



www.techglobalschool.com

2860 S River Rd Suite 350, Des Plaines IL 60018

<i>Cucumber Interview Questions</i>	2
What is Cucumber?	2
In which language is Cucumber software written?.....	2
What language is used by cucumber?.....	2
What are the various keywords that are used in Cucumber for writing a scenario?.....	2
What is the use of Behavior Driven Development in Agile methodology?	2
What are the major advantages of the Cucumber framework?	3
What is a feature file?.....	3
When do you use Background keyword in Cucumber?.....	3
What is a Scenario in Cucumber?.....	4
What is a Scenario Outline in Cucumber?.....	4
What is the purpose of Examples keyword in Cucumber?	4
What symbol is used for parameterization in Cucumber?	4
What is tag in cucumber?	4
What are the files required to execute a Cucumber test scenario?.....	4
What is Step Definition in Cucumber?	5
What is the starting point of execution for feature files?	5
What is Runner class in Cucumber and what do you have there?	5
What is the use of features property under the @CucumberOptions tag?.....	5
What is the use of glue property under the @CucumberOptions tag?	5
What is the use of plugin property under the @CucumberOptions tag?	5
What is the use of tag property under the @CucumberOptions tag?	6
What is the use of dryRun property under the @CucumberOptions tag?	6
What is the use of monochrome property under the @CucumberOptions tag?	6
What is difference between Examples table and Data table?.....	6
Can you provide an example of the Runner class in Cucumber?	6
Can you provide an example of the feature file in Cucumber?.....	7
Can you provide an example of the Scenario Outline in Cucumber?.....	7
Can you provide an example of the Step Definition in Cucumber?	7
What are the Hooks in Cucumber?	7

Cucumber Interview Questions

What is Cucumber?

- Cucumber is a tool that is used to provide Behavior Driven Development (BDD) framework
- **REMEMBER:** The main aim of the Behavior Driven Development framework is to make various project roles such as Business Analysts, Quality Assurance, Developers, etc., understand the application without diving deep into the technical aspects
- Basically, we can say that "Cucumber is a software tool used by the testers to develop test cases for the testing of behaviors of the software."

In which language is Cucumber software written?

- Cucumber itself is written in Ruby programming language

What language is used by cucumber?

- **Gherkin** is the language that is used by the Cucumber tool
- It is a simple English representation of the application behavior
- **Gherkin language** uses several keywords to describe the behavior of applications such as **Feature, Scenario, Scenario Outline, Given, When, Then, And, etc.**

What are the various keywords that are used in Cucumber for writing a scenario?

- Keywords that are used for writing a scenario:
 - **Given** is used to specify a precondition for the scenario
 - **When** is used to specify an operation to be performed
 - **Then** is used to specify the expected result of a performed action
 - **And** is used to join one or more statements together into a single statement
 - **But** is used to denote a logical OR relationship between two propositions. OR can be combined with the Given, When, and Then statements

What is the use of Behavior Driven Development in Agile methodology?

- The advantages of Behavior Driven Development are best realized when non-technical users such as Business Analysts use it to write requirements to be implemented by developers or testers
- In Agile methodology, user stories with their requirements can be written in the Gherkin format which will help all team members especially developers and testers to understand easily to implement and test
- Cucumber is a tool that uses Behavior Driven Development to write acceptance tests of an application. It is used to bridge the communication gap between various project stakeholders
- The main use of Cucumber lies in its simplicity to be understood by non-technical users as requirements are written in English like sentences with Gherkin keywords

What are the major advantages of the Cucumber framework?

- Cucumber is an open-source tool, and it allows quick and easy setup and execution
- Cucumber syntax Gherkin makes it easy to understand application flow in the Agile environment. It is easy to use as even non-technical users can also understand the scenarios because of its plain text representation
- It is helpful to involve business stakeholders who cannot read the code
- It bridges the communication gap between various project stakeholders such as Business Analysts, Developers, and Quality Assurance personnel
- Automation test cases developed using the Cucumber tool are easier to maintain and understand
- Easy to integrate with other tools such as Selenium
- Cucumber tool provides support for multiple programming languages such as Java, .Net, Ruby etc.

What is a feature file?

- A feature file contains a high-level description of the test scenarios in Gherkin language
- A feature file consists of the following components like:
- **Feature** is used to define relevant scenarios and provide a high-level description of a software feature
- **Background:** is used to define common steps that will be executed before each scenario in a feature file
- **Scenario** is used to define steps and expected outcome for a specific test case
- **Scenario outline** is used to execute a scenario against multiple sets of data (used with Examples)
- **Examples** is used to define test data to be executed by a single scenario (used with Scenario Outline)
- **Given** is used to specify a precondition for the scenario
- **When** is used to specify an operation to be performed
- **Then** is used to specify the expected result of a performed action
- **And** is used to join one or more statements together into a single statement
- **But** is used to denote a logical OR relationship between two propositions. OR can be combined with the Given, When, and Then statements
- **NOTE:** The first line of the feature file must start with the keyword **Feature** followed by the description of the software feature
- A feature file may include multiple scenarios within the same file (ideal is up to 10)
- The extension of the feature file must be **.feature**

Example: TechGlobalLogin.feature

When do you use Background keyword in Cucumber?

- Background keyword is used only if you have all scenarios in the feature starts with same steps
- It groups common steps and increases the code reusability
- **NOTE:** Background can be used only for Before steps. You cannot use it for common After steps
- **NOTE:** Do not use Background if even a single scenario in the feature does not need the steps in the background

EXAMPLE

Background:

Given the authenticated user is on the application home page

What is a Scenario in Cucumber?

- Scenario is keyword that defines steps and expected outcome for a specific test case
- We may have multiple scenarios in a feature file (ideally up to 10)
- Each Scenario must have a description and related steps written in Gherkin language using Given, When, Then, And keywords
- If there is a Background in a feature file, steps defined with Background will be executed first before the Scenario steps
- If a Scenario required to be executed with multiple data sets, then it is better to it to a Scenario Outline

What is a Scenario Outline in Cucumber?

- Scenario outline is a way of parameterization of scenarios
- This is ideally used when the same scenario needs to be executed for multiple sets of data, however, the test steps remain the same.
- **Scenario Outline must be followed by the keyword Examples**, which specify the set of values for each parameter

What is the purpose of Examples keyword in Cucumber?

- Examples keyword is used to specify values for each parameter used in the Scenario Outline
- Scenario Outline keyword must always be followed by the keyword Examples

What symbol is used for parameterization in Cucumber?

- **Pipe symbol |** is used to specify one or more parameter values in a feature file

What is tag in cucumber?

- Tags in Cucumber are used to organize your test suites and makes it easy to run suites
- We can define each scenario with a useful tag and later, in the Runner class, we can specify which specific tag (and so as the scenario(s)) we want to execute
- Tag starts with @

NOTE: if you run your tests with no tag in the Runner class, then all your feature files will be executed respectively

EXAMPLES: @Smoke, @Regression

What are the files required to execute a Cucumber test scenario?

- There are 3 files required to execute a Cucumber test scenario
 - feature file: it is what we create first and define our test scenario using Gherkin language
 - step definition: it is the class that we write implementation behind the Gherkin steps
 - runner: it is the class that we link our feature files and their step implementation and run scenarios

NOTE: From these files first 2 are must, but Runner is highly recommended. Keep in mind that we can execute Cucumber scenarios without Runner as well

What is Step Definition in Cucumber?

- A step definition class in Cucumber is used to implement the Gherkin steps in the feature files
- Each step of the feature file can be implemented with a corresponding method on the Step Definition class using programming languages
- While feature files are written in an easily understandable language (Gherkin), Step Definition classes are written in programming languages such as Java, .Net, Ruby, etc.
- A step definition is the actual code implementation of the feature file Gherkin steps

What is the starting point of execution for feature files?

- It is the Runner class

What is Runner class in Cucumber and what do you have there?

- Runner is the class that we link our feature files and their step implementation and run scenarios
- A Runner class is generally an empty class
- There are **2 important annotations** we used in the Runner class
 1. **@RunWith()**
 2. **@CucumberOptions()**

@RunWith()

- This annotations makes the class Runner, without it you cannot execute the suites in the class

@CucumberOptions()

- This tag is used to provide a link between the feature files and step definition files
- Each step of the feature file is mapped to a corresponding method on the step definition file

EXAMPLE:

```
@CucumberOptions(  
    features="featuresPath",  
    glue={"stepDefinitionsPath"}  
)
```

What is the use of features property under the @CucumberOptions tag?

- **features property** is used to let the Cucumber framework identify the location of the feature files

What is the use of glue property under the @CucumberOptions tag?

- **glue property** is used to let the Cucumber framework identify the location of step definition files

What is the use of plugin property under the @CucumberOptions tag?

- **plugin property** is used to produce reports that contain information about what scenarios have passed or failed

What is the use of tag property under the @CucumberOptions tag?

- **tag property** is used to define which test scenarios will be executed

What is the use of dryRun property under the @CucumberOptions tag?

- **dryRun property** is used to compile cucumber features files and step definitions
- It is run to find any compilation errors
- It also gives you the steps that are found as not implemented yet

What is the use of monochrome property under the @CucumberOptions tag?

- **monochrome property** is used to make the Cucumber test console output more readable
- If it is set to false by default, and you can make it true to get the console output more readable

What is difference between Examples table and Data table?

- Examples table can be used with Scenario Outline and it is used to run a scenario against multiple sets of data
 - Scenario Outline will be executed for each row in the Examples section beneath it (not counting the first header row).
 - This works for the whole scenario
 - Example tables always have a header row because the compiler needs to match the header columns to the placeholders in the Scenario Outline's steps
- Data table is used to define the test data for a particular step in the scenario and it does not use any keyword
 - Data table works only for a particular step
 - Data table does not necessarily have a header row

Can you provide an example of the Runner class in Cucumber?

```
@RunWith(Cucumber.class)
@CucumberOptions(
    features={"featuresPath"},
    glue={"stepDefinitionsPath"},
    plugin = {"reportType1:path", "reportType2:path"},
    tag = {"@Smoke"},
    dryRun = false,
    monochrome = true
)
public class Runner {

}
```

Can you provide an example of the feature file in Cucumber?

Feature: Application Home page validation

Background:

Given the authenticated user is on the application home page

Scenario: Validate logo is displayed

Then validate the logo is displayed

Can you provide an example of the Scenario Outline in Cucumber?

Feature: Application Home page validation

Background:

Given the user is on the application home page

Scenario Outline: Validate member account login

When user navigates to login page

And user enters <username> and <password>

And user selects login button

Then validate the user is logged in

Examples:

username	password
techglobal1	1234
techglobal2	Abcd

Can you provide an example of the Step Definition in Cucumber?

- Let's take below feature step as an example and write implementation for it
When user navigates to TechGlobal School website

- Step Implementation of above When step could be as below

```
@When("user navigates to TechGlobal School website")
public void user_navigates_to_TechGlobal_School_website(){
    driver.get("www.techglobalschool.com");
}
```

What are the Hooks in Cucumber?

- Cucumber supports hooks, which are blocks of code that run before or after each scenario
- You can define them anywhere in your project or step definition
- It allows us to run before and after tasks using the annotations **@Before** and **@After**
- Cucumber Hooks allows us to better manage the code workflow and helps us to reduce the code redundancy