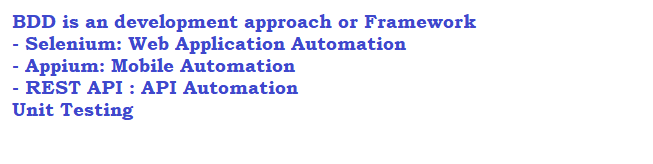
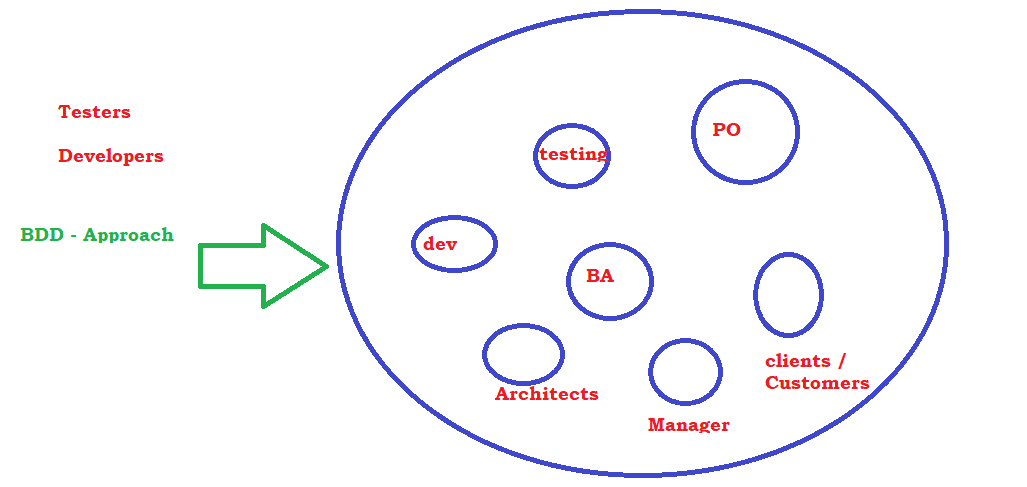
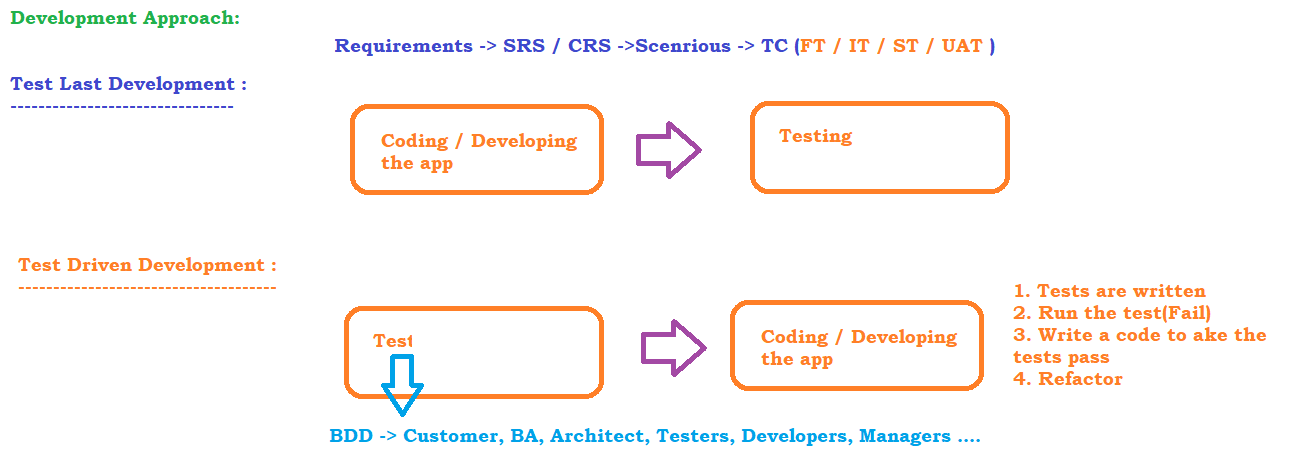
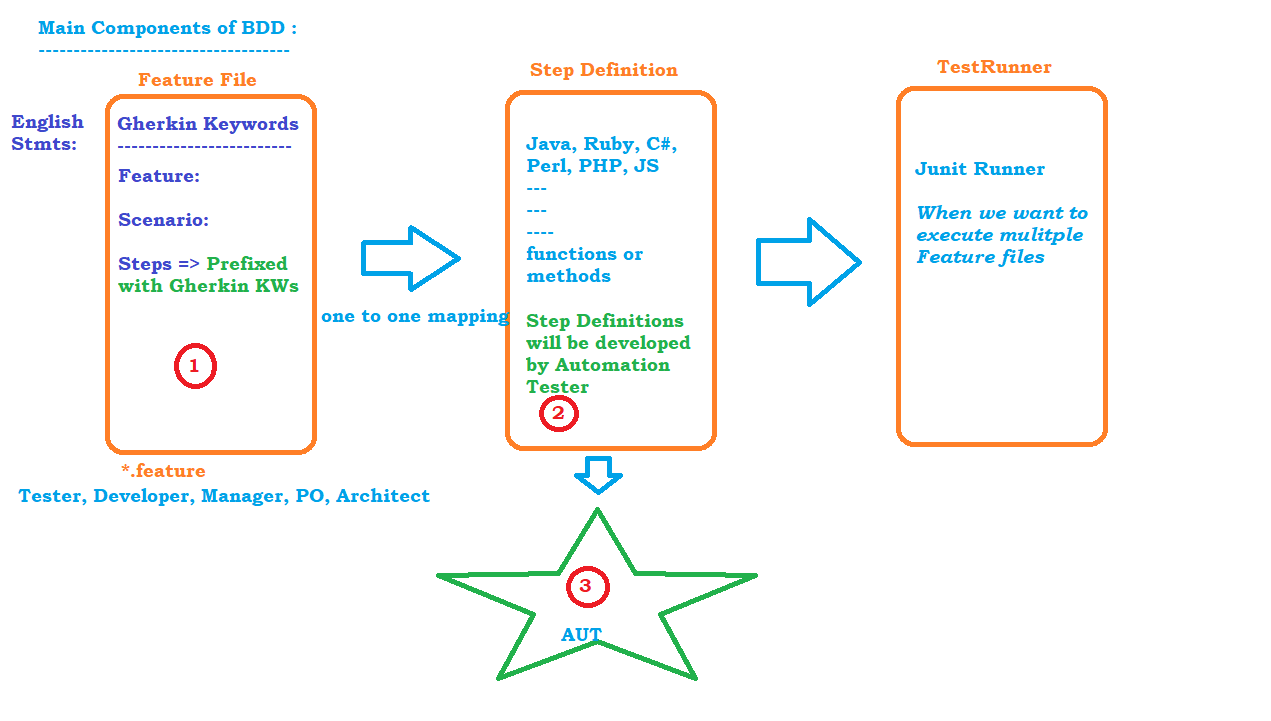
BDD Cucumber

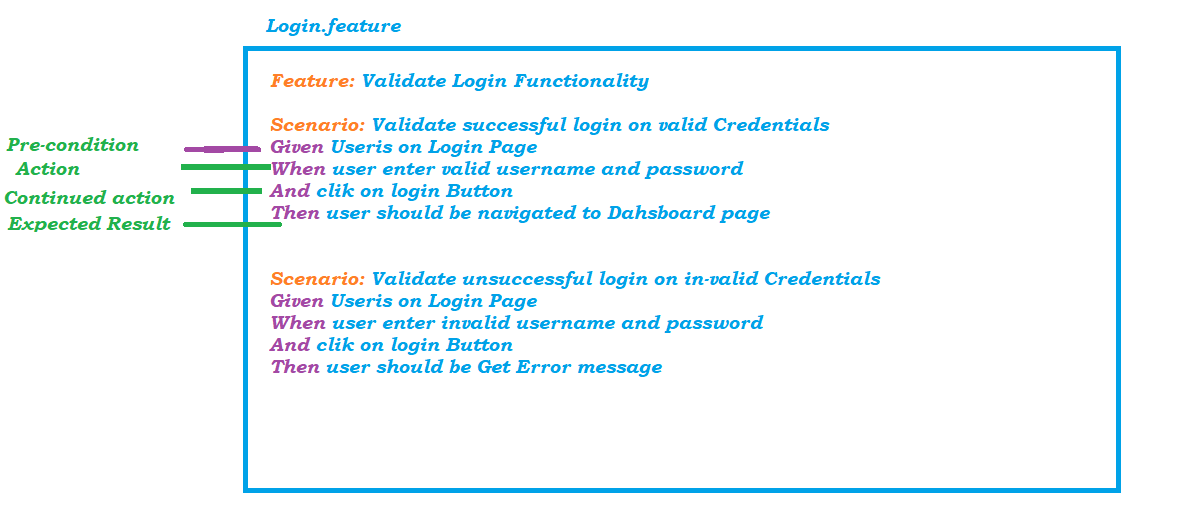








Sample Feature file



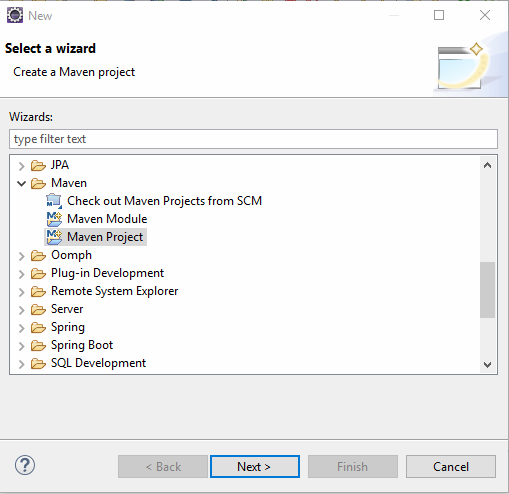
Eclipse Setup for Maven and BDD

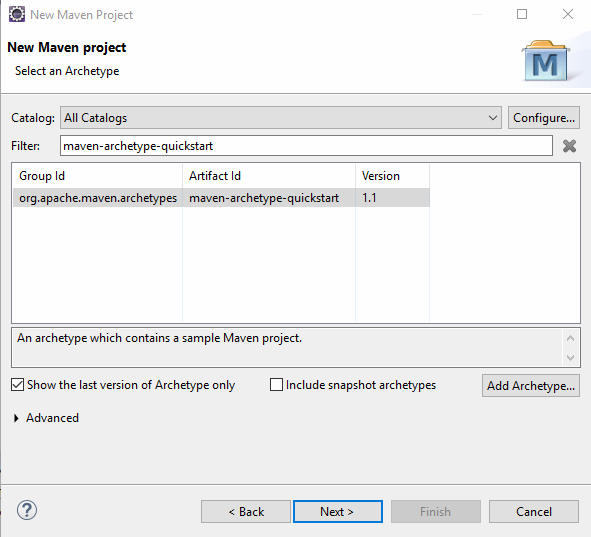
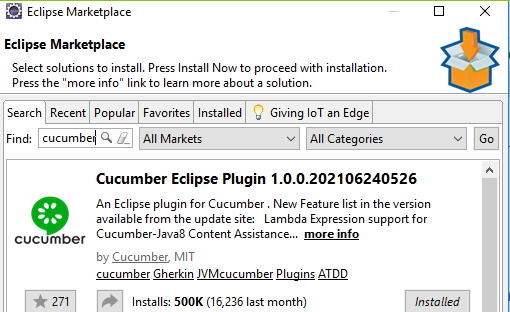
1. Download maven <https://maven.apache.org/download.cgi>
2. Unzip the downloaded zip file
3. Update the Environment Variables
   1. MAVEN\_HOME - D:\maven\apache-maven-3.6.1
   2. M2 - D:\maven\apache-maven-3.6.1\bin

OR %MAVEN\_HOME%\bin

* 1. PATH - D:\maven\apache-maven-3.6.1\bin

1. Download Eclipse from <https://www.eclipse.org/downloads/packages/release/2022-03/r/eclipse-ide-java-developers>
2. Create a Maven Project in Eclipse



1. select the archetype
   1. 
2. Update JDK And JRE
   1. Update JDK Version
      1. Right click on project -> select Java Compiler ->update the correct version
   2. Update JRE
      1. Remove the old JRE
      2. ADD JRE from JDK location
         1. Add library
         2. Click on JRE System library
         3. Alternate JRE
         4. Click on installed JRE
         5. Click on ADD -> Standard VM
         6. Browse and select JRE from JDK Location
         7. Click on apply and finish
   3. Update pom.xml for the dependencies
      1. cucumber-java : <https://mvnrepository.com/artifact/io.cucumber/cucumber-java>
      2. cucumber-junit : <https://mvnrepository.com/artifact/io.cucumber/cucumber-junit/7.3.4>
      3. junit: <https://mvnrepository.com/artifact/junit/junit/4.13.2>
   4. Install Cucumber Plugin
      1. go to Help ->marketplace
      2. 
      3. Write your first feature file

Test Runner

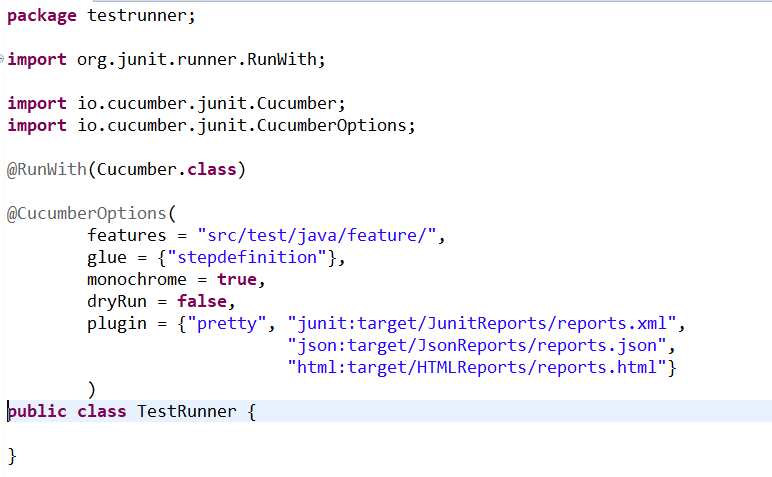
features : location of the feature files

glue: location of the step definition

monochrome: to remove the junk characters

dryrun: to check all steps in the feature file has a step definition

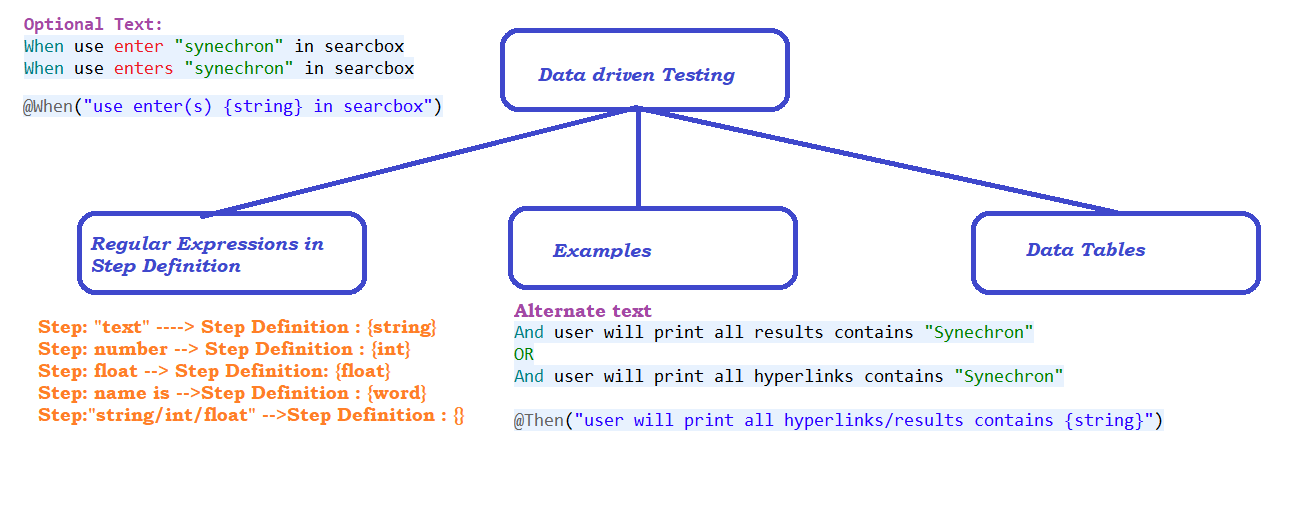
plugin: to get the reports , cucumber supports junit, json and html reporting.



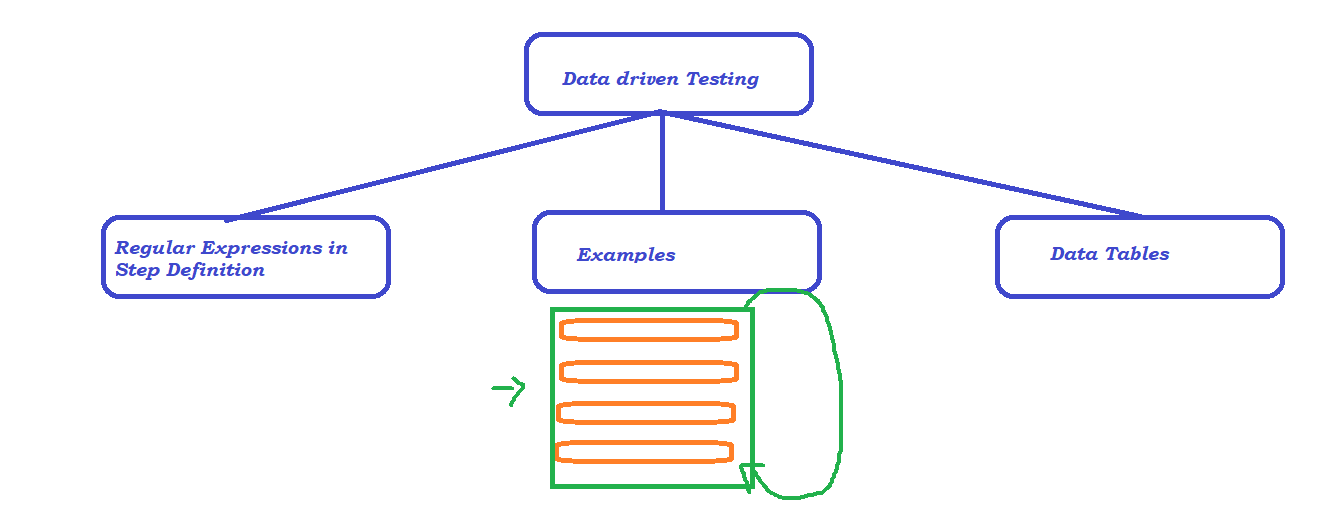
# Selenium Tests using BDD:

1. Update pom.xml with selenium dependencies
   1. WebdriverManager : is to automatically download the driver executable <https://mvnrepository.com/artifact/io.github.bonigarcia/webdrivermanager/5.2.0>

Data driven Testing

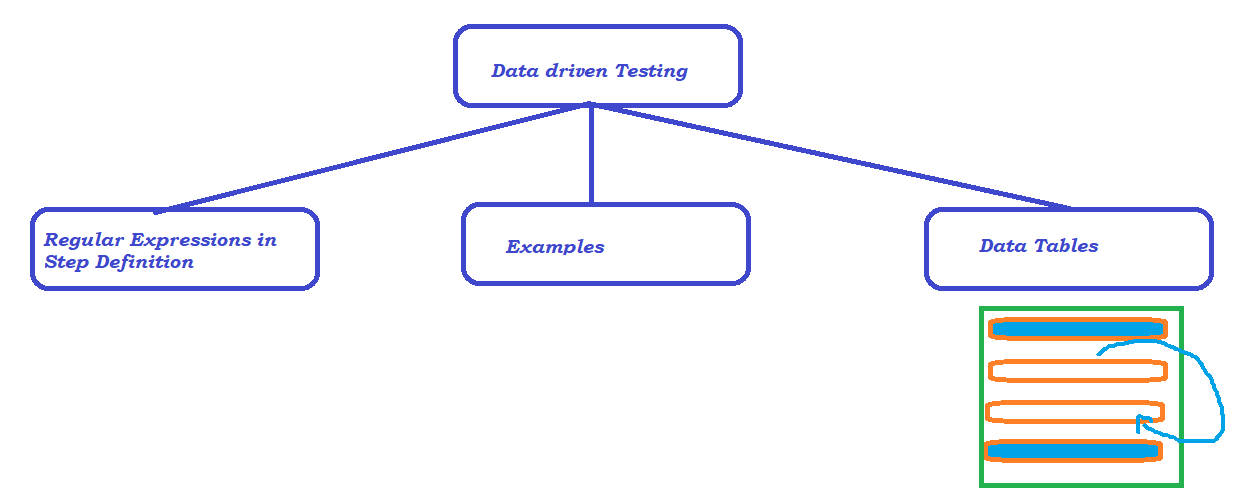


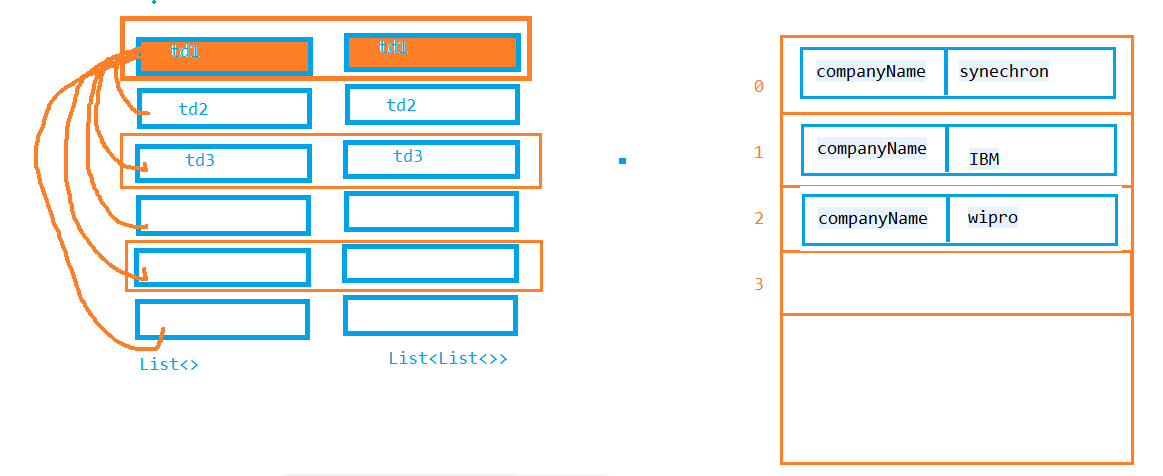
Examples :



Scenario using examples for google search

Data table:

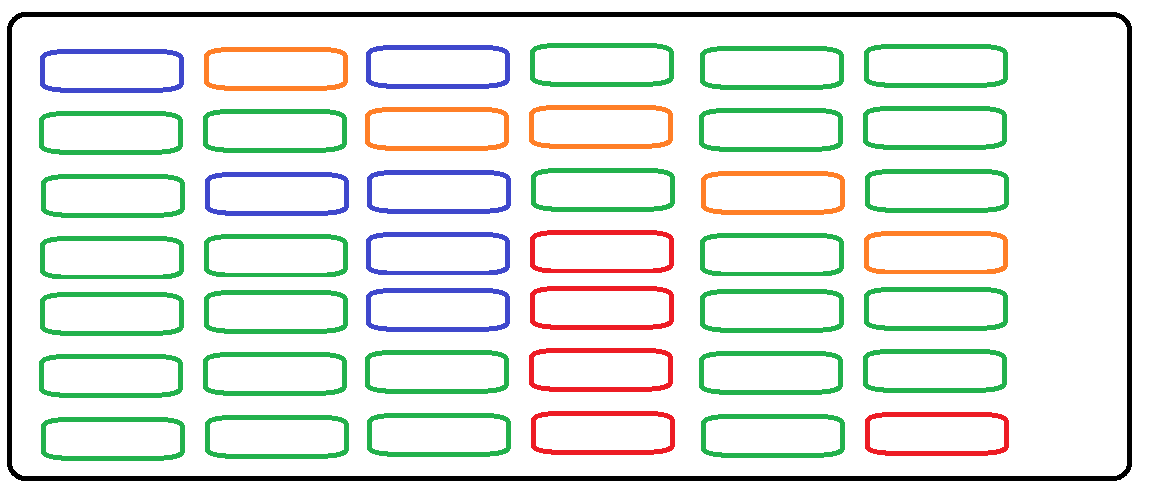




Tags

Tags are mainly used to group the tests based on the functionality

* smoke
* sanity
* uat
* production
* regression





Hooks

the main drawback of background is only precondition can be added.if we want to put both pre and post condition then we have to go with hooks.

to use hooks we have to use before and after tags, this is in a stepdefinition

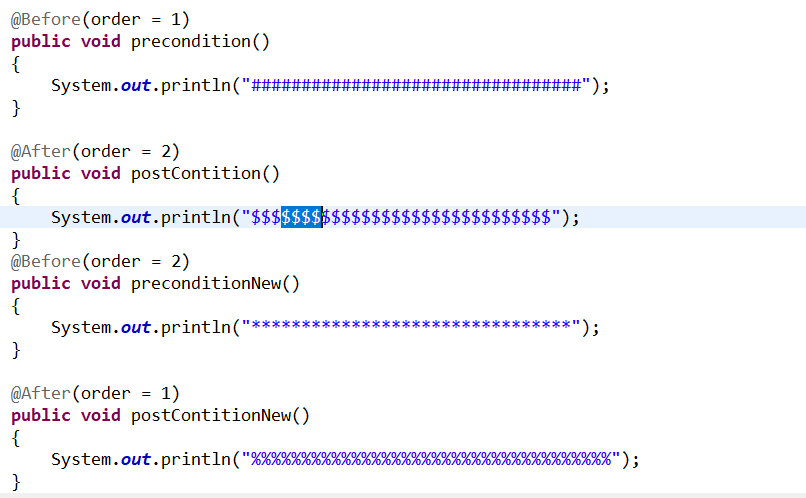
before

* creating driver
* launching application
* DB connection
* enable some cookies or proxy settings

after

* logout
* closing browser
* closing DB connection
* resetting all the settings

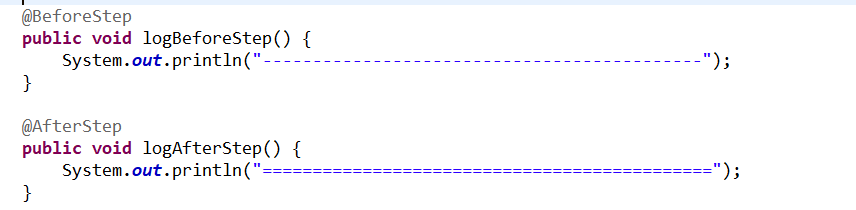
# - Order of Execution



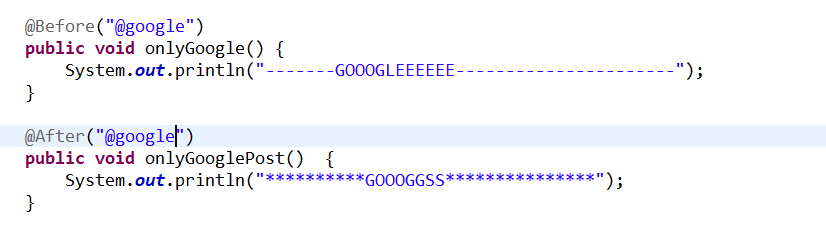
# Hooks for Steps:

If we want to execute any code before each step is executed then we can use

@BeforeStep and @AfterStep



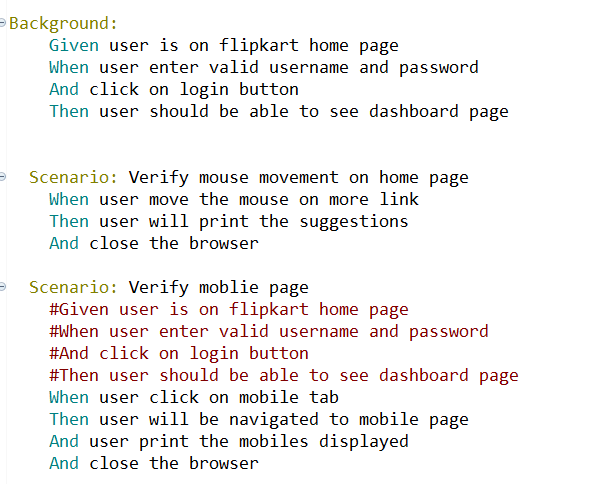
# - Conditional Hooks



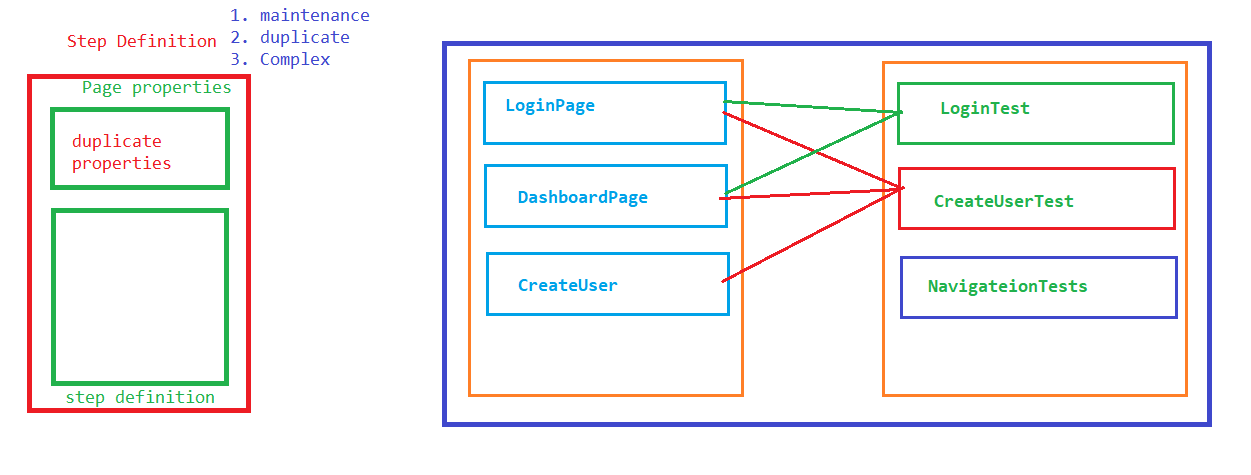
Background

if we have any pre condition to be executed, instead of keeping it in all tests / scenarios we can keep it in One place that is Background,

Before executing any scenario Background will be executed first. also background will be shown in Cucumber Report



POM



Execute tests using CLI

echo "Automation Run"

cd D:\Synechron\_BDD\_13\_June\Eclipse\_WS\BDDTraining

d:

mvn test -Dcucumber.filter.tags="@pom"