

## NodeJS와 MongoDB 연동



2016년도 2학기 2학년 방과후학교

# Node.js와 MongoDB 연동

- Node.js에서 몽고DB를 연동하는 방법
  - ✓ Node.js MongoDB Driver : MongoDB용 Node.js 드라이버 모듈 이용
  - ✓ 몽구스(Mongoose) : ODM (Object Document Mapper) 도구 이용

The logo for Mongoose, featuring the word "mongoose" in a bold, lowercase, red sans-serif font.

elegant **mongodb** object modeling for **node.js**

# MongoDB Driver

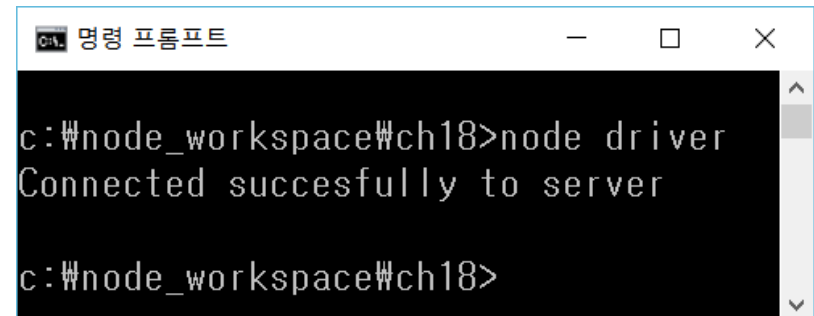
- MongoDB 콘솔에서 사용하는 명령어를 그대로 사용한다.
- `npm install mongodb`
- <http://mongodb.github.io/node-mongodb-native/2.1/>

# MongoDB Driver

```
var MongoClient = require('mongodb').MongoClient;

// Connection URL
var url = 'mongodb://localhost:27017/example';

// Use connect method to connect to the server
MongoClient.connect(url, function(err, db) {
  console.log("Connected succesfully to server");
  // 데이터 CRUD 로직 추가
  // 1. 콜백 함수 방식
  // 2. Promise 방식
  db.close();
});
```

A screenshot of a Windows Command Prompt window titled "명령 프롬프트". The window shows the command "node driver" being executed in the directory "c:\node\_workspace\ch18". The output is "Connected succesfully to server".

```
명령 프롬프트
c:\node_workspace\ch18>node driver
Connected succesfully to server
c:\node_workspace\ch18>
```

# Mongoose란?

- 몽구스(Mongoose)는 ODM(Object Document Mapper)의 약자로 ORM과 비슷하게 데이터를 다룬다.
- 몽구스에서는 스키마를 이용해서 다루는 도큐먼트의 형식을 정의한다.
- 도큐먼트에 정해진 형식 없이 저장할 수 있는 것이 몽고DB의 장점이지만, 정해진 형식이 없다는 것이 단점으로 동작할 때도 많다.
- 몽구스에서는 스키마를 이용해서 정해진 형식에 맞는 데이터만 다룰 수 있다.
- `npm install mongoose`

# Mongoose 처리 순서 - 1

- 데이터베이스 연결

```
var mongoose = require('mongoose');
mongoose.connect('mongodb://localhost:27017/example');

var db = mongoose.connection;

db.on('error', console.error.bind(console, 'connection error:'));
db.once('open', function() {
  console.log('Database connected!!');
});
```

## Mongoose 처리 순서 - 2

- 스키마 정의 (테이블의 구조와 유사)
- 스키마에서 모델 클래스 얻기 (모델은 MongoDB의 컬렉션에 해당)

```
// 스키마 정의
```

```
var fruitSchema = mongoose.Schema({ name: String, price: Number});
```

```
var Fruit = mongoose.model('fruit', fruitSchema);
```

## Mongoose 처리 순서 - 3

- 모델에서 문서 생성

```
var melon = new Fruit({ name: "melon", price: 5000 });
```

- MongoDB에 반영

```
// 콜백 함수 방식
melon.save(function (err, result) {
  // 콜백 처리
});

// Promise 방식
melon.save().then(resolved, rejected);
```



# [실습] 영화 관리 어플리케이션 (Mongoose)

- MongoDB로 직접 영화 데이터를 CRUD 처리해보자. (RESTful API 사용)

GET

127.0.0.1:3000/movie

Params

Send

Save

Authorization

Headers (1)

Body

Pre-request Script

Tests

Generate Code

Type

No Auth

Body

Cookies

Headers (6)

Tests

Status: 200 OK

Time: 21 ms

Pretty

Raw

Preview

JSON

```
1 {
2   {
3     "_id": "57dbf4ae424f7c29543c94a1",
4     "title": "아바타",
5     "director": "제임스 카메론",
6     "year": 2009,
7     "poster": "avata.jpg",
8     "synopsis": "인류의 마지막 희망, 행성 판도라! 이 곳을 정복하기 위한 'O
시작된다! 가까운 미래, 지구는 에너지 고갈 문제를 해결하기 위해 대L
자원을 채굴하기 시작한다. 하지만 판도라의 독성을 지닌 대기로 인해
된 인류는 판도라의 토착민 '나비(Na'vi)'의 외형에 인간의 의식을 주입
새로운 생명체 '아바타'를 탄생시키는 프로그램을 개발한다.",
9     "__v": 0,
10    "created": "2016-09-16T13:33:34.055Z"
11  },
12  {
13    "_id": "57dbf4c2424f7c29543c94a2",
14    "title": "스타워즈",
15    "director": "조지 루카스",
16    "year": 1977,
17    "poster": "starwars.jpg",
18    "synopsis": "평화롭던 은하계에서 타린총독(피터 커싱)이 왕정에 저항하며
우주정거장인 죽음의 별을 완성하고 은하계의 작은 나라들을 점령하고자
반란군은 레아 공주(캐리 피셔)를 보내 죽음의 별 설계를 입수하려고
잡는다. 대신 공주는 도움을 요청하기 위해 제다이 기사 케노비(알렉 그
19    "__v": 0,
20    "created": "2016-09-16T13:33:54.139Z"
21  }
22 }
```

GET

127.0.0.1:3000/movie/57dbf4ae424f7c29543c94a1

Params

Send

Save

Authorization

Headers (1)

Body

Pre-request Script

Tests

Generate Code

Type

No Auth

Body

Cookies

Headers (6)

Tests

Status: 200 OK

Time: 26 ms

Pretty

Raw

Preview

JSON

```
1 {
2   {
3     "_id": "57dbf4ae424f7c29543c94a1",
4     "title": "아바타",
5     "director": "제임스 카메론",
6     "year": 2009,
7     "poster": "avata.jpg",
8     "synopsis": "인류의 마지막 희망, 행성 판도라! 이 곳을 정복하기 위한 '아바타 프로젝트'가 시작된다!
가까운 미래, 지구는 에너지 고갈 문제를 해결하기 위해 머나먼 행성 판도라에서 대체 자원을
채굴하기 시작한다. 하지만 판도라의 독성을 지닌 대기로 인해 자원 획득에 어려움을 겪게 된
인류는 판도라의 토착민 '나비(Na'vi)'의 외형에 인간의 의식을 주입, 원격 조종이 가능한 새로운
생명체 '아바타'를 탄생시키는 프로그램을 개발한다.",
9     "__v": 0,
10    "created": "2016-09-16T13:33:34.055Z"
11  }
12 }
```