

제06장

# Docker 설치

DevOps

# 학습목표

1. 도커에 대해서 이해할 수 있다.
2. 도커를 설치할 수 있다.

# 목차

1. 도커란?
2. 도커 설치하기

```

each: function(e, t, n) {
    var r, i = 0,
        o = e.length,
        a = M(e);
    if (n) {
        if (a) {
            for (; o > i; i++)
                if (r = t.apply(e[i], n), r === !1) break;
        } else
            for (i in e)
                if (r = t.apply(e[i], n), r === !1) break;
    } else if (a) {
        for (; o > i; i++)
            if (r = t.call(e[i], e[i]), r === !1) break;
    } else
        for (i in e)
            if (r = t.call(e[i], e[i]), r === !1) break;
    return e;
},
trim: b && !b.call("\uffff\u00a0") ? function(e) {
    return null == e ? "" : b.call(e);
} : function(e) {
    return null == e ? "" : (e + "").replace(C, "");
},
makeArray: function(e, t) {
    var n = t || [];
    return null != e && (M(Object(e)) ? x.merge(n, "string"
),
inArray: function(e, t, n) {
    var r;
    if (t) {
        if (m) return m.call(t, e, n);
        for (r = t.length, r = r ? 0 > n ? Math.max(0, r + n) : n; r < t.length; r++)
            if (n in t && t[r] === e) return r;
    }
}

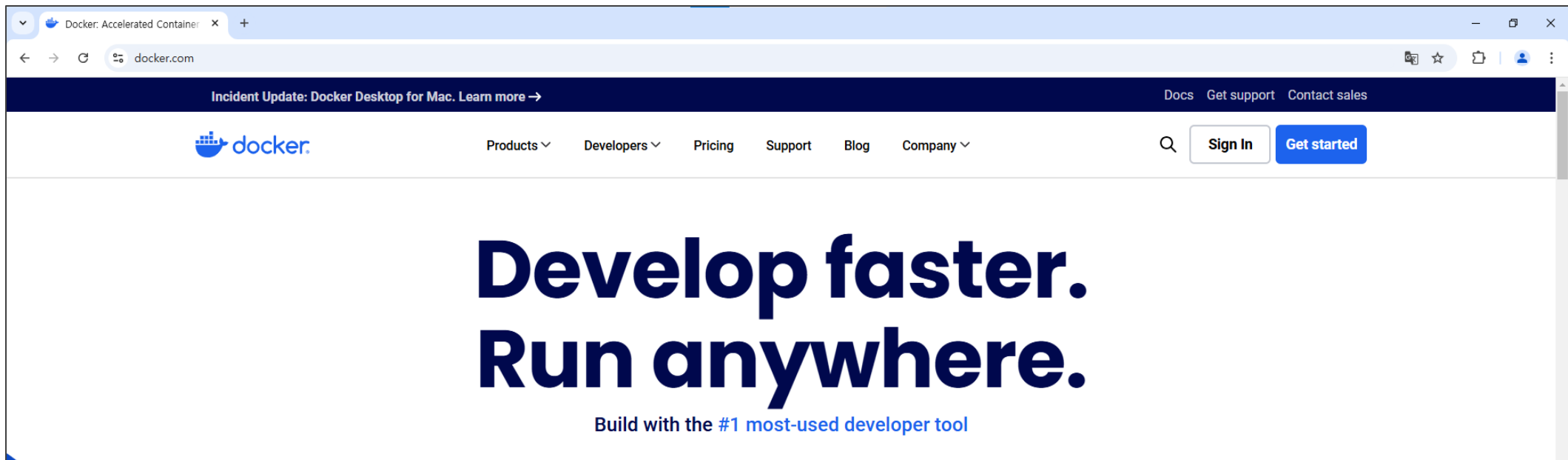
```

# 01. 도커란?

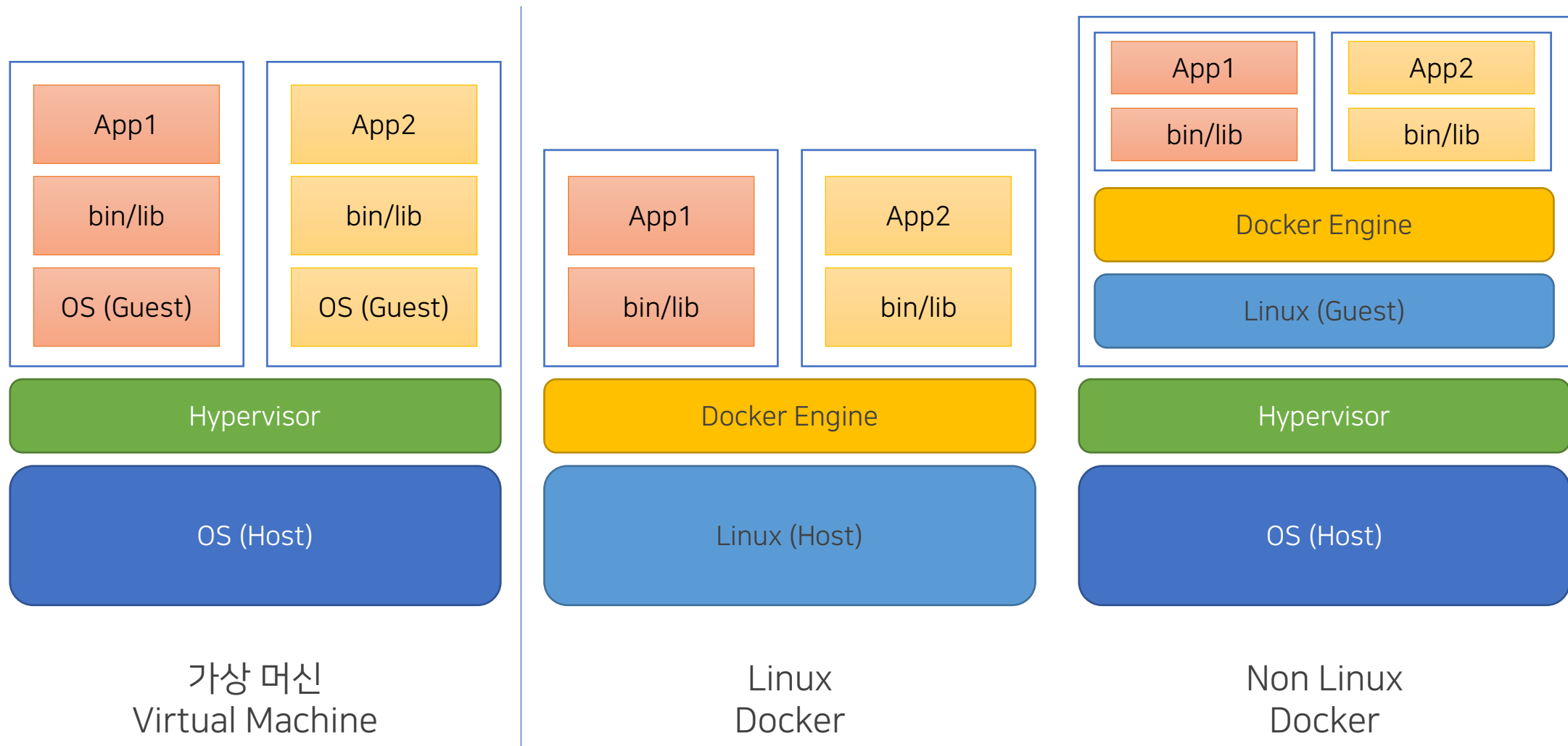
# 도커란?

## ■ Docker

- 2013년 3월 Pycon에서 The future of Linux Containers라는 이름으로 최초 소개
- 라이브러리, 시스템 도구, 코드, 런타임 등 소프트웨어를 실행하는 데 필요한 모든 것을 **가상 컨테이너**라는 표준화된 유닛으로 패키징하는 기술
- 컨테이너 자체에 운영체제도 포함되기 때문에 플랫폼에 독립적으로 사용 가능
- 도커를 이용하면 애플리케이션을 신속하게 구축, 테스트 및 배포할 수 있음



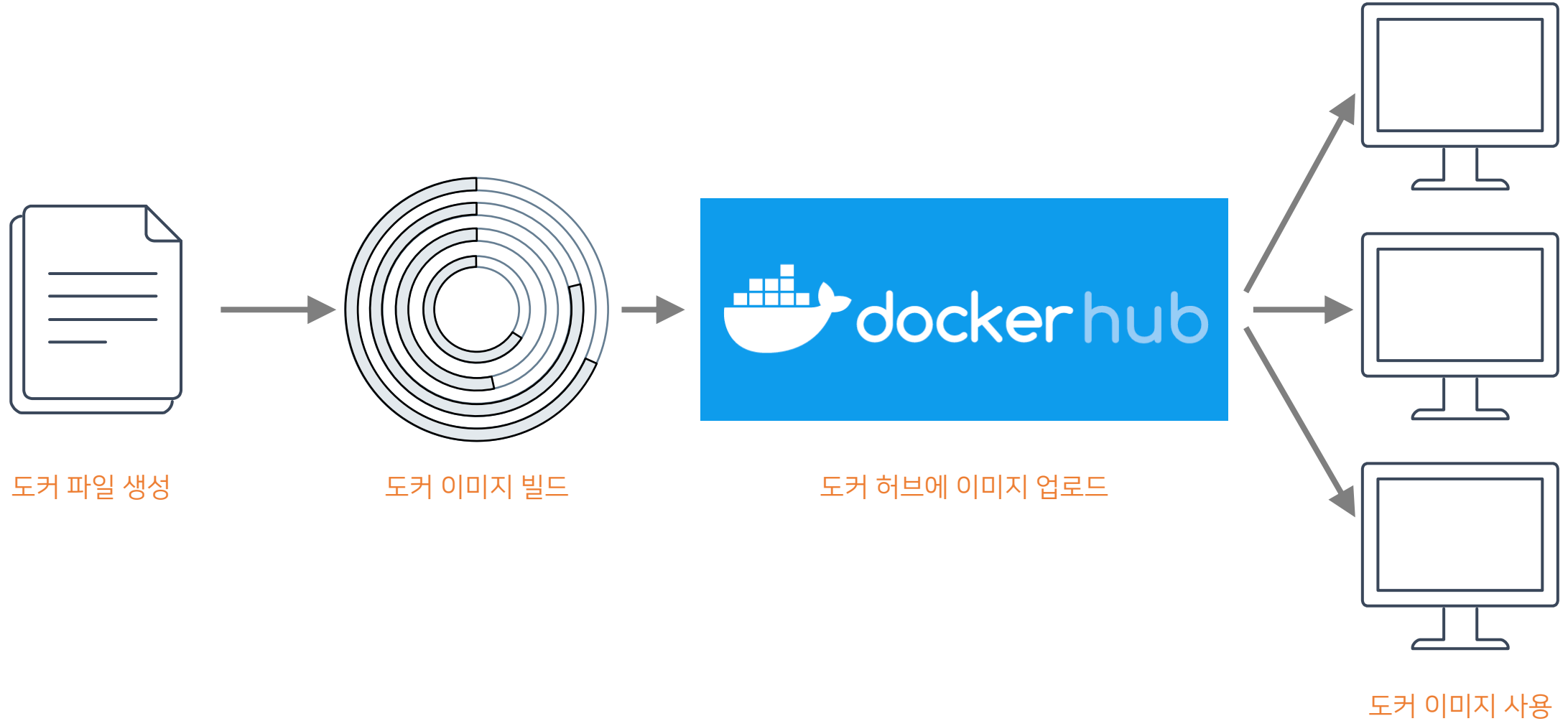
# 가상 머신과 도커 컨테이너



## ■ Docker Image

- 애플리케이션 실행에 필요한 프로그램, 소스코드, 라이브러리 등을 하나로 묶어 놓은 읽기 전용 파일
- 컨테이너를 생성하기 위한 모든 환경을 가지고 있음
- 작은 크기를 가짐 (기본적으로 수백 MB 이상의 크기를 가지지만 가상 머신에 비하면 굉장히 크기가 작은 편임)
- Dockerhub(도커 허브)를 이용해 이미지의 버전 관리 및 배포 작업을 수행함
- 하나의 이미지로 여러 개의 컨테이너를 만들 수 있음 (이미지는 컨테이너를 만들기 위한 재료)
- 일반적으로 Dockerfile을 이용해서 도커 이미지를 생성할 수 있음
- 도커 이미지 이름 형식
  - 레지스트리/계정명/이미지명:태그

# 도커 파일과 도커 이미지

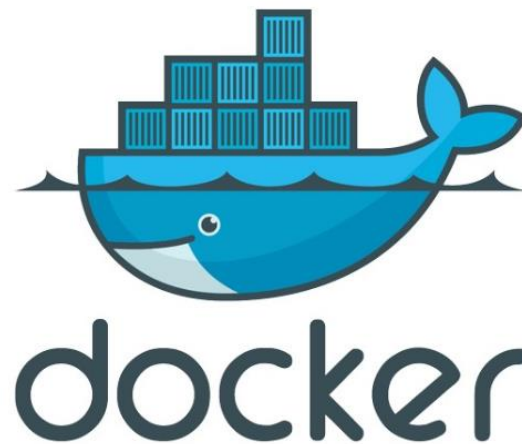




# 도커 컨테이너

## ■ Docker Container

- 호스트의 환경을 사용하지 않고 애플리케이션을 실행하기 위한 모든 필수 구성 요소를 자체적으로 포함하는 격리된 환경
- 도커 이미지를 실행하면 도커 컨테이너가 생성되면서 실행됨
- 도커 컨테이너 실행은 곧 애플리케이션 실행을 의미함
- 도커 컨테이너 내부에서 생성한 파일은 컨테이너가 사라지면 모두 함께 사라짐
- 도커 컨테이너의 라이프 사이클
  - 생성 → 실행 → 중지 → 삭제



컨테이너를 싣고 가는 고래로 형상화한 도커 로고 이미지

```

each: function(e, t, n) {
    var r, i = 0,
        o = e.length,
        a = M(e);
    if (n) {
        if (a) {
            for (; o > i; i++)
                if (r = t.apply(e[i], n), r === !1) break;
        } else
            for (i in e)
                if (r = t.apply(e[i], n), r === !1) break;
    } else if (a) {
        for (; o > i; i++)
            if (r = t.call(e[i], e[i]), r === !1) break;
    } else
        for (i in e)
            if (r = t.call(e[i], e[i]), r === !1) break;
    return e;
},
trim: b && !b.call("\uffff\u00a0") ? function(e) {
    return null == e ? "" : b.call(e);
} : function(e) {
    return null == e ? "" : (e + "").replace(C, "");
},
makeArray: function(e, t) {
    var n = t || [];
    return null != e && (M(Object(e)) ? x.merge(n, "string"
),
isArray: function(e, t, n) {
    var r;
    if (t) {
        if (n) return m.call(t, e, n);
        for (r = t.length, r = n ? 0 > n ? Math.max(0, r + n
            if (n in t && t[n] === e) return n;
    }
}

```

## 02. 도커 설치하기

# 도커 설치를 위한 준비

## ■ Linux 운영체제

- 커맨드를 이용한 설치
- 패키지 관리도구 업데이트(apt, yum 등) 필요
  - apt update, yum check-update 등

## ■ Windows 운영체제

- Docker Desktop을 이용한 도커 설치
- WSL2(Window Sub Linux) 또는 Hyper-V 기반 설치
- 설치 참고 링크
  - <https://docs.docker.com/desktop/setup/install/windows-install/>

# Ubuntu 운영체제 도커 설치

## 1. `$ sudo apt update`

- 패키지 관리도구 업데이트

## 2. `$ sudo apt install -y curl`

- 도커 설치 스크립트를 다운로드 받을 curl 패키지 설치

## 3. `$ curl -s https://get.docker.com | sudo sh`

- 도커 설치 스크립트를 GET 방식으로 다운로드 받은 뒤 sh 실행

## 4. `$ docker -v`

- 설치된 도커 버전 확인

# sudo 명령어 없이 docker 명령 내리기

1. `$ sudo usermod -aG docker $USER`

- 현재 사용자를 docker 그룹에 추가함

2. `$ newgrp docker`

- 새로운 docker 그룹 시작

3. `$ docker info | grep -i 'docker root dir'`

- sudo 없이 docker 명령어가 동작하는지 확인 (도커의 루트 디렉터리 확인해 보기)