



**CI/CD**

**Continuous Integration Continuous  
Deployment**



# Definition

- In very simple terms, CI is a modern software development practice in which incremental code changes are made frequently and reliably.
- Automated build-and-test steps triggered by CI ensure that code changes being merged into the repository are reliable.
- The code is then delivered quickly and seamlessly as a part of the CD process. In the software world, the CI/CD pipeline refers to the automation that enables incremental code changes from developers' desktops to be delivered quickly and reliably to production.



# Why is CI/CD important?

CI/CD allows organizations to ship software quickly and efficiently. CI/CD facilitates an effective process for getting products to market faster than ever before, continuously delivering code into production, and ensuring an ongoing flow of new features and bug fixes via the most efficient delivery method.





# What is the Benefits of CI and CD?

CI is practice that involves developers making small changes and checks to their code. Due to the scale of requirements and the number of steps involved, this process is automated to ensure that teams can build, test, and package their applications in a reliable and repeatable way.

CD is the automated delivery of completed code to environments like testing and development. CD provides an automated and consistent way for code to be delivered to these environments.



# What is the Benefits of CI and CD?

Continuous Deployment is the next step of continuous delivery. Every change that passes the automated tests is automatically placed in production, resulting in many production deployments.

Continuous deployment should be the goal of most companies that are not constrained by regulatory or other requirements.