

=====
Jenkins CI CD
=====

Jenkins workshop video : <https://www.youtube.com/watch?v=4cG7dWKbrC8>

- 1) Build & Deployment process
- 2) Challenges with Manual build & deployment
- 3) Jenkins Introduction
- 4) Jenkins Setup
- 5) Jenkins CI CD Pipeline
- 6) Git + Maven + Docker + Jenkins

=====
What is Build & Deployment
=====

- 1) Take latest source code from Git Hub
- 2) Compile project source code
- 3) Execute Unit Test cases
- 4) Package the application (jar / war)
- 5) Build Docker Image
- 6) Create Docker Container

=====
Challenges in Manual build & deployment
=====

- 1) Every day we need to deploy latest code
- 2) Deploy code in multiple environments
- 3) Takes lot of time
- 4) Repeated Work
- 5) Error Prone

=====
What is Jenkins ?
=====

- 1) Jenkins is free & open source software
- 2) Jenkins developed using Java language
- 3) Jenkins is used to automate Build & Deployment process
- 4) Using Jenkins we can implement CI CD

=====
Jenkins Setup
=====

<https://github.com/ashokitschool/DevOps-Documents/blob/main/04-Jenkins-Docker-Project.md>

=====
Jenkins Declarative Pipeline Syntax
=====

```

pipeline {
    agent any

    stages {
        stage('Hello'){
            echo 'welcome to ashokit'
        }
    }
}

```

===== Git + Maven + Docker + Jenkins Integration =====

Git Hub Repo :: <https://github.com/ashokitschool/maven-web-app>

- 1) Configure maven as global tool
- 2) Install docker in jenkins machine
- 3) Restart jenkins server
- 4) Create CI CD pipeline
- 5) Execute CI CD pipeline
- 6) Enable host port in ec2 vm security group inbound rules
- 7) Access our application in browser

URL : <http://public-ip:host-port/maven-web-app/>

===== final ci cd pipeline =====

```

pipleline {
    agent any

    tools {
        maven "M3"
    }

    stages{
        stage('Git clone'){
            steps{
                git 'https://github.com/ashokitschool/maven-web-app.git'
            }
        }
        stage('Maven Build'){
            steps{
                sh 'mvn clean package'
            }
        }
        stage('Build Image'){
            steps{
                sh 'docker build -t ashokitapp .'
            }
        }
        stage('Deployment'){
            steps{
                sh 'docker run -d -p 9090:8080 --name ashokitc ashokitapp'
            }
        }
    }
}

```

=====
Real-Time workflow
=====

=> DevOps team will setup Jenkins server in linux vm

=> DevOps team will manage users in jenkins server
(role based access)

=> For Development team members only jobs read & execute access will be available.

=> Development team will send request to DevOps team to create jenkins pipeline for the project

=> Based on Dev Team request, DevOps team will create CI CD pipeline for the project.

=> Dev Team members can run CI CD pipeline for project build & deployment process.

Note: If CI CD job execution got failed then we need to check job execution logs (console output).

Note: For production deployment seperate release team will be available.

Note: Production Jenkins server will be seperate.