**What is BDD?**

BDD is one of the development approach involves the usage of common language that enhances communication between various tech and non-tech teams. Tests are more user-focused and based on the System’s behavior. In BDD, “ Given-When-Then” is the proposed approach for writing test cases.

**Given:** I am at the landing Page of Ecommerce application.

**When:** I searched for products which does not exist.

**Then:** No results message should be displayed.

**What is Cucumber?**

Cucumber is a tool/framework that supports behavior-Driven Development.

**Difference Between Cucumber & Selenium?**

Selenium is the tool to design web automation testcases.

Cucumber helps to design framework for selenium automation tests written in BDD standard.

**What is Gherkin?**

It is a language to define the testcases.

It is also known as Business Readable, Domain specific language.

**Keywords Used in Cucumber:**

Scenario, Feature, Feature file, Scenario outline, Step Definition.

**Scenario:**

In Cucumber TestCases are represented as Scenarios:

Scenarios contain steps which are equivalent to test steps and use the following Keywords(“Gherkin Syntax”) to denote them: Given, When, Then, But and And(“ Case Sensitive”).

**Given:** Precondition are mentioned in the Given Keyword.

**When:**  The purpose of the When steps is to describe the user action.

**Then:** The Purpose of the Then steps is to observe the expected output. The observations should be related to the business value/benefit of your Feature description.

**Scenario:** Make minimum Due payment

**Given:** user is on Pay credit card page.

**When:** usr fills all details and select Minimum amount option **And** user clicks on Pay button.

**Then:** Credit Card confirmation page is displayed.

When we specify a business requirement, sometimes there are multiple pre-conditions, user actions and expected outcomes.

**We are going to add one more scenario and will use the And and But Keywords:**

**And:** This is used for statements that are an addition to the previous Steps and represent positive statements.

**But:** This is used for statements that are an addition to previous Steps and represent negative statements.

**Feature and Feature File:**

Feature represents Business Requirement.

Feature File acts as a Test Suite which consists of all Scenarios.

In Cucumber, Feature files contain Scenarios. We can simply create feature file with, Feature extension.

Scenarios belonging to specific area of Application will be grouped into one Feature file.

The text that immediately follows the feature Keyword, and is in the same time, is the Title of the Feature file.

Feature file should contain either Scenario Outline. The naming conventions for feature files should be lowercase with feature extension.

**Dependencies for Cucumber Project:**

Cucumber JVM: Java

Cucumber JVM: JUnit

**What is the main component/terminology for automation in Cucumber?**

1. Feature File: Where you will be describe the Test case what you need to be automated.
2. Step Definition: How the tescases to be automated you can see in step Definition.
3. Test Runner: To connect these two we need testRunner.

**@CucumberOptions:** It is a annotation in cucumber that takes two files:- Features & step definition file to run.

Ex:- @CucumberOptions(

features="src/test/java/features",

glue="stepDefinition", stepNotifications = true)

**glue is a parameter that gives the path of your step definition file.**

**For run our test Runner file with TestNG we need to extends one file which name:- AbstractTestNGCucumberTests.**

**Difference between Junit Runner & TestNg Runner:-**

1. In Junit Runner we have to use @RunWith annotation for run our features file where as in TestNG Runner there is no need to use @RunWith annotation.
2. In TestNg runner we have to extends **AbstractTestNGCucumberTests** class where in Junit there is no need to extends it.

**Agar hum cucumber me kuch “ ” me likhte h to use cucumber manta h ki ye dynamic h.**

**Ex:- “John” & “1234”**

**Data Table to handle in Cucumber:**

In these cases we use pipeline method jese hmare pass sign up form hota h jisme first Name, Last Name, dob etc or bhi field hote h to isme hm niche diye hue trike se likhenge

| Abhay | Goel | 16-09-1997 | Shamli | --------------- -> Cucumber me ise Data Table khte h

**What is Parameterization?**

If you want to repeat your same test case multiple times with the different type of Data set then it is called Parameteriztion.

Parameterization k liye hme :- Scenario Outline use krna hota h. Parameterization k liye hm Examples: keyword use krte h jese:

Examples:

|username|password|

|user1 |Pass1 |

|user2 |Pass2 |

|user3 |Pass3 |

**Tagging Feature to control Test Cases in Cucumber:**

Maanlo hmar feature file me bahut saare testcases h lkin hme kuch testcases hi run krne h bss to uske liye hm tagging feature ka use krte h....

Ese case hm feature file k andr jkr @RegTest or @SmokeTest add kr denge phir hm apni testRunner file me jkr **@CucumberOptions** k andr glue tags k andr likh denge ki:

**glue=”stepDefinitions”, tags=”@SmokeTest or @RegTest”** isse ye dono trigger ho jayenge or agr hm ek ek lgayenge to kvl ek hi trigger hoga.

Agr hm chahte h ki ye vale na chle mtb negatition to hm likhenge ki

glue=”stepDefinitions”, tags=”not @SmokeTest”

**Importance of Background Keyword in Cucumber Scenario:**

**Imp: What is the purpose of Background keyword in Cucumber?**

**Usage of Hooks in setting up tear down and setup Scripts.**

**Imp: Hooks and Background ek sath use nhi kr skte h.**

**Difference between Background and Hooks:**

**Ques: What Keyword is used if you want to Parameterize the Test with multiple Data combinations?**

Example

**Ques: Which feature helps us to data feed into scenario?**

DataTable

**Ques: What Pre requiste feature can be used if you have multiple scenarios in cucumber feature file where each scenario lands on different portals?**

Hooks with Tag

**Ques: Which feature helps us to run only selected scenarios instead of complete feature file?**

Tagging

**Generating reports of cucumber scenarios:**

Agr hme cucumber me testcases ki report bnani ho to uske liye hm **@CucumberOptions** me ek keyword add kr dete h jiska naam h pretty.

Syntax:

@CucumberOptions(

Features=”src/test/java/features”,

glue=”stepDefinitions”,tags=”@PortalTest”,

plugin={“pretty”,”html:target/cucumber.html”}

**For Json File:-** plugin={“pretty”,”json:target/cucumber.json”}

**For Junit file:-** plugin={“pretty”,junit:target/cucumber.xml}

)

**Importance of dryRun and tags attribute in cucumber?**

**Importance of monochrome attribute in cucumber?**

**Importance of strict attribute in cucumber?**

**How we have addressed common problems with cucumber features?**

**What is Cucumber dependency Injection?**

It is done by pico-container.

**What is single responsibility Principle and loosely coupled classes?**