Cucumber Framework

Q. Can you explain about Cucumber framework?

A.*Yah, we will define the behaviour in a feature file. In feature file we will write the set of scenarios, by using the keyword "given", "when", "then", "but ". So, this language is called as Gherkin Language.

***We will write the all-feature files under src/test/resources.**

*** In cucumber project the main packages are test runner, step definitions, Pages, and html reporting folder.**

***We will implement all keywords in the step definition.java file and we will execute all test scenarios by using testrunner.java file.**

*** In TestRunner.java file. We will define @cucumber options. We have to specify the list of scenarios what's need to be executed in class path section.**

***In the glue section we have to specify where the step definiton.java file package is implemented and if I want to execute particular group of scenarios we have to specify the group name in the tag section.**

***If I want to generate html reports. I have to specify Plugin section as extent report html plugin or cucumber report html plugin. So, that reports will be generated automatically in cucumber. Like that test runner.java file which contains @cucumber options.**

*** In the feature file we will write the scenario and scenario outline and Background key section.**

***And coming to Scenario outline. I want to execute my scenario file with different set of input data. I can use scenario outline. There i have to specify the parameter in the example section.**

*Coming to 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 it will be executed first for every scenario.

* Hooks concept - If i want to execute some snippet of code globally before and after scenario I will use hooks concept and I have to write @Before and @After steps implementation in hooks.java.

*For parameterization purpose we can also use data tables and we can also specify input data with column headers for required steps in the scenario file. By using data tables, we can achieve parameterization.

* If I want read input values from scenario file, in step definition. java file. I will specify the input parameters with the curling brackets as String arguments like that.

*By using input arguments, we are able to read input values from the data tables. Like that the behaviour can be separated in the feature file and the implementation can be implemented in step definition.java file.

* By using test runner.java file we can execute scenario files. Like that we can develop BDD cucumber framework. Yeah, that is the overview on the BDD Cucumber framework.