

| Docker

#docker

| Docker 란?

- 애플리케이션을 신속하게 구축, 테스트 및 배포할 수 있는 소프트웨어 플랫폼
- **컨테이너**라는 표준화된 유닛으로 패키징
 - 컨테이너에는 라이브러리, 시스템 도구, 코드, 런타임 등 소프트웨어를 실행하는 데 필요한 모든 것이 포함되어있음

| Docker의 작동 방식

- Docker는 컨테이너를 위한 운영 체제로 컨테이너는 서버 운영 체제를 가상화함
- Docker는 각 서버에 설치되며 컨테이너를 구축, 시작 또는 중단하는 데 사용할 수 있는 간단한 명령을 제공

| Docker의 장점

- 코드를 더 빨리 전달 가능
- 애플리케이션 운영을 표준화함
 - 작은 컨테이너식 애플리케이션을 사용하면 손쉽게 배포하고 문제를 파악하고 수정을 위해 롤백할 수 있음
- 코드를 원활하게 이동
- 리소스 사용률을 높여 비용 절감 가능
- 어디서나 안정적으로 실행할 수 있는 단일 객체 확보
- 간단한 구문을 사용해 완벽하게 제어 가능

| Docker의 단점

-

| Docker Image란?

- 컨테이너를 정의하는 읽기 전용 템플릿/ 컨테이너로 실행될 소프트웨어 모음
- 코드에 필요한 라이브러리 및 종속성에 대한 정의를 비롯해 실행되는 코드가 포함되어 있음
- Docker 컨테이너는 인스턴스화된 Docker 이미지

