

# 9월 29일 (수)

## 1. 코딩애플 - Node.js & MongoDB - Part 2

- MongoDB 셋팅하기 (무료 호스팅 받기)

The screenshot shows a Visual Studio Code interface with a dark theme. There are two tabs open: 'server.js' and 'write.html'. The 'server.js' tab contains the following code:

```
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
  응답.send('전송완료');
  console.log(요청.body.date);
  console.log(요청.body.title);
});
```

On the right side of the screen, there is a summary of the course content:

웹사이트 기능만들기 기본  
1. 서버로 데이터 전송할 수 있는 UI 만들고  
2. 서버에서 원하는대로 정보를 처리해주면 됨

Ln 16, Col 9 (78 selected) Spaces: 2 UTF-8 CRLF JavaScript ⚡

```
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
  응답.send('전송완료');
  console.log(요청.body.date);
  console.log(요청.body.title);
});
```

】 중요 정보니까 어딘가 저장하는게 좋겠군요  
.txt 파일 만들어서 거기 저장해도 가능

데이터를 다루기 힘들고 대량의 데이터 못 다룸

```
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
  응답.send('전송완료');
  console.log(요청.body.date);
  console.log(요청.body.title);
});
```

】 중요 정보니까 어딘가 저장하는게 좋겠군요  
엑셀 파일 만들어서 거기 저장해도 가능

엑셀로 가로줄 1억개 저장 불가능 너무 느려짐 초대용량의 데이터 저장 불가

File Edit Selection View Go Run Terminal Help

server.js - todoapp1 - Visual Studio Code

write.html JS server.js X

```
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
  응답.send('전송완료');
  console.log(요청.body.date);
  console.log(요청.body.title); ] Database에 저장합니다
});
```

customerid name email

5	Rosalyn Rivera	rosalyn@adatum.com
6	Jayne Sargent	jayne@contoso.com
7	Dean Luong	dean@contoso.com

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF JavaScript ⚡

MySQL, MariaDB, Oracle, MS SQL Server 의 관계형

File Edit Selection View Go Run Terminal Help

server.js - todoapp1 - Visual Studio Code

write.html JS server.js X

```
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('listening on 8080') 데이터 이름 달고 →
});                                실제 데이터 기입 →
```

customerid name email

5	Rosalyn Rivera	rosalyn@adatum.com
6	Jayne Sargent	jayne@contoso.com
7	Dean Luong	dean@contoso.com

app.post('/add', function (요청, 응답) {
 응답.send('전송완료');
 console.log(요청.body.date);
 console.log(요청.body.title);
});

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF JavaScript ⚡

DB 종류1. 관계형

```

const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('Listening on 8080');
  app.post('/add', (req, res) => {
    res.send('전송되었습니다.');
    console.log(req.body);
    console.log(res);
  });
});
  
```

DB 종류1. 관계형  
- 대부분 SQL이라는 언어 써야함

```

const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('Listening on 8080');
  app.post('/add', (req, res) => {
    res.send('전송되었습니다.');
    console.log(req.body);
    console.log(res);
  });
});
  
```

File Edit Selection View Go Run Terminal Help

server.js - todoapp1 - Visual Studio Code

write.html JS server.js X

```
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
  응답.send('전송완료');
  console.log(요청.body.date);
  console.log(요청.body.title);
});

```

DB 종류2. NoSQL  
- Object 자료형으로 입출력 가능

자료1

자료2

자료3

이름 : "Kim",  
나이 : 22

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF JavaScript ⚡

File Edit Selection View Go Run Terminal Help

server.js - todoapp1 - Visual Studio Code

write.html JS server.js X

```
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
  응답.send('전송완료');
  console.log(요청.body.date);
  console.log(요청.body.title);
});

```

DB 종류2. NoSQL  
- Object 자료형으로 입출력 가능

Dynamo  
Oracle NoSQL  
MongoDB  
Redis  
Cassandra

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF JavaScript ⚡

DB 종류2. NoSQL  
- 데이터 입출력에만 신경쓸 수 있음

```

const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
  응답.send('전송완료');
  console.log(요청.body.date);
  console.log(요청.body.title);
});

```

행과 열 신경안써도 된다는 점이 아주 편하다.

MongoDB  
무료 호스팅을 받아쓰자

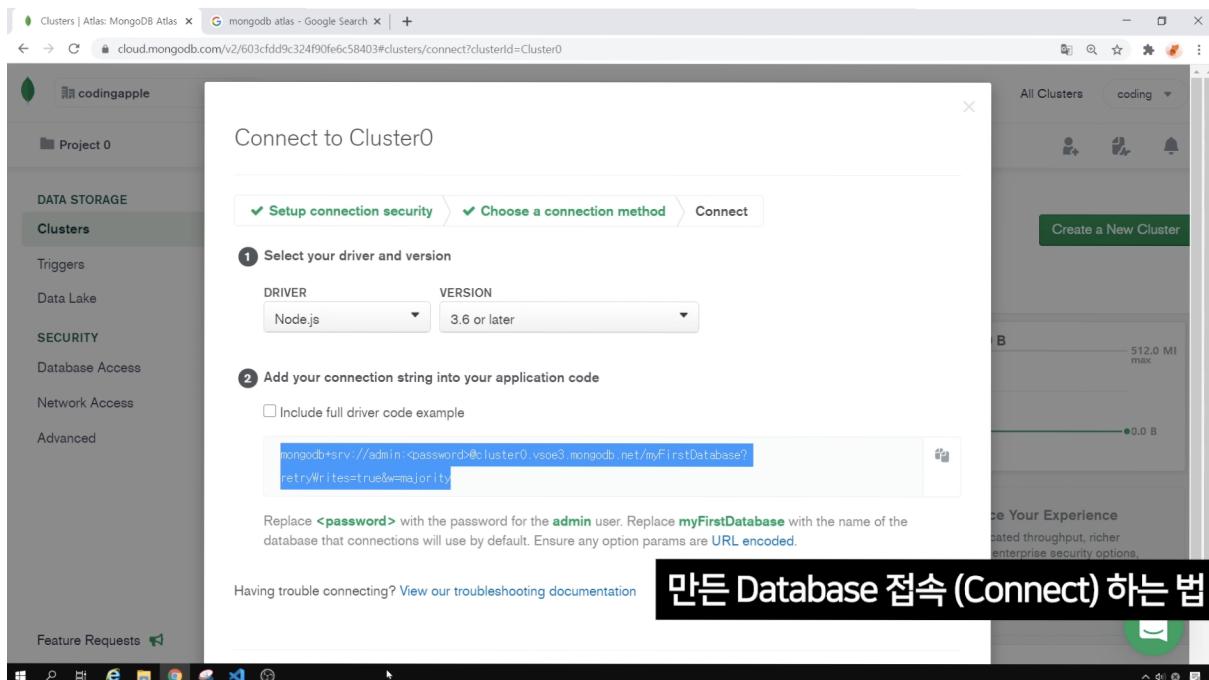
```

const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

http.listen(8080, function () {
  console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
  응답.send('전송완료');
  console.log(요청.body.date);
  console.log(요청.body.title);
});

```



The screenshot shows Visual Studio Code with a file named 'server.js' open. The code includes:

```
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

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

To the right of the code, the text 'npm install mongodb' is displayed.

The terminal window at the bottom shows the command being run in a Windows PowerShell:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 크로스 플랫폼 PowerShell 사용 https://aka.ms/powershell
PS C:\Users\harry\Desktop\todoapp\todoapp1> npm install mongodb
```

A screenshot of Visual Studio Code showing the 'server.js' file. The code is a Node.js application for a todo list. It includes importing 'body-parser', setting up middleware, connecting to MongoDB, and defining a POST endpoint for adding tasks. The code is written in English.

```
File Edit Selection View Go Run Terminal Help
write.html JS server.js
server.js > ...
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

const MongoClient = require('mongodb').MongoClient;
MongoClient.connect('URL', function(에러, client){
    })

http.listen(8080, function () {
    console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
    응답.send('전송완료');
    console.log(요청.body.date);
    console.log(요청.body.title);
});

In 11, Col 25 Spaces: 2 UTF-8 CRLF JavaScript ⚡ 🔍
```

A screenshot of Visual Studio Code showing the 'server.js' file. The code is identical to the one in the previous screenshot, but it includes a database connection string for MongoDB. A callout box points to the connection string with the text 'Database Access 메뉴에서 만든 아이디:비번 입력필요' (Input required for ID:password from Database Access menu).

```
File Edit Selection View Go Run Terminal Help
write.html JS server.js
server.js > ...
const bodyParser = require('body-parser');
app.use(bodyParser.urlencoded({ extended: true }));

const MongoClient = require('mongodb').MongoClient;
MongoClient.connect('mongodb+srv://admin:<password>@cluster0.vsoe3.mongodb.net/myFirstDatabase?retryWrites=true&w=majority', function(에러, client){

})
Database Access 메뉴에서 만든
아이디:비번 입력필요

http.listen(8080, function () {
    console.log('listening on 8080')
});

app.post('/add', function (요청, 응답) {
    응답.send('전송완료');
    console.log(요청.body.date);
    console.log(요청.body.title);
});

In 11, Col 36 (16 selected) Spaces: 2 UTF-8 CRLF JavaScript ⚡ 🔍
```

```
const MongoClient = require('mongodb').MongoClient;
MongoClient.connect('mongodb+srv://admin:qwer1234@cluster0.vsoe3.mongodb.net/myFirstDatabase?retryWrites=true&w=majority', function(에러, client){

    app.listen(8080, function () {
        console.log('listening on 8080')    전에 했던 app.listen() 여기로 옮기셈
    });

})

app.post('/add', function (요청, 응답) {
    응답.send('전송완료');
    console.log(요청.body.date);
    console.log(요청.body.title);
});
```

Line 11, Col 123 (111 selected) Spaces: 2 UTF-8 CRLF JavaScript ⚡ ⌂

```
const MongoClient = require('mongodb').MongoClient;
MongoClient.connect('mongodb+srv://admin:qwer1234@cluster0.vsoe3.mongodb.net/myFirstDatabase?retryWrites=true&w=majority', function(에러, client){ database 접속이 완료되면
    app.listen(8080, function () {
        console.log('listening on 8080')    내부 코드 실행해주셈~
    });
});
```

▲ nodejs 서버띄우는 코드였음

```
app.post('/add', function (요청, 응답) {
    응답.send('전송완료');
    console.log(요청.body.date);
    console.log(요청.body.title);
});
```

Line 15, Col 6 Spaces: 2 UTF-8 CRLF JavaScript ⚡ ⌂