# Jenkins - Code Analysis

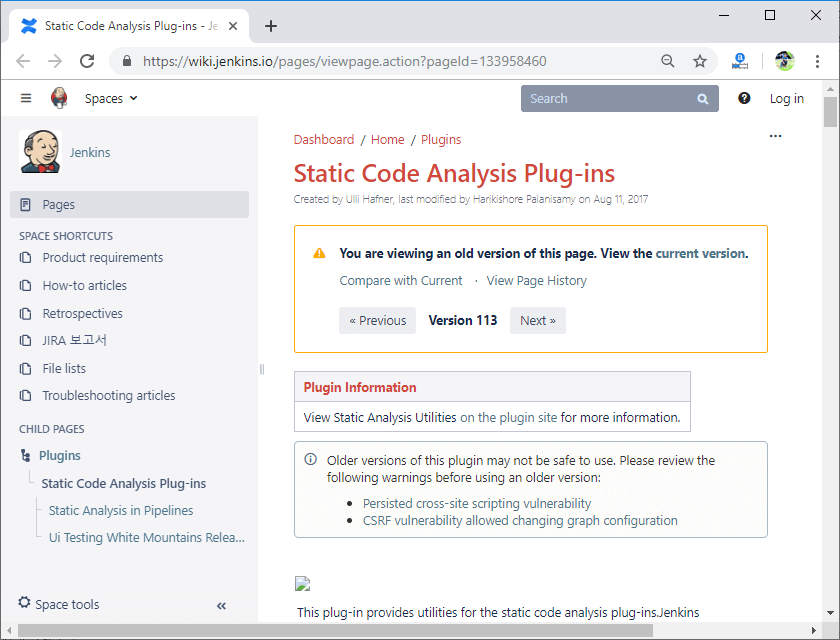
Now a day we are writing code. So it is important that the quality of the code must be good to have better performance.

The implementation of static analysis job within Jenkins, the most popular open- source tool for Continuous Integration (CI), and Continuous Deployment made it more interesting.

## Why Code Analysis?

Sometimes there may be a situation when compilers often fail to identify all the bugs. Static code analysis helps in improving the position a little.

Jenkins has a host of code Analysis plugin. The various plugins can be found at the official site of Jenkins: [https://wiki.jenkins.io/pages/viewpage.action?pageId=133958460](https://wiki.jenkins.io/pages/viewpage.action?pageId=133958460" \t "https://www.javatpoint.com/_blank)



These plugins provide utilities for the static code analysis plugins. Jenkins can parse the results file from various Code Analysis tools such as PMD, CheckStyle, FindBugs, etc. For each corresponding code analysis tool, a plugin in Jenkins requests to be installed.

Furthermore the add-on plugin **Static Analysis Collector** is available that combines the individual results of these plugins into a single trend graph and view.

The plugins can provide information such as:

* The total number of warnings in a job
* Trend Reports showing the number of warnings per build
* Overview of the found warnings per package, module, category, or type
* Detailed reports of the established warnings optionally filtered by severity (or new and fixed)