

BigData

- 실습 소스 및 참고

<MongoDB>

[MongoDB 켜기]

```
C:\Users\admin>mongod --version
```

```
db version v7.0.0
```

```
Build Info: {
```

```
  "version": "7.0.0",
```

```
  "gitVersion": "37d84072b5c5b9fd723db5fa133fb202ad2317f1",
```

```
  "modules": [],
```

```
  "allocator": "tcmalloc",
```

```
  "environment": {
```

```
    "distmod": "windows",
```

```
    "distarch": "x86_64",
```

```
    "target_arch": "x86_64"
```

```
  }
```

```
}
```

Linux, Mac에서의 설치 및 실행은 책 참고

MongoDB 구조

- Document, Collection, DataBase

한 행의 데이터 = Document

Document에 객체를 넣을 수 있음.

BSON(Binary JSON)의 형태로 저장

필드에는 문제열만 들어가지만 값에 들어갈 수 있는 형식에는 배열, 숫자, 3차원 위치 좌표로부터 필드 값을 가지는 또 다른 오브젝트도 값으로 가질 수 있음.

BSON 구조는 거의 모든 정보를 표현 할 수 있도록 다양한 값의 형식을 지원한다.

Collection과 DataBase

<MongoDB 사용 및 Collection과 DataBase 조회>

Mongo

```
use testDB
```

```
switched to db testDB
```

```
db.myCollection.insertOne({x:1})
```

```
db
```

```
testDB
```

```
show dbs
```

```
show collections
```

<도큐먼트 삽입, 수정, 삭제>

<MongoDB 게시판 작성하기>

<데이터 입력>

<한개의 데이터 입력>

```
예> db.article.insertOne({  
  board_id: "secretboard_id",
```

```
    title: 'my Secret Title',
    content: 'hi, hello1',
    author: 'noname'
  })
}
```

<게시판 도큐먼트 생성하기>

```
use board
freeboard_result = db.board.insertOne({name: "자유게시판"})
freeboard_id = freeboard_result.insertedId
```

<게시판 데이터 입력>

```
db.article.insertMany([
  {
    board_id: "freeboard_id",
    title: 'hello',
    content: 'hi, hello1',
    author: 'Karoid'
  },
  {
    board_id: "freeboard_id",
    title: 'hi',
    content: 'hi, hello2',
    author: 'Jeong'
  },
  {
    board_id: "freeboard_id",
    title: 'hi',
    content: 'hi, hello3',
    author: 'Hong',
    comments: [
      {
        author: 'karoid',
        content: 'hello Hong!'
      }
    ]
  }
])
```

<비밀게시판 만들기>

```
secretboard_id = db.board.insertOne({name: '비밀게시판'}).insertedId
db.article.insertOne({
  board_id: "secretboard_id",
  title: 'my Secret Title',
  content: 'hi, hello1',
  author: 'noname'
})
```

<글을 불러오기 - 조회>

```
use board // Document 조회
freeboard_id = db.board.find({name: '자유게시판'}).toArray()[0]._id
```

```
db.article.find({board_id: "freeboard_id"},{
  _id: false, board_id: false, author: false, comments: false})
```

<글 수정하기>

```
use board // Document 조회
db.article.updateMany({},{$set: {upvote: 0}})
```

<글 추천수 증가>

```
secretboard_id = db.board.find({name: "비밀게시판"}).toArray()[0]._id
db.article.updateMany({board_id: secretboard_id}, {$inc: {upvote: 1}})
```

<댓글이 달린 자유게시글의 id 저장>

```
freeboard_id = db.board.find({name: "자유게시판"}).toArray()[0]._id
doc_id = db.article.find({board_id: "freeboard_id", author: "Hong"}).toArray()[0]._id
```

<Content 값 수정>

```
db.articel.updateOne({_id: 'doc_id'}, {$set: {content: 'updated'}})
```

<comments 필드의 요소 추가>

```
db.article.updateOne(
  { _id: 'doc_id' }, {
    $push: {comments: {author: "Quote", content: "reply"}}
  })
```

```
db.article.updateOne(
  { _id: 'doc_id' }, {
    $set: { "comments.$[karoidcomment].upvote": 0 }
  },
  {
    arrayFilters: [{ "karoidcomment.author": "Quote" }]
  }
)
```

<Document 삭제>

```
db.article.deleteMany({}) // Collection 내의 모든 Document 삭제
db.article.drop() // Collection 삭제(article 컬렉션)
db.board.drop() // Collection 삭제(board 컬렉션)
db.dropDatabase() // 선택된 DB 삭제
```

순차적으로 하고 show collections 하면 비어있는 것 확인

<Web 환경으로 연결>

Python - Django(장고)
Node.js - Express(익스프레스) 등이 있음.

Spring Boot로도 연결할 수 있음.

