1. 컨테이너

|  |
| --- |
| 컨테이너 학습 이유    소프트웨어 운영 플랫폼이 이렇게 바뀌고 있음  리눅스에서 컨테이너를 돌리는 이유  독립된 공간 형성  isolate 기능 지원  필요한만큼 HW 지원  즉, 리눅스 커널 기능을 사용하기 위해  그냥 프로그램과 컨테이너의 차이점  하는 일은 같지만 구조가 다르다.      도커를 쓰는 이유?  개발자가 만든 그대로 어디서든 돌릴 수 있다.  확장/축소가 쉽고 MSA, DevOps에 적합하다. |

2. 도커 컨테이너 설치

|  |
| --- |
| 컴퓨터(BareMetal 또는 Virtualmachine)  cpu: 2core, memory: 2GB이상  운영체제 (리눅스 또는 Windows)  도커 설치  도커 서비스 실행 |

3. virtualbox 설치 및 Network 구성

|  |
| --- |
|  |

3. ubuntu 20.04 설치

|  |
| --- |
|  |

4.centos7 설치

|  |
| --- |
|  |

5. docker설치

|  |
| --- |
| 계정 추가  루트계정 접속  $usermod -a -G docker guru |

6. 컨테이너

|  |
| --- |
| 컨테이너? 컨테이너 이미지?  컨테이너는 하나의 Application 프로세스    3개의 레이어를 가지고 만들어진 컨테이너 이다.  컨테이너 이미지는 여러 개의 레이어로 구성되어 하나의 Application이 잘 실행될 수 있도록 모아져 있는 이미지들의 조합이 하나의 컨테이너 이미지가 된다.    컨테이너 동작 방식 |
| 실습  1. Docker Hub에서 컨테이너 이미지 검색  2. 컨테이너 이미지 다운로드 후 image layer 보기  3. 컨테이너 실행하고 확인해 보기  $docker search nginx  $docker pull nginx:latest  $docker run --name web -d -p 80:80 nginx  $docker ps  $docker stop web  $docker rm web  $docker rmi nginx |