

# Elasticsearch의 기본 동작 및 플러그인(2회차)

---

# 목차

- Elasticsearch 기본 동작
- Elasticsearch 플러그인
- Q & A

# Elasticsearch 기본 동작

인덱스 생성 및 삭제, 조회

문서 색인 및 조회

문서 갱신 및 삭제

클러스터 정보 확인하기

# Elasticsearch 기본 동작 - 인덱스 생성 및 삭제, 조회

인덱스는 문서(도큐먼트)들의 모음

인덱스를 생성하는 세 가지 방법

- 1) 인덱스의 settings 를 정의
- 2) 인덱스의 mappings 를 정의
- 3) 사용자 정의된 도큐먼트를 인덱싱

# Elasticsearch 기본 동작 - 인덱스 생성 및 삭제, 조회

## 인덱스의 settings

### 1) Static index settings

- `number_of_shards` - primary 샤드 갯수 설정

### 2) Dynamic index settings

- `number_of_replicas` - replica 샤드 갯수 설정
- `refresh_interval` - 시스템 OS 버퍼 캐시에 있는 데이터를 어느 주기로 세그먼트로 내릴지 결정하는 설정
- `index.routing.allocation.enable` - 인덱스의 샤드들의 라우팅 허용 설정

### 3) other settings..

- Analysis, Mapping, Slowlog..

# Elasticsearch 기본 동작 - 인덱스 생성 및 삭제, 조회

인덱스의 settings 를 정의하여 인덱스 생성하기  
- PUT Method 를 사용

PUT twitter

```
{
  "settings" : {
    "index" : {
      "number_of_shards" : 3,
      "number_of_replicas" : 1
    }
  }
}
```

```
{
  "acknowledged": true,
  "shards_acknowledged": true,
  "index": "twitter"
}
```

		twitter		
		size: 690B (1.35ki) docs: 0 (0)		
		<a href="#">Info</a> <a href="#">Actions</a>		
●	EWVXzG_	<a href="#">Info</a>	<a href="#">Actions</a>	<div>0</div> <div>1</div> <div>2</div>
★	NXLq-qi	<a href="#">Info</a>	<a href="#">Actions</a>	<div>0</div> <div>1</div> <div>2</div>

# Elasticsearch 기본 동작 - 인덱스 생성 및 삭제, 조회

flat\_settings 형태로도 인덱스 생성 가능

PUT twitter

```
{
  "settings" : {
    "index.number_of_shards" : 3,
    "index.number_of_replicas" : 1
  }
}
```

```
{
  "acknowledged": true,
  "shards_acknowledged": true,
  "index": "twitter"
}
```

		twitter		
		size: 690B (1.35ki)		
		docs: 0 (0)		
		<a href="#">Info</a> <a href="#">Actions</a>		
●	EWVXzG_	<a href="#">Info</a>	<a href="#">Actions</a>	<div>012</div>
★	NXLq-qi	<a href="#">Info</a>	<a href="#">Actions</a>	<div>012</div>

# Elasticsearch 기본 동작 - 인덱스 생성 및 삭제, 조회

## 인덱스 삭제하기

- DELETE Method 를 사용

## DELETE twitter

- 삭제는 조심해서..
- nginx 를 앞단에 두어 DELETE Method 를 특정 IP 에서만 허용하기도..

- CLI 사용하기

`curl -XDELETE -H 'Content-Type: application/json' http://{ES_URL}:9200/{index}`

```
{  
  "acknowledged": true  
}
```



# Elasticsearch 기본 동작 - 인덱스 생성 및 삭제, 조회

HEAD twitter

- 인덱스가 존재하는지 확인

GET twitter/\_settings

- 인덱스의 세팅을 확인

GET twitter/\_mappings

GET twitter/\_mapping

- 인덱스의 매핑을 확인

200 - OK

```
{
  "twitter": {
    "settings": {
      "index": {
        "creation_date": "1539620123793",
        "number_of_shards": "3",
        "number_of_replicas": "1",
        "uuid": "woejXzXgTT-3TMGV3GBe5A",
        "version": {
          "created": "6040299"
        },
        "provided_name": "twitter"
      }
    }
  }
}
```

# Elasticsearch 기본 동작 - 인덱스 생성 및 삭제, 조회

GET twitter/\_stats

- 인덱스의 상태를 확인(사이즈, 문서수, 실행된 명령 정보들..)

```
{
  "_shards": {
    "total": 6,
    "successful": 6,
    "failed": 0
  },
  "_all": {
    "primaries": {
      "docs": {
        "count": 0,
        "deleted": 0
      },
      "store": {
        "size_in_bytes": 690
      },
      "indexing": {
        "index_total": 0,
        "index_time_in_millis": 0,
        "index_current": 0,
        "index_failed": 0,
        "delete_total": 0,
        "delete_time_in_millis": 0,
        "delete_current": 0,
        "noop_update_total": 0,
        "is_throttled": false,
        "throttle_time_in_millis": 0
      },
      "get": {
        "total": 0,
        "time_in_millis": 0,

```

# Elasticsearch 기본 동작 - 인덱스 생성 및 삭제, 조회

GET twitter/\_segments

- 인덱스의 샤드 및 세그먼트 정보들

```
{
  "_shards": {
    "total": 6,
    "successful": 6,
    "failed": 0
  },
  "indices": {
    "twitter": {
      "shards": {
        "0": [
          {
            "routing": {
              "state": "STARTED",
              "primary": false,
              "node": "NXLq-qizQNe8GX2cLqI2EQ"
            },
            "num_committed_segments": 0,
            "num_search_segments": 0,
            "segments": {}
          },
          {
            "routing": {
              "state": "STARTED",
              "primary": true,
              "node": "EWVXzG_hTwaIj1ajCvJrTw"
            },
            "num_committed_segments": 0,
            "num_search_segments": 0,
            "segments": {}
          }
        ]
      }
    }
  },
}
```

# Elasticsearch 기본 동작 - 인덱스 생성 및 삭제, 조회

GET \_cat/indices?v

GET \_cat/indices/twitter?v

- 인덱스 요약 정보

health	status	index	uuid	pri	rep	docs.count	docs.deleted	store.size	pri.store.size
green	open	data	0CzFMKMDRH-4B-MEX7lDHA	5	1	1	0	7.6kb	3.8kb
green	open	.my-kibana	e6EpJm60TlCxV69mPpNdtw	1	1	23	0	105.1kb	52.5kb
green	open	twitter	woejXzXgTT-3TMGV3GBe5A	3	1	0	0	1.5kb	783b
green	open	multipath	-ecLfCbYQlWuuUys2Spp-Q	5	1	7	0	41.5kb	20.7kb
green	open	tdex	5KNRbc0wTCyJrkZsPK-Grg	5	1	1	0	7.7kb	3.8kb
green	open	kibana_sample_data_flights	aprYW5X1Rsu_hMsyrY4UQ	1	0	13059	0	6.4mb	6.4mb

health	status	index	uuid	pri	rep	docs.count	docs.deleted	store.size	pri.store.size
green	open	data	0CzFMKMDRH-4B-MEX7lDHA	5	1	1	0	7.6kb	3.8kb

# Elasticsearch 기본 동작 - 문서 색인 및 조회

인덱스는 미리 정의된 샤드 갯수에 의해 나뉘어짐

한 번 설정한 샤드 갯수는 변경 불가

6.x 까지는 기본 Primary Shard 개수가 5

7.x 부터는 기본 Primary Shard 개수가 1

문서는 색인될 때 랜덤한 string 을 문서 id 로 할당받거나,  
사용자가 정의한 문서 id 로 생성됨

사용자는 생성된 문서 id 를 통해 문서를 가져올 수 있음



# Elasticsearch 기본 동작 - 문서 색인 및 조회

## 샤드 할당 알고리즘

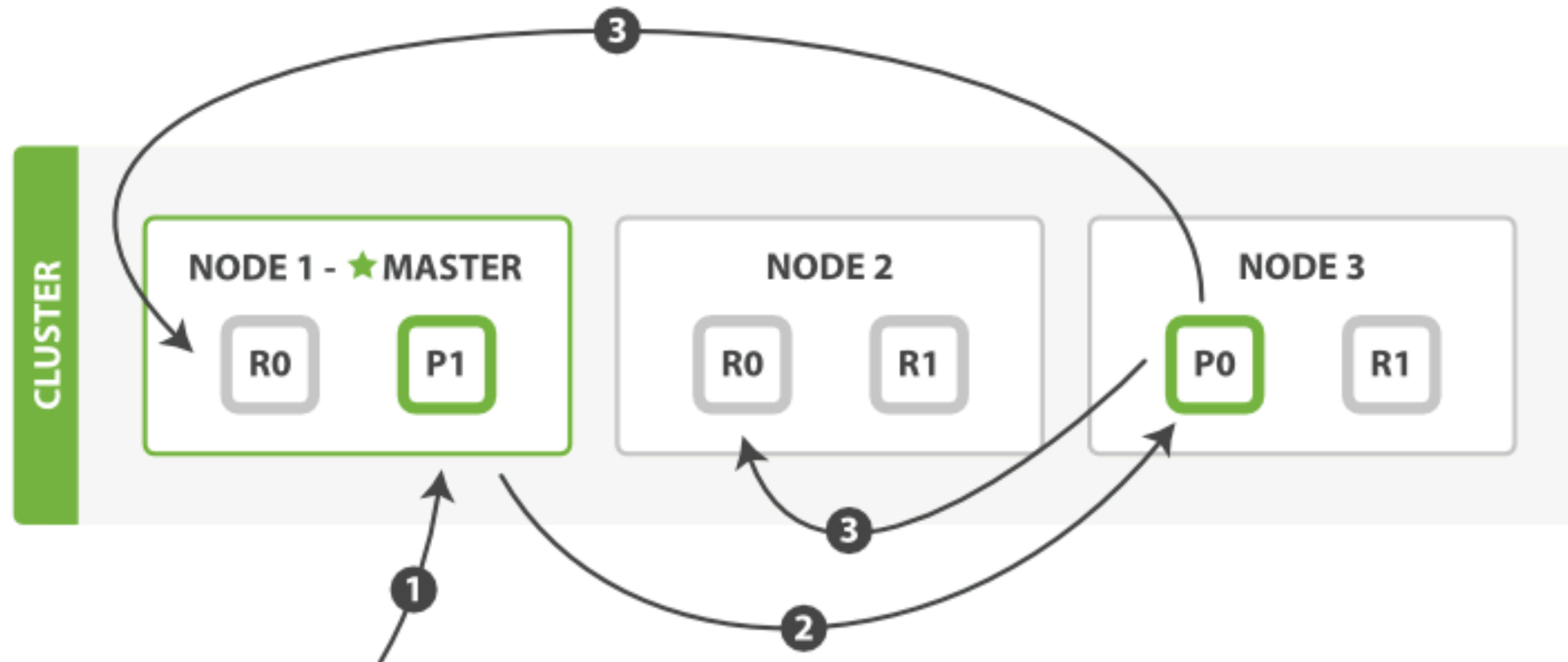
$\text{shard} = \text{hash}(\text{routing}) \% \text{number\_of\_primary\_shards}$

ES 내부 해싱 알고리즘에 의해 문서의 id 를 샤드 갯수로 나눈 나머지 값을 할당

# Elasticsearch 기본 동작 - 문서 색인 및 조회

## 인덱싱의 필수 조건

- 1) 프라이머리 샤드가 항상 제일 먼저 writing 되어야 한다
- 2) 프라이머리 샤드가 writing 이 전부 완료된 이후에 리플리카 샤드로 복제를 한다



# Elasticsearch 기본 동작 - 문서 색인 및 조회

사용자 정의된 도큐먼트를 ID 를 지정하여 인덱싱

- PUT Method 를 이용하면 지정된 ID 로 생성됨

PUT twitter/\_doc/1

```
{  
  "user" : "kimchy",  
  "post_date" : "2009-11-15T14:12:12",  
  "message" : "trying out Elasticsearch"  
}
```

7.x 버전부터는 \_doc 이외의 타입 이름은 권고되지 않음

```
{  
  "_index": "twitter",  
  "_type": "_doc",  
  "_id": "1",  
  "_version": 1,  
  "result": "created",  
  "_shards": {  
    "total": 2,  
    "successful": 2,  
    "failed": 0  
  },  
  "_seq_no": 0,  
  "_primary_term": 1  
}
```



# Elasticsearch 기본 동작 - 문서 색인 및 조회

ID 가 없을 때만 인덱싱

PUT twitter/\_doc/1?op\_type=create

```
{
  "user" : "kimchy",
  "post_date" : "2009-11-15T14:12:12",
  "message" : "trying out Elasticsearch"
}
```

PUT twitter/\_doc/1/\_create

```
{
  "user" : "kimchy",
  "post_date" : "2009-11-15T14:12:12",
  "message" : "trying out Elasticsearch"
}
```

```
{
  "error": {
    "root_cause": [
      {
        "type": "version_conflict_engine_exception",
        "reason": "[_doc][1]: version conflict, document already exists (current version [1])",
        "index_uuid": "woejXzXgTT-3TMGV3GBe5A",
        "shard": "2",
        "index": "twitter"
      }
    ],
    "type": "version_conflict_engine_exception",
    "reason": "[_doc][1]: version conflict, document already exists (current version [1])",
    "index_uuid": "woejXzXgTT-3TMGV3GBe5A",
    "shard": "2",
    "index": "twitter"
  },
  "status": 409
}
```

# Elasticsearch 기본 동작 - 문서 색인 및 조회

사용자 정의된 도큐먼트를 ID 지정 없이 인덱싱

- POST Method 를 이용하면 문서 ID 가 랜덤하게 생성됨

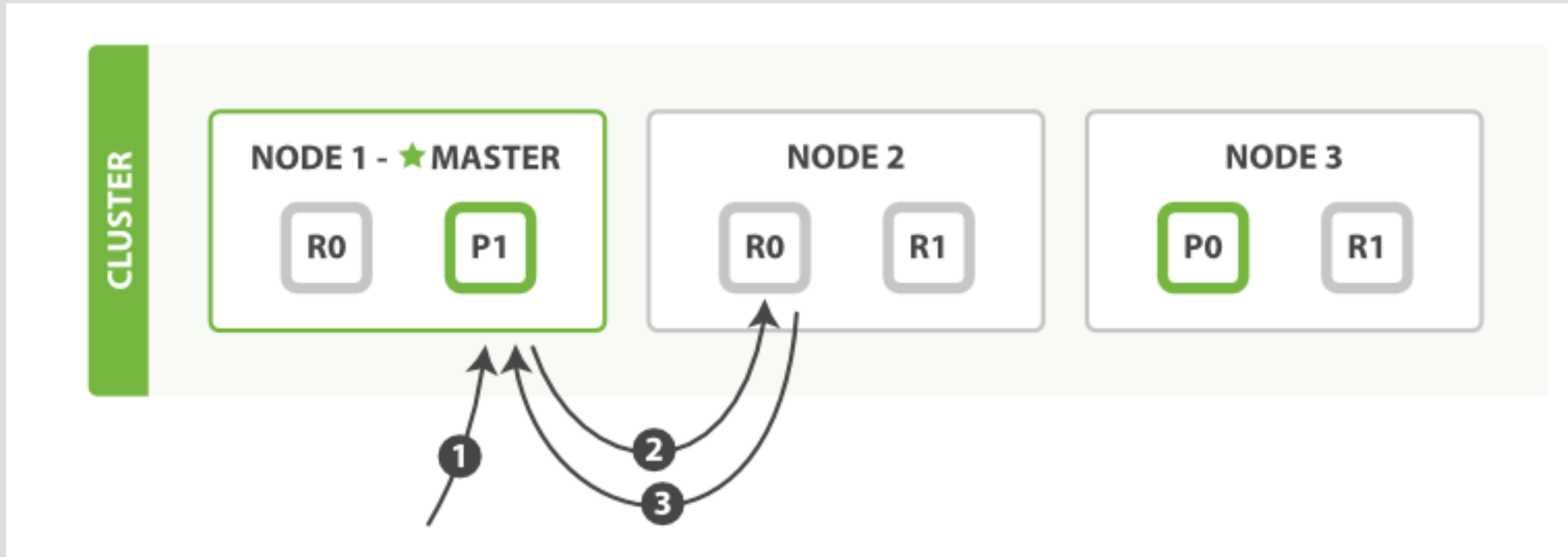
POST twitter/\_doc

```
{  
  "user" : "kimchy",  
  "post_date" : "2009-11-15T14:12:12",  
  "message" : "trying out Elasticsearch"  
}
```

```
{  
  "_index": "twitter",  
  "_type": "_doc",  
  "_id": "w9uWeGYBdUHV5YPFQ6Jt",  
  "_version": 1,  
  "result": "created",  
  "_shards": {  
    "total": 2,  
    "successful": 2,  
    "failed": 0  
  },  
  "_seq_no": 1,  
  "_primary_term": 1  
}
```

# Elasticsearch 기본 동작 - 문서 색인 및 조회

## 문서 조회



- 조회 요청을 Round Robin 으로 처리
- 최초 요청받은 노드가 응답을 주면 제일 좋을 것 같지만..
- 데이터를 가진 NODE1 이 계속 요청을 클라이언트에게 리턴
- 문서를 가진 다른 노드는 상대적으로 계속 유힤해지는 단점
- 요청 자체도 앞단에서 Balancing 권고

# Elasticsearch 기본 동작 - 문서 색인 및 조회

GET twitter/\_doc/1

- 문서 ID 를 통해 문서 조회

```
{
  "_index": "twitter",
  "_type": "_doc",
  "_id": "1",
  "_version": 1,
  "found": true,
  "_source": {
    "user": "kimchy",
    "post_date": "2009-11-15T14:12:12",
    "message": "trying out Elasticsearch"
  }
}
```

# Elasticsearch 기본 동작 - 문서 색인 및 조회

실제 문서 데이터인 `_source` 오브젝트만 조회

GET twitter/\_source/1

6.x 까지는 타입 명시

GET twitter/\_doc/1/\_source

```
{  
  "user": "kimchy",  
  "post_date": "2009-11-15T14:12:12",  
  "message": "trying out Elasticsearch"  
}
```

# Elasticsearch 기본 동작 - 문서 갱신 및 삭제

## 문서 갱신

- 동일한 문서 ID 를 통해 문서 갱신

PUT twitter/\_doc/1

```
{  
  "user" : "kimchy",  
  "post_date" : "2009-11-15T14:12:12",  
  "message" : "trying out Elasticsearch"  
}
```

```
{  
  "_index": "twitter",  
  "_type": "_doc",  
  "_id": "1",  
  "_version": 2,  
  "result": "updated",  
  "_shards": {  
    "total": 2,  
    "successful": 2,  
    "failed": 0  
  },  
  "_seq_no": 2,  
  "_primary_term": 1  
}
```



# Elasticsearch 기본 동작 - 문서 갱신 및 삭제

DELETE twitter/\_doc/1

- 문서 id 를 통해 삭제

```
{
  "_index": "twitter",
  "_type": "_doc",
  "_id": "1",
  "_version": 3,
  "result": "deleted",
  "_shards": {
    "total": 2,
    "successful": 2,
    "failed": 0
  },
  "_seq_no": 3,
  "_primary_term": 1
}
```

# Elasticsearch 기본 동작 - 클러스터 정보 확인하기

GET \_cluster/health

- 클러스터 상태정보(노드갯수, 샤드 갯수 등)

```
{
  "cluster_name": "elasticsearch",
  "status": "green",
  "timed_out": false,
  "number_of_nodes": 2,
  "number_of_data_nodes": 2,
  "active_primary_shards": 20,
  "active_shards": 39,
  "relocating_shards": 0,
  "initializing_shards": 0,
  "unassigned_shards": 0,
  "delayed_unassigned_shards": 0,
  "number_of_pending_tasks": 0,
  "number_of_in_flight_fetch": 0,
  "task_max_waiting_in_queue_millis": 0,
  "active_shards_percent_as_number": 100
}
```



# Elasticsearch 기본 동작 - 클러스터 정보 확인하기

GET \_cluster/settings

- 클러스터 사용자 설정 정보

```
{  
  "persistent": {},  
  "transient": {}  
}
```

# Elasticsearch 기본 동작

Elasticsearch 기본 동작 More..

<https://www.elastic.co/guide/en/elasticsearch/reference/current/getting-started-explore.html>

[https://www.elastic.co/guide/en/elasticsearch/reference/current/docs-index\\_.html](https://www.elastic.co/guide/en/elasticsearch/reference/current/docs-index_.html)

<https://www.elastic.co/guide/en/elasticsearch/reference/current/docs-get.html>

<https://www.elastic.co/guide/en/elasticsearch/reference/current/docs-delete.html>

# Elasticsearch 플러그인

플러그인이란?

플러그인 설치

head 플러그인 활용 - 한눈에 클러스터를 보기 위한 도구

HQ 플러그인 활용 - 한눈에 클러스터의 사용률을 보기 위한 도구

# Elasticsearch 플러그인 - 플러그인이란?

## 플러그인이란?

- ES 기능을 커스텀 설정에 의해 좀 더 강화하여 사용하는 방법
- 커스터마이징 할 기능을 가진 플러그인을 별도로 설치하여 사용
- 전체 노드에 설치해야하는 플러그인들이 있음
- 플러그인을 로딩하려면 클러스터 재시작이 필요
- root 계정으로 설치(sudo 활용)

# Elasticsearch 플러그인 - 플러그인이란?

## Core Plugins

- Elasticsearch 에서 공식적으로 지원하는 플러그인
- ES 버전이 올라갈 때마다 같이 버전 업데이트가 지원됨
- ES 에서 공식적으로 권고됨

## Community contributed

- 개인 개발자나 회사에 의해 지원되는 플러그인

# Elasticsearch 플러그인 - 플러그인 설치

```
cd /usr/share/elasticsearch/
```

```
sudo bin/elasticsearch-plugin install [plugin_name]
```

- 인터넷이 잘 되는 환경이라면..

ex) `sudo bin/elasticsearch-plugin install analysis-nori`

```
sudo bin/elasticsearch-plugin install file:///path/to/plugin.zip
```

- 잘 안된다면 파일로 받아서 설치

```
sudo bin/elasticsearch-plugin install [url]
```

- 혹은 파일서버에 올려두고 설치

```
sudo bin/elasticsearch-plugin list
```

- 설치된 플러그인 리스트 확인

```
sudo bin/elasticsearch-plugin remove [pluginname]
```

- 설치된 플러그인 제거

# Elasticsearch 플러그인 - 플러그인 활용

## head 플러그인 활용

- 한눈에 클러스터를 보기 위한 도구

	twitter		.ben-lectes-kibana
	size: 1.27ki (2.54ki) docs: 0 (0)		size: 48.2ki (96.4ki) docs: 22 (44)
	<a href="#">Info</a> <a href="#">Actions</a>		<a href="#">Info</a> <a href="#">Actions</a>
★ -Vxovpx	<a href="#">Info</a> <a href="#">Actions</a>		
● 71_q5IB	<a href="#">Info</a> <a href="#">Actions</a>		
● Krvd6u4	<a href="#">Info</a> <a href="#">Actions</a>		
● node-1	<a href="#">Info</a> <a href="#">Actions</a>	0 1 2 3	
● node-2	<a href="#">Info</a> <a href="#">Actions</a>	0 1 4 0	
● node-3	<a href="#">Info</a> <a href="#">Actions</a>	2 3 4 0	

# Elasticsearch 플러그인 - 플러그인 활용

head 플러그인 설치

<https://github.com/mobz/elasticsearch-head>

```
sudo yum -y install git
```

```
git clone https://github.com/mobz/elasticsearch-head.git
```

```
cd elasticsearch-head/
```

```
sudo yum -y install bzip2 epel-release
```

```
sudo yum -y install npm
```

```
npm install
```

```
npm run start
```

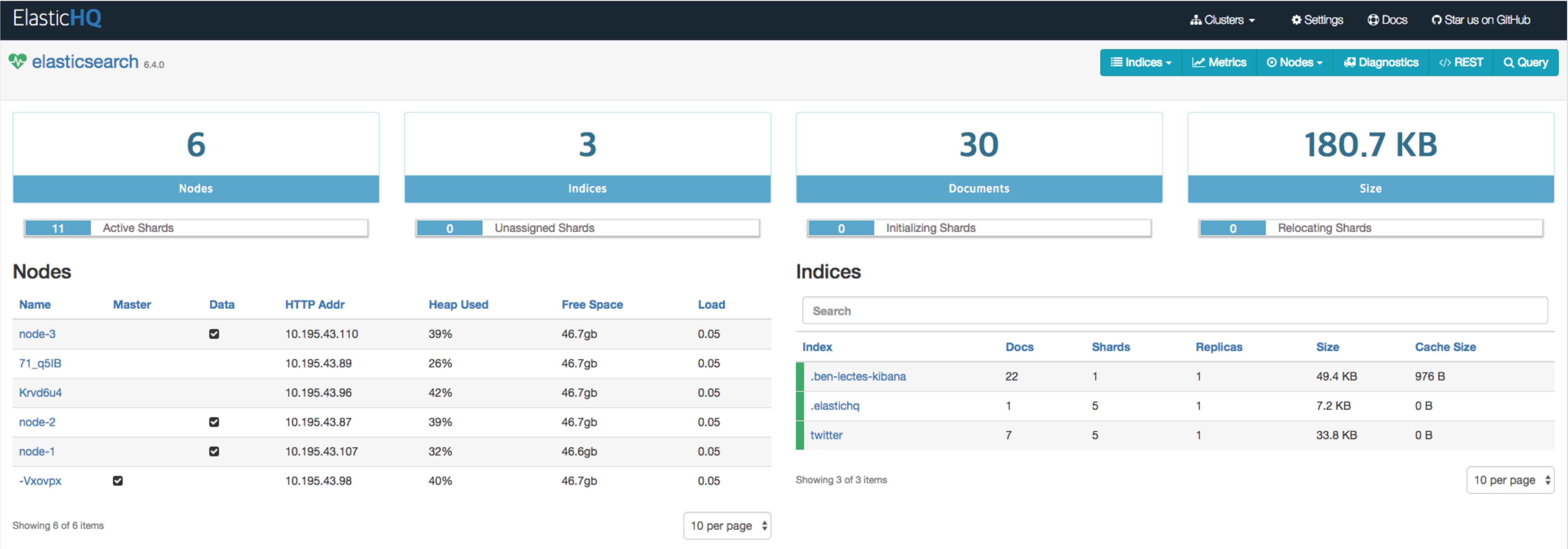
<http://{Server FQDN}:9100>



# Elasticsearch 플러그인 - 플러그인 활용

## HQ 플러그인 활용

- 한눈에 클러스터 지표를 보기 위한 도구



# Elasticsearch 플러그인 - 플러그인 활용

## HQ 플러그인 설치

<https://github.com/ElasticHQ/elasticsearch-HQ>

```
sudo yum -y install git  
git clone https://github.com/ElasticHQ/elasticsearch-HQ.git  
cd elasticsearch-HQ/
```

```
sudo yum -y install python34 python34-pip  
sudo pip3 install -r requirements.txt  
python3.4 application.py
```

<http://{Server FQDN}:5000>

Elasticsearch Plugin Install

<https://github.com/benjamin-btn/ES-Tutorial/tree/master/ES-Tutorial-2>

esbot

<https://github.com/benjamin-btn/ES-Tutorial/tree/master/ES-Tutorial-2/tools/esbot>

Q & A

Q & A