

# Dockek 05

## [ Docker Compose ]

따라하며 배우는 도커

John Ahn

2023.02.20

---

## Docker Compose란?

docker compose 는 다중 컨테이너 도커 어플리케이션을 정의하고 실행하기 위한 도구

간단한 어플리케이션을 만들어서 하나로

레디스를 사용해서 간단한 페이지 만들기

**Redis (REmote Dictionary Server)**는 메모리 기반의 키-값 구조 데이터 관리 시스템이며, 모든 데이터를 메모리에 저장하고 빠르게 조회할 수 있는 비 관계형 데이터베이스(NoSql)이다.

### Why Redis?

메모리에 저장하기 때문에 빠르고 영속적으로도 관리가 가능하다.

레디스 서버 + node.js 앱을 사용하게끔 두개의 컨테이너를 띄우는 어플리케이션을 만들것.

```
const express = require("express");
const redis = require("redis");

//레디스 클라이언트
const client = redis.createClient({
  //도커환경이 아니라면 https://redis-server.com 처럼 자신의 url을 레디스
  호스트로 주면된다.
  //도커환경이라면 redis-server (도커 컨테이너 이름) 으로 넣어주면 된다. - 이는
  yml 파일에서 정의해줄것.
  host:"redis-server",
  port: 6379
});

//숫자는 0 부터 시작합니다.
```

```

client.set("number", 0);

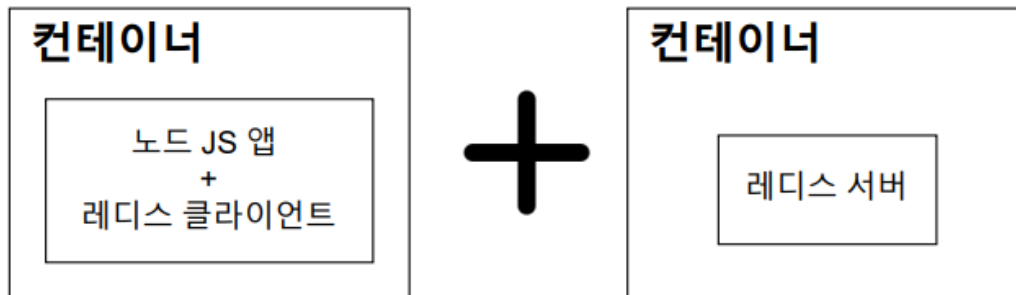
app.get('/', (req, res) => {
  client("number", (err, number) => {
    //현재 숫자를 가져온 후에 1씩 올려줍니다.
    client.set("number", parseInt(number) + 1);
    res.send("숫자가 1씩 올라갑니다. 숫자 : " + number);
  })
})

const app = express();
app.listen(9090);
console.log('Server is running');

```

→ server.js 부분

간단한 redis를 이용한 / 루트 페이지 호출시 number + 1시켜주고 보여주는 페이지



의 형태로 실행 시켜줄 것. 레디스 서버를 실행시키고 → 왼쪽의 앱을 사용한다.

```

D:\StudyResouce\dockerResource\docker_example\docker-compose-app>docker run redis
1:C 20 Feb 2023 11:57:11.599 # o000o000o000o Redis is starting o000o000o000o
1:C 20 Feb 2023 11:57:11.599 # Redis version=7.0.7, bits=64, commit=00000000, modified=0, pid=1, just started
1:C 20 Feb 2023 11:57:11.599 # Warning: no config file specified, using the default config. In order to specify a config
file use redis-server /path/to/redis.conf
1:M 20 Feb 2023 11:57:11.599 * monotonic clock: POSIX clock_gettime
1:M 20 Feb 2023 11:57:11.601 * Running mode=standalone, port=6379.
1:M 20 Feb 2023 11:57:11.602 # Server initialized
1:M 20 Feb 2023 11:57:11.602 # WARNING Memory overcommit must be enabled! Without it, a background save or replication m
ay fail under low memory condition. Being disabled, it can also cause failures without low memory condition, see htt
ps://github.com/jemalloc/jemalloc/issues/1328. To fix this issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and
then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to take effect.
1:M 20 Feb 2023 11:57:11.603 * Ready to accept connections

```

>docker run redis

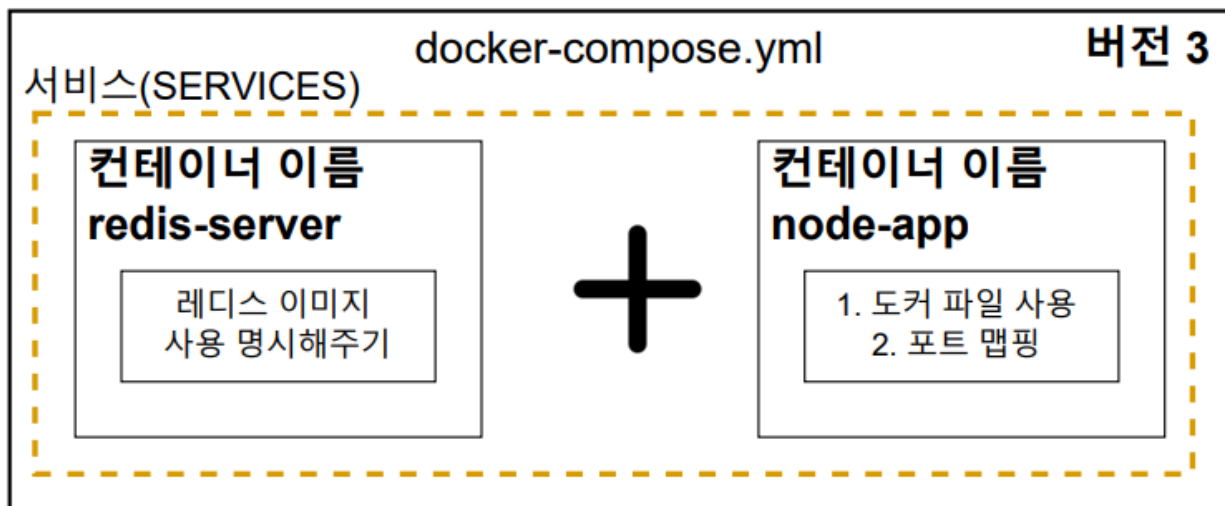
레디스 실행

## Docker Compose 파일은 확장자가 yml이나 yaml인데 이것은 무엇인가요?

YAML ain't markup language의 약자이며,  
일반적으로 구성 파일 및 데이터가 저장되거나 전송되는 응용 프로그램에서 사용되고

원래는 XML이나 json 포맷으로 많이 쓰였지만,  
좀 더 사람이 읽기 쉬운 포맷으로 나타난 게 yaml입니다.

xml , json 보다 읽기 쉽게 yaml 로 만든것 !



컨테이너들을 감싼 서비스를 정의해주는 설정파일 = docker-compose.yml 이라고 이해하자

```

docker-compose-app > docker-compose.yml
1  version: '3'
2  services:
3  #서비스에 있는 컨테이너들 redis-server , node-app
4      redis-server:
5          image: "redis"
6      node-app:
7          build:
8              context: .
9              dockerfile: DockerFile
10         ports:
11             - "5000:9090"

```

docker-compose up으로 실행 하면된다.

## Docker compose 중단시키기

시작 - docker-compose up

중지 - docker-compose down

```

D:\StudyResource\dockerResource\docker_example\docker-compose-app>docker-compose down
[+] Running 3/3
- Container docker-compose-app-node-app-1      Removed
- Container docker-compose-app-redis-server-1  Removed
- Network docker-compose-app_default           Removed

```

.yml 파일이있는 위치에서 실행해 줘야한다.

아니면 옵션을

docker-compose -d up 으로 실행 하는 방법이 있다 output이 표시되지않음. [ 백그라운드 실행]

docker-compose up --build <=> 무조건 재빌드 하는법

---