* [What is Selenium?](https://www.interviewbit.com/blog/selenium-frameworks/" \l "what-is-selenium)
* [What are Selenium Frameworks?](https://www.interviewbit.com/blog/selenium-frameworks/#what-are-selenium-frameworks)
* [Top Selenium Frameworks](https://www.interviewbit.com/blog/selenium-frameworks/#top-selenium-frameworks)

**What is Selenium?**

Web applications are validated using Selenium, a free (open-source) automated testing framework that you can use on various browsers and systems. In addition, you may write selenium Test Scripts in a variety of programming languages such as Java, C#, Python, and other similar languages. Selenium testing is a term used to describe testing performed using the Selenium testing tool.

Selenium Software is more than just a single tool; it is a collection of software; each component caters to a certain organization’s Selenium QA testing requirements. In this article, we will discuss top selenium frameworks and what selenium frameworks are.

**What are Selenium Frameworks?**

The [**Selenium Framework**](https://www.interviewbit.com/blog/selenium-frameworks/) is a coding framework that simplifies and streamlines the process of code maintenance. Users may store “code” and “data” in the same area if frameworks are not used, making the code and data neither reusable nor understandable. In addition to producing desirable effects such as greater code reusability, increased portability, decreased script maintenance cost, and improved code readability, frameworks can cause negative consequences.

**Top Selenium Frameworks**

**1. Linear Scripting Framework**

Linear Scripting Framework is a basic level test automation framework that is implemented in the form of “Record and Playback” and is executed linearly. Also known as the “Record and Playback” framework, this framework is used to record and playback audio. This sort of framework is used to test apps that are of a small size.

In this testing method, the preparation and execution of test scripts are done on an individual basis for each test case.

The testers record each test step, such as browsing, navigation, user inputs, and imposing checkpoints. The testers then play the scripts back and use them to carry out the tests.

The following are some of the benefits of using Linear Scripting Automation Framework:

* It is possible to develop test scripts (record and playback) without spending a lot of time preparing ahead.
* It is not necessary to be familiar with coding.
* A simple method for quickly generating test scripts

The following are some of the disadvantages of the Linear Scripting Automation Framework:

* Because of auto generated scripts, there is a lack of reusability.
* We are unable to execute with numerous data sets since the data has been hardcoded.
* Finally, the maintenance cost is substantial — even minor adjustments need a significant amount of work.

**2. Modular Testing Framework**

In the modular testing framework, testers construct test scripts module by module, breaking down the whole application under test into smaller, independent tests. Modular Testing Framework:

The testers break down the programme into many parts and develop test scripts for each module independently, to put it another way. Combining these separate test scripts to create bigger ones by using a master script to produce the desired situations in each case. This master script is used to call the separate modules to perform end-to-end test scenarios from start to finish.

The primary rationale for using this framework is to provide an abstraction layer that protects the master module from modifications made in individual tests, which may be dangerous. In this framework, testers create function libraries that may be called upon whenever necessary. This is referred to as a modularity framework or a module-based framework in certain circles.

The following are the advantages of using a modular testing framework:

* Because the whole programme has been broken down into various modules, it has been made more scalable and simpler to maintain.
* Can develop test scripts on their own initiative.
* When a change is made in one module, it has little or no influence on the other modules.

The following are the disadvantages of the Modular Testing Framework:

* First, it takes more time to examine the test cases and find reusable processes than it does to write them.
* Numerous data sets are not allowed because the test scripts include hard coded data.
* Setting up the framework necessitates the use of coding abilities.

**3. Library Architecture Testing Framework**

Also known as “Structured Scripting” or “Functional Decomposition,” the Library Architecture Testing Framework is a framework for testing library architecture. It has some extra benefits since it is built on a modular foundation.

Similar to splitting the programme under test into modules in the modular testing framework, we identify the common tasks and combine them into functions in the functional testing framework (Figure 1). Once the functions have been grouped, the groupings will be stored in a library of sorts. Then, the test scripts re-use these libraries to generate new test cases.

The following are some of the benefits of using a LIbrary Architecture Testing Framework:

* The script’s upkeep is straightforward.
* Scalability is straightforward.
* The functions library is reusable, and you can re-use it.

The following are the disadvantages of using a LIbrary Architecture Testing Framework:

* Coding skills are necessary.
* Preparing test scripts takes more time than usual.
* The scripts include a defined set of test data that has been hardcoded into them.

**4. Data-driven Framework**

The data-driven test automation framework is concerned with separating the logic of the test scripts from the data utilized in the tests themselves. It enables us to generate test automation scripts by giving the script generator multiple sets of test data.

It is necessary to save the test data set in external files or resources such as Microsoft Excel sheets, Microsoft Access tables (MS Access database), SQL database (SQL database), XML files, etc. The test scripts establish connections with other services to get test data. We were able to quickly and simply make the test scripts function correctly for various kinds of test data by using this framework. When compared to the module-based architecture, this framework lowers the number of test scripts by a substantial amount. Because of the use of reusable tests, this framework provides increased test coverage while also providing more flexibility in the execution of tests just when necessary and by altering only the input test data.

It is dependable in that it does not affect tests when the test data is changed, but it has its own limitations, such as the need for testers who work on this framework to have hands-on programming experience in order to build test scripts.

The following are some of the advantages of a data-driven framework:

* It is compatible with a variety of data sets.
* Making changes to the test scripts will have no effect on the test data.
* There is no need to hardcode test data.
* It saves time by running a greater number of tests.

A data-driven framework has the following disadvantages:

* Coding abilities are required.
* More time is required for the setup of the framework and test data.
* Framework design requires the expertise of professional automation testers.

**5. Keyword Driven Testing Framework**

Keyword Driven Testing Framework (also known as table-driven testing or action word based testing) is a kind of testing framework that uses keywords to drive the testing process. We utilize a table structure in keyword-driven testing to declare keywords or action words for each function or method we want to test before executing it. Then, it runs automation test scripts on the basis of the keywords given in the excel sheet and reports the results.

Testers with no programming expertise can work on test automation scripts using this framework, which allows them to work with keywords to construct whatever test automation script they choose.

The code to read keywords from an external excel sheet and call the appropriate action specified in the sheet is included inside the main class. Keyword-driven testing is similar to data-driven testing in that keywords drive it.

Even though working on this framework does not need much programming knowledge, the initial setup (i.e., putting the framework into operation) necessitates more competence.

The following are some of the advantages of keyword-driven frameworks:

* First, it is not necessary to be an expert in order to develop test scripts.
* It is possible to re-use the code in this situation. For example, we may direct the separate scripts to the same Keyword by using a variable.
* Even though the application changes, test scripts remain unchanged.
* You may create tests prior to the development of the application.
* Test scripts may be used independently of the application under test by making just minor changes.
* It is not reliant on test tools.

Disadvantages of Keyword-Driven Frameworks include the following:

* It will take more time to design since the initial cost will be expensive.
* Employees with strong test automation abilities are required.

**6. Hybrid Driven Testing Framework:**

A hybrid-driven testing framework is a mix of two or more of the frameworks discussed above. It is a kind of test automation framework. It makes an effort to make use of the strengths and advantages of other frameworks for the specific test environment that it handles. In the present market, most teams are constructing this hybrid-driven structure.

**7. Behavior Driven Development Testing Framework**

This Testing Framework for Behavior Driven Development aims to develop a platform that enables everyone (including Business Analysts, Developers, and Testers, among others) to participate actively. More communication between the Development and Testing Teams is required. It does not need the knowledge of a programming language on the part of the consumers. In order to construct test requirements, we employ non-technical, everyday language. JBehave, Cucumber, and other similar tools are available on the market for Behavior Driven Development.

**Conclusion**

While there are many different kinds of test automation frameworks, we have only encountered a few. The Data-driven Framework, the Keyword driven Framework, and the Hybrid Framework are the most popular test automation frameworks available. When working with the Data-Driven Framework, we maintain test data in excel sheets and utilize TestNG’s data provider to run test cases on the data. In a keyword-driven framework, we create keywords in an excel sheet, and the code will refer to this file in order to run the test cases defined in the spreadsheet.

The hybrid framework is a blend of data-driven and keyword-driven frameworks in structure and functionality.

Automation testing with Selenium seems to be the most effective solution given the requirements for rapid and reliable testing that extends across several browsers and devices to ensure compatibility. Furthermore, the Selenium Framework, which is widely used in the industry, has advantages over alternative testing frameworks in terms of practicality, adaptability, and cost-effectiveness. This makes it an excellent method for providing a consistent end-user experience that is compliant with the continuous delivery approach to development.

**FAQs**

**Q1: Which Selenium framework is mostly used?**  
**Ans:** TestNG is one of the most widely used frameworks for the **Selenium WebDriver**, and it includes data-driven functionality. This paradigm is typically used in the supply chain and financial industries, where data is critical for operational efficiency. Advantages: Because test data is stored independently from the test script, it is simple to make modifications to the test script.

**Q2: Why do we use Framework in Selenium?**  
**Ans:** The Selenium Framework is a coding format that makes code maintenance easier than it otherwise would be. Benefits of using Frameworks include enhanced code re-use, greater portability, lower script maintenance costs, and higher code readability, among other things.

**Q3: Why does Selenium include Cucumber?**  
**Ans:** Cucumber and Selenium make it easy for managers, non-technical stakeholders, and business analysts to understand the test cases since they are written in simple feature files.

**Q4: Does Gherkin is supported by Selenium?**  
**Ans:** Top companies utilise Selenium because it’s straightforward to combine with Gherkin and allows for both comprehension and automation of the flow to be done simultaneously.