**Apache Log4j.**

If you ask an expert developer about the most annoying thing about an application, the answer might be related to logging. If there is no suitable logging in an application, maintenance will be a nightmare. Most of the application go through Development testing, unit testing, integration testing. But when it comes to production, you will always face unique scenarios and exception. So the only way to figure out what happened in a specific case is to debug through the logs. Many frameworks provide some way of default logging, but it’s always best to go with the industry standard logging mechanism. Apache Log4j is one of the most widely used logging frameworks. Apache Log4j 2 is the next version, that is far better than Log4j.

**Log4j2 Overview**

Using Logging API in application isn’t a luxury, it’s a must have. [Log4j](https://logging.apache.org/) is an open source library that’s published and licensed under **Apache Software**. You can debug an application using Eclipse Debugging or some other tools, but that is not sufficient and feasible in a production environment. Logging mechanism will provide you several benefits that you will not find in normal debugging.

Various logging frameworks are available in java such as SLF4J and Log4J. **Apache Log4j** is a very popular and old logging framework. It is a reliable, flexible, and fast logging framework or APIs are written in Java developed in early 1996. It is distributed under the Apache software license. Log4J has been ported to the Python, Perl, and C, C++, C #, Ruby and Eiffel languages. This tool is used for small to large scale Selenium Automation projects.

Log4j views the logging process in levels of priorities and to offer mechanisms to direct logging information to a great variety of destinations, such as a console, database, file, UNIX Syslog, etc.