CI/CD

What is CI/CD

Continuous integration (CI) and continuous delivery (CD), also known as CI/CD, embodies a culture, operating principles, and a set of practices that application development teams use to deliver code changes more frequently and reliably.

Continuous Integration

The practice of merging all developers' working copies to a shared mainline several times a day. It's the process of "**Making**". Everything related to the code fits here, and it all culminates in the ultimate goal of CI: a high quality, deployable artifact! Some common CI-related phases might include:

- Compile
- Unit Test
- Static Analysis
- Dependency vulnerability testing
- Store artifact

Continuous Deployment

A software engineering approach in which the value is delivered frequently through automated deployments. Everything related to deploying the artifact fits here. It's the process of "**Moving**" the artifact from the shelf to the spotlight. Some common CD-related phases might include:

- Creating infrastructure
- Provisioning servers
- Copying files
- Promoting to production
- Smoke Testing (aka Verify)
- Rollbacks

Benefits of CI and CD

- It helps reduce cost by catching compile errors on time
- It increases revenues by reducing the time to market by using automated deployments
- It helps return production to the last working state using automated rollbacks to protect revenue
- Automated deployments help release new features to the market faster
- Security vulnerabilities are detected on time which prevents embarrassing and costly security holes