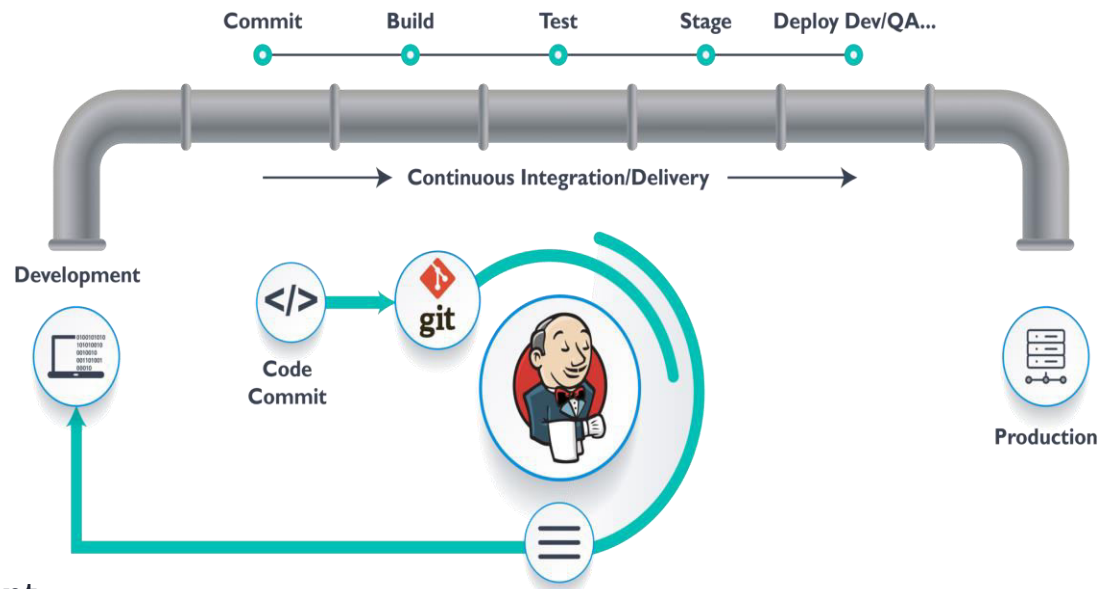# Jenkins

- To implement the Entire DevOps process, we will automate the pipeline in order to make the entire software development lifecycle in DevOps/automated mode. For this, we will need automation tools.

- **Jenkins** provides us with various interfaces and tools in order to automate the entire process.

- We have a Git repository where the development team will commit the code. From Git, Jenkins pulls the code and then Jenkins moves it into the **commit phase**, where the code is committed from every branch. The **build phase** is where we compile the code. If it is Java code, we use tools like maven in Jenkins and then compile that code, which can be deployed to run a series of tests.

- To install Jenkins
    - Download jenkins.war file from the Jenkins.io site (http://mirrors.jenkins.io/war-stable/latest/jenkins.war).
    - Open up a terminal in the download directory.
    - Run java -jar jenkins.war --httpPort=8080 (by default 8080)
    - Browse to http://localhost:8080.

- Most of the above tasks are featured and facilitated by appropriate **Plugins** e.g. Plugin for Git SCM Checkout, triggering an Email Post informing the status of the build to the concerned parties, etc. This is done with **manage plugins** tab in the Jenkins UI.

- Automation in Jenkins starts with a Jenkins Job

Plugins

BitBucket plugin:

Bitbucket plugin is designed to offer integration between Bitbucket and Jenkins. It exposes a single URI endpoint that you can add as a WebHook within each Bitbucket project you wish to integrate with. This single endpoint receives a full data payload from Bitbucket upon push (see their documentation), triggering compatible jobs to build based on changed repository/branch.

Email extension Plugin:

This plugin allows you to configure every aspect of email notifications. You can customize when an email is sent, who should receive it, and what the email says.

GIT plugin:

This plugin allows use of Git as a build SCM, including repository browsers for several providers.

NodeJS plugin:

Provides Jenkins integration for NodeJS & npm packages.

Parameterized Trigger Plugin:

This plugin lets you trigger new builds when your build has completed, with various ways of specifying parameters for the new build. You can add multiple configurations: each has a list of projects to trigger, a condition for when to trigger them (based on the result of the current build), and a parameters section.

Pipeline Plugin:

Jenkins Pipeline is a suite of plugins which supports implementing and integrating *continuous delivery pipelines* into Jenkins.

Sonar Quality Gates Plugin:

Jenkins plugin that fails the build if the predefined sonar quality gates are not green.

Credentials Plugin:

The credentials plugin provides a standardized API for other plugins to store and retrieve different types of credentials.

GitHub plugin:

This plugin integrates Jenkins with Github projects.