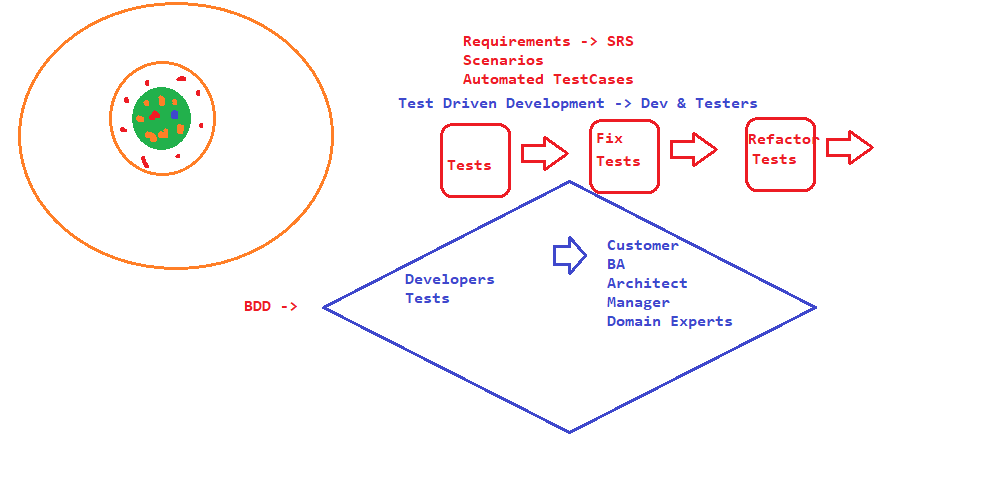
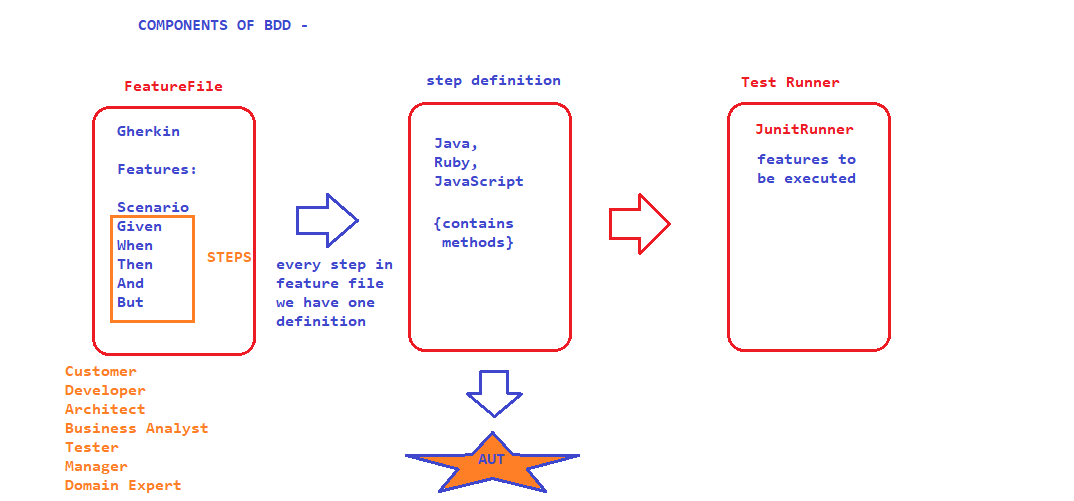
BDD Cucumber :

Pre req:

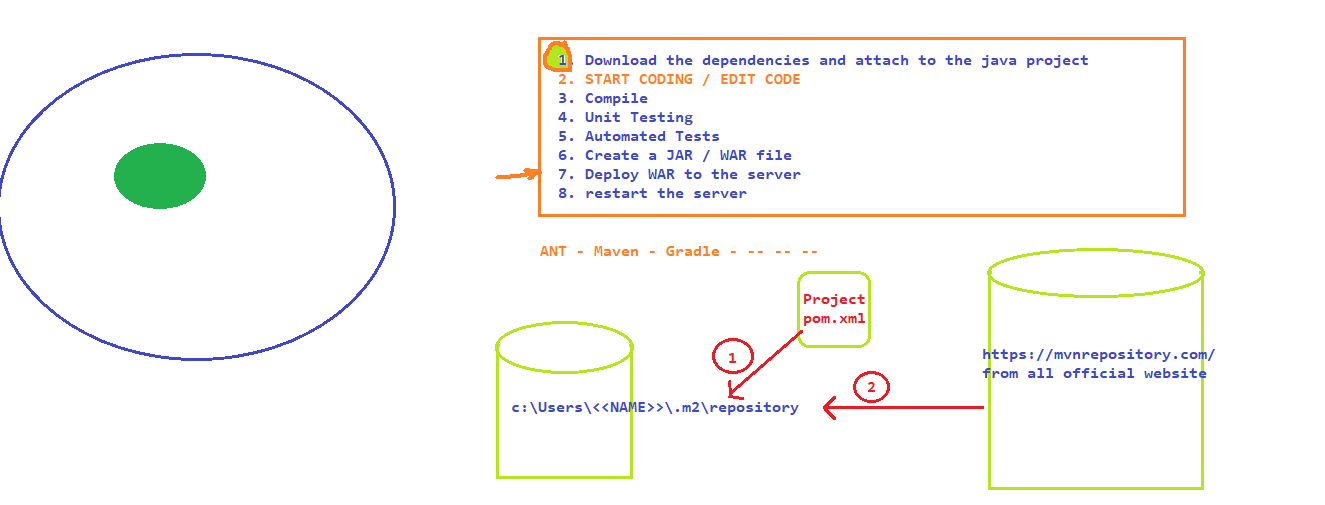
* Java
* Selenium – WebDriver
  + Object identification ->**xpath**
    - basic
      * Syntax - //htmltag[@attribute=’value’]
      * Ex - //input[@id='file-upload-field']
    - with Logical operations or / and / not
      * Syntax - //htmltag[@attribute1 =’value’ and @attribure2 = ‘value’]
      * //htmltag[@attribute1 =’value’ or @attribure2 = ‘value’]
      * Ex - //input[@type='text' or @type='password'] 🡪 2 elements
      * //input[@type='text' and @type='password'] 🡪 0 elements
      * //td[text()='15' and not (@class='past day')]

OR

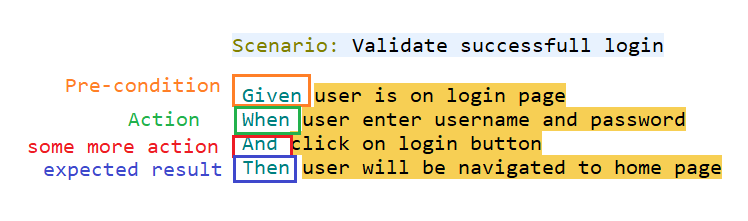
* + - * //td[(@class='wd day' or @class='we day' or @class='current day') and text()='14']
    - functions -> Web Element which has **text** or which is **hyperlink**
      * text()
        + Syntax - //html[text()=’exact text’]
        + Example - //td[text()='15']
      * contains(arg1, arg2)
        + arg1 -> attribute or text()
        + arg2-> partial value corresponding to arg1
        + Syntax - //htmltag[contains(@attibure,’partialValue’)]
        + Ex - //img[contains(@src,'timer')]
        + //label[contains(text(),'Keep')]
      * starts-with(arg1, arg2)
        + arg1 -> attribute or text()
        + arg2-> partial value corresponding to arg1
        + Syntax - //htmltag[starts-with(@attribute,’startingvalue’)]
        + Example - //h3/span[starts-with(text(),'Synechron')]
        + //htmltag[starts-with(text(),’startingvalue’)]
        + //\*[starts-with(@id,'ext-gen')]
    - Travers from parent to child
      * Immediate child - //h3**/**span[contains(text(),'Wiki')]
      * Not Immediate child - //div[@class='tF2Cxc']**//**span[contains(text(),'Wiki')]
    - Travers from child to parent – *whenever we have dependent and independent elements in the webpage*
      * 1. write xpath to the child element (independent ele)
      * 2. put the complete expression (excluding //) inside square bracket and specify parent html tag
      * Syntax - //parenttag[xpath\_of\_child]
      * //parent2[parenttag1[xpath\_of\_child]]
      * //div[div[div[h3[contains(text(),'Retreat to')]]]]//p[contains(@class,'blackText ')]
    - Axes Functions
      * traverse to following siblings
        + Syntax - //htmltag[@attribute=’value’]/following-sibling::SIBLINGTAG
        + //th[text()='Directed by']/following-sibling::td
      * traverse to preceding sibling
        + Ex - //li[a[text()='rPool ']]/preceding-sibling::li
      * till end of the page from any particular location
        + Ex - //span[contains(text(),'Wiki')]/following::a
      * till beginning of the page from any particular location
        + Ex - //span[contains(text(),'Wiki')]/preceding::a
      * traverse to parent
        + //th[text()='Directed by']/parent::tr//a
      * traverse to child
        + //h3/child::span
      * traverse to ancestor
        + //h3[contains(text(),'Retreat to')]/ancestor::div[contains(@class,'packageListing ')]//p[contains(@class,'blackText ')]
* BDD Cucumber
  + 
  + Frameworks
    - Cucumber
    - JBehave
    - BeanSpec
    - Esay B ..
* Components of BDD



* Maven – BAT



* Sample Feature file



Maven Project Steps

1. Create a Maven Project using QuickStart archetype

2. Update JDK and JRE to the installed version on machine

3. Add dependencies - *cucumber-java, cucumber-junit, junit*

*4.* Install Cucumber plugin

5. Created a Feature file (File-> New->General->File) save file a \*.properties

6. Execute the Feature file (Rightclick on the feature file and Run As -> cucumber feature

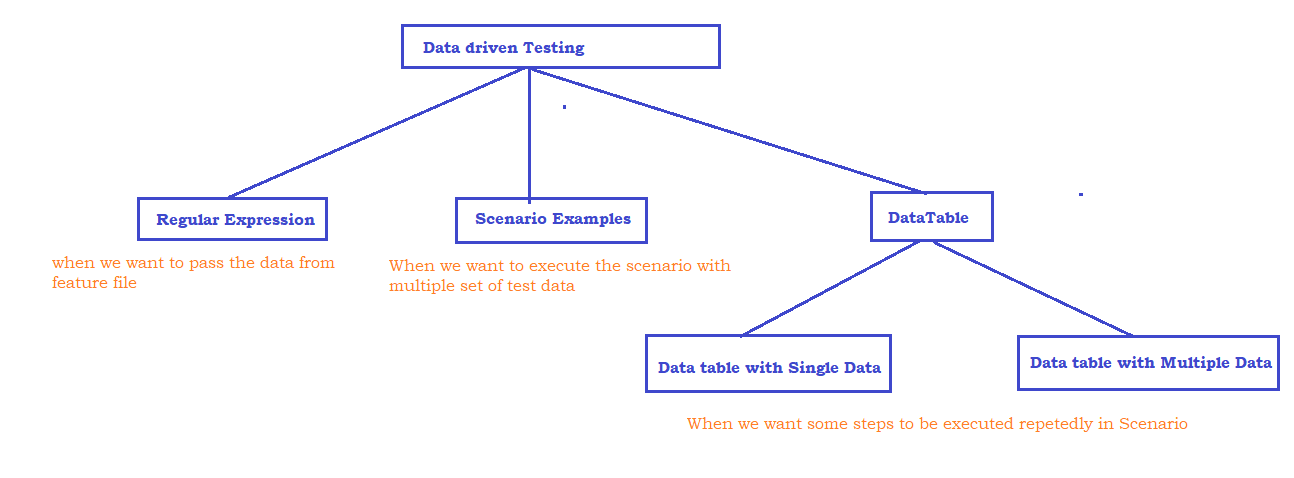
7. Add New Stepdefinitions folder / new package under src/test/java

8. copy all the missing step definitions and put it in a new java file under stepdefinitions

9. Execute and make sure it works

Selenium Automation / Step Definition

Data driven Testing in BDD



Regular Expression :

==================

1. {string} --> "IBM" or "Wipro" or 'Synechron'

2. {int} --> 4 or 5 -1 -----

3. {float} -->5.8 or 75.5

4.{} -->matches everthing

5.{word} --> synechron is a good place to work

Optional Text

------------------

Then application should dispaly more results

Then application should dispaly no result

@Then("application should dispaly more/no result(s)")

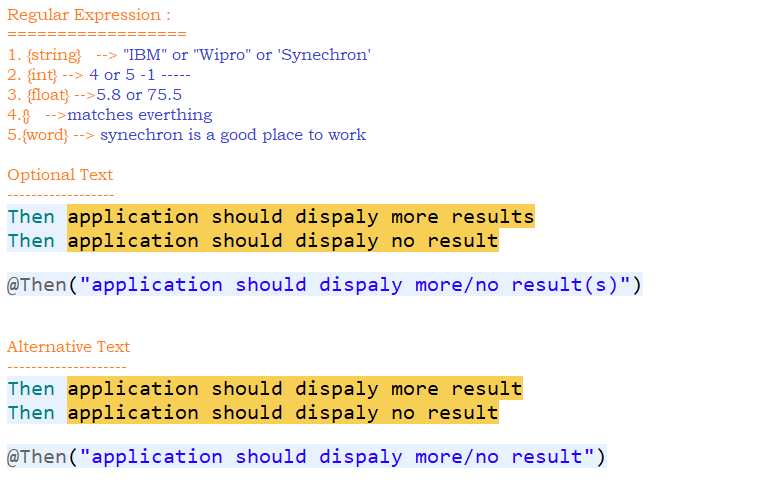
Alternative Text

--------------------

Then application should dispaly more result

Then application should dispaly no result

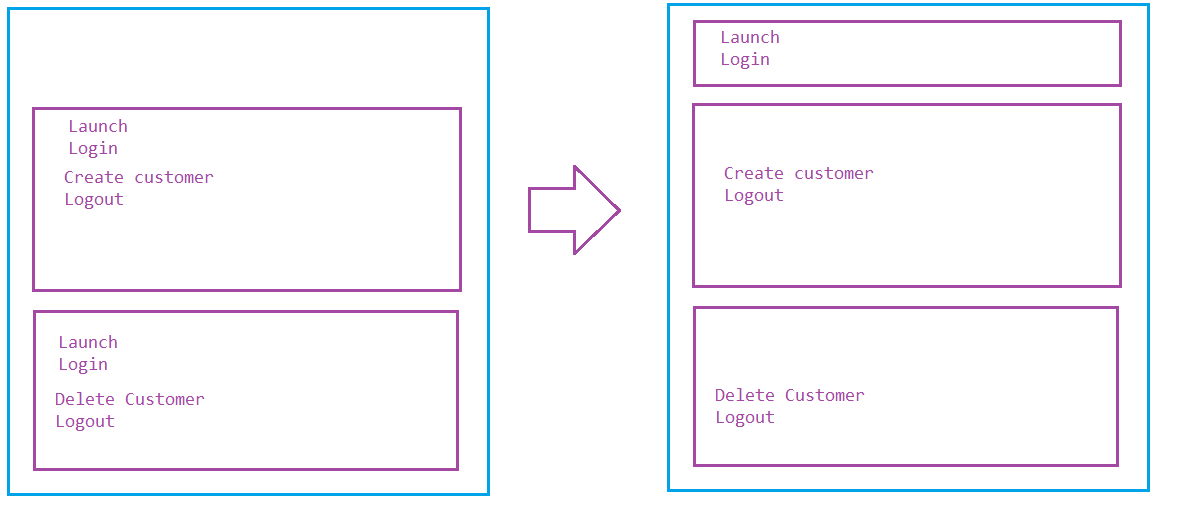
@Then("application should dispaly more/no result")



TAGS

* Regression
* sanity
* uat
* production
* end2end
* functional
* tags = "@underdevelopment",
* tags = "@underdevelopment and @regression",
* tags = "@underdevelopment or @regression",
* tags = "@underdevelopment and not @regression",

BackGround - it’s a preconditon to the test or Scenario ex – launch / login etc



Hooks

