

# Docker

# Docker

LXC(리눅스 컨테이너스)라는 커널 컨테이너 기술을 이용하여 만든 컨테이너 기술 중 하나. 물론 지금은 LXC에 전적으로 의존하고 있지는 않으나, LXC 옵션 등을 사용할 수 있는 것을 감안한다면 아예 분리된 별개의 기술이라고 할 수는 없다.

2015년 이래 리눅스 컨테이너 기술 부분에서 사실상 업계 표준이 되어 가고 있다.

## <특징>

- **구성 단순화** Docker는 하나의 Configuration으로 모든 플랫폼에서 실행할 수 있습니다. Configuration 파일을 코드에 넣고 환경 변수를 전달하여 다른 환경에 맞추어 사용할 수 있습니다. 따라서 하나의 Docker 이미지를 다른 환경에서 사용할 수 있습니다.
- **코드 관리** Docker는 일관된 환경을 제공하여 개발 및 코딩을 훨씬 편안하게 만들어줍니다. Docker 이미지는 변경이 불가하기에 개발환경에서 운영 환경까지 애플리케이션 환경이 변경되지 않는 이점이 존재합니다.
- **개발 생산성 향상** 개발 환경을 운영 환경에 최대한 가깝게 복제할 수 있습니다. Docker를 사용하면 코드가 운영 환경의 컨테이너에서 실행될 수 있으며 VM과 달리 Docker는 오버 헤드 메모리 용량이 적기에 여러 서비스를 실행하는데 도움이 됩니다. 또한 Docker의 Shared Volume을 사용하여 호스트에서 컨테이너의 어플리케이션 코드를 사용할 수 있도록 할 수 있습니다. 이를 통해 개발자는 자신의 플랫폼 및 편집기에서 소스 코드를 편집할 수 있으며 이는 Docker내에서 실행 중인 환경에 반영됩니다.
- **애플리케이션 격리** Web Server(e.g. Apache, Nginx)와 연결된 API 서버를 격리할 필요가 있는 경우가 있습니다. 이 경우 다른 컨테이너에서 API를 서버를 실행할 수 있습니다.
- **빠른 배포** 컨테이너가 OS를 부팅하지 않고 어플리케이션을 실행하기 때문에 Docker 컨테이너를 매우 빠르게 만들 수 있습니다.