**Robot Framework**

**Robot Framework** is a generic open-source automation framework. It can be used for test automation and robotic process automation (RPA).

Robot Framework is actively supported, with many industry-leading companies using it in their software development.

Robot Framework is test automation framework for acceptance testing and acceptance test-driven development. It follows different test case styles – keyword-driven, behaviour-driven and data-driven for writing test cases. This feature makes it very easy to understand. Test cases are written using keyword style in a tabular format. Robot Framework provides good support for external libraries, tools that are open source and can be used for automation. The most popular library used with Robot Framework is Selenium Library used for web development & UI testing.

Robot Framework is open and extensible and can be integrated with virtually any other tool to create powerful and flexible automation solutions. Being open-source also means that Robot Framework is free to use without licensing costs.

Robot Framework has easy syntax, utilizing human-readable keywords. Its capabilities can be extended by libraries implemented with Python or Java. The framework has a rich ecosystem around it, consisting of [**libraries**](https://robotframework.org/#libraries) and [**tools**](https://robotframework.org/#tools) that are developed as separate projects.

Robot Framework project is hosted on [**GitHub**](https://github.com/robotframework/robotframework) where you can find further documentation, source code, and issue tracker. Downloads are hosted at **[PyPI](https://pypi.python.org/pypi/robotframework" \t "_blank)**.

Robot Framework is operating system and application independent. The core framework is implemented using [**Python**](http://python.org/) and also runs on **[Jython](http://jython.org/" \t "_blank)** (JVM) and **[IronPython](http://ironpython.net/" \t "_blank)** (.NET).

Robot Framework itself is open-source software released under [**Apache License 2.0**](http://www.apache.org/licenses/LICENSE-2.0.html), and most of the libraries and tools in the ecosystem are also open source. The framework was initially developed at [**Nokia Networks**](http://networks.nokia.com/) and was open sourced in 2008.

Robot Framework Features

In this section, we will look at the different features offered by Robot.

Tabular format for test cases

The robot framework comes with a simple tabular format where the test cases are written using keywords. It is easy for a new developer to understand and write test cases.

Keywords

Robot framework comes with built-in keywords available with robot framework, keywords available from the libraries like Selenium Library (open browser, close browser, maximize browser, etc.). We can also create user-defined keywords, which are a combination of other user-defined keywords or built-in or library keywords. We can also pass arguments to those keywords, which make the user-defined keywords like functions that can be reused.

Variables

Robot framework supports variables – scalar, list and dict. Variables in robot framework are easy to use and are of great help while writing complex test cases.

Libraries

Robot framework has support for a lot of external libraries like SeleniumLibrary, Database Library, FTP Library and http library. SeleniumLibrary is mostly used as it helps to interact with the browsers and helps with web application and UI testing. Robot framework also has its own built-in libraries for strings, date, numbers etc.

Resources

Robot framework also allows the import of robot files with keywords externally to be used with test cases. Resources are very easy to use and are of great help when we need to use some keywords already written for other test projects.

Data driven test cases

Robot framework supports keyword driven style test cases and data driven style. Data driven works with high-level keyword used as a template to the test suite and the test cases are used to share data with the high-level keyword defined in the template. It makes the work very easy for testing UI with different inputs.

Test Case Tagging

Robot framework allows to tag test-cases so that we can either run the tags test-cases or skip the tagged testcases. Tagging helps when we want to run only a group of test cases or skip them.

Reports and Logs

Robot framework provides all the details of test suite, test case execution in the form of report and logs. All the execution details of the test case are available in the log file. The details like whether the test case has failed or passed, time taken for execution, steps followed to run the test case are provided.

RIDE

This editor available with Robot framework helps in writing and running test cases. The editor is very easy to install and use. RIDE makes life easy for writing test cases by providing framework specific code completion, syntax highlighting, etc. Creation of project, test suite, test case, keywords, variables, importing library, executing, tagging the test case is easily done in the editor. Robot framework also provides plugins for eclipse, sublime, Textmate, Pycharm that has support for robot test cases.

Robot Framework Advantages

Robot framework is open source, so anyone who wants to try out can easily do so.

* It is very easy to install and helps in creating and executing test cases. Any new comer can easily understand and does not need any high level knowledge of testing to get started with robot framework.
* It supports keyword-driven, behaviour-driven and data-driven style of writing test cases.
* It is a good support for external libraries. Most used is Selenium Library, which is easy to install and use in robot framework.

Robot Framework Limitations

Robot lacks support for if-else, nested loops, which are required when the code gets complex.

Conclusion

Robot Framework is an open-source test automation framework for acceptance testing and acceptance test-driven development. The test cases in Robot Framework are based on keywords written in tabular format, which makes it clear and readable, and conveys the right information about the intention of the test case. For example, to open browser, the keyword used is “Open Browser”.

Robot Framework is a Test Automation tool in which the test cases are written using keywords that makes it easy to learn and use.

These keywords are written in a tabular form. With Robot Framework, the Test Scripts are replaced by a few keywords thereby replacing the need for large pieces of code.

Let us understand the keyword-driven approach of this Framework with a simple example.

Example: Suppose, I want to test a website say Google.com, for which the very first step would be to open a Browser and open the ‘Google.com’ webpage. Now to automate this step using Robot Framework, we have a keyword called “Open Browser”.

The script for this step would look as shown below:

|  |  |  |
| --- | --- | --- |
| Open Browser | Google.com | Chrome |

Do you agree that this code looks quite simple?

I am sure, your answer would be ‘yes’. This simplicity makes it easy to learn and use the tool. We have pre-defined Keywords and Libraries that can be used. Just by knowing these available keywords, we can automate our test cases and greatly reduce our testing efforts.

You must be wondering about the language in which the Framework is written. It has been written using the Python programming language. Hence, to install Robot Framework it is necessary to have Python installed in your system. There are also a few other sets of software that are required to use this Framework.

We will see more about this as we proceed further with this tutorial. Robot Framework provides good support for External Libraries as well as Functions. The most popular library used with the Framework is the Selenium Library that is used for web development & UI testing.

With this brief understanding of what Robot Framework is, let us take a look at the other features of this Framework.

Features Of Robot Framework

Enlisted below are the main features of the Robot Framework:

* Robot Framework is used for Acceptance Testing and Acceptance Test-driven development.
* The Framework uses the keyword-driven approach where small understandable words (either pre-defined or user-defined) are used for writing scripts.
* It supports Test Automation for different data sets thereby supporting data-driven testing.
* It shuns the use of large code and follows a behavior-driven testing approach.
* Test cases are written using keyword (pre-defined or user-defined) in a tabular format.
* Users have the option of creating their keywords.
* It supports the use of Variables.
* It can interact with third-party libraries and functions.
* It allows tagging of test cases that come handy while trying to run either of the Smoke Test Cases, Regression Test Cases, System Test Cases, etc.
* It provides detailed reports and logs of the execution status which is very helpful in case of failure of the script.
* The reports and logs are generated after every build execution.

Pros & Cons Of Robot Framework

Advantages:

Some of the advantages that contribute to the popularity of this Framework are:

* Being open-source, it is readily available for use by everyone.
* Its capability gets extended by the use of External Libraries and Functions.
* Besides the available Library Keywords, high-level keywords can also be created by the user which further enhances its use.
* As Robot Framework does not require the user to write a complicated piece of code, it is easy to learn and use the tool.

Disadvantages:

The most likely disadvantage of using this Test Automation Framework is that it does not allow nested loops and thus it comes as a constraint to test complex scenarios where multilevel looping is to be used.

With this basic idea of this Framework, let us move ahead to the next topic which will help you set up your system to use Robot Framework. We would now be going ahead with the understanding and installation of various software that is required to use the Framework.

Let’s get going!

Robot Framework – Installation Instructions

A couple of Software are required when it comes to using the Robot Framework.

The following software are required for us to be able to use the Framework.

1. Python
2. Pip
3. Robot Framework
4. WxPython
5. Ride