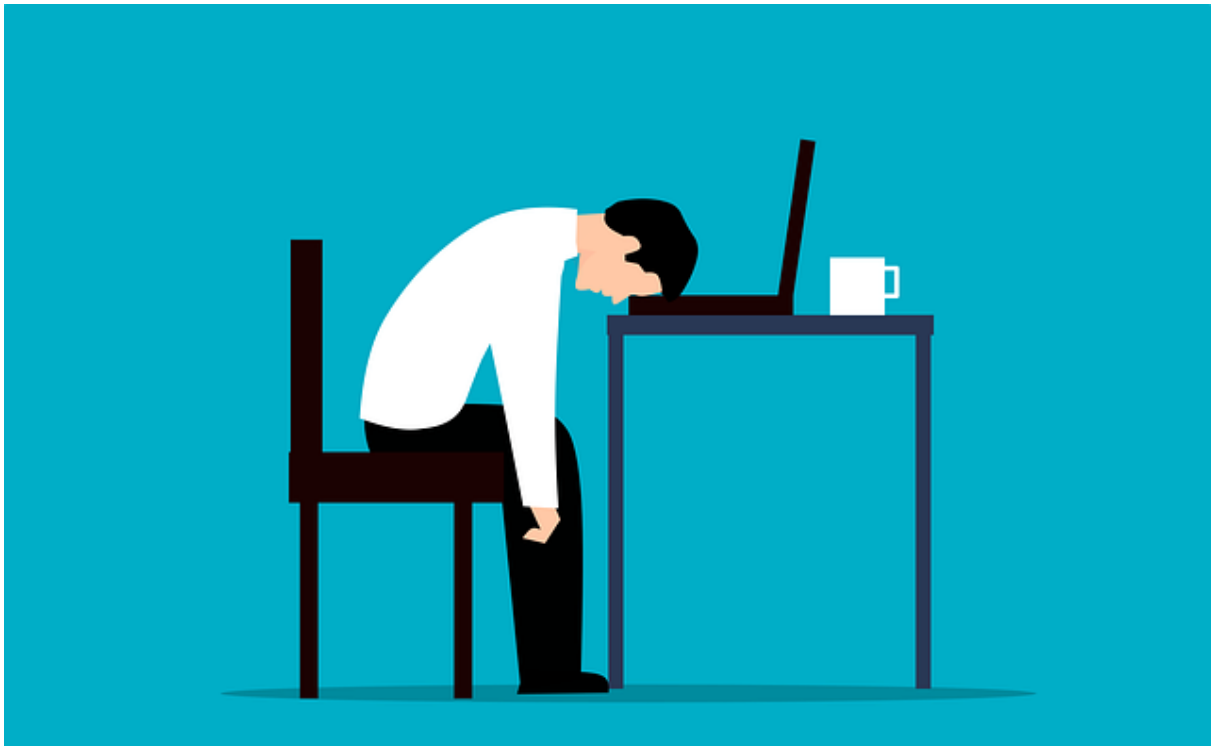




# CI/CD : 도커(Docker)를 사용하는 지속적 통합 / 지속적 배포 해보기 - 왜 도입해야할까?

## 도입

혹시 개발 도중에 이런 경험을 하신 적이 있으십니까?



1. 서로 **코드 통합에 대한 책임**(빌드 실패, 기능 동작 실패)을 떠넘기거나,
2. 주기적으로 또는 일시적으로 **배포 작업**을 한사람에게 떠맡기거나,
3. 서버에 **각종 패키지를 계속해서 추가**하여 파악하기 힘들 정도로 난장판이 되거나,
4. 정적 리소스나 소스코드 빌드 시에만 필요한 **라이브러리, 모듈로 인한 더미 데이터가 엄청나게 축적**되거나,
5. 추가되는 서비스에 의해 **폴더 구조 또는 네트워크 포트가 꼬이**거나...

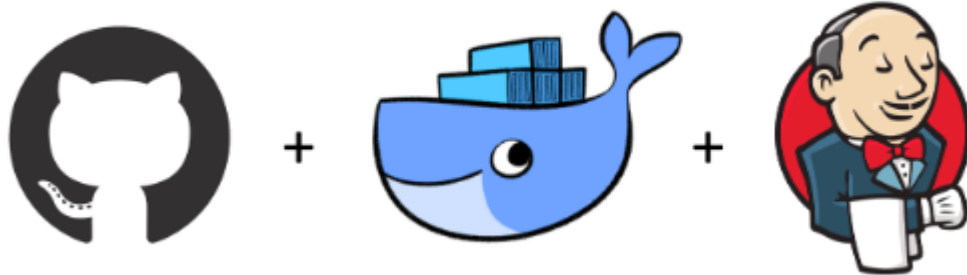
등등 더 많은 일을 빌드와 배포 시에 겪으시면서 팀원의 먹살을 잡아보시고 싶으셨을 겁니다...

(아직 겪어보시지 않으셨다면, 미리 해당 글을 읽으셔서 많은 도움을 받으셨으면 좋겠습니다.)

---

## 🤔 그래서 무엇을 해야하죠?

위와 같은 문제에 빠지지 않기 위해 빌드와 배포 과정에 대해 지속적이며, 자동적으로 진행할 수 있는 서비스를 도입해주는 것이 좋은 해결 방안이 될 수 있습니다.

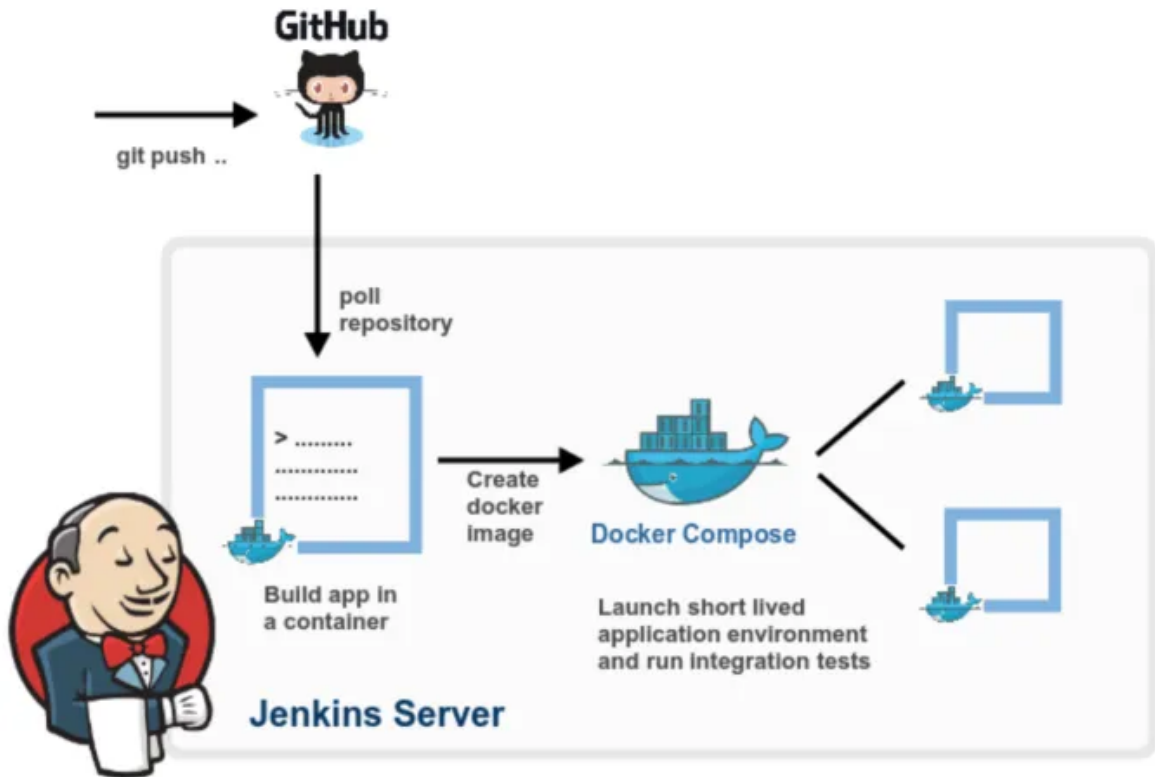


이미지 출처 : [blogs oracle | cloud-infrastructure | Build a Continuous Integration pipeline using GitHub, Docker and Jenkins on Oracle Cloud Infrastructure](#)

**젠킨스(Jenkins)**를 사용하여 빌드와 배포 과정을 파이프라인으로 만들고, **도커(Docker)**를 사용하여

각 목적에 따라 이미지를 받아, 구분한 인스턴스 컨테이너를 실행하여 작업을 마무리하는 것입니다.

(위의 개념을 처음 접해보았다면, 하단을 더 참고해주세요.)



이미지 출처 : linkedin | Malepati Yogesh Kumar | publish: 2020년 5월 7일 | GIT-JENKINS-DOCKER INTEGRATION

- 진행 순서

1. 작업한 내역을 커밋 후, **git push**하여 github에 갱신한다.
2. github에 걸어둔 이벤트 트리거로 **webhook**을 통해 **jenkins 서버에 알려준다**.
3. jenkins는 설정해둔 **스케줄 작업 파이프라인을 실행**하며, 빌드와 배포 과정을 진행한다.
  - 각 과정을 stage라고 구분지어, 일련의 쉘 명령어를 통해 실행하는 것과 유사하며, git프로젝트에 **Jenkinsfile** 파일을 작성하여 로드하는 것으로 대체할 수 있다.
  - 서버 빌드와 배포(실행)은 명령어를 통해 Docker 컨테이너를 생성하거나 인스턴스를 실행하여 배포하게 된다.

결국 상위의 **Jenkins + Docker**(+확장성을 고려하여, 다중의 인스턴스, 서버를 활용하는 회사에서는 쿠버네티스를 추가적으로 활용한다)를 사용하여 지속적으로 통합 및 배포를 유지할 수 있다. 이를 CI/CD라는 조금 더 포괄적인 용어를 사용한다.



하단은 CI/CD에 대한 설명인데, 시스템 설계를 그릴 때에와 배포 구조 설계에서 사내 PT를 할 때에 "~를 사용하여 CI/CD를 구축하였습니다" 라는 용어로 설명할 때에가 있는데 어떤 의미인지 알아두길 권한다.



## 지속적 통합, 지속적 배포 : CI/CD (Continuous Integration / Continuous Delivery & Continuous Deployment)

- 지속적 통합과 지속적 배포 또는 지속적 배포를 결합한 방식

→ 애플리케이션 개발 단계를  
자동화하는 것으로 보다 짧은 주기로

고객에게 서비스를 제공하는 방법

- 💡 이점은?

→ **인테그레이션 헬** (Integration Hell = 통합 지옥)을 사전에 해결할 수 있다.



\*인테그레이션 헬 : 프로덕션 시점에서 구성원들의 개별 코드를 통합하는 과정에 일어나는 매끄럽지 않게 진행되는 일련의 프로세스들을 의미합니다. 이런 통합 과정에서 일어나는 문제는 몇 시간 또는 며칠이 소요될 수 있습니다.

CI (지속적 통합)은 팀 구성원이 매시간 또는 최소 매일 통합 할 수 있도록 활성화, 장려하므로 **인테그레이션 헬**이 일어날 상황을 방지할 수 있습니다.

## 참고자료

- 위키백과 CI/CD | <https://en.wikipedia.org/wiki/CI/CD>

- CI/CD란 무엇인가 (Feat. DevOps 엔지니어) | 개발자 김모씨의 성장 일기 | <https://artist-developer.tistory.com/24>
- integration-hell 용어 설명 | solutionsiq | <https://www.solutionsiq.com/agile-glossary/integration-hell/>
- Red Hat, CI/CD(지속적 통합/지속적 제공): 개념, 방법, 장점, 구현 과정 | <https://www.redhat.com/ko/topics/devops/what-is-ci-cd>
- How to Build a Docker image using Jenkins | foxutech | <https://foxutech.com/how-to-build-a-docker-image-using-jenkins/>