一、实验01（不含富文本）

打开终端

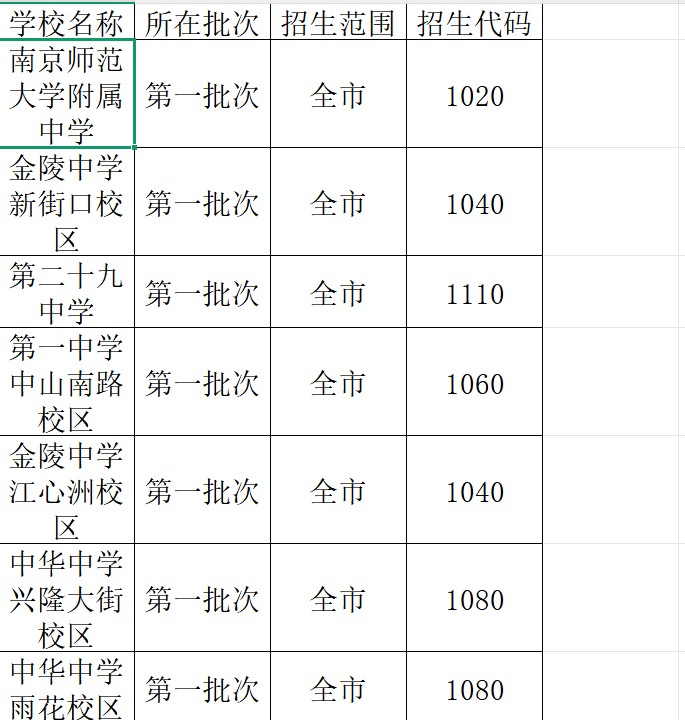
输入npm install express body-parser mysql2安装项目依赖

输入node server.js启动node.js服务器

打开浏览器，访问<http://localhost:3000。>

！！！注意：excel表格中的列名与顺序要和数据库中的完全相同一一对应。数据库中的“创建时间”列会在导入时自动创建，不用提前建好。





附代码：

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Excel to Database</title>

</head>

<body>

<h1>导入Excel到数据库</h1>

<input type="file" id="excelFile" accept=".xlsx, .xls" style="display: none;" />

<button onclick="document.getElementById('excelFile').click()">导入Excel</button>

<script src="https://cdnjs.cloudflare.com/ajax/libs/xlsx/0.16.9/xlsx.full.min.js"></script>

<script>

document.getElementById('excelFile').addEventListener('change', function(event) {

const file = event.target.files[0];

if (file) {

const reader = new FileReader();

reader.onload = function(e) {

const data = new Uint8Array(e.target.result);

const workbook = XLSX.read(data, {type: 'array'});

const firstSheetName = workbook.SheetNames[0];

const worksheet = workbook.Sheets[firstSheetName];

const json = XLSX.utils.sheet\_to\_json(worksheet, {header: 1});

// 发送数据到后端

fetch('/upload', {

method: 'POST',

headers: {

'Content-Type': 'application/json',

},

body: JSON.stringify(json),

})

.then(response => response.json())

.then(data => {

console.log('Success:', data);

alert('导入成功！');

})

.catch((error) => {

console.error('Error:', error);

alert('导入失败！');

});

};

reader.readAsArrayBuffer(file);

} else {

alert('请选择一个Excel文件');

}

});

</script>

</body>

</html>

Server.js

const express = require('express');

const bodyParser = require('body-parser');

const mysql = require('mysql2');

const app = express();

const path = require('path');

app.use(bodyParser.json());

// 创建数据库连接

const connection = mysql.createConnection({

host: 'localhost',

user: 'root',

password: '你的密码',

database: 'school'

});

connection.connect();

// 处理上传的Excel数据

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

const data = req.body;

// 假设Excel的第一行是列名

const columns = data[0];

const rows = data.slice(1);

// 获取本地时间并格式化为 MySQL 的 DATETIME 格式

const getLocalTime = () => {

const now = new Date();

const year = now.getFullYear();

const month = String(now.getMonth() + 1).padStart(2, '0'); // 月份从 0 开始，需要加 1

const day = String(now.getDate()).padStart(2, '0');

const hours = String(now.getHours()).padStart(2, '0');

const minutes = String(now.getMinutes()).padStart(2, '0');

const seconds = String(now.getSeconds()).padStart(2, '0');

return `${year}-${month}-${day} ${hours}:${minutes}:${seconds}`;

};

const currentTime = getLocalTime(); // 获取本地时间

columns.push('创建时间'); // 添加时间字段名

rows.forEach(row => row.push(currentTime)); // 为每一行数据添加当前时间

// 构建SQL插入语句

const tableName = '你的数据表名';

const sql = `INSERT INTO ${tableName} (${columns.join(', ')}) VALUES ?`;

connection.query(sql, [rows], (error, results) => {

if (error) {

console.error('Error inserting data:', error);

res.status(500).json({

//success: false,

status:'error',

message: 'Error inserting data',

error: error.message

});

} else {

console.log('Data inserted successfully');

res.status(200).json({

//success: true,

status:'success',

message: 'Data inserted successfully',

results: results

});

}

});

});

// 设置静态文件目录（假设你的 HTML 文件在 "public" 文件夹中）

app.use(express.static(path.join(\_\_dirname, 'public')));

// 设置根路由

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

res.sendFile(path.join(\_\_dirname, 'public', 'index.html'));

});

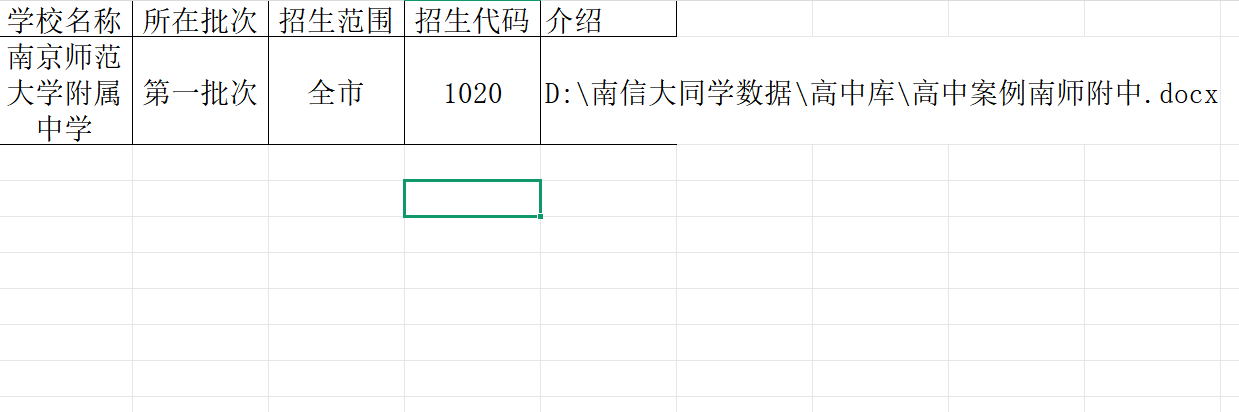
app.listen(3000, () => {

console.log('Server is running on port 3000');

});

1. 实验02（含富文本）

在excel表格中新增一列存放富文本的word文件路径



同样，也在数据库新增一列，我用的字段类型是mediumtext。

在终端输入npm install multer mammoth mysql2安装依赖

输入node server.js启动服务器，以下内容同实验01.

附代码：

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Excel to Database</title>

</head>

<body>

<h1>导入Excel到数据库</h1>

<input type="file" id="excelFile" accept=".xlsx, .xls" style="display: none;" />

<button onclick="document.getElementById('excelFile').click()">导入Excel</button>

<script src="https://cdnjs.cloudflare.com/ajax/libs/xlsx/0.16.9/xlsx.full.min.js"></script>

<script>

document.getElementById('excelFile').addEventListener('change', function(event) {

const file = event.target.files[0];

if (file) {

const reader = new FileReader();

reader.onload = function(e) {

const data = new Uint8Array(e.target.result);

const workbook = XLSX.read(data, {type: 'array'});

const firstSheetName = workbook.SheetNames[0];

const worksheet = workbook.Sheets[firstSheetName];

const json = XLSX.utils.sheet\_to\_json(worksheet, {header: 1});

// 发送数据到后端

fetch('/upload', {

method: 'POST',

headers: {

'Content-Type': 'application/json',

},

body: JSON.stringify(json),

})

.then(response => response.json())

.then(data => {

console.log('Success:', data);

alert('导入成功！');

})

.catch((error) => {

console.error('Error:', error);

alert('导入失败！');

});

};

reader.readAsArrayBuffer(file);

} else {

alert('请选择一个Excel文件');

}

});

</script>

</body>

</html>

Server.js

const express = require('express');

const bodyParser = require('body-parser');

const mysql = require('mysql2');

const mammoth = require('mammoth');

const fs = require('fs');

const path = require('path');

const app = express();

app.use(bodyParser.json());

// 创建数据库连接

const connection = mysql.createConnection({

host: 'localhost',

user: 'root',

password: '你的密码',

database: 'school'

});

connection.connect();

// 处理上传的Excel数据

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

const data = req.body;

// 假设Excel的第一行是列名

const columns = data[0];

const rows = data.slice(1);

// 获取本地时间并格式化为 MySQL 的 DATETIME 格式

const getLocalTime = () => {

const now = new Date();

const year = now.getFullYear();

const month = String(now.getMonth() + 1).padStart(2, '0'); // 月份从 0 开始，需要加 1

const day = String(now.getDate()).padStart(2, '0');

const hours = String(now.getHours()).padStart(2, '0');

const minutes = String(now.getMinutes()).padStart(2, '0');

const seconds = String(now.getSeconds()).padStart(2, '0');

return `${year}-${month}-${day} ${hours}:${minutes}:${seconds}`;

};

const currentTime = getLocalTime(); // 获取本地时间

columns.push('创建时间'); // 添加时间字段名

rows.forEach(row => row.push(currentTime)); // 为每一行数据添加当前时间

// 找到“介绍”列的索引

const introductionIndex = columns.indexOf('介绍');

// 遍历每一行数据

for (let i = 0; i < rows.length; i++) {

const row = rows[i];

const wordFilePath = row[introductionIndex];

if (wordFilePath) {

try {

// 读取Word文档内容并转换为HTML

const result = await mammoth.convertToHtml({ path: wordFilePath });

const htmlContent = result.value;

// 将HTML内容替换到当前行的“介绍”列

row[introductionIndex] = htmlContent;

} catch (error) {

console.error('Error reading Word file:', error);

row[introductionIndex] = ''; // 如果读取失败，设置为空字符串

}

}

}

// 构建SQL插入语句

const tableName = '数据表名'; // 填写你的数据表名

const sql = `INSERT INTO ${tableName} (${columns.join(', ')}) VALUES ?`;

connection.query(sql, [rows], (error, results) => {

if (error) {

console.error('Error inserting data:', error);

// 返回JSON格式的错误响应

res.status(500).json({

status: 'error',

message: 'Error inserting data into database',

error: error.message

});

} else {

console.log('Data inserted successfully');

// 返回JSON格式的成功响应

res.status(200).json({

status: 'success',

message: 'Data inserted successfully',

insertedRows: results.affectedRows

});

}

});

});

// 设置静态文件目录（假设你的 HTML 文件在 "public" 文件夹中）

app.use(express.static(path.join(\_\_dirname, 'public')));

// 设置根路由

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

res.sendFile(path.join(\_\_dirname, 'public', 'index.html'));

});

app.listen(3000, () => {

console.log('Server is running on port 3000');

});