

# AWS 컨테이너 서비스로 나만의 AI 기반 서비스 배포하기



# 오늘은..

## 컨테이너 먼데이거

컨테이너란 무엇인지에 대해서 간단하게 알아봅니다.

## 컨테이너 관련 AWS 서비스

AWS 서비스들 중에서, 컨테이너와 관련된 서비스에 대해 알아봅니다.

## Hands-on

오늘 배운 서비스들을 가볍게 사용해보며 친해져봅니다.

AWS 컨테이너 서비스로 나만의 AI 기반 서비스를 배포해보아요!

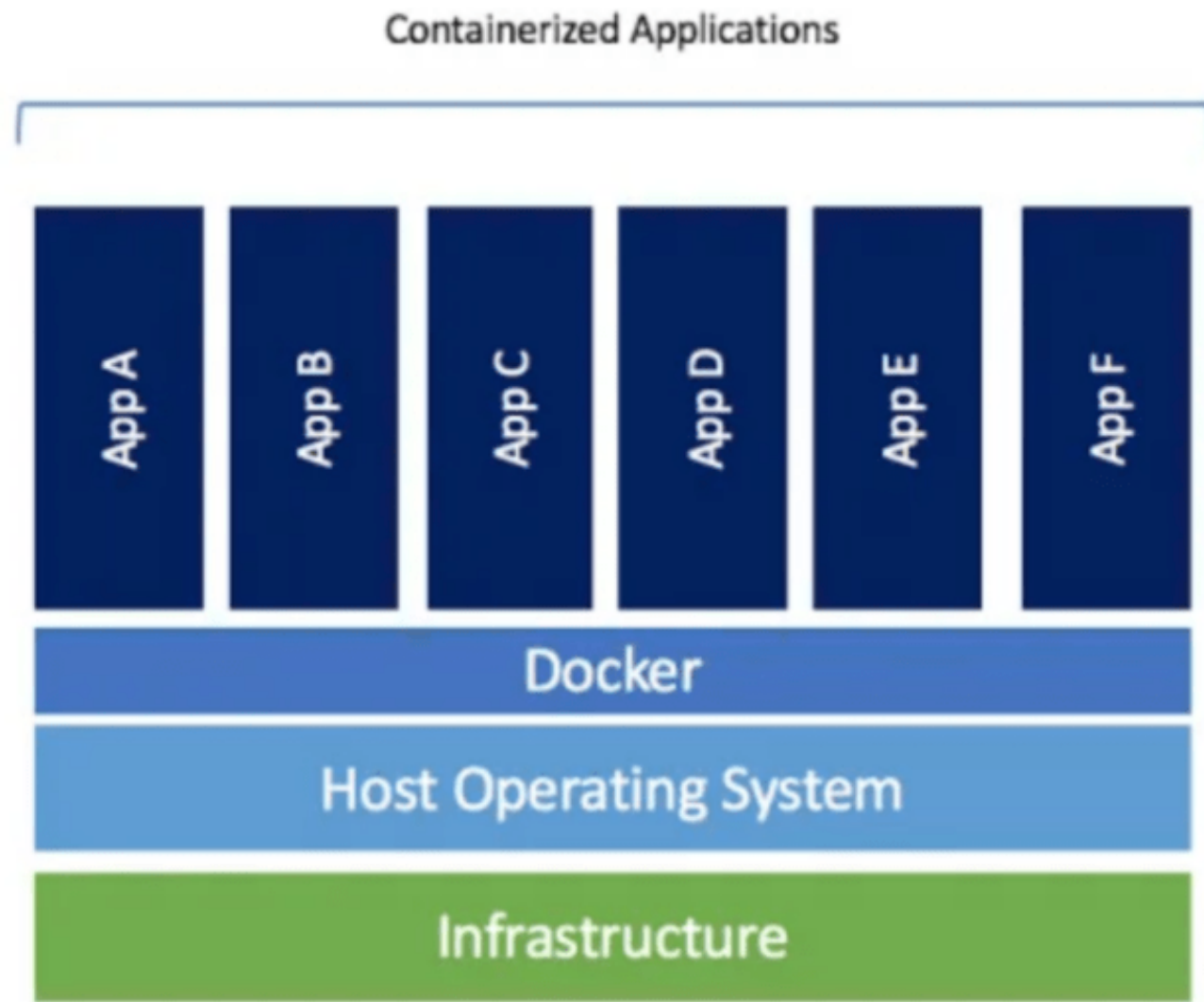


# 컨테이너란

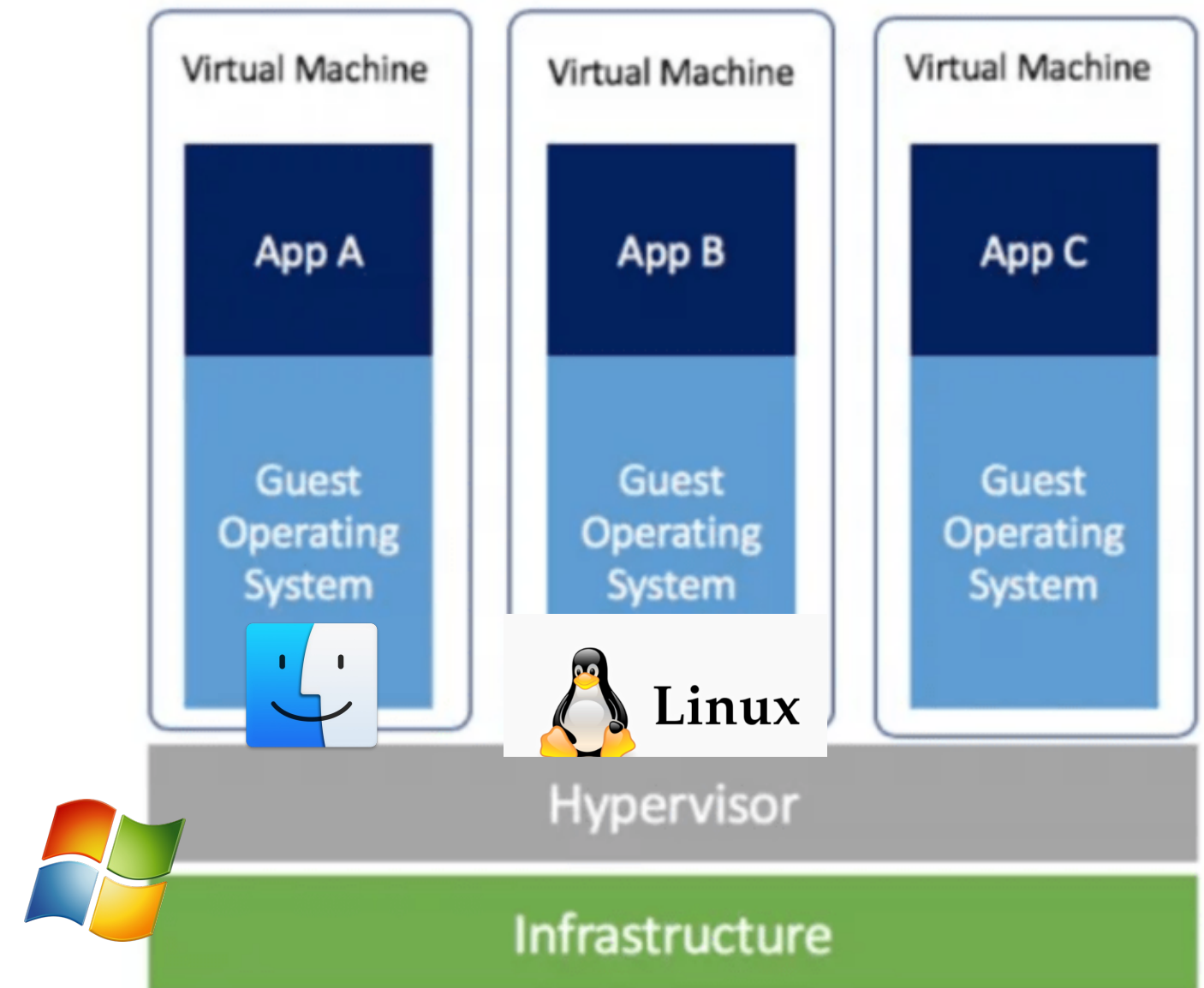
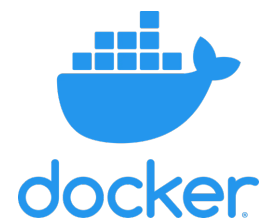
다른 프로세스와 **격리**된 상태로 OS에서 소프트웨어를 실행하는 기술



# 하이퍼바이저 vs 컨테이너



컨테이너 가상화



하이퍼바이저 가상화




# 컨테이너 특징 및 장점

1. 운영체제 수준의 가상화
2. 빠른 속도와 효율성
3. 높은 이식성 (Portability)
4. 상태를 가지지 않음 (Stateless)


- 
- 
- 



# Docker의 기본적인 흐름



```
# Dockerfile Example
FROM alpine
COPY app.sh /
CMD ["/app.sh"]
```



```
# Dockerfile을 기반으로 이미지 빌드
docker build -t username/app-name:tag .
# 빌드된 이미지를 이미지 레지스트리(Docker Hub, ECR)에 PUSH
docker push username/app-name:tag
# 이미지 레지스트리로부터 특정 컨테이너 이미지를 PULL
docker pull username/app-name:tag
# 특정 컨테이너 이미지로 컨테이너를 생성, 실행
docker run username/app-name:tag
```

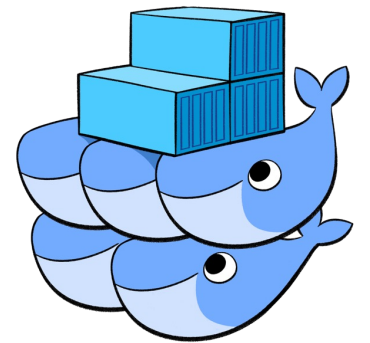


# 오케스트레이션?

컨테이너 배치 관리  
컨테이너 부하 분산  
컨테이너 상태 감시 및 자동  
복구  
컨테이너 배포



**kubernetes**



# AWS의 컨테이너 관련 서비스



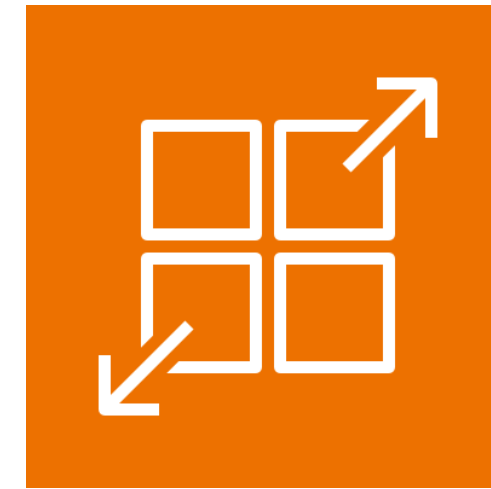
Amazon  
Elastic  
Container  
Service



Amazon  
Elastic  
Kubernetes  
Service



Amazon  
Elastic  
Container  
Registry



AWS  
App  
Runner





# Amazon Elastic Container Service

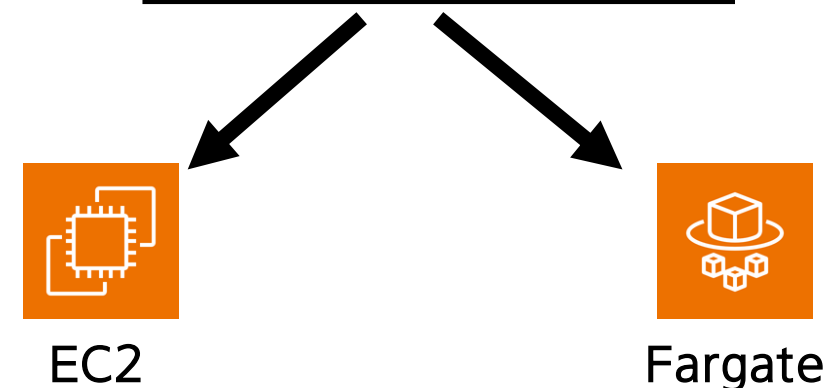


## Amazon ECS

완전 관리형 컨테이너 오케스트레이션 서비스. AWS 상에서 컨테이너 기반의 애플리케이션을 호스팅하는 가장 일반적인 방법

Amazon EKS는 완전관리형 **쿠버네티스** 서비스!

**컨테이너를 동작시키는 실행 환경 서비스가 아니다!**



# Amazon Elastic Container Service

▼ Infrastructure [Info](#)

Serverless

Your cluster is automatically configured for AWS Fargate (serverless) with two capacity providers. Add Amazon EC2 instances.

☒ AWS Fargate (serverless)

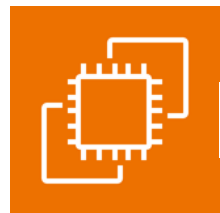
Pay as you go. Use if you have tiny, batch, or burst workloads or for zero maintenance overhead. The cluster has Fargate and Fargate Spot capacity providers by default.

☐ Amazon EC2 instances

Manual configurations. Use for large workloads with consistent resource demands.

ⓘ

External instances using **ECS Anywhere** can be registered after cluster creation is complete.



## EC2

- 직접 관리가 필요함(스케줄링 등)
- VPC 안에 생성
- 비용이 높음



## Fargate

- 서버리스!!
- VPC 밖에 생성, VPC에서 접근
- EC2에 비해서 비교적 비용 관리가 Easy~



# AWS App Runner



## AWS App Runner

경량 컨테이너 기반 애플리케이션 실행 서비스  
⇒ ECS + Fargate 의 조합을 랩핑해놓은 서비스

정말 빠르게 컨테이너 배포 가능

하지만.. **서울 리전 지원 X**



# Amazon Elastic Container Registry



## Amazon ECR

완전 관리형 컨테이너 저장소

컨테이너 이미지를 간단하게 저장, 관리할 수 있다.

ECS, EKS와 합쳐졌을 때 엄청난 시너지

AWS의 다른 서비스와의 연계 & 보안 설정

➔ Docker Hub < Amazon ECR



## (부록) 실습에 사용하는 서비스



### Amazon Bedrock

완전 관리형 생성형 AI 서비스

통합 엔드포인트로 AI21 Labs, Anthropic, Cohere, Stability AI 등의 다양한 파운데이션 모델을 사용 가능



# Hands-on

## 실습 #1

AWS Lambda, 컨테이너로 띄워보기

<https://bit.ly/container-handson-1>

## 실습 #2

AWS App Runner로 나만의 AI 기반 서비스 배포하기 (Amazon Bedrock)

<https://bit.ly/container-handson-2>

## 실습 #3

Amazon ECS로 나만의 AI 기반 서비스 배포하기 (Amazon Bedrock)

<https://bit.ly/container-handson-3>



# References

유튜브 강의 [AWS의 컨테이너 서비스를 정리합니다](#) - AWS 강의실

인프런 강의 [개발자를 위한 쉬운 도커 - 데브위키](#)

AWS Docs [What is Amazon Elastic Container Service?](#)

AWS Docs [What is Amazon Elastic Container Service?](#)

AWS Workshop [EKS Workshop](#)

