**Name: PARTH DAS**

**Roll No: C111**

**DIV: C-22**

**Sap ID: 60004220185**

**EXPERIMENT NO: 3**

**What is Jenkins?**

Jenkins is a tool that is used for automation. It is mainly an open-source server that allows all the developers to build, test and deploy software. It is written in Java and runs on java only. By using Jenkins we can make a continuous integration of projects(jobs) or end-to-endpoint automation

**Why Jenkins?**

Jenkins facilitates the automation of several stages of the software development lifecycle, including application development, testing, and deployment. Operating within servlet containers like Apache Tomcat, the technology is server-based.Continuous delivery (CD) and integration (CI) pipelines can be created and managed with Jenkins. The development, testing, and deployment of software applications are automated using CI/CD pipelines. You will be able to release software more regularly and with fewer problems if you do this.

1. jenkins is flexible.
2. You can add the n no.of plugins you want to add to the jenkins.
3. You can automate the proceses of CI/CD pipelines of all the projects.

**What is Jenkins CI/CD Pipeline?**

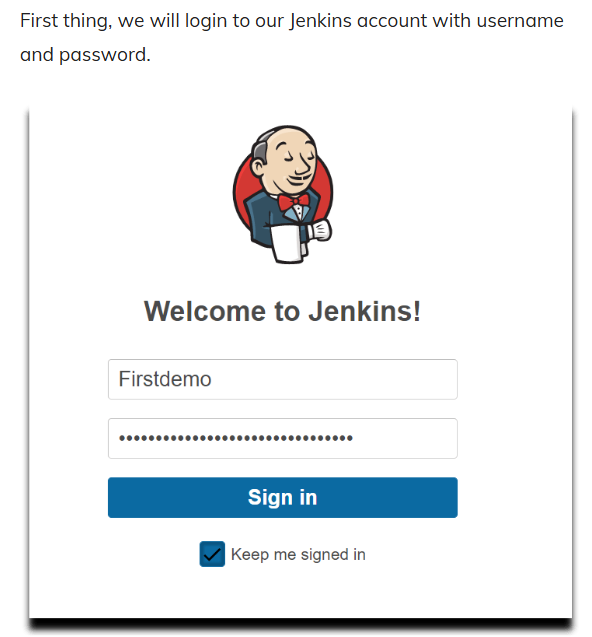
Jenkins CI/CD stands for Continuous Integration / Continuous Deployment first let us try to understand what is a pipeline. In computing, a pipeline is a set of stages or processes linked together to form a processing system. Each stage in the pipeline takes an input, processes it in accordance with a set of rules, and then sends the outputs to the stage that follows. Frequently, the pipeline’s overall output is its final step’s output. like the procedures outlined below

1. Test code
2. Build Application
3. Push Repository
4. Deploy to Server

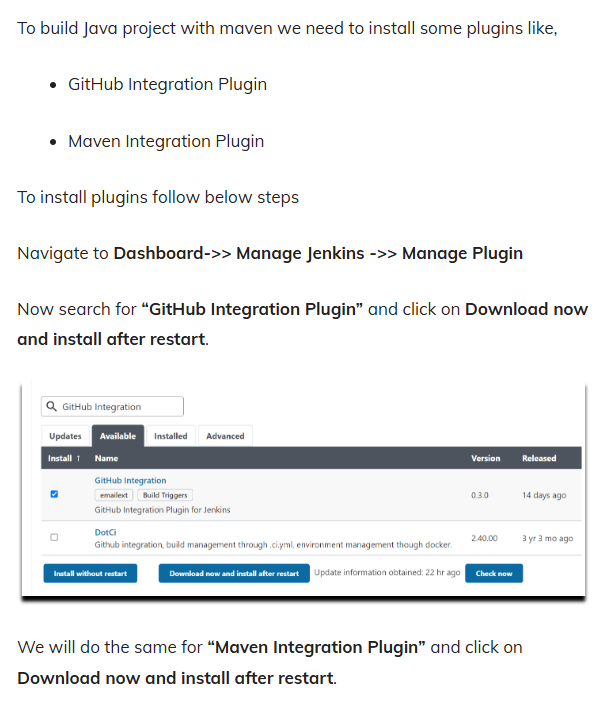
All the steps mentioned above will perform in sequence one after the other if any step/stage get failed it will not move forward to another step/stage until the previous step got a success. To master Jenkins and its integration in CI/CD pipelines, the DevOps Engineering – Planning to Production course provides detailed lessons and real-world projects.

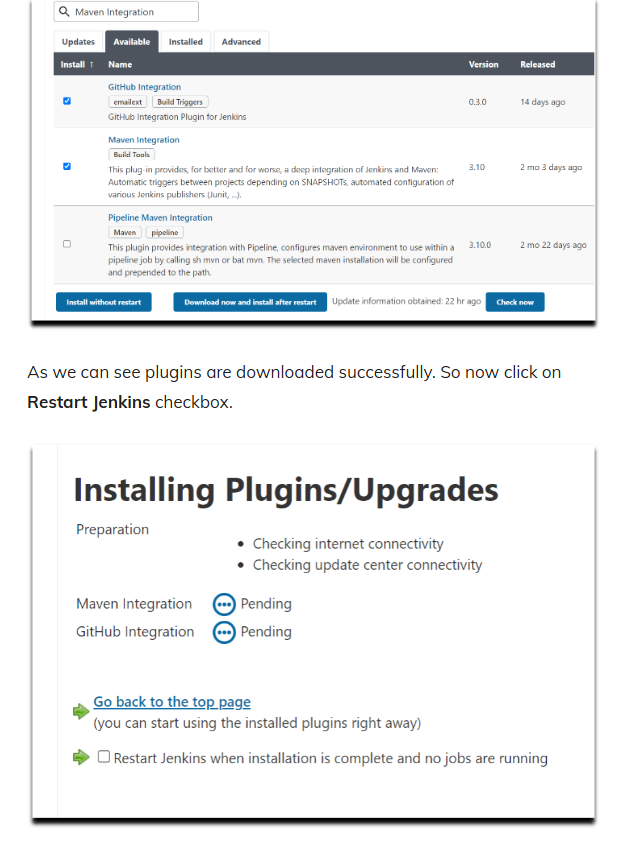
**How to Build Java Project using Maven in Jenkins?**

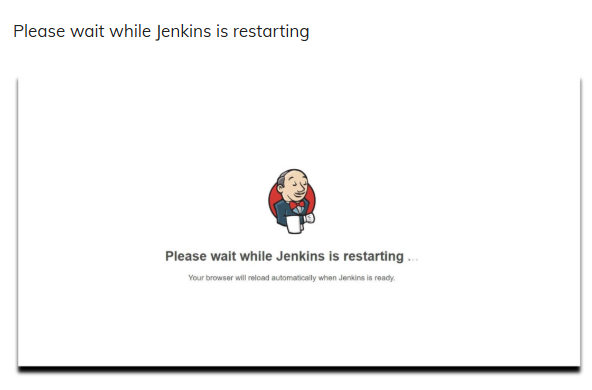
1. **Login to Jenkins**

****

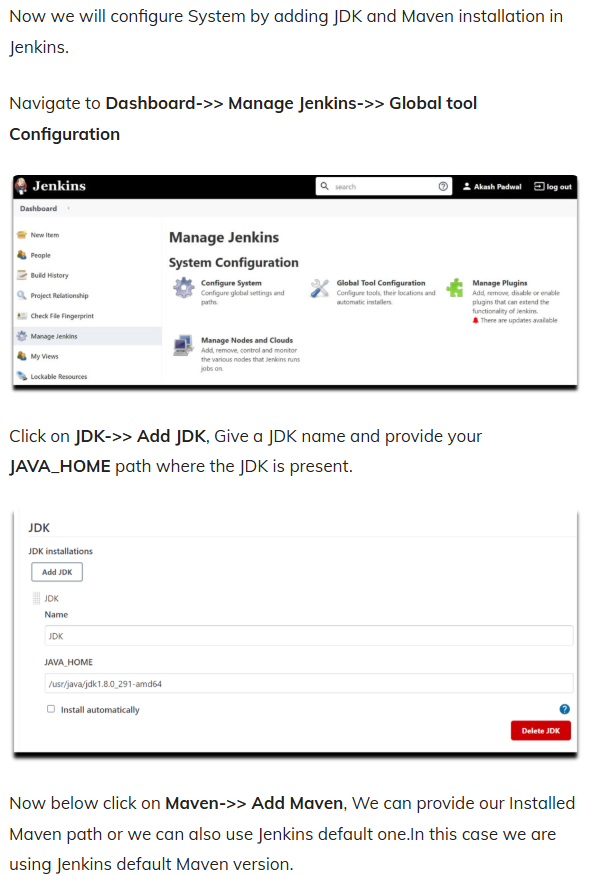
1. **Install GitHub and Maven Integration Plugins in Jenkins**

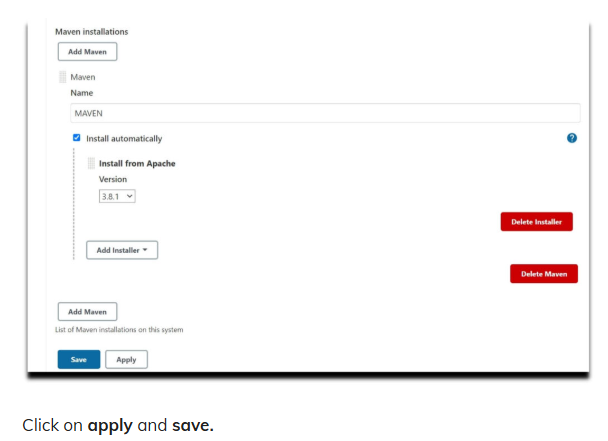
****

****

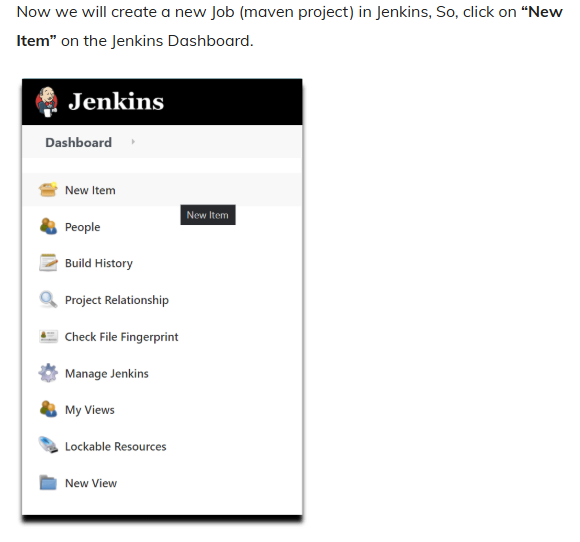
****

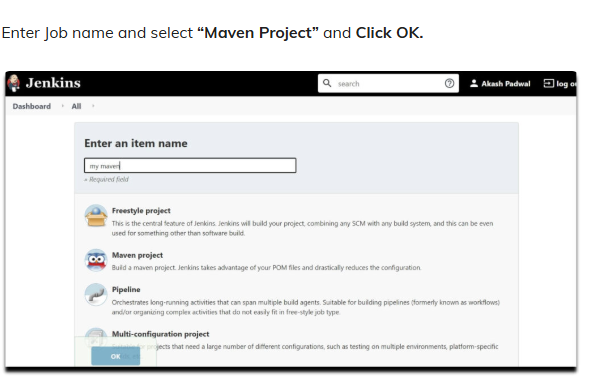
1. **Global Tool configuration in Jenkins**

****

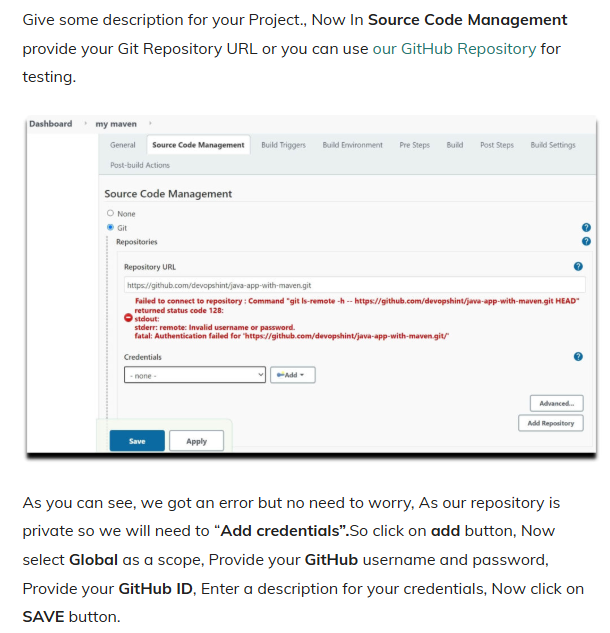
****

1. **Create Maven Project in Jenkins**

****

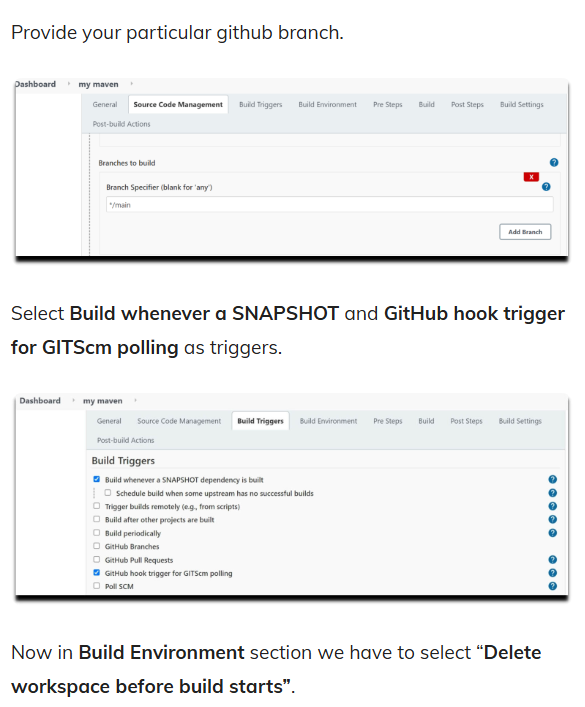
****

1. **Enter Project Details in Jenkins (source code management).**

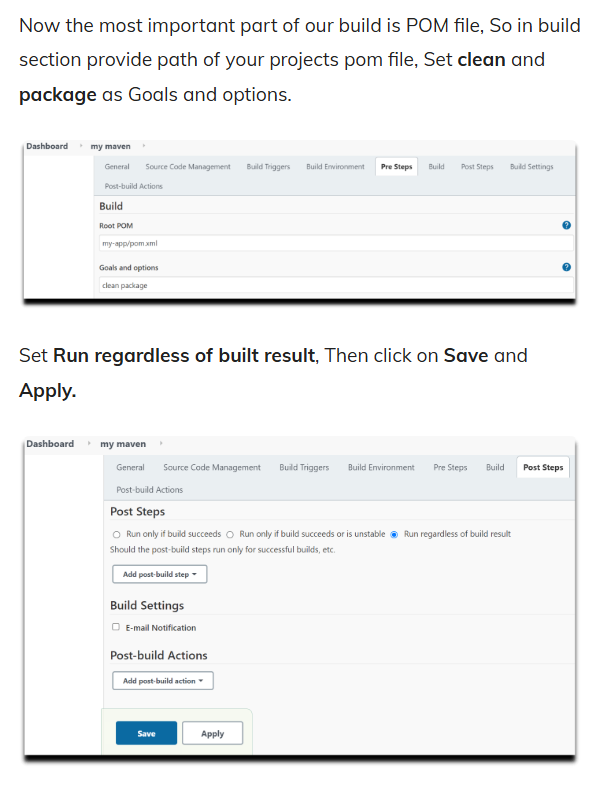
****

****

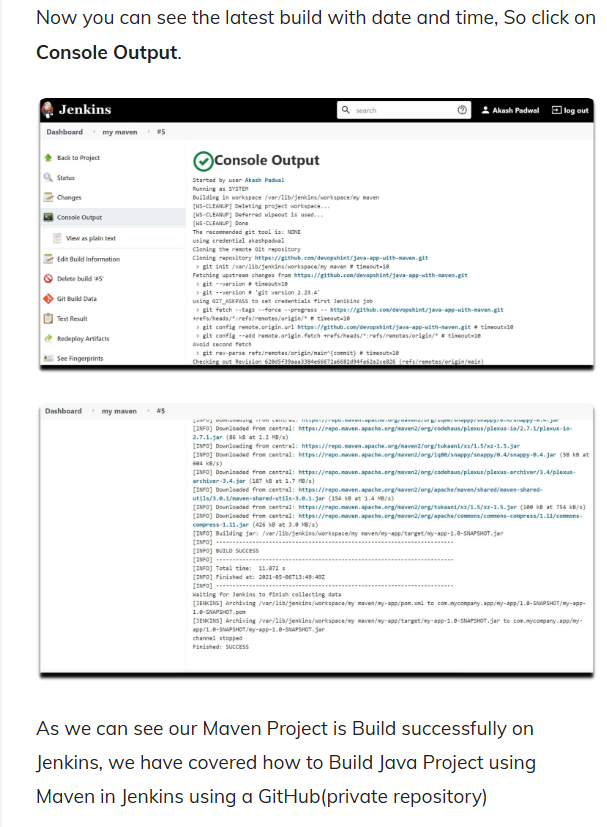
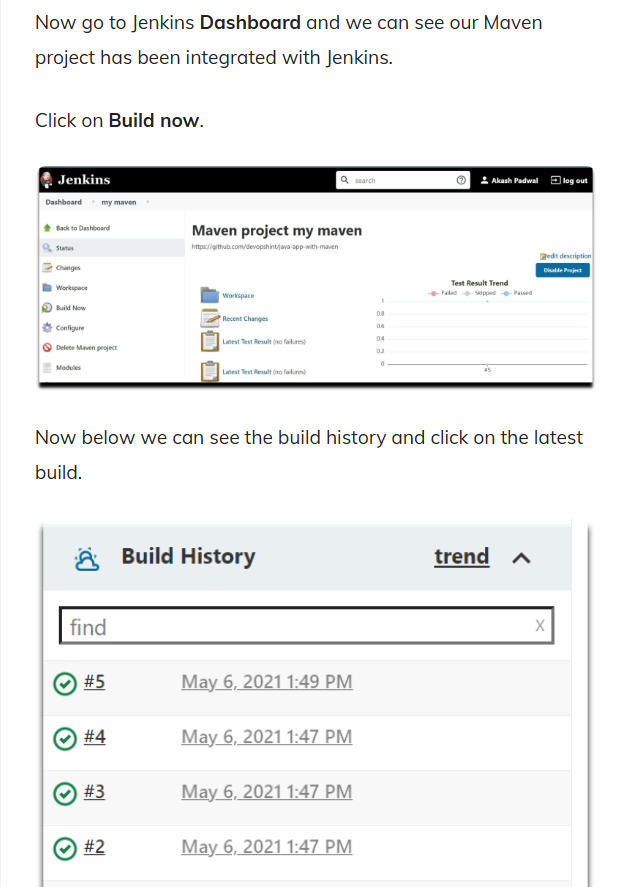
1. **Set Branch and Build Triggers in Jenkins**

****

1. **Provide POM file for Maven Project in Jenkins**

****

1. **Build maven project and check Console Output**

****