Linus Koh Zhijie
1006011

Task 2: Count Staff using MySQL

Step 1: Set up the MySQL database:

- Create a MySQL database named CountStaff.
- Create tables for departments and staffs.

Enter the following SQL code:

```
CREATE TABLE departments (
  id INT AUTO_INCREMENT PRIMARY KEY,
  code VARCHAR(255) NOT NULL
);

CREATE TABLE staffs (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(255) NOT NULL,
  dept VARCHAR(255) NOT NULL,
  FOREIGN KEY (dept) REFERENCES departments(code)
);
```

Step 2: Install required packages:

Enter the following command in the terminal of visual studio code:

```
npm install mysql2
```

Step 3: Connect to the MySQL database in db.js:

Enter the following JavaScript code:

```
const mysql = require('mysql2');

// Define the MySQL connection
const connection = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: 'password',
  database: 'CountStaff'
```

```
});

// Connect to MySQL
connection.connect(error => {
  if (error) throw error;
  console.log('Connected to the MySQL database');
});

module.exports = connection;
```

Step 4: Set up the route to count staff by department in routes/departments.js

Enter the following JavaScript code:

```
const express = require('express');
const router = express.Router();
const connection = require('../db');

// Endpoint to get the count of staff members by department
router.get('/count', (req, res) => {
  const query = `
    SELECT dept, COUNT(*) as count
    FROM staffs
    GROUP BY dept
  `;

  connection.query(query, (error, results) => {
    if (error) throw error;
    res.json(results);
  });
});

module.exports = router;
```

Step 5: Test the API

- Start the Express.js server.
- Use Postman or a web browser to test the endpoints.