**Jenkins configuration**

* Plugins to install
  + Cucumber reports
  + HTML Publisher plugin
  + Ant Plugin
  + Build Timeout
  + Email Extension Plugin
  + GitHub Branch Source Plugin
  + Gradle Plugin
  + HTML Publisher plugin
  + Maven Integration plugin
  + Pipeline
  + Pipeline: GitHub Groovy Libraries
  + SSH Slaves plugin
  + Subversion Plug-in
  + TestNG Results Plugin
  + Timestamper
  + Workspace Cleanup Plugin
* Global Tool Configuration
  + Jenkins 🡪Manage Jenkins 🡪 Global Tool Configuration
    - Set Git path “C:\Program Files\Git\bin\git.exe”
    - Set name as “maven” and Maven path as “D:\Training\TrainingContent\SeleniumJars\_Software\apache-maven-3.5.0”

**Jenkins + Extent Report**

* If Extent report is not properly displayed in the browser (because of content security setting of Jenkins, please perform below steps)
  + Go to Jenkins installation directory i.e. C:\Users\admin\.jenkins
  + Open Jenkins.xml file
  + In the arguments tag please add ------hudson.model.DirectoryBrowserSupport.CSP="sandbox allow-scripts; script-src \* 'self' 'unsafe-inline' 'unsafe-eval'"
  + Save the file and restart the Jenkins from services

**TDD VS BDD**

* TDD
  + Tests written by developer
  + Run tests by dev, test failed
  + Develop code
  + Run Tests, test should pass
  + Refactor tests if required
  + Mostly written to test each unit level code/method (Unit Tests)
  + Covers low level design scenarios
* BDD
  + Stakeholders/Non-Technical defines behavior of application in simple English language
  + Design high level scenario to test expectation from application
  + Design normally for acceptance testing

<http://www.thetestroom.com/cucumber-parameter-tutorial/>

^ - This marks the beginning of the string

$ - This marks the end of the string

\ - This is used to allow using special characters in a Regex pattern such as " and .

\* - This is used to mean either zero, one of more

( ) - The text in these brackets are used to identify grouped characters

[ ] - This text is in these square brackets are used to match a single character using an or expression

. - This text will be used represent any character

|  |  |  |
| --- | --- | --- |
| **No.** | **Character Class** | **Description** |
| 1 | [abc] | a, b, or c (simple class) |
| 2 | [^abc] | Any character except a, b, or c (negation) |
| 3 | [a-zA-Z] | a through z or A through Z, inclusive (range) |
| 4 | [a-d[m-p]] | a through d, or m through p: [a-dm-p] (union) |
| 5 | [a-z&&[def]] | d, e, or f (intersection) |
| 6 | [a-z&&[^bc]] | a through z, except for b and c: [ad-z] (subtraction) |
| 7 | [a-z&&[^m-p]] | a through z, and not m through p: [a-lq-z](subtraction) |

|  |  |
| --- | --- |
| **Regex** | **Description** |
| . | Any character (may or may not match terminator) |
| \d | Any digits, short of [0-9] |
| \D | Any non-digit, short for [^0-9] |
| \s | Any whitespace character, short for [\t\n\x0B\f\r] |
| \S | Any non-whitespace character, short for [^\s] |
| \w | Any word character, short for [a-zA-Z\_0-9] |
| \W | Any non-word character, short for [^\w] |
| \b | A word boundary |
| \B | A non word boundary |

**Description:**

Do you want to write BDD style automation implementation with Cucumber, Gherkin and Java? You are at right place. We will cover starting from all basic setup and configuration required

This course will start from very basic level with no prior experience required and cover End-To-End automation with Jenkins. Cucumber is a Behavior Driven Development framework and will be used along with Gherkin, Selenium, Java, Maven, Ecllipse.

This course will also cover how you can push your written code to GIT (Version Control System) and use that code in Jenkins (CI) to run your automation and generate report. This is called as End-To-End Automation.

Basic understanding of Java or any other OOP language is required to understand the code.