**Jenkins**

* Jenkins is a java application. It is platform independent and it doesn’t require any specific platform. It can run on windows/mac/Linux.
* Jenkins is used for continuous integration and continuous delivery.
  + - Continuous integration means whenever any resource updates their code and pushes in a shared repository, it is Jenkins’s job to trigger the build and inform the concerned team if there are any errors or not as soon as the build completes.
    - By using Jenkins, we will be able to find out which code has created issue, and can fix it asap, without wasting time in finding out which code in which commit has created the issue.
    - If the build has been deployed successfully, we can also accommodate unit tests and integration tests in our application, they can also be added as post build steps, and Jenkins will trigger them afterwards.
    - The report of post build steps is sent to us.
* Download Jenkins from “Jenkins.io”, place the downloaded war file in some drive in your system.
* Go to command prompt and go to the folder path where ‘Jenkins.war’ is placed and run command ‘java -jar jenkins.war’. This will extract the Jenkins war and it will be up and running in few minutes and will be accessible from any browser on port 8080.
* When you will get the massage on command prompt that “Jenkins initial setup is required”. Copy the password provided by Jenkins as it will be used for login in admin account. Go to browser and open “http://localhost:8080/” and give the password you copied to unlock Jenkins.

CLI: Command Line Integration for Jenkins: Easier, faster and more efficient. It consumes less memory than using the UI.i.e memory management is efficient.

1. To start Jenkins from command line on a different port, use command

‘java -jar jenkins.war –httpPort=9090’

1. Go to ‘Manage Jenkins’ on browser on port <http://localhost:9090/>.
2. Go to ‘Configure global security’ and make sure ‘Enable Security’ is checked.
3. Go to <http://localhost:9090/cli/> , and download Jenkins-cli jar. Place at any location in your system.
4. To test if the Jenkins command line is working, go to command prompt and go to the location where you have placed the Jenkins-cli jar.
5. Run ‘java -jar Jenkins-cli.jar –s http://localhost: 9090/ help’, If it asks for a “passphrase”, go to your browser where Jenkins is opened, go to ‘admin’ drop down and ‘configure’, you will find ‘SSH Public Keys’, whatever you set there, will be your passphrase.
6. On the browser, you will find multiple commands. Run Jenkins restart command and if it works, you can use Jenkins cli.