For automation testing, Selenium Webdriver is unquestionably the best automation tool available. But it’s a known fact that this is mostly a programmer-centric tool. Some of its users say that it’s a programmer’s delight but a nightmare for the software testers. Since it demands the users to possess technical skills which could become a challenge for the testers. You can curb these limitations by using Cucumber which is a behavior-driven development (BDD) framework. It reduces the gaps between a software developer and the tester. In this post, you’ll find the top Selenium Webdriver Cucumber interview questions and answers for practice.

The reason we are advocating for Cucumber is its ability to enable writing test cases in plain English language. It defines a set of three clauses to start writing a test case. With these three verbs *Given, When, and Then*, you can write test cases in a syntax which is far easier to learn than a standard programming language. The driving force behind Cucumber is the Gherkin language which infuses support of the English like syntax in the tool.
No doubt that reading tutorials and referring online study material give you an edge over your competitors. But the interview questions and quizzes organize your ideas and give the strength to remember the concepts much longer. At this blog, we host some of the best Java and Selenium Webdriver questions and quizzes to help software testers in preparing for their interviews. Here are some of the useful links for you to read from this blog.

Unlock The Top List Of Selenium Webdriver Cucumber Interview Questions.

In the latest edition of questions and answers series, we’ve added another blog post covering the Selenium Webdriver Cucumber interview. Since it’s relatively a less talked topic, so we tried hard to come up with the most relevant questions. We wish you make the most out of the below list. Let’s start now.

1- General Selenium Webdriver Cucumber Interview Questions.

Q-1: What is a test framework?
In general, a framework is an entity which binds several modules in a logical sequence to cover the end-to-end flows of an application. The objective of investing into a framework is to test a product which has a recurring roadmap and regular release cycle.

Q-2: What are the apparent advantages of a test framework?
Following are the possible benefits of using a test framework.

1- It reduces the complexity of using a variety of technologies inculcated in a product.

2- It organizes the unit and functional testing efforts of a developer and tester.
3- Provides early feedback on the quality of the code.
4- Helps in tracking test coverage as well as code coverage.
5- Results in easy debugging and reduces chances of errors.

Q-3: What type of tests Selenium can run?
1- You can use Selenium for the functional, regression, and load testing of the web-based applications.
2- You can employ this tool for doing the post-release validation.
3- Integrate it with the continuous integration tools like Jenkins, Hudson, QuickBuild or CruiseControl.
The above two questions were a little basic which sometimes we miss explaining during the interview. So, we thought it was worth starting with them.

Must Read – Top 20 Interview Questions for Automation Testing in Selenium.

Moving down, you can see all the questions below are specific to the Selenium Webdriver Cucumber interview.

2- Basic Level Selenium Webdriver Cucumber Interview Questions.

Q-4: What are the prereqs for building a Selenium Cucumber automation framework?
You might like to consider the following facts while creating a productive and scalable test framework.

1- Identify the type of application you are going to test. Is it a Web app, support mobile devices or runs on a desktop.
2- Would it require backend testing? e.g. Databases or SDK.
3- Decide on the input format. Is it static or dynamic?
4- Do you need to test the app for internationalization?
5- It must have a report which can help you trace a failure with minimum efforts.
6- It must support auto-generation of parametrization tests.
7- Have a config file to define any setup related settings or the global properties.
8- Apply abstraction at every level to separate the functionality.
If you follow the above rules, then you'll land up with a product which is easy to maintain and free to scale.

Q-5: List down the advantages of using Selenium as a testing tool?
1- It's an open source, so you save a lot on the cost side.
2- It gives you options to choose from a list of programming languages. e.g. Java, Python, C-Sharp, Ruby, and Python.

3- It offers easy and powerful dom-level testing.
4- You can use it in either of Agile or waterfall environment.
5- Easy integration with Jenkins, Bamboo, and some other notable CI tools.

Q-6: List down the mobile device which Selenium supports?
1- It supports Safari browser via a third-party driver. It is experimental and comes with limited functionality.
2- It provides an Android driver to run tests on its native mobile browser.

3- Intermediate Level Selenium Webdriver Cucumber Interview Questions.

Q-7: How to integrate Cucumber with Selenium Webdriver?
It's the most obvious Selenium Webdriver Cucumber interview question which you must know. And it's better if you give a step by step reply to the interviewer. It'll leave a positive impression on him as you'll show the depth of your knowledge.

1- Cucumber is a testing framework to run acceptance test cases. It creates scripts using the BDD approach.
2- It makes use of a feature file which describes the test cases in plain text format.
3- Here you write tests in simple English. And later use the Selenium Webdriver to run the test scripts.
4- To start Cucumber with Selenium, first of all, you require creating a Maven project in Eclipse.
5- In the Maven's POM file, you add the Cucumber dependency which brings the support of annotations like the <*Given*>, <*When*>, and <*Then*> and many other.

```
<dependency>
    <groupId>info.cukes</groupId>
    <artifactId>cucumber-core</artifactId>
    <version>1.1.5</version>
</dependency>

<dependency>
    <groupId>info.cukes</groupId>
    <artifactId>cucumber-java</artifactId>
    <version>1.1.5</version>
</dependency>

<dependency>
    <groupId>info.cukes</groupId>
    <artifactId>cucumber-junit</artifactId>
    <version>1.1.5</version>
</dependency>
```

6- Similarly, you can introduce the Selenium dependency into the above project. Alternatively, you can download the latest version of Selenium standalone jar from their website. And then, add to your project as an external jar file.
If you want to do it via the POM file, then add the following entry.

```
<dependency>
  <groupid>org.seleniumhq.selenium</groupid>
  <artifactid>selenium-java</artifactid>
  <version>2.53.0</version>
</dependency>
```

Q-8: Are there any readymade Selenium-Cucumber frameworks available?
Yes, there are a few we are listing down below. Though, we recommend building one of your own as it gives you more freedom.

1- Selenium-Cucumber framework for testing the web and android apps.
2- Another one is an acceptance testing framework using Cucumber and Selenium Webdriver.

Restore self-confidence, run through online tests.
Online Selenium Webdriver Quiz to Practice Before Interview.

4- Advanced Level Selenium Webdriver Cucumber Interview Questions.

Now Let’s start to discuss some of the internal Selenium Webdriver Cucumber interview questions.

Q-9: What are the two files which you need to run a Cucumber test scenario?
If you want to execute a Cucumber test, then make sure it has the following two files.

1- A feature file.
2- A step definition file.

Q-10: What does a feature file contain?
A feature file in cucumber specifies parameters and conditions for executing the test code. It can combine any of the following.

1- A feature.
2- A user scenario.
3- The scenario outline.

4- A <*Given*> clause.
5- A <*When*> clause.
6- A <*Then*> clause.

Q-11: What is a profile in cucumber?
You can create Cucumber profiles to run a set of features and step definitions. Use the following command to execute a cucumber profile.

```
cucumber features -p <profile_name>

#Example: cucumber features -p acceptance
```

Q-12: What are before, after, beforeStep and afterStep hooks?
1- Before: executes before the feature file execution.
2- After: executes after the feature file execution.
3- BeforeStep: executes before the each step execution.
4- AfterStep: executes after the each step execution.

Q-13: What are cucumber tags? And why do we use them?
Cucumber tags help in filtering the scenarios. We can tag the scenarios and then run them based on tags.

1- We can add tags to scenarios with <@> symbol.
2- We can use the following command to run a cucumber tagged scenario.

```
cucumber features -t @<tag_name>

#Example: cucumber features -t @test
```

Read more interview questions and plan in advance.
Appium Mobile Testing Interview Questions.

5- Expert Level Selenium Webdriver Cucumber Interview Questions.

Q-14: What is the purpose of cucumber dry-run?
We use to compile the cucumber feature files and step definitions. If there occur any compilation errors, then it shows them on the console.

Q-15: Why do you use the scenario outline?

We use it to execute the same scenario with different test data.

Q-16: What if you don’t use the cucumber keywords in test steps?

Please note that it’s not mandatory to write keywords in test steps.

For example, we can build a test step like the one shown in the next line.

e.g.- We are testing using cucumber.

Q-17: List out some of the main differences between Jbehave and Cucumber?

However, the Cucumber and Jbehave share the same perspective, but there are few key differences.

1- Jbehave is Java-based and Cucumber is Ruby-based.
2- Jbehave is story-driven whereas the Cucumber is feature-driven.

Q-18: When would you use RSpec and when to use Cucumber?

1- RSpec is more successful in doing unit testing.
2- As you know that Cucumber is a behaviour-driven development tool. So you can use it for System and Integration testing.

Q-19: What are the steps to generate a report in Cucumber?

We run the following command to produce HTML reports.

```
cucumber <featurename>.feature --format html --out report.html
--format pretty
```

Q-20: What is the right way to execute a specific scenario from the feature file?

We can select the target scenario from a feature file by providing its line number.

```
cucumber features/test.feature:10 --format html >
testfeature.html
```