



## CI/CD

어플리케이션 개발단계부터 배포까지 이 모든 단계들을 자동화를 통해서 조금 더 효율적이고 빠르게 사용자에게 빈번히 배포할 수 있도록 만드는 것

### CI:Continuous Integration 지속적인 통합

- 코드 변경사항을 주기적으로 빈번하게 머지해야 한다.

최대한 작은 단위로 머지해야된다.

- 통합을 위한 단계(빌드,테스트,머지)의 자동화

개발 생산성 향상

문제점을 빠르게 발견

버그 수정 용이

코드의 퀄리티 향상

빌드 + 테스트 : 둘 중 하나가 오류가 뜬다면 오류 구문이 나옴.

두 과정이 통과가 되면 머지가 됨

### CD:Continuous Delivery/Deployment 지속적인 전달, 지속적인 배포

CI가 성공되고 prepare Release가 완성이 된다면 실제 서버에 적용하기 전에 이 서비스가 잘 되고 있는지 확인을 한다.

사람이 수동적으로 확인을 하는 경우는 Continuous Delivery

기계가 자동적으로 확인을 하는 경우는 Continuous Deployment

CI와 CD를 거쳐서 배포를 한다.

code → build → test (CI) → release → deploy (CD)

CI,CD를 이용할 때 쓰는 툴

Jenkins, Buildkite, GitLab CI/CD, GitHub Actions, Bitbucket Pipelines, circleci