16.

Unix Command

Ls – list of files in current directory

Cd – cd takes you back to your home

Mv file1 file2 – moves delete file1 and create new file file2

Cp file1 file2 – makes a copy of the file1 and calls it file2

Man – the manual of inputted command

Mkdir dirname – make directory with name ‘dirname’

Rmdir dirname– removes the directory with name ‘dirname’

Touch file1– allow user to make ‘file1’ touch testfile.txt

Rm file1 – removes file1 rm testfile.txt

Rm –I file1 – asks for confirmation that you want to delete file1

Rm –rf dirname – removes ‘dirname’

Rm –rf/ never use it, to delete all the directories from you Linux/os

Whoami – obvious

Cat files – concatenate and display files

Grep pattern files – search files for pattern

1. If you are currently in insert or append mode, press Esc .
2. Press : (colon). The cursor should reappear at the lower left corner of the screen beside a colon prompt.
3. Enter the following: q!

ctrl + shift + w to close a terminal tab

ctrl + shift + q to close the entire terminal

Sql

Describ “tablename”

Describe Employees;

Describe Region;

Select \* from Customers Where country=’Germany’ and City=’Berlin’;

Where Not Country=’Germany’ and Not Country =’USA’

Where city=’Berlin’ OR city =’München’

Union to join multiple queries

Here are the different types of the JOINs in SQL:

* (INNER) JOIN: Returns records that have matching values in both tables
* SELECT Orders.OrderID, Customers.CustomerName  
  FROM Orders  
  INNER JOIN Customers ON Orders.CustomerID = Customers.CustomerID;
* LEFT (OUTER) JOIN: Return all records from the left table, and the matched records from the right table SELECT Customers.CustomerName, Orders.OrderID  
  FROM Customers  
  LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID  
  ORDER BY Customers.CustomerName;
* RIGHT (OUTER) JOIN: Return all records from the right table, and the matched records from the left table SELECT Orders.OrderID, Employees.LastName, Employees.FirstName  
  FROM Orders  
  RIGHT JOIN Employees ON Orders.EmployeeID = Employees.EmployeeID
* FULL (OUTER) JOIN: Return all records when there is a match in either left or right table SELECT Customers.CustomerName, Orders.OrderID  
  FROM Customers  
  FULL OUTER JOIN Orders ON Customers.CustomerID=Orders.CustomerID

Cucumber- gherkin format

BDD Behavior driven development: we define the behavior on the basis of behavior test cases. Most important thing you need to have test cases, feature file will convert your test cases into gherkin format. Gherkin(language not a programing language) format goes step by step make it very easy and simple for everybody on project like developers, project manager and even junior tester which has no automation exposure or someone no it background can easily pick it up. Easy to convert. Each step find some keyword. You need to have add of Cucumber Editor in order to create feature file. Start with

Feature: testing Login Functionallity:

Scenarios: User are able to login successfully with valid credentials

Steps Given, when, And and Then.

If you have 10 cases, you need to create 10 feature files. If you have 10 sprint, you can create 10 folder for sprint1, Sprint 2 and each folder/each sprint you need to have specific feature file.

Feature: Name of feature

Background: it will always run background step first then go to scenario.

Scenario- one feature file can have multiple scenario.

Then steps

Feature file- login.feature, in this particular feature file, we will be using gherkin keywords. We have when, then, as, And , Given, But with the help of these keywords we will create scenario or scenario outline.

Step Definition- in step definition file we will writing our code by using Selenium, java and different annotations.

Test Runner- we will be writing test runner in for of junit. To execute your feature. To generate and output and reports.

Cucumber dependencies- cucumber-java, cucumber-jvm, cucumber-junit, cucumber-jvm-deps, cucumber-picocontainer, gherkin, cucumber reporting, junit. We also have to installed the Natural(BDD/Gherkin) plugin from eclipse market place to support the feature file.

To create pretty format console and report: ~~format~~ = {"pretty", "html:test-output"} in your runner class.

What are cucumber Options available?

@CucumberOptions

dryRun – to check if mapping is perfect or not, or if am missing any scenarios in stepDefinition class.

Features – file path of your feature file

Glue – path of your stepDefinition, you can simply enter your package name of your “stepDefinition”

Monochrome – Display the console output in a proper readable format

Format – it also shows pretty format in console. You can generate, xml report, html, json report.

Strict – It will strictly follow the mapping between feature file and step-definition.

and all these options will be define in runner class. apart from tag. All tags need to be define in future file.

tags = {"@RegressionTest”} = any test case with RegressionTest Tag

tags = {"@RegressionTest, @SmokeTest"} any test case with regression or smoke test.

tags = {"@RegressionTest”, “@SmokeTest"} execute test cases only tagged as SmokeTest and RegressionTet.

How to avoid test cases.

tags = {"~@RegressionTest, @SmokeTest"} by using ~ special character you can skip

tags = {"~@RegressionTest” “~@SmokeTest"} you can skip any tag with reg or smoke.

Data Driven Testing Using cucumber:

* Simple data driven – without examples keyword
* With examples + Scenario outline
* Using data-tables

Using data driven approach without examples keyword. (In your stepDef you have to use reg exp). In your method, you must pass the arguments, depending on number of parameters.

Reg exp =  \"(.\*)\" if you have 2 parameters you will pass 2 reg exp

And user enters "bravo1516" and "abcd1234"

@And("^user enters \"(.\*)\" and \"(.\*)\"$")

Data driven Testing with help of scenario outline- With examples keyword you always use scenario outline. You use reg exp in your step def for scenario outline approach as well. In your method, you must pass the arguments, depending on number of parameters.

Data driven testing approach without using scenario outline and examples keyword. We can use data table. Test data table created instantly after the step, where you want to use the data. Examples keywords are applicable to entire test case where with scenario outline, only applicable to specific step.

We need to import data-table class and raw method, it will return List<List<String>(List-object) of values. List object you can only pass one parameter.

We can also use data-table concept with help of Map-Object, we can define thousand set of data, more readable format and behavioral format.

Cucumber Tags – it is simply annotation. lot of confusion can be ignored with help of tagging in feature file. When you have 100’s of scenario with in your feature file and you would segregate some scenario for different type of testing, like some for smoke or sanity testing, or some for regression or end to end testing that’s when you use tagging.

Cucumber Hooks – pre-conditions or pre-requisites cucumber annotation @Before and @after.

Cucumber tagged hooks – if you want to execute any pre-condition for specific scenario. global hooks(annotations @before and @ after) and local hooks(@Before and @After), global is applicable for all the scenarios. And local would only be implement to specific scenario.

Environment Variables: Config.properties, URL, username, password, browser, browser stack.

Cucumber with page object model

WebDriver Properties(Test base)- Page Layer, Feature, stepDef, Test Runner

Eclipse = short cut to import cntr+shift+Os

Camel case: method name start with lower case.

Gradle- it’s a build management tool.

GroupID: us.piit(domain name in reverse order) ArtifactID: project name

Use Auto-import. Go to preferences, go to gradle and check create directories for empty content.

Build.gradle is just like pom.xml in maven, you can store all dependencies heart of gradle.

For Cross browser testing WebDriver driver = new RemoteWebdriver():

Desired