

[Selenium](#) [Java](#) [CUCUMBER](#) [Guide](#) [Wire](#) [Testing](#) [API](#) [Testing](#) [SQL](#) [Agile](#) [Scrum](#) [HR](#) [Round](#) [Profiles](#)

Cucumber_Options	Cucumber_Tags	Cucumber_Hooks	Back Ground Key	Scenario_vs_Scenario_outline
Cucumber_Datatable	Cucumber_Advantages_Pom.xml	Cucumber Basics	Framework	Parameters
Glue	Pending	Pending	Pending	Pending

Feature File & Scenario:

We will define the behavior in features file ,
in the feature file , we will write the scenarios by using GIVEN , WHEN, AND ,
THEN keywords ,
this language is called as a Jerkin language .

We will write all features files under src/test/resources
In Cucumber project the main packages are test runner package, step
definitions package , pages , HTML Reporting Folder.

We will implement all key words in step definitions. Java file
and We will execute all test scenarios by using Test Runner .java file

In Test Runner.java file , We will define @cucumber options

1. We have to specify the list of scenarios , what needs to be executed in the class path section
2. in The glue section , I have to specify where my step definitions. Java file package has been implemented
3. If I want to execute particular group of scenario's , I have to specify

the group name in the tag section

4. If I want to generate HTML reports , I have to specify in plugins section as extent report HTML plug in or cucumber html report plug in etc.

Scenario Outline :

If I want to execute my scenario file with the different sets of input data I can use scenario outline , there I have to specify the parameters in example section .

Back Ground Key :

In background key section , we will write some common reusable steps, which will be executed for every scenario.

If a feature file contains multiple scenarios , then background key section will be executes first for every scenario.

Hooks :

If I want to execute some snippet code globally before and after scenario , I will use hooks concept

I have to write @ before and @ after steps implementation in the hook. Java

Data tables;

I can specify my input data with column headers for required steps in scenario file by using data tables

If I want to read my input values from the scenario file, in the step

definitions.java file , I will specify input parameters within the curly brackets as string arguments.