(IoT Network) Practice -10-

MongoDB Server and Web UI

Index

I. Practice Overview

II. Installation of MongoDB Server

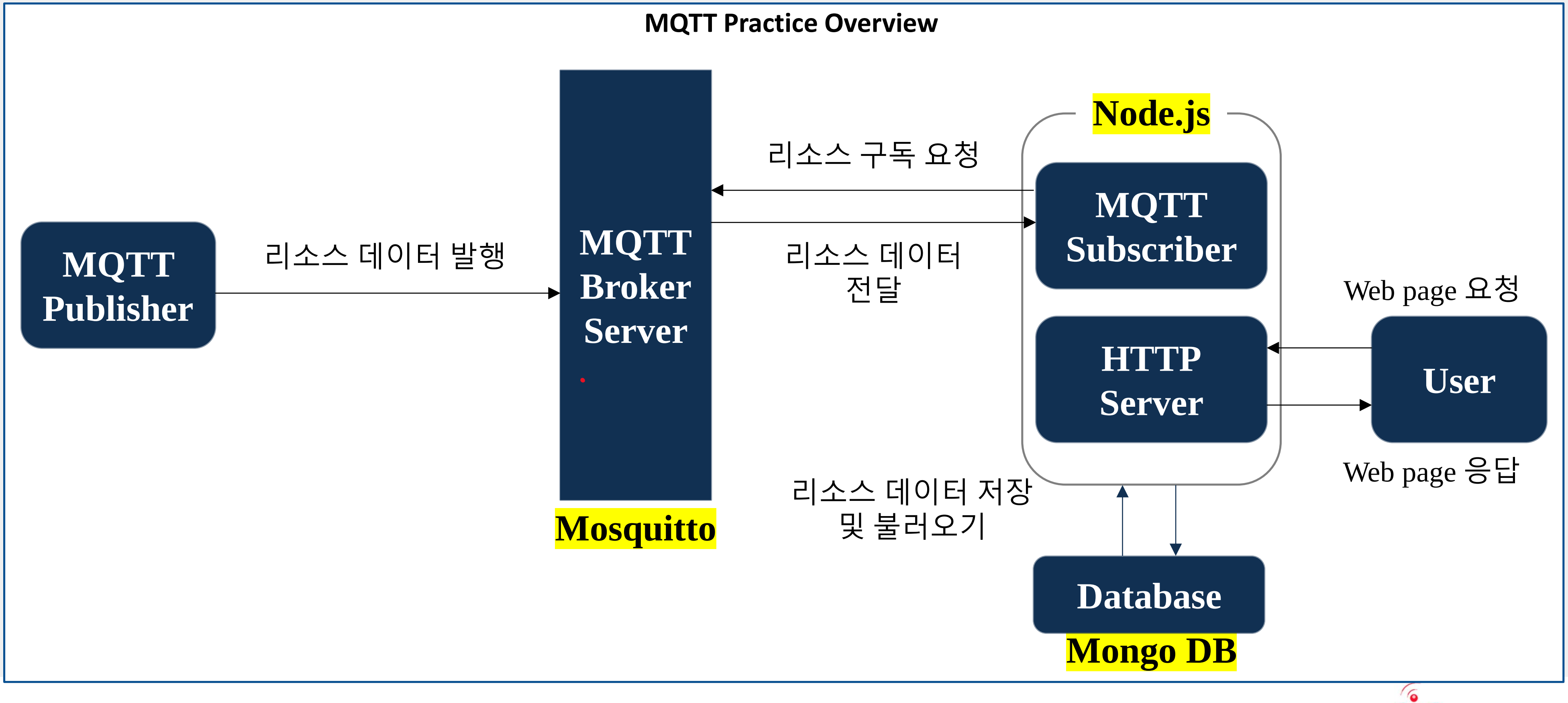
III. MongoDB Server Setting

IV. Forwarding Resource Data to Database

V. Web UI for Resource Monitoring

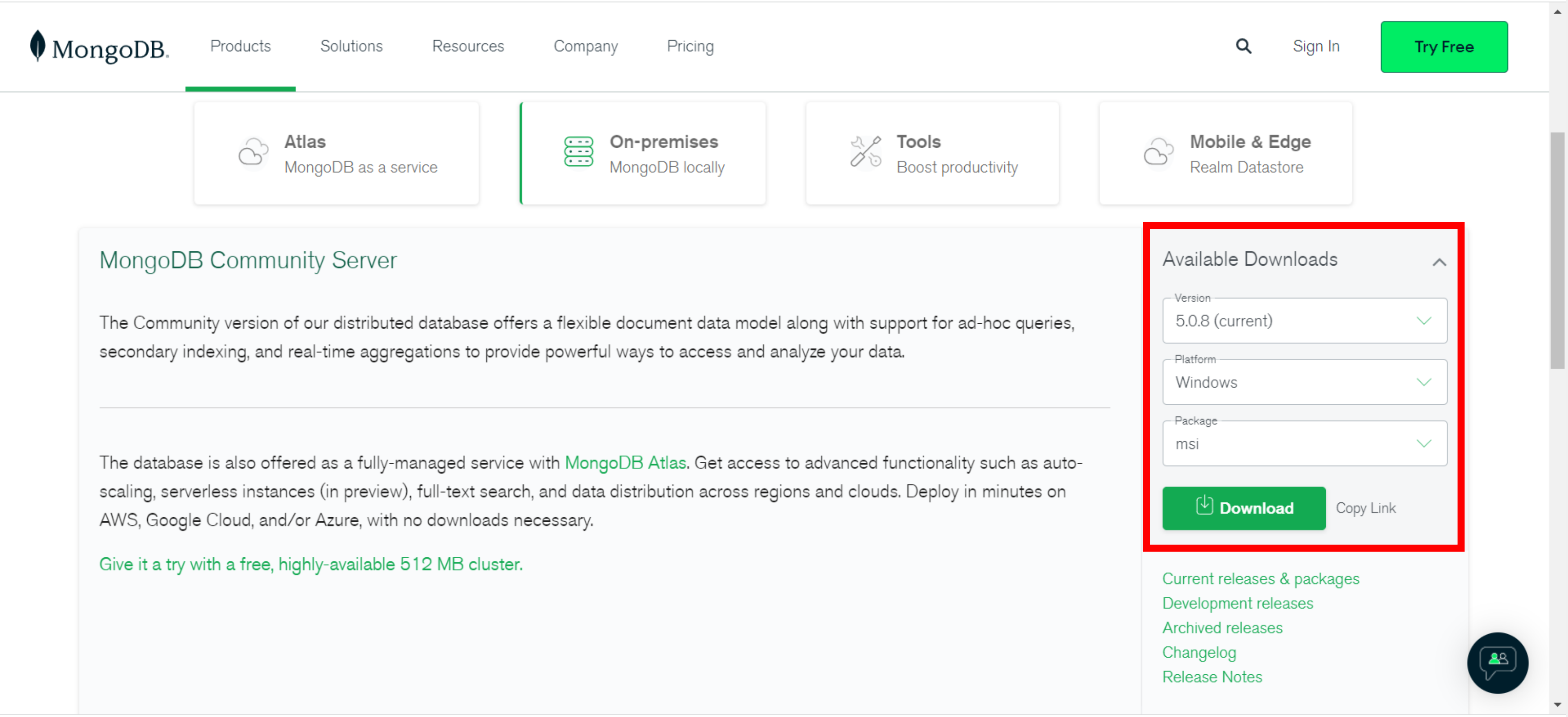
Practice Overview

- **실습목표**
  - MongoDB Server 설치 및 환경설정
  - Publish된 Resource Data를 Database에 저장
  - Database에 저장된 Resource Data 모니터링을 위한 Web UI 작성



Installation of MongoDB Server

- **Download MongoDB Server**
  - MongoDB 공식 홈페이지에서 Community Server 다운로드
    - ✓ https://www.mongodb.com/try/download/community



MongoDB Community Server

The Community version of our distributed database offers a flexible document data model along with support for ad-hoc queries, secondary indexing, and real-time aggregations to provide powerful ways to access and analyze your data.

The database is also offered as a fully-managed service with MongoDB Atlas. Get access to advanced functionality such as auto-scaling, serverless instances (in preview), full-text search, and data distribution across regions and clouds. Deploy in minutes on AWS, Google Cloud, and/or Azure, with no downloads necessary.

Give it a try with a free, highly-available 512 MB cluster.

Available Downloads

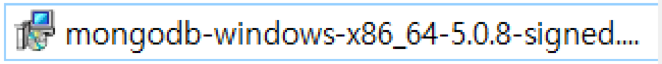
Version: 5.0.8 (current)

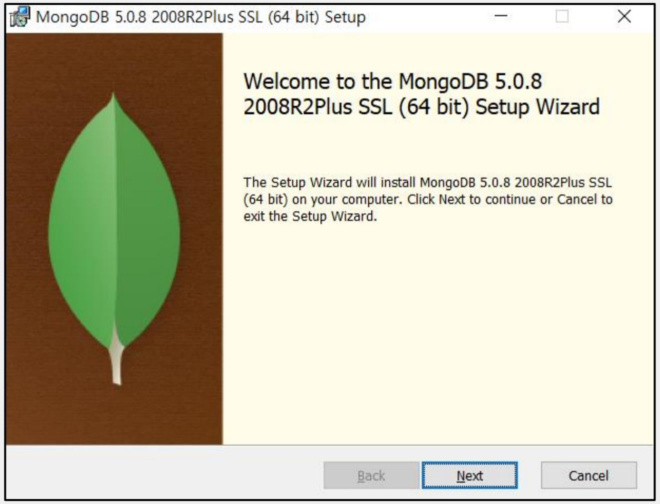
Platform: Windows

Package: msi

Installation of MongoDB Server

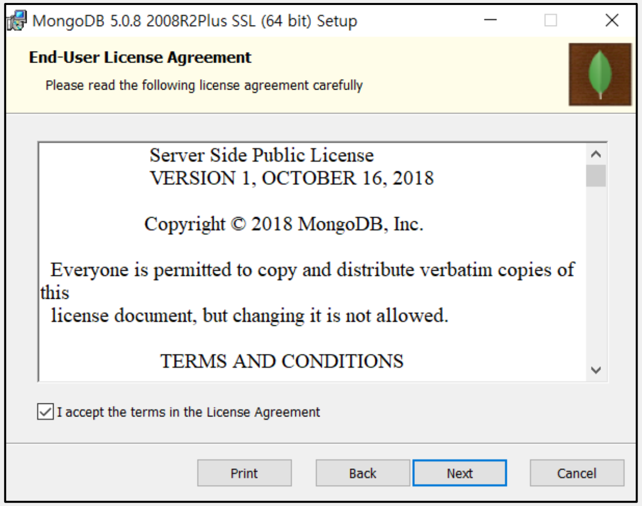
- **Installation of MongoDB Server**
  - 설치파일 실행


mongodb-windows-x86_64-5.0.8-signed....


MongoDB 5.0.8 2008R2Plus SSL (64 bit) Setup

Welcome to the MongoDB 5.0.8 2008R2Plus SSL (64 bit) Setup Wizard

The Setup Wizard will install MongoDB 5.0.8 2008R2Plus SSL (64 bit) on your computer. Click Next to continue or Cancel to exit the Setup Wizard.


MongoDB 5.0.8 2008R2Plus SSL (64 bit) Setup

End-User License Agreement

Please read the following license agreement carefully

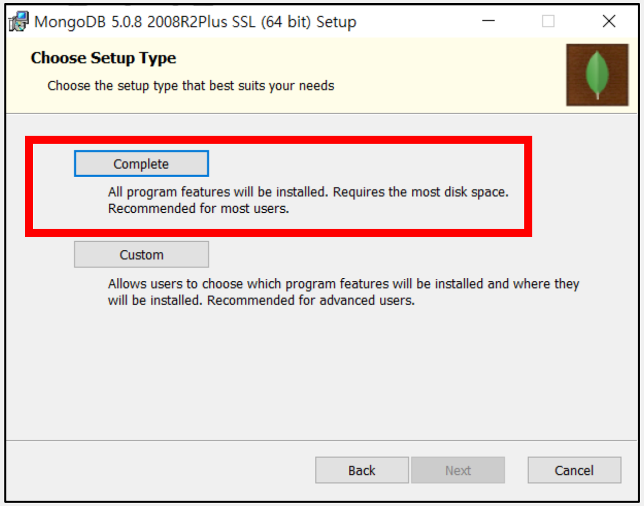
Server Side Public License
VERSION 1, OCTOBER 16, 2018

Copyright © 2018 MongoDB, Inc.

Everyone is permitted to copy and distribute verbatim copies of this
license document, but changing it is not allowed.

TERMS AND CONDITIONS

☑ I accept the terms in the License Agreement


MongoDB 5.0.8 2008R2Plus SSL (64 bit) Setup

Choose Setup Type

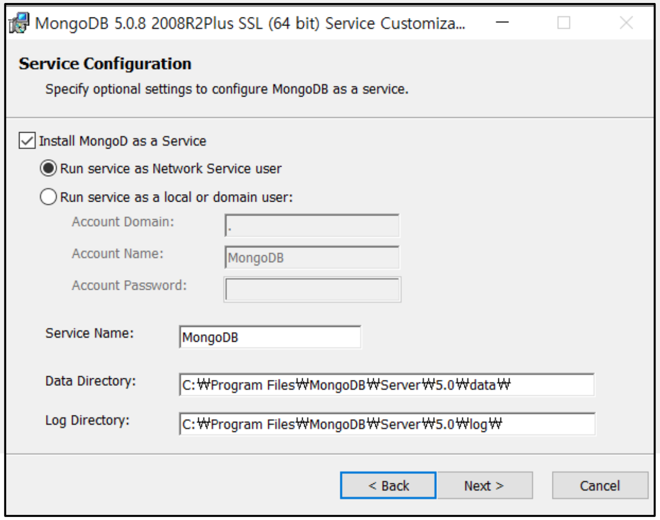
Choose the setup type that best suits your needs

Complete

All program features will be installed. Requires the most disk space. Recommended for most users.

Custom

Allows users to choose which program features will be installed and where they will be installed. Recommended for advanced users.


MongoDB 5.0.8 2008R2Plus SSL (64 bit) Service Customiza...

Service Configuration

Specify optional settings to configure MongoDB as a service.

☑ Install MongoD as a Service

● Run service as Network Service user

○ Run service as a local or domain user:

Account Domain: .

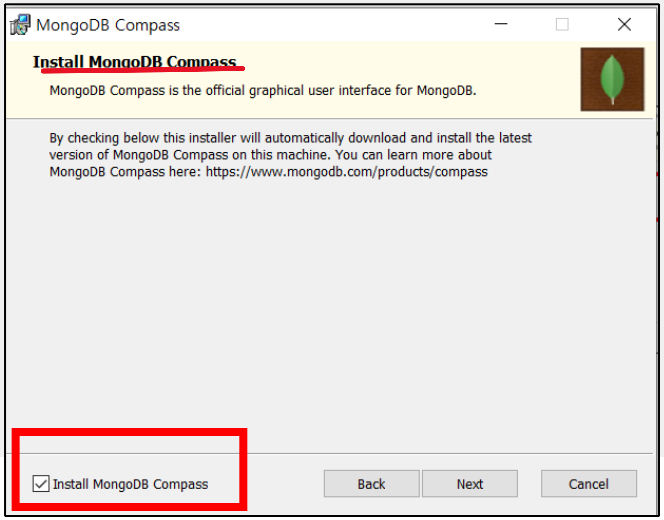
Account Name: MongoDB

Account Password:

Service Name: MongoDB

Data Directory: C:₩Program Files₩MongoDB₩Server₩5.0₩data₩

Log Directory: C:₩Program Files₩MongoDB₩Server₩5.0₩log₩

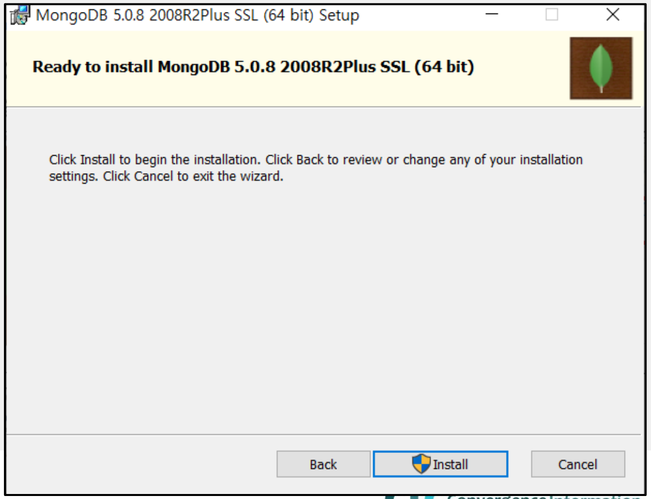

MongoDB Compass

Install MongoDB Compass

MongoDB Compass is the official graphical user interface for MongoDB.

By checking below this installer will automatically download and install the latest version of MongoDB Compass on this machine. You can learn more about MongoDB Compass here: https://www.mongodb.com/products/compass

☑ Install MongoDB Compass

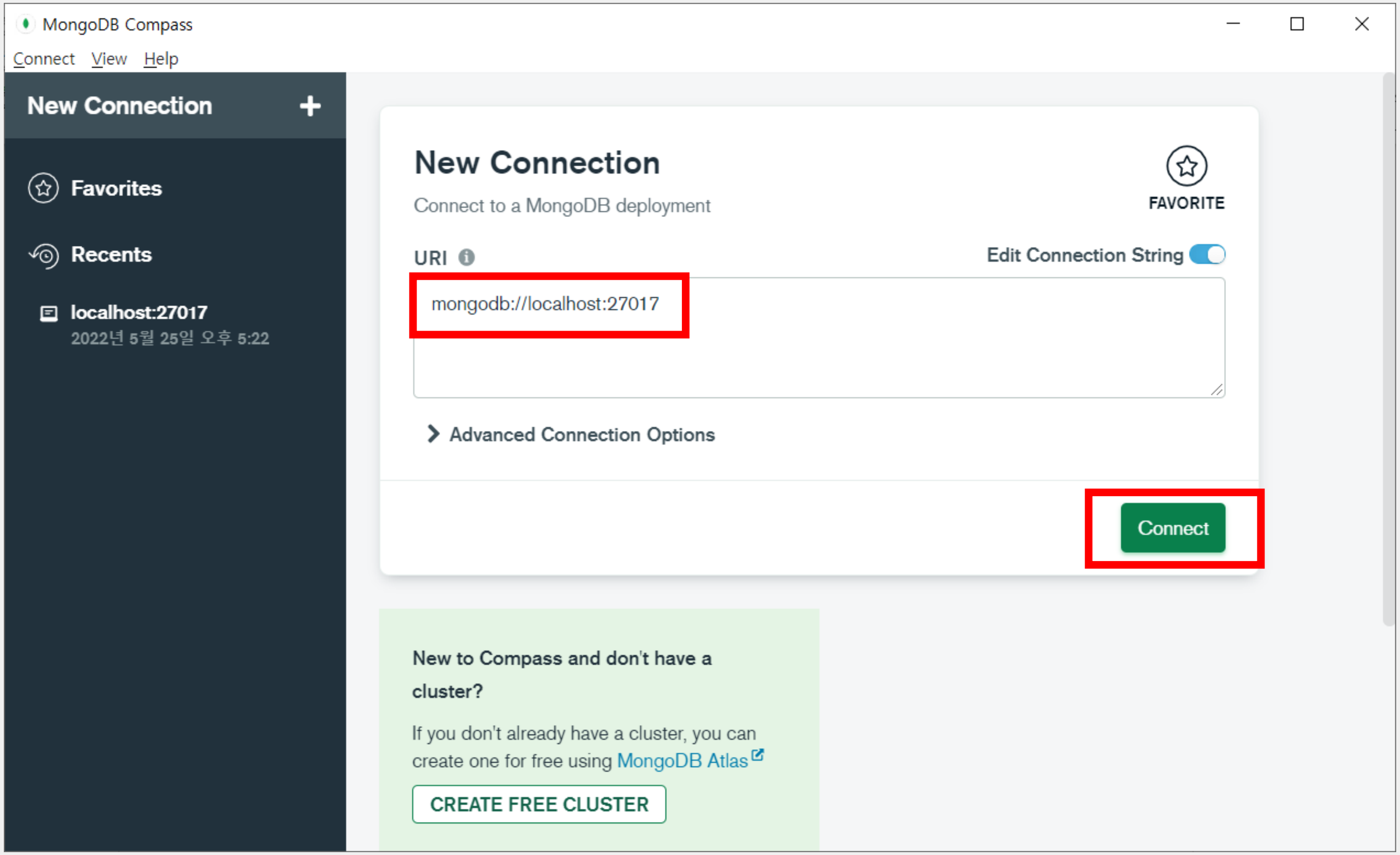

MongoDB 5.0.8 2008R2Plus SSL (64 bit) Setup

Ready to install MongoDB 5.0.8 2008R2Plus SSL (64 bit)

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

MongoDB Server Setting

- **Connecting to MongoDB Server**
  - MongoDB GUI Client인 Compass를 통해 MongoDB 접속



MongoDB Compass

Connect View Help

New Connection +

Favorites

Recents

localhost:27017
2022년 5월 25일 오후 5:22

New Connection

Connect to a MongoDB deployment

FAVORITE

URI

Edit Connection String

```
mongodb://localhost:27017
```

Advanced Connection Options

Connect

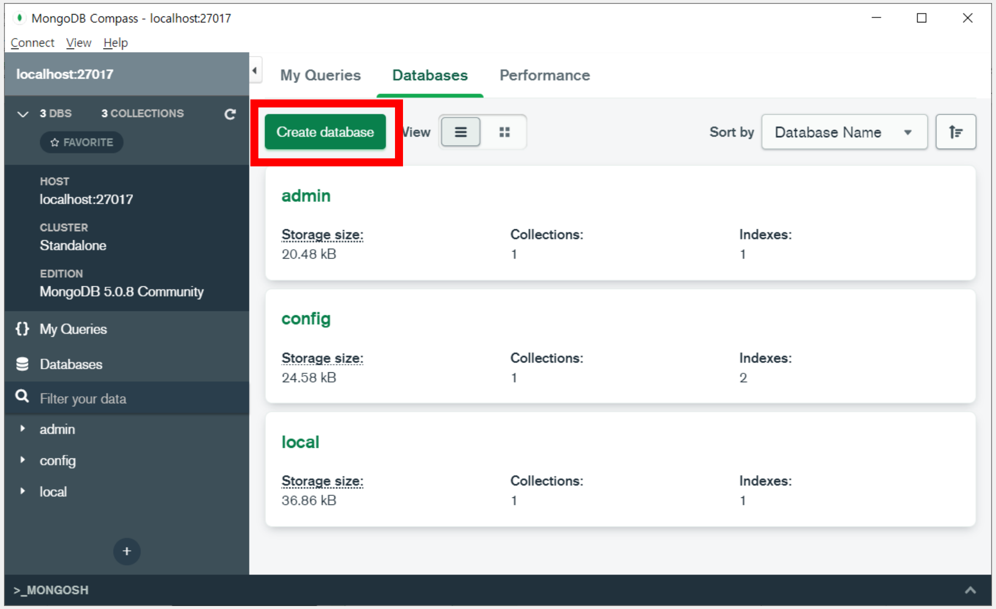
New to Compass and don't have a cluster?

If you don't already have a cluster, you can create one for free using MongoDB Atlas

CREATE FREE CLUSTER

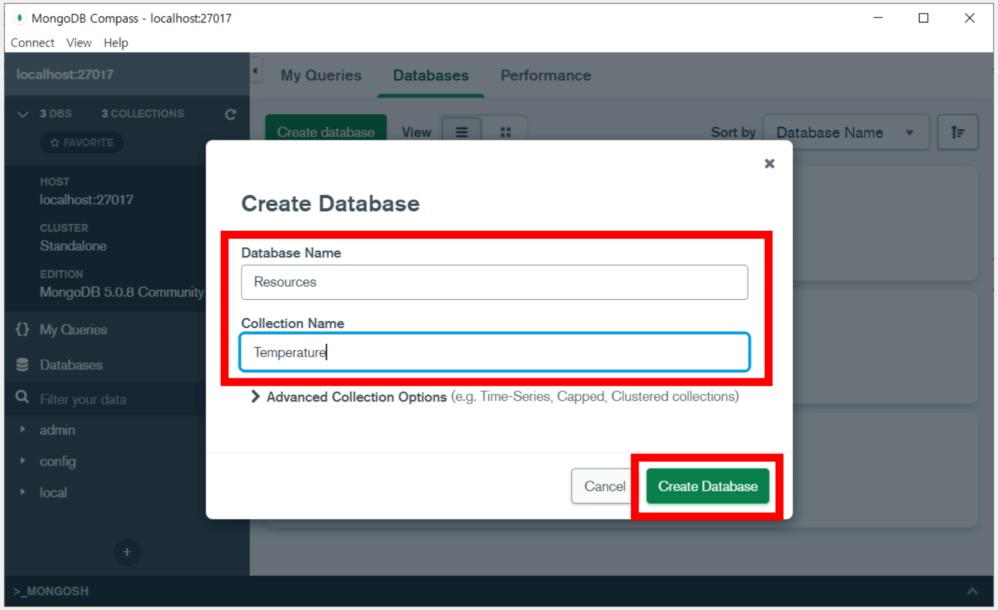
MongoDB Server Setting

- **MongoDB Server Setting**
  - Server에 새로운 Database 생성
    - ✓ Database 이름 → Resources
    - ✓ Database에 포함되는 Collection → Temperature



HOST localhost:27017
CLUSTER Standalone
EDITION MongoDB 5.0.8 Community

Database	Storage size:	Collections:	Indexes:
admin	20.48 kB	1	1
config	24.58 kB	1	2
local	36.86 kB	1	1



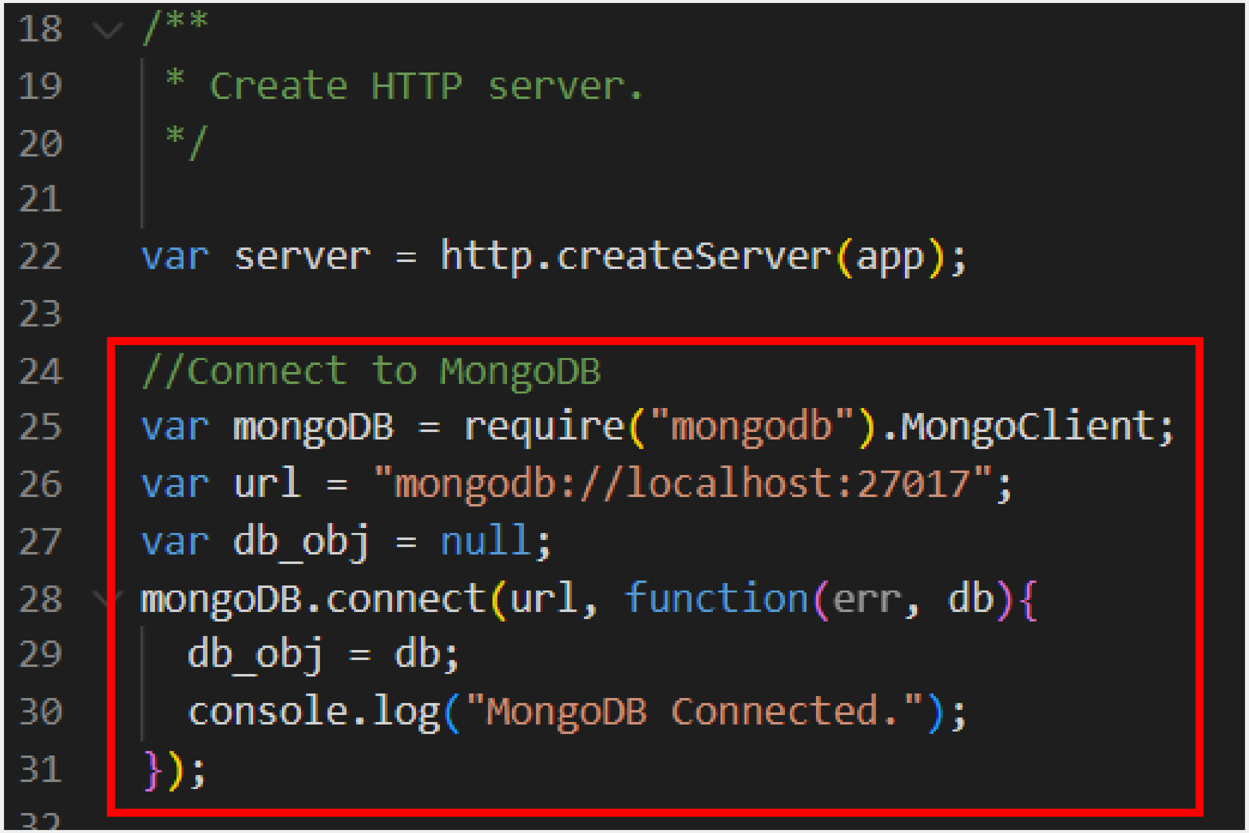
Database Name: Resources

Collection Name: Temperature

> Advanced Collection Options (e.g. Time-Series, Capped, Clustered collections)

Forwarding Resource Data to Database

- **Express Project의 www 수정**
  - MongoDB Server 접속



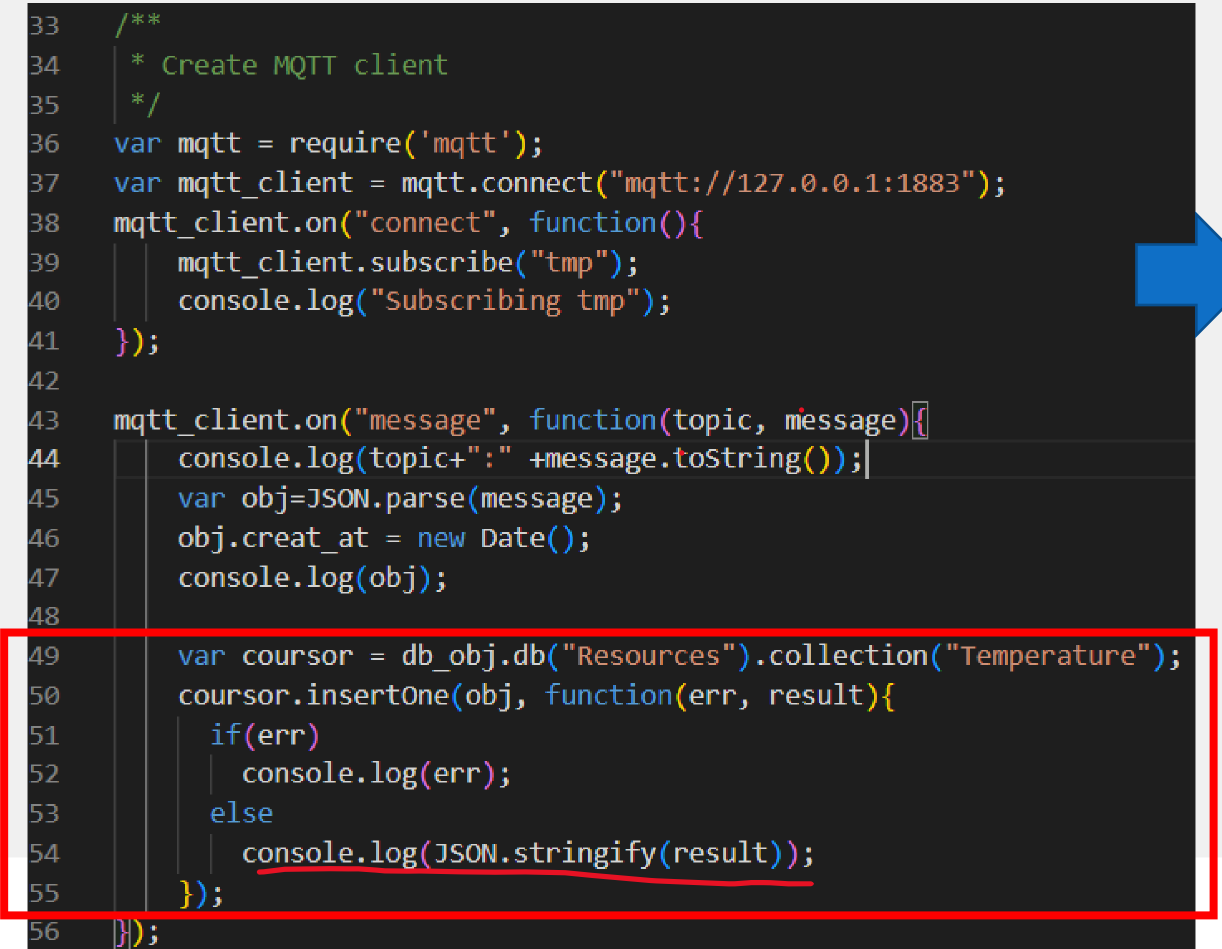
```
/**
 * Create HTTP server.
 */

var server = http.createServer(app);

//Connect to MongoDB
var mongoDB = require("mongodb").MongoClient;
var url = "mongodb://localhost:27017";
var db_obj = null;
mongoDB.connect(url, function(err, db){
  db_obj = db;
  console.log("MongoDB Connected.");
});
```

Forwarding Resource Data to Database

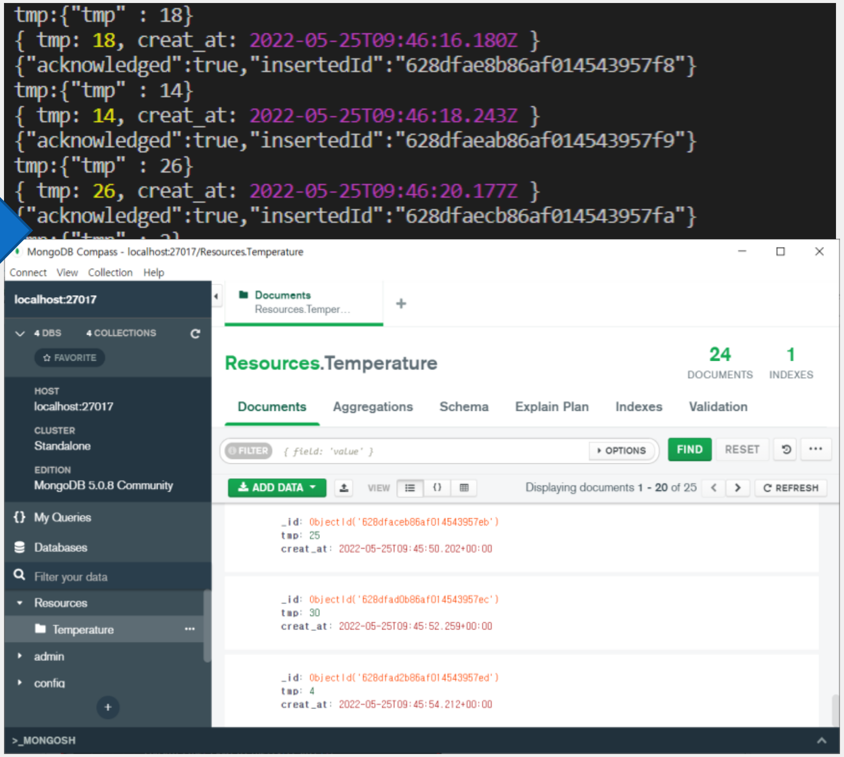
- **Express Project의 www 수정**
  - MQTT를 통해 전달받은 Resource Data를 MongoDB Server에 저장
    - ✓ 저번주에 작성한 “simple_publisher.bat” 및 Node.js 서버 (명령어 → npm start) 실행
    - ✓ DB와 Collection 명의 대소문자 주의! **(e.g., Resources O, resources X)**



```
/**
 * Create MQTT client
 */
var mqtt = require('mqtt');
var mqtt_client = mqtt.connect("mqtt://127.0.0.1:1883");
mqtt_client.on("connect", function(){
    mqtt_client.subscribe("tmp");
    console.log("Subscribing tmp");
});

mqtt_client.on("message", function(topic, message){
    console.log(topic+":" +message.toString());
    var obj=JSON.parse(message);
    obj.creat_at = new Date();
    console.log(obj);

    var coursor = db_obj.db("Resources").collection("Temperature");
    coursor.insertOne(obj, function(err, result){
      if(err)
        console.log(err);
      else
        console.log(JSON.stringify(result));
    });
});
```



```
tmp:{"tmp" : 18}
{ tmp: 18, creat_at: 2022-05-25T09:46:16.180Z }
{"acknowledged":true,"insertedId":"628dfae8b86af014543957f8"}
tmp:{"tmp" : 14}
{ tmp: 14, creat_at: 2022-05-25T09:46:18.243Z }
{"acknowledged":true,"insertedId":"628dfaeab86af014543957f9"}
tmp:{"tmp" : 26}
{ tmp: 26, creat_at: 2022-05-25T09:46:20.177Z }
{"acknowledged":true,"insertedId":"628dfaecb86af014543957fa"}
```

MongoDB Compass - localhost:27017/Resources.Temperature

HOST: localhost:27017

CLUSTER: Standalone

EDITION: MongoDB 5.0.8 Community

Resources.Temperature

24 DOCUMENTS 1 INDEXES

Displaying documents 1 - 20 of 25

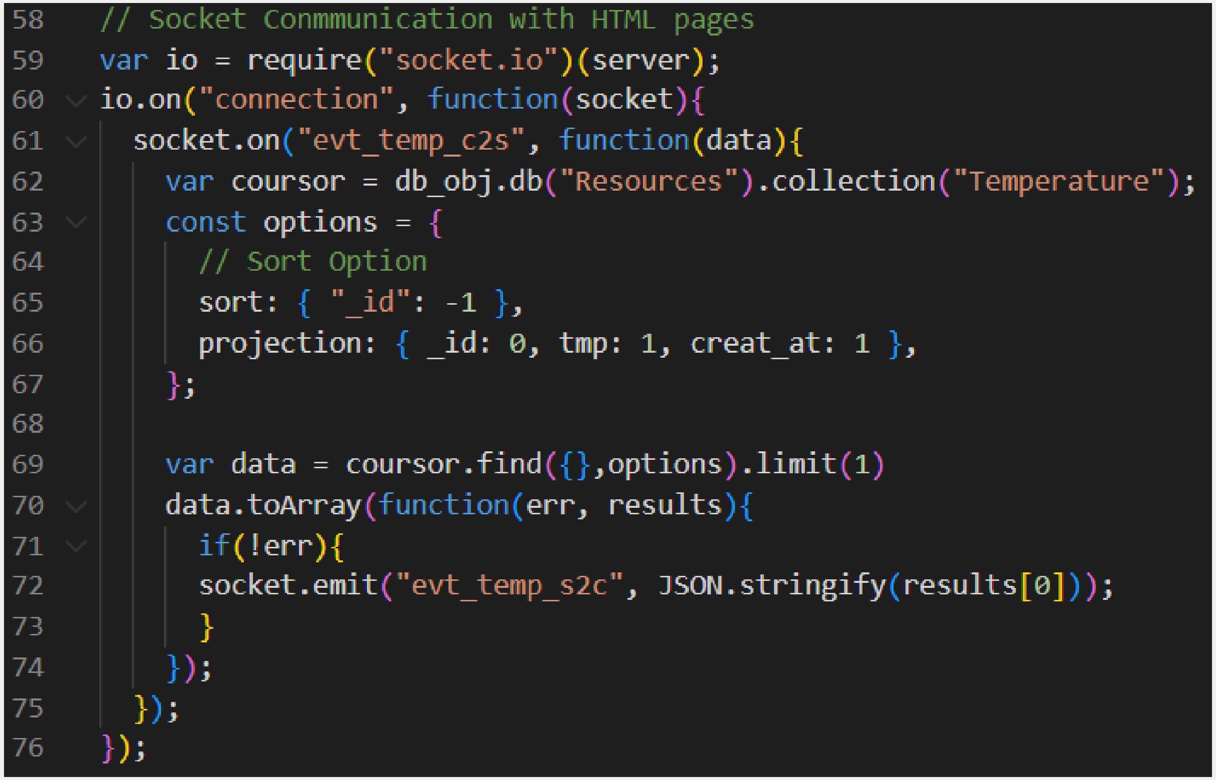
```
_id: ObjectId('628dfaceb86af014543957eb')
tmp: 25
creat_at: 2022-05-25T09:45:50.202+00:00

_id: ObjectId('628dfad0b86af014543957ec')
tmp: 30
creat_at: 2022-05-25T09:45:52.259+00:00

_id: ObjectId('628dfad2b86af014543957ed')
tmp: 4
creat_at: 2022-05-25T09:45:54.212+00:00
```

Web UI for Resource Monitoring

- **Express Project의 www 수정**
  - www에 아래 구문 추가
    - ✓ 소켓 선언 → MongoDB에서 최신 data 읽어오기 → 소켓을 통해 HTML page로 data 전송

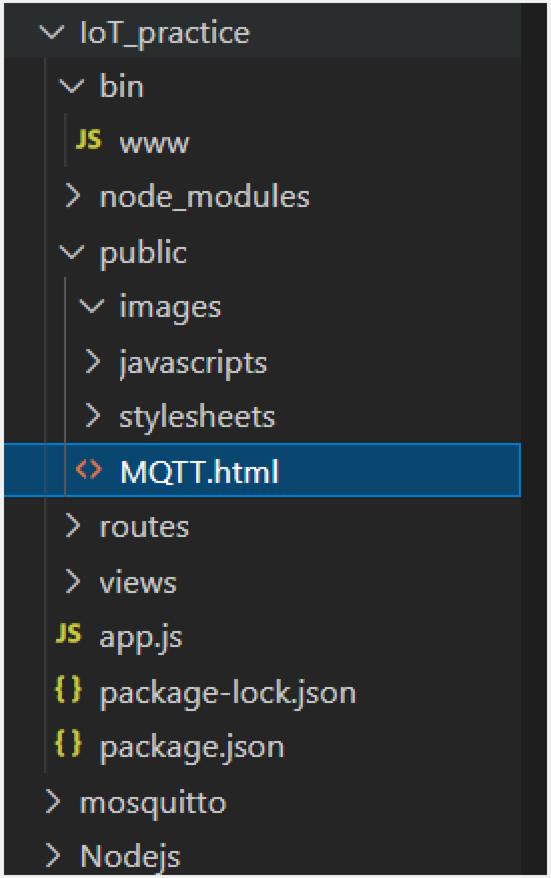


```
// Socket Conmmunication with HTML pages
var io = require("socket.io")(server);
io.on("connection", function(socket){
  socket.on("evt_temp_c2s", function(data){
    var coursor = db_obj.db("Resources").collection("Temperature");
    const options = {
      // Sort Option
      sort: { "_id": -1 },
      projection: { _id: 0, tmp: 1, creat_at: 1 },
    };

    var data = coursor.find({},options).limit(1)
    data.toArray(function(err, results){
      if(!err){
      socket.emit("evt_temp_s2c", JSON.stringify(results[0]));
      }
    });
  });
});
```

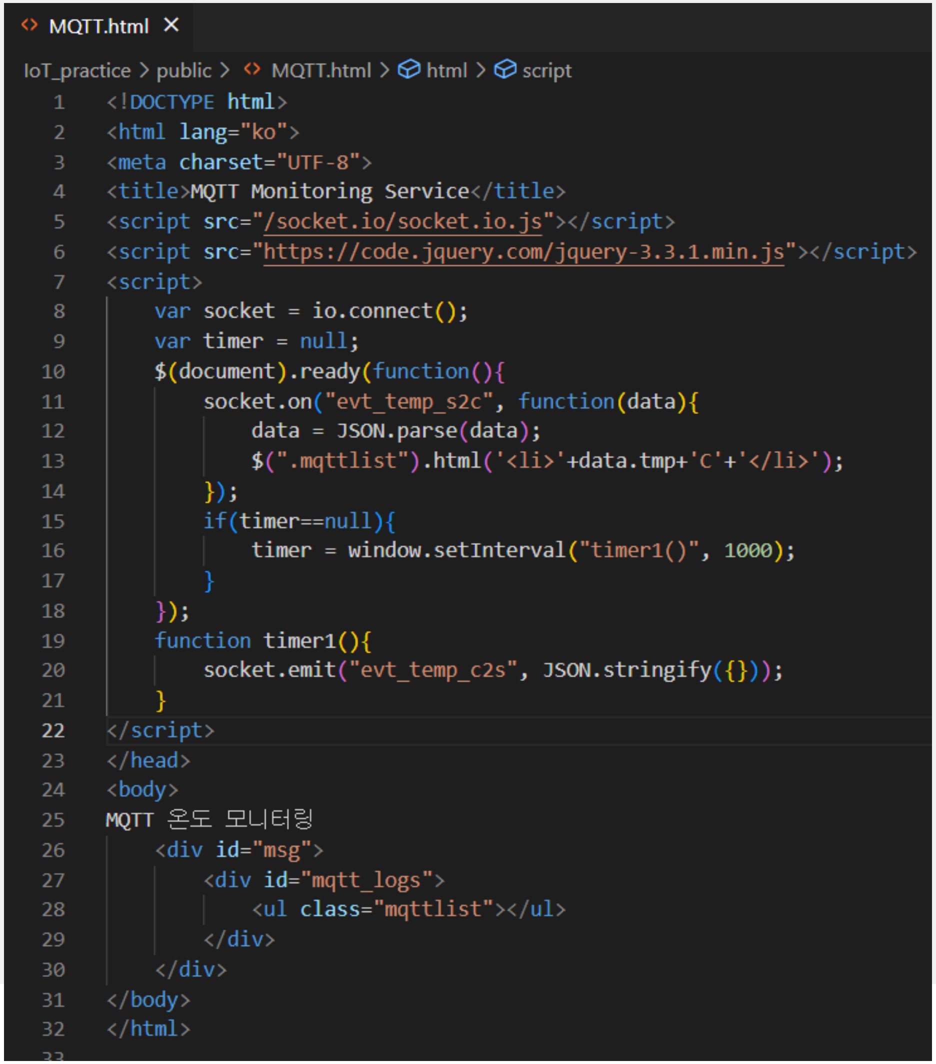
Web UI for Resource Monitoring

- **MQTT.html 생성 및 작성**
  - Express 프로젝트 Public 폴더에 MQTT.html 파일 새로 생성



Web UI for Resource Monitoring

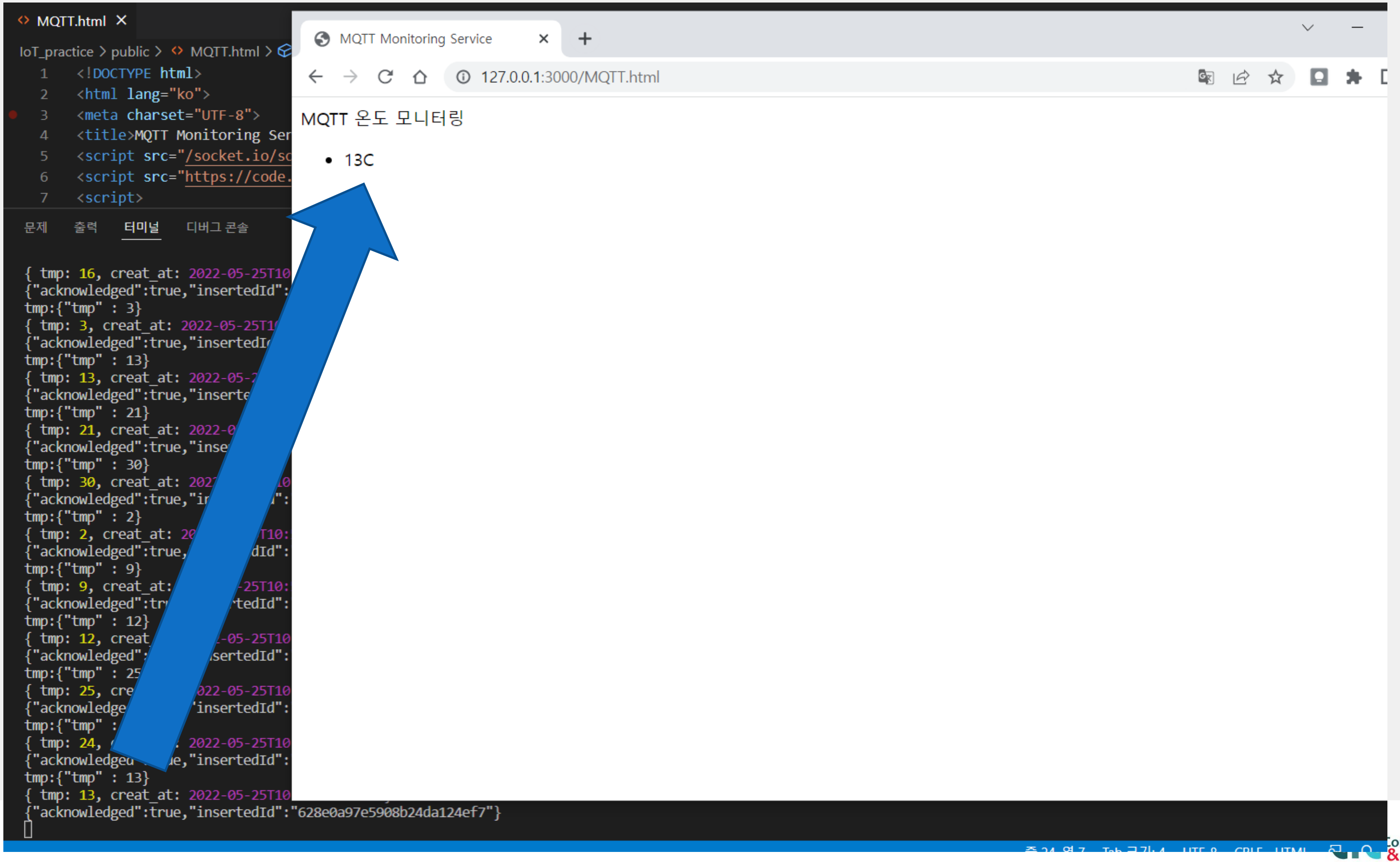
- **MQTT.html 생성 및 작성**
  - MQTT.html 파일 다음과 같이 작성



```html
<!DOCTYPE html>
<html lang="ko">
<meta charset="UTF-8">
<title>MQTT Monitoring Service</title>
<script src="/socket.io/socket.io.js"></script>
<script src="https://code.jquery.com/jquery-3.3.1.min.js"></script>
<script>
    var socket = io.connect();
    var timer = null;
    $(document).ready(function(){
        socket.on("evt_temp_s2c", function(data){
            data = JSON.parse(data);
            $(".mqttlist").html('<li>'+data.tmp+'C'+'</li>');
        });
        if(timer==null){
            timer = window.setInterval("timer1()", 1000);
        }
    });
    function timer1(){
        socket.emit("evt_temp_c2s", JSON.stringify({}));
    }
</script>
</head>
<body>
MQTT 온도 모니터링
    <div id="msg">
        <div id="mqtt_logs">
            <ul class="mqttlist"></ul>
        </div>
    </div>
</body>
</html>
```

Web UI for Resource Monitoring

- **실습결과 확인**
  - http://127.0.0.1:3000/MQTT.html



```
<!DOCTYPE html>
<html lang="ko">
<meta charset="UTF-8">
<title>MQTT Monitoring Ser
<script src="/socket.io/so
<script src="https://code.
<script>
```

```
{ tmp: 16, creat_at: 2022-05-25T10
{"acknowledged":true,"insertedId":
tmp:{"tmp" : 3}
{ tmp: 3, creat_at: 2022-05-25T1
{"acknowledged":true,"insertedI
tmp:{"tmp" : 13}
{ tmp: 13, creat_at: 2022-05-2
{"acknowledged":true,"inserte
tmp:{"tmp" : 21}
{ tmp: 21, creat_at: 2022-0
{"acknowledged":true,"inse
tmp:{"tmp" : 30}
{ tmp: 30, creat_at: 202         10
{"acknowledged":true,"ir        d":
tmp:{"tmp" : 2}
{ tmp: 2, creat_at: 20         T10:
{"acknowledged":true,        dId":
tmp:{"tmp" : 9}
{ tmp: 9, creat_at:        -25T10:
{"acknowledged":tr        tedId":
tmp:{"tmp" : 12}
{ tmp: 12, creat        -05-25T10
{"acknowledged"        sertedId":
tmp:{"tmp" : 25
{ tmp: 25, cre        022-05-25T10
{"acknowledge        "insertedId":
tmp:{"tmp" :
{ tmp: 24,        : 2022-05-25T10
{"acknowledged        ue,"insertedId":
tmp:{"tmp" : 13}
{ tmp: 13, creat_at: 2022-05-25T10
{"acknowledged":true,"insertedId":"628e0a97e5908b24da124ef7"}
```

MQTT 온도 모니터링

- 13C