텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

**dbConnect.js**

const mongooes = require('mongoose');

require('dotenv').config();

// async, await

const dbConnect = async() => {

    try {

        const connect = await mongooes.connect(process.env.DB\_CONNECT);

        console.log('DB Connected');

    } catch(err) {

        console.log(err);

    }

}

module.exports = dbConnect;

**Images.js**

const mongoose = require('mongoose');

// 스키마 정의

const ImageSchema = new mongoose.Schema({

    // 게시물 제목

    title: {

        type: String,

        required: true,

    },

    // 게시물 내용

    body: {

        type: String,

        required: true,

    },

    filename: {

        type: String,

        required: true,

    },

    path: {

        type: String,

        require: true,

    },

    // 게시물 작성일

    createdAt: {

        type: Date,

        default: Date.now(),

    },

});

module.exports = mongoose.model('Image', ImageSchema);

**server.js**

require('dotenv').config({ path: './path/to/.env' });

const express = require('express');

const multer = require('multer');

const path = require('path');

const fs = require('fs');

const mongoose = require('mongoose');

const Image = require('./models/Images');

const dbConnect = require('./config/dbConnect');

dbConnect();

// Express 앱 초기화

const app = express();

const port = 3000;

// EJS 템플릿 엔진 설정

app.set('view engine', 'ejs');

//EJS 파일 위치 지정

app.set('views', path.join(\_\_dirname, 'views'));

// 업로드 폴더 확인 후 생성

const uploadDir = path.join(\_\_dirname, 'uploads');

if (!fs.existsSync(uploadDir)) {

    fs.mkdirSync(uploadDir);

}

// Multer 저장 설정 (디스크 저장 방식)

const storage = multer.diskStorage({

    destination: function (req, file, cb) { // callback

        cb(null, uploadDir);

    },

    filename: function(req, file, cb) {

        cb(null, Date.now() + path.extname(file.originalname));

        // 파일 이름을 현재 시간과 확장사로 지정

    }

});

// Multer 미들웨어 설정

const upload = multer({ storage: storage });

// 업로드된 파일을 정적 파일로 제공

app.use('/uploads', express.static(uploadDir));

// 파일 업로드 폼을 EJS 템플릿으로 렌더링

app.get('/', (req, res) => {

    // views/index.ejs 파일을 렌더링

    res.render('index');

});

// 파일 업로드를 처리하는 라우터

app.post('/upload', upload.single('file'), (req, res) => {

    if (!req.file) {

        return res.status(400).send('파일이 업로드 되지 않았습니다.');

    }

    const { title, body } = req.body;

    console.log('파일 정보:', req.file);

    console.log('제목:', title);

    console.log('내용:', body);

       // MongoDB

   const newImage = new Image({

    title: title,

    body: body,

    filename: req.file.filename,

    path: path.join('uploads', req.file.filename),

 });

 newImage.save()

    .then(() => {

       console.log('파일 업로드 완료');

       // EJS에 파일과 제목, 내용을 모두 전달

       res.render('index', { uploadedFile: { title, body, filename: req.file.filename, path: req.file.path } });

    })

    .catch((err) => {

       console.error('오류:', err);

       res.status(500).send('파일 업로드 오류');

    });

});

// 서버 시작

app.listen(port, () => {

    console.log(`서버가 http://localhost:${port}에서 실행 중입니다.`)

});

**Index.ejs**

<!DOCTYPE html>

<html lang="ko">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>파일 업로드</title>

</head>

<body>

    <h1>파일 업로드</h1>

    <!-- 업로드 폼 -->

     <form action="/upload" method="post" enctype="multipart/form-data">

        <label for="file">파일선택:</label>

        <input type="file" id="file" name="file" required />

        <label for="title">제목:</label>

        <input type="text" id="title" name="title" placeholder="제목">

        <label for="body">내용:</label>

        <textarea name="body" id="body" placeholder="내용"></textarea>

        <button type="submit">업로드</button>

     </form>

     <br>

     <!-- 업로드된 파일 정보가 있을 경우 표시 -->

      <% if (typeof uploadedFile !== 'undefined') { %>

        <h2>업로드된 파일:</h2>

        <p>제목: <%= uploadedFile.title %></p>

        <p>내용: <%= uploadedFile.body %></p>

        <p>파일명: <%= uploadedFile.filename %></p>

        <p>파일 경로: <%= uploadedFile.path %></p>

        <!-- 이미지 미리보기 -->

         <h3>업로드된 이미지:</h3>

         <img src="/uploads/<%= uploadedFile.filename %>" alt="Uploaded Image" style="max-width: 500px;">

    <% } %>

</body>

</html>