

DATE : 12.05.2025
DT/NT : DT
LESSON : JENKINS
**SUBJECT: INTRODUCTION AND
INSTALLATION**
BATCH : B 303

AWS-DEVOPS



TECHPRO
EDUCATION



techproeducation.com



+1 (585) 304 29 59





Table of Contents

- ▶ CI/CD
- ▶ Continuous Delivery and Continuous Deployment
- ▶ Jenkins



Continuous Integration

Software Development

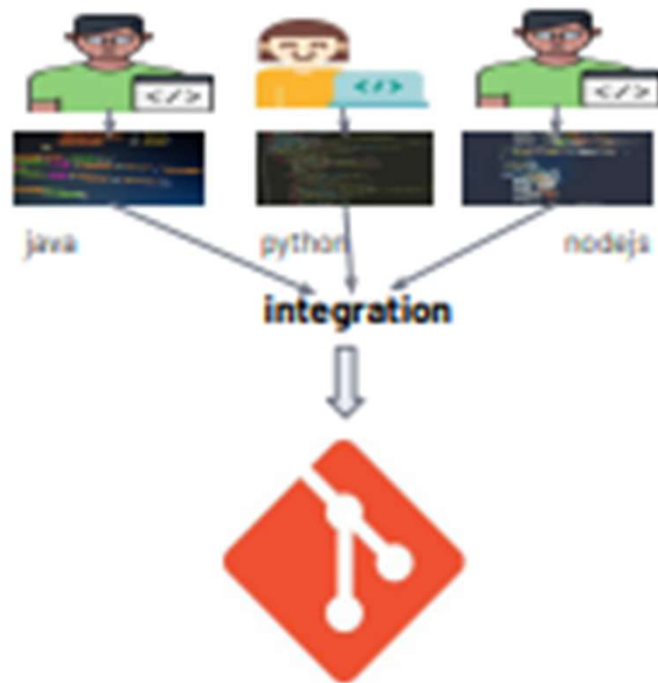
1. Developers write the code, then make unit and integration tests
2. QA and staging
3. Operation (deploy to production)



Integration Hell



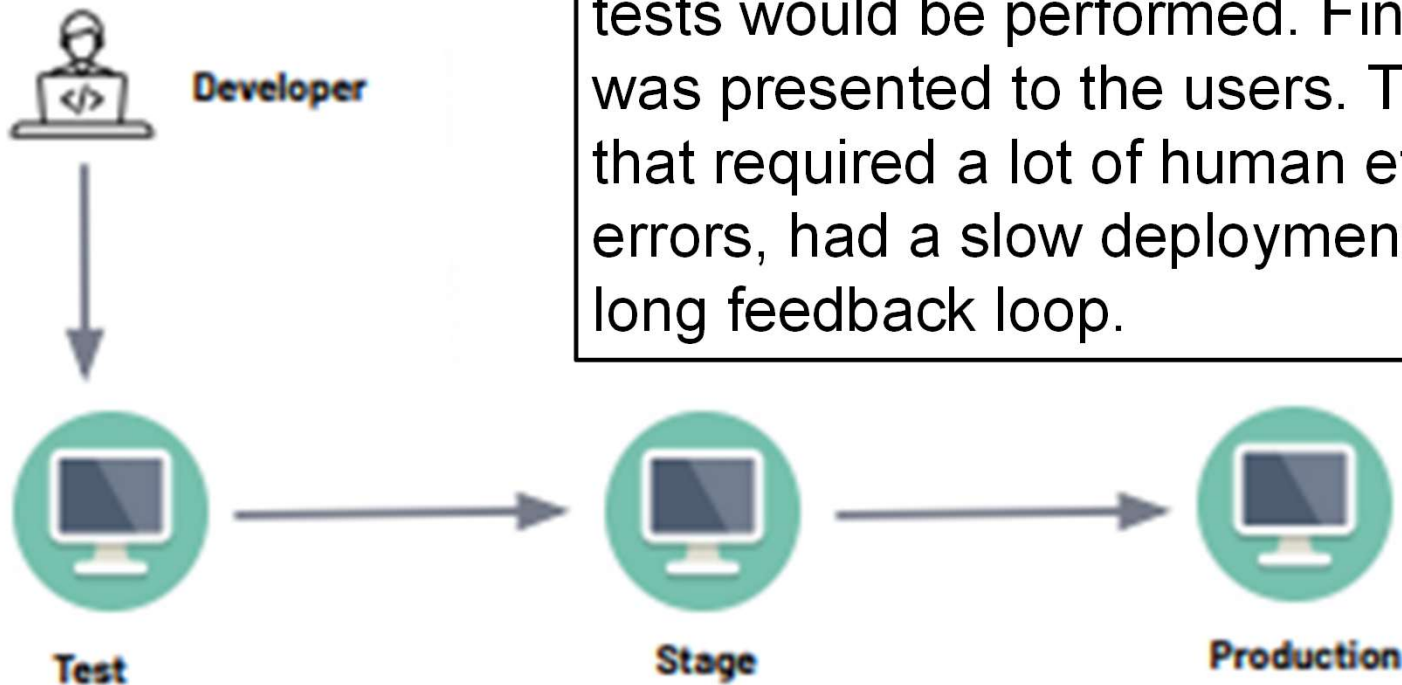
Merge the code



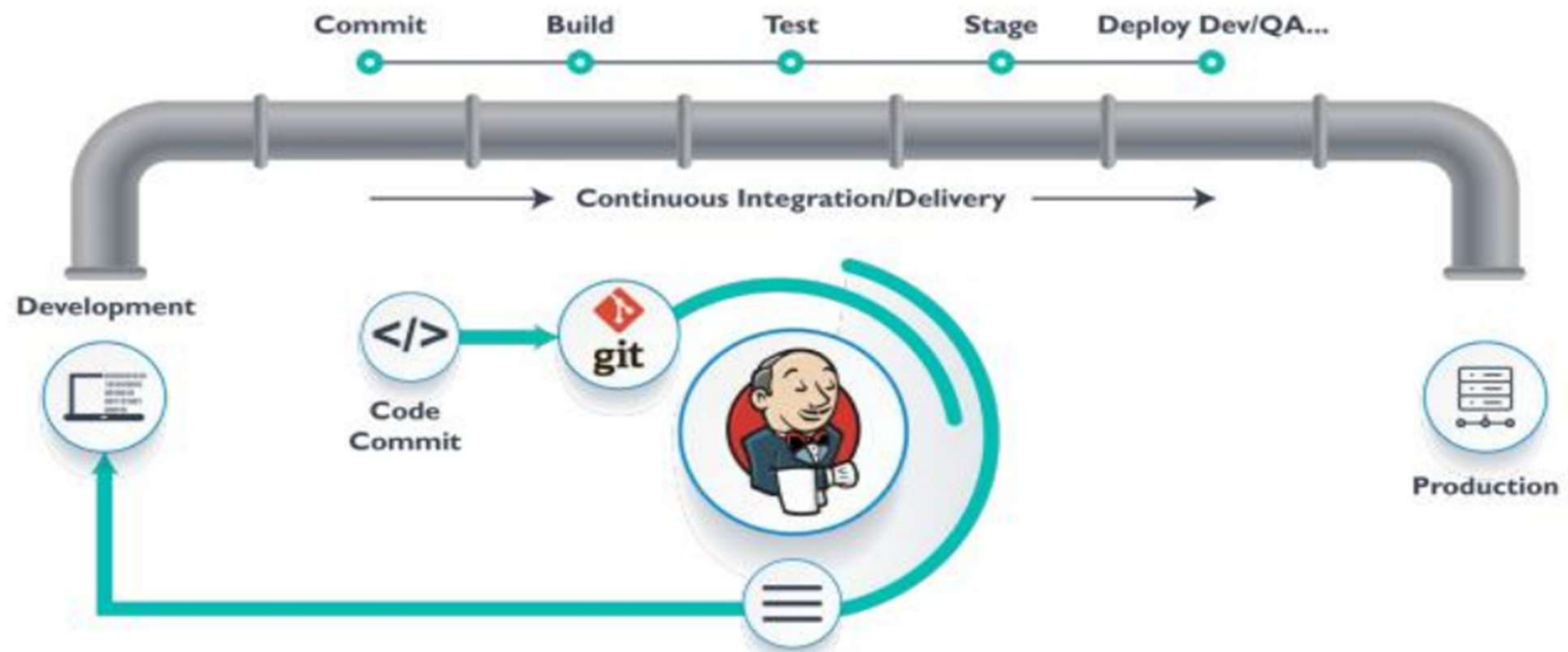
CI/CD Server

Before CI

Before, software development process used to be manual. Code was written later on, unit and integration tests were done. Then, after passing the tests, quality assurance and acceptance tests would be performed. Finally, the final code was presented to the users. This was a process that required a lot of human effort, was prone to errors, had a slow deployment process and a long feedback loop.



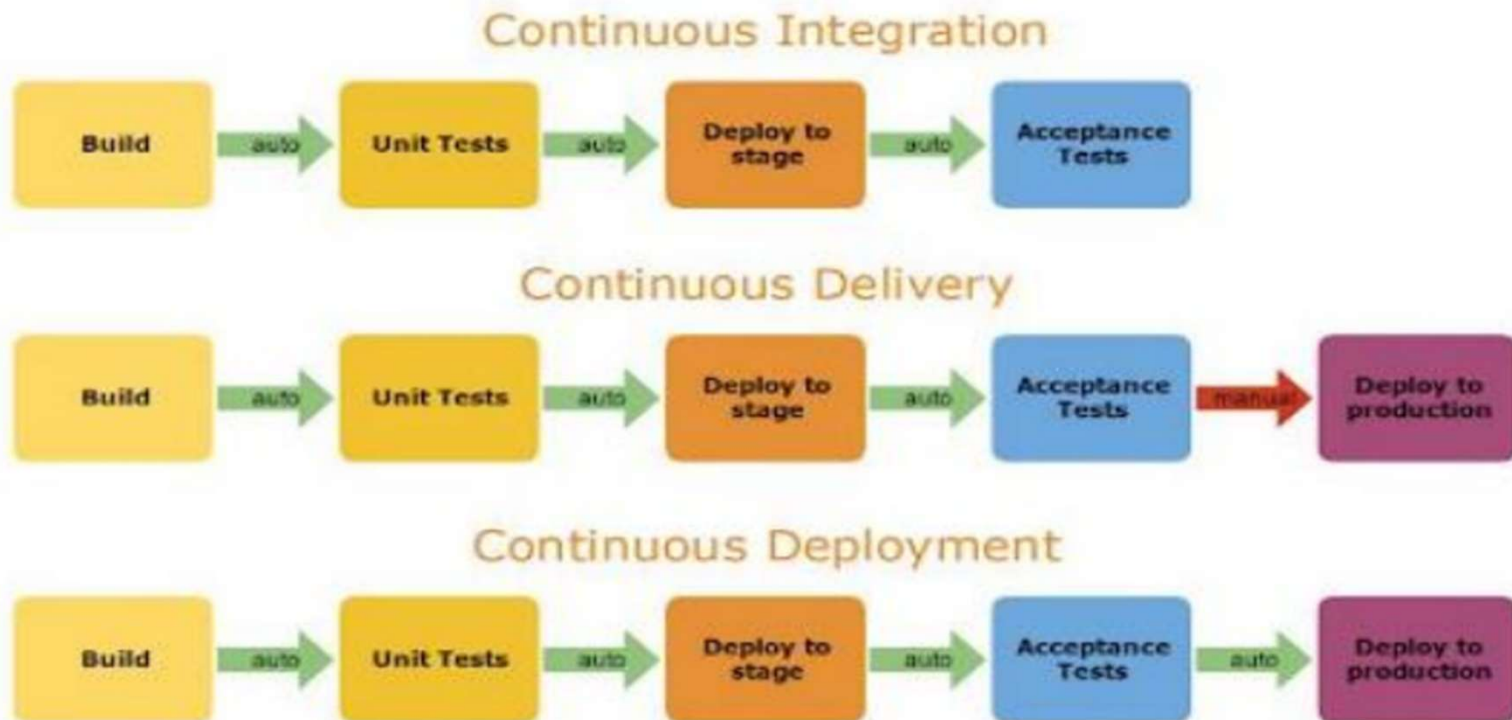
Monolith vs Microservices





Continuous Delivery and Continuous Deployment

Continuous Delivery and Continuous Deployment



Why CI/CD

- Automate quality control and deployment processes
- Fixing bugs in the source code early in the process
- A quick meeting of software developments with the user
- Make the software better
- Guarantee the software's compliance with standards
- Increasing user satisfaction

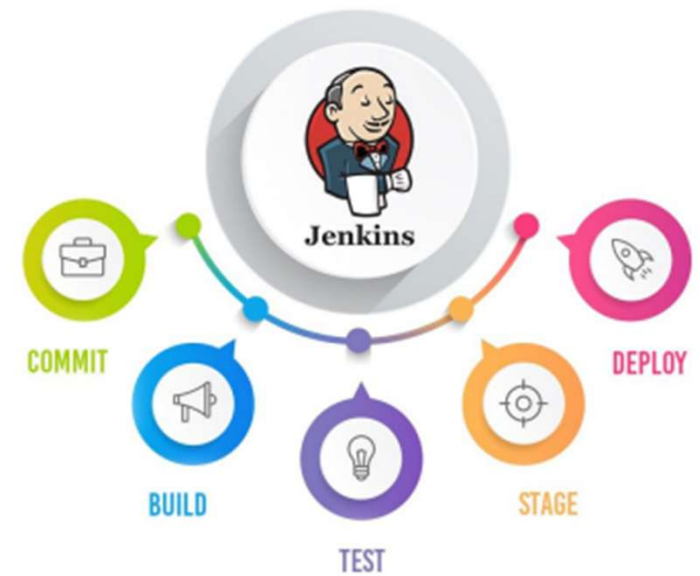




Jenkins

What is Jenkins

- Jenkins is a self-contained, open source automation server.
- Automates all sorts of tasks related to building, testing, and delivering or deploying software.





Job Types-Freestyle Project

A simple GUI-based method for creating projects.
Configuration is done through a user-friendly interface.

Use Cases:

Ideal for small and straightforward projects.
Provides quick setup for basic build and deployment processes.

Features:

Step-by-step configuration (build) processes can be added.
Easy integration with version control systems.
Can include steps for testing, building, and deploying.

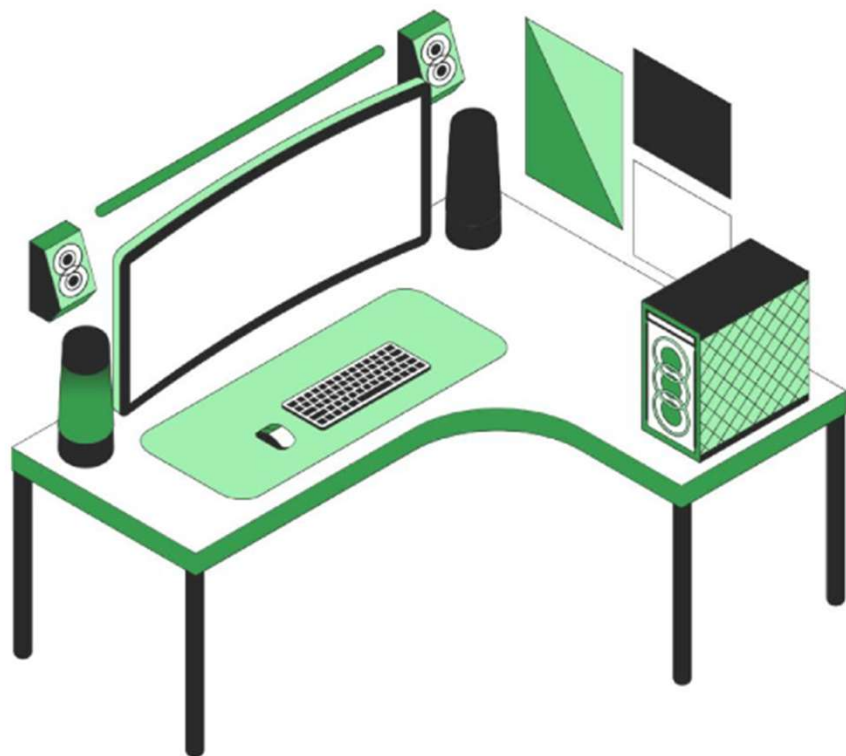
Job Types-Pipeline Project

```
• pipeline {  
•   agent any  
•  
•   stages {  
•     stage('Build') {  
•       steps {  
•         sh 'mvn clean package'  
•       }  
•     }  
•     stage('Test') {  
•       steps {  
•         sh 'mvn test'  
•       }  
•       post {  
•         always {  
•           junit 'target/surefire-reports/*.xml'  
•         }  
•       }  
•     }  
•     stage('Deploy') {  
•       when {  
•         branch 'main'  
•       }  
•       steps {  
•         sh 'kubectl apply -f k8s/deployment.yaml'  
•       }  
•     }  
•   }  
•  
•   post {  
•     success {  
•       slackSend channel: '#ci', message: 'Build başarılı!'  
•     }  
•     failure {  
•       slackSend channel: '#ci', message: 'Build başarısız!'  
•     }  
•   }  
• }
```

A project type that offers a more complex and flexible configuration, typically defined in a file called Jenkinsfile.

Use Cases: Ideal for large and complex projects. Suitable for managing continuous integration and continuous deployment (CI/CD) processes.

Features: Allows defining step-by-step processes as code, which can be version controlled. Offers more customization and flexibility. Can manage parallel operations and more complex workflows.



Do you have any questions?

Send it to us! We hope you learned something new.