Course Syllabus -What you are going to Learn?

Please note that this course goal is to Learn Cucumber and implement framework for Selenium tests. So basic Selenium knowledge is prerequisite skill to have before you see part 2 of the course.

This course is divided into 2 Parts.

In part 1 , You will get complete understanding of all Cucumber features and its framework functionalities which does not require any prerequisite. Below are the topics discussed in part 1

Understand Gherkin Syntax

Understand Cucumber Scenarios

Feature files & Step Definition files

cucumber Test Runner (Junit & TestNg)

Cucumber regular expressions

Cucumber tags & Background keyword

Cucumber Scenario outline for parameterization

Cucumber Datatables

Cucumber hooks

Tagged Hooks mechanism

Cucumber plugins

cucumber reports

Cucumber attributes

Cucumber Parallel execution

In Part 2- We will inject Selenium code and develop end to end Cucumber Selenium Framework from Scratch. Below are the topics

Build Maven Project with all the framework dependencies

Develop End to end Selenium Web Scenarios in the feature file

Implement TestNG /Junit Runner to run the framework

Understand the Single responsibility principle to distribute the implementation into multiple Step definitions.

Importance of Dependency injection to share the data between the Steps file

Understand the Page object Pattern to drive the locators from page files

Implement Factory Design pattern by writing PageObjectManager class to create objects of all PO classes

How driver can be configured and distributed across the files in the framework

How to run Cucumber Scenarios in Parallel Mode using Cucumber TestNG Runner

Build Test utilities for reusable Selenium methods

Implement Cucumber Hooks to capture Screenshots on test failures

Implement Cucumber Tags to run selected tests in the framework

Parameterizing the test data to run the Scenarios with multiple data sets

Creating HTML & Extent reports to run Cucumber Selenium tests

Running tests in parallel mode and generate Extent reports with screenshots attached for failed Scenarios

Run the cucumber tests using Maven & Command line options

Integrate the Framework to CI/CD Jenkins and Schedule the Jobs on regular time Interval

Create Parameterized Jenkin job to dynamically send the global properties at run time of Job execution