

# 클러스터 구성 실습

## 1 클러스터 설정 파일 이해하기

## 클러스터 구성 실습

# 1.

## 클러스터 설정 파일 이해하기

### 클러스터 설정 파일

- `cluster-enabled <yes/no>`: 클러스터 모드로 실행할지 여부를 결정
- `cluster-config-file <filename>`: 해당 노드의 클러스터를 유지하기 위한 설정을 저장하는 파일로, 사용자가 수정하지 않음.
- `cluster-node-timeout <milliseconds>`
  - 특정 노드가 정상인 것으로 판단하는 기준 시간
  - 이 시간동안 감지되지 않는 **master**는 **replica**에 의해 **failover**가 이루어짐

## 클러스터 구성 실습

### 1.

클러스터 설정  
파일 이해하기

### 클러스터 설정 파일

- `cluster-replica-validity-factor <factor>`
  - `master`와 통신한지 오래된 `replica`가 `failover`를 수행하지 않게 하기 위한 설정
  - $(\text{cluster-node-timeout} * \text{factor})$ 만큼 `master`와 통신이 없었던 `replica`는 `failover` 대상에서 제외된다.
- `cluster-migration-barrier <count>`:
  - 한 `master`가 유지해야 하는 최소 `replica`의 개수
  - 이 개수를 충족하는 선에서 일부 `replica`는 `replica`를 가지지 않은 `master`의 `replica`로 `migrate`될 수 있다.

## 클러스터 구성 실습

### 1.

클러스터 설정  
파일 이해하기

### 클러스터 설정 파일

- `cluster-require-full-coverage <yes/no>`:
  - 일부 **hash slot**이 커버되지 않을 때 **write** 요청을 받지 않을지 여부
  - **no**로 설정하게 되면 일부 노드에 장애가 생겨 해당 **hash slot**이 정상 작동하지 않더라도 나머지 **hash slot**에 대해서는 작동하도록 할 수 있다.
- `cluster-allow-reads-when-down <yes/no>`:
  - 클러스터가 정상 상태가 아닐 때도 **read** 요청은 받도록 할지 여부
  - 어플리케이션에서 **read** 동작의 **consistency**가 중요치 않은 경우에 **yes**로 설정할 수 있다.

# 클러스터 구성 실습

## 2 클러스터 구성해보기

# 클러스터 구성 실습

## 3 Spring을 이용한 클러스터 사용