

Quá trình hoàn thiện bài tập.

Họ và tên: Ma Bách Duy

Mssv: K205480106010

1. Quá trình cài đặt và môi trường

- Firstly, sử dụng anaconda để chạy visual studio code để sử dụng không gian riêng quản lý bởi anaconda
- Secondly, trong terminal của workspace cài đặt nodeJS và fastapi
<https://fastapi.tiangolo.com/#installation>
<https://nodejs.org/en/download/current>

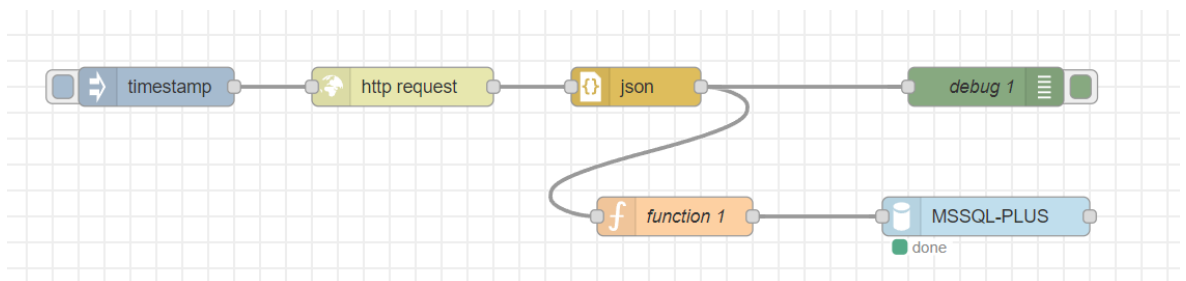
để cài đặt node-red: yêu cầu cài nodejs trước để kích hoạt npm sau đó:
npm install -g --unsafe-perm node-red

- Finally, cài XAMPP

2. Bắt đầu

- Input: Thông tin thô từ https://home.openweathermap.org/api_keys
- Output: Biểu đồ thông tin
- Quá trình tìm hiểu và nghiên cứu có thời gian khá ngắn nên không hiểu chi tiết về từng code và công dụng nên phải hỏi chatGPT khá nhiều.
- Quy trình bài toán sẽ là: Python lấy dữ liệu thô từ weatherapi về -> sử dụng fastapi để gửi dữ liệu lên <http://127.0.0.1:8000/weather> -> node-red lấy dữ liệu và xử lý + insert vào db.

```
127.0.0.1:8000/weather
[{"Date": "2024-05-14", "Temperature": 26.95, "Humidity": 85, "Pressure": 1013, "City": "Thai Nguyen"}, {"Date": "2024-05-15", "Temperature": 25.84, "Humidity": 86, "Pressure": 1014, "City": "Thai Nguyen"}, {"Date": "2024-05-16", "Temperature": 23.82, "Humidity": 92, "Pressure": 1015, "City": "Thai Nguyen"}, {"Date": "2024-05-17", "Temperature": 23.95, "Humidity": 88, "Pressure": 1012, "City": "Thai Nguyen"}, {"Date": "2024-05-18", "Temperature": 24.23, "Humidity": 97, "Pressure": 1008, "City": "Thai Nguyen"}, {"Date": "2024-05-19", "Temperature": 22.84, "Humidity": 98, "Pressure": 1007, "City": "Thai Nguyen"}]
```

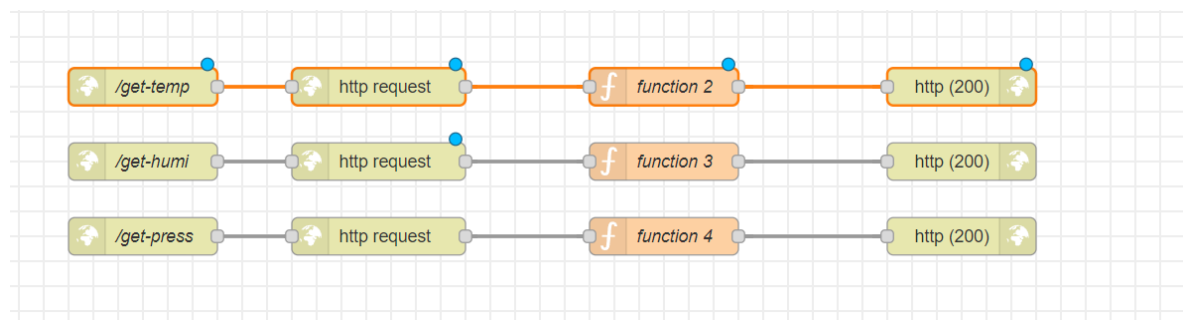


Từ db lấy dữ liệu ra bằng php api bằng cách gọi sp_getthoietdata

```
thaycop > getdata.php
1 <?php
2 // Kết nối đến cơ sở dữ liệu
3 $serverName = "DUYVPRO";
4 $connectionOptions = array(
5     "Database" => "thaycop",
6     "Uid" => "sa",
7     "PWD" => "makaeenm1"
8 );
9 $conn = sqlsrv_connect($serverName, $connectionOptions);
10
11 // Kiểm tra kết nối
12 if ($conn === false) {
13     die(print_r(sqlsrv_errors(), true));
14 }
15
16 // Gọi stored procedure
17 $sql = "{CALL GetThoiTietData}";
18 $stmt = sqlsrv_query($conn, $sql);
19
20 // Kiểm tra và xử lý kết quả
21 if ($stmt === false) {
22     die(print_r(sqlsrv_errors(), true));
23 } else {
24     $result = array();
25     while ($row = sqlsrv_fetch_array($stmt, SQLSRV_FETCH_ASSOC)) {
26         $result[] = $row;
27     }
28
29     // Trả về kết quả dưới dạng JSON
```

```
CREATE PROCEDURE GetThoiTietData
AS
BEGIN
    SELECT * FROM thoi_tiet;
END
```

Sau khi gọi api thì sử dụng http request lấy thông tin và đưa vào endpoint



<http://127.0.0.1:1880/get-temp>

<http://127.0.0.1:1880/get-humi>

<http://127.0.0.1:1880/get-press>

Cuối cùng dùng js để lấy dữ liệu và vẽ biểu đồ

```
// Hàm để vẽ biểu đồ nhiệt độ
function drawTemperatureChart() {
  loadData('http://127.0.0.1:1880/get-temp', function(data) {
    // Tạo mảng dữ liệu cho biểu đồ
    var chartData = [];
    chartData.push(['Date', 'Temperature']); // Thêm tiêu đề cột

    // Lặp qua dữ liệu JSON và thêm vào mảng dữ liệu
    data.forEach(function(entry) {
      chartData.push([new Date(entry.Date), entry.Temperature]);
    });

    // Tạo DataTable từ mảng dữ liệu
    var dataTable = google.visualization.arrayToDataTable(chartData);

    // Cấu hình tùy chọn cho biểu đồ
    var options = {
      title: 'Temperature Chart',
      curveType: 'function',
      legend: { position: 'bottom' }
    };

    // Vẽ biểu đồ
    var chart = new google.visualization.LineChart(document.getElementById('temperature_chart'));
    chart.draw(dataTable, options);
  });
}
```

- Nhớ chạy node-red trong terminal để có thể mở palette, và fastapi để cập nhật dữ liệu

3. Kết quả

1. SQL

Data:

thaycop.sql - DUYV...O.thaycop (sa (85))			SQLQuery1.sql - DU...O.thaycop (sa (72))*		
	City	Date	Temperature	Humidity	Pressure
▶	Thai Nguyen	2024-05-14	26,28	85	1013
	Thai Nguyen	2024-05-15	25,84	86	1014
	Thai Nguyen	2024-05-16	23,82	92	1015
	Thai Nguyen	2024-05-17	23,95	88	1012
	Thai Nguyen	2024-05-18	24,23	97	1008
	Thai Nguyen	2024-05-19	22,84	98	1007
*	NULL	NULL	NULL	NULL	NULL

Struct:

SP:

```
CREATE PROCEDURE GetThoiTietData
AS
BEGIN
    SELECT * FROM thoi_tiet;
END
```

Query:

```
use thaycop
go
create table thoi_tiet(
    City NVARCHAR(255),
    Date DATE,
    Temperature FLOAT,
    Humidity FLOAT,
    Pressure FLOAT,
)
```

2. Xử lý api và html, js

Html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Charts</title>
    <!-- Bao gồm thư viện Google Charts -->
```

```

    <script type="text/javascript"
src="https://www.gstatic.com/charts/loader.js"></script>
    <script src="draw_charts.js"></script>
</head>
<body>
    <h1>Vẽ biểu đồ thời tiết</h1>
    <!-- Đây là nơi để hiển thị các biểu đồ -->
    <div id="temperature_chart" style="width: 900px; height: 500px;"></div>
    <div id="humidity_chart" style="width: 900px; height: 500px;"></div>
    <div id="pressure_chart" style="width: 900px; height: 500px;"></div>
</body>
</html>

```

Getdata.php:

```

<?php
// Kết nối đến cơ sở dữ liệu
$serverName = "DUYVPRO";
$connectionOptions = array(
    "Database" => "thaycop",
    "Uid" => "sa",
    "PWD" => "makaeenm1"
);
$conn = sqlsrv_connect($serverName, $connectionOptions);

// Kiểm tra kết nối
if ($conn === false) {
    die(print_r(sqlsrv_errors(), true));
}

// Gọi stored procedure
$sql = "{CALL GetThoiTietData}";
$stmt = sqlsrv_query($conn, $sql);

// Kiểm tra và xử lý kết quả
if ($stmt === false) {
    die(print_r(sqlsrv_errors(), true));
} else {
    $result = array();
    while ($row = sqlsrv_fetch_array($stmt, SQLSRV_FETCH_ASSOC)) {
        $result[] = $row;
    }

    // Trả về kết quả dưới dạng JSON
    header('Content-Type: application/json');
    echo json_encode($result);
}

// Đóng kết nối

```

```
sqlsrv_free_stmt($stmt);
sqlsrv_close($conn);
?>
```

Draw_charts:

```
// Hàm để tải dữ liệu từ URL JSON

google.charts.load('current', {'packages':['corechart']});
google.charts.setOnLoadCallback(drawCharts);
function drawCharts() {
    drawTemperatureChart();
    drawHumidityChart();
    drawPressureChart();
}
function loadData(url, callback) {
    fetch(url)
        .then(response => response.json())
        .then(data => callback(data))
        .catch(error => console.error('Error loading data:', error));
}

// Hàm để vẽ biểu đồ nhiệt độ
function drawTemperatureChart() {
    loadData('http://127.0.0.1:1880/get-temp', function(data) {
        // Tạo mảng dữ liệu cho biểu đồ
        var chartData = [];
        chartData.push(['Date', 'Temperature']); // Thêm tiêu đề cột

        // Lặp qua dữ liệu JSON và thêm vào mảng dữ liệu
        data.forEach(function(entry) {
            chartData.push([new Date(entry.Date), entry.Temperature]);
        });

        // Tạo DataTable từ mảng dữ liệu
        var dataTable = google.visualization.arrayToDataTable(chartData);

        // Cấu hình tùy chọn cho biểu đồ
        var options = {
            title: 'Temperature Chart',
            curveType: 'function',
            legend: { position: 'bottom' }
        };

        // Vẽ biểu đồ
        var chart = new
google.visualization.LineChart(document.getElementById('temperature_chart'))
;
        chart.draw(dataTable, options);
    });
}
```

```

}

// Hàm để vẽ biểu đồ độ ẩm
// Hàm để vẽ biểu đồ độ ẩm
function drawHumidityChart() {
    loadData('http://127.0.0.1:1880/get-humi', function(data) {
        // Tạo mảng dữ liệu cho biểu đồ
        var chartData = [];
        chartData.push(['Time', 'Humidity']); // Thêm tiêu đề cột

        // Lặp qua dữ liệu JSON và thêm vào mảng dữ liệu
        data.forEach(function(entry) {
            chartData.push([new Date(entry.Date), entry.Humidity]);
        });

        // Tạo DataTable từ mảng dữ liệu
        var dataTable = google.visualization.arrayToDataTable(chartData);

        // Cấu hình tùy chọn cho biểu đồ
        var options = {
            title: 'Humidity Chart',
            curveType: 'function',
            legend: { position: 'bottom' }
        };

        // Vẽ biểu đồ
        var chart = new
google.visualization.LineChart(document.getElementById('humidity_chart'));
        chart.draw(dataTable, options);
    });
}

// Hàm để vẽ biểu đồ áp suất
function drawPressureChart() {
    loadData('http://127.0.0.1:1880/get-press', function(data) {
        // Tạo mảng dữ liệu cho biểu đồ
        var chartData = [];
        chartData.push(['Time', 'Pressure']); // Thêm tiêu đề cột

        // Lặp qua dữ liệu JSON và thêm vào mảng dữ liệu
        data.forEach(function(entry) {
            chartData.push([new Date(entry.Date), entry.Pressure]);
        });

        // Tạo DataTable từ mảng dữ liệu
        var dataTable = google.visualization.arrayToDataTable(chartData);

        // Cấu hình tùy chọn cho biểu đồ
        var options = {

```

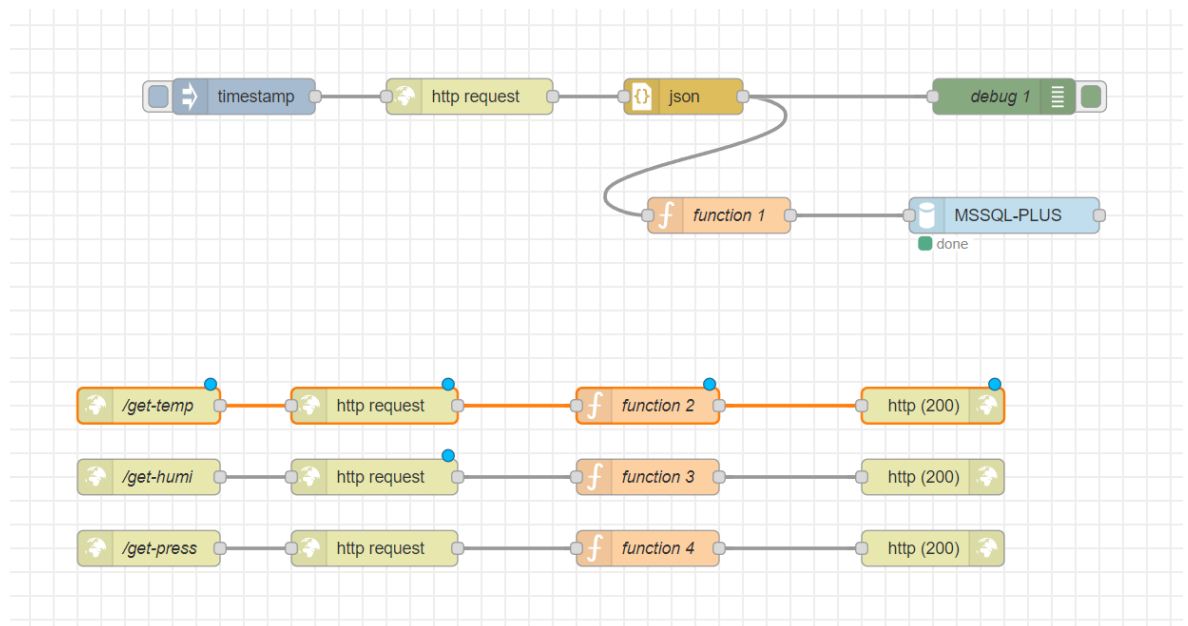
```

        title: 'Pressure Chart',
        curveType: 'function',
        legend: { position: 'bottom' }
    };

    // Vẽ biểu đồ
    var chart = new
google.visualization.LineChart(document.getElementById('pressure_chart'));
    chart.draw(dataTable, options);
    });
}

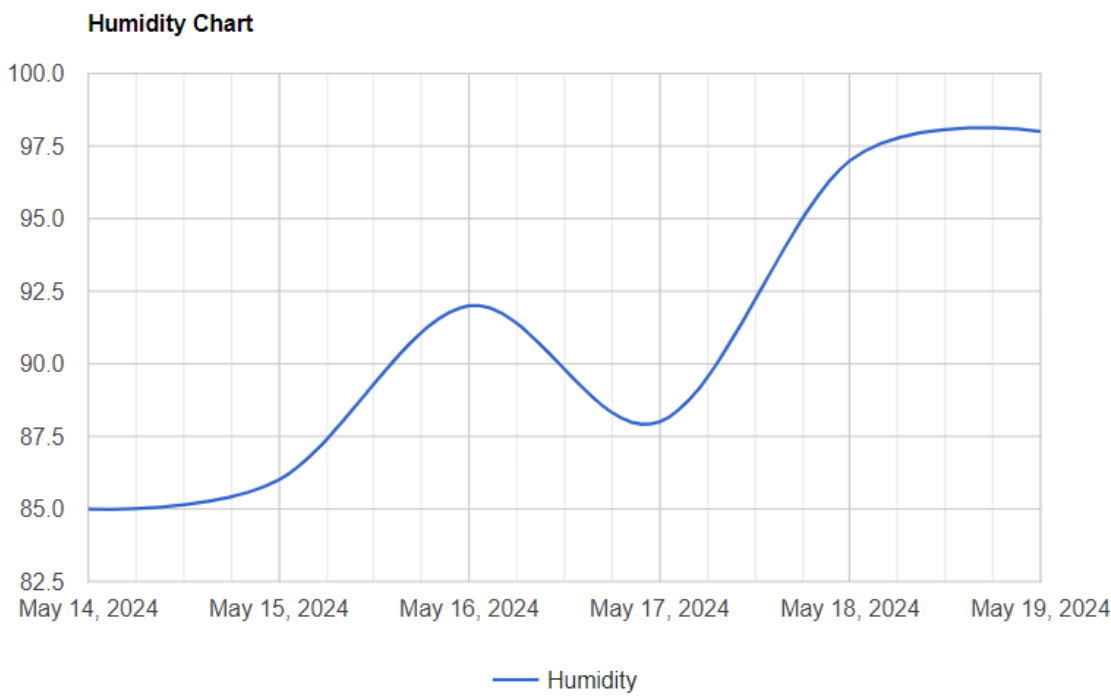
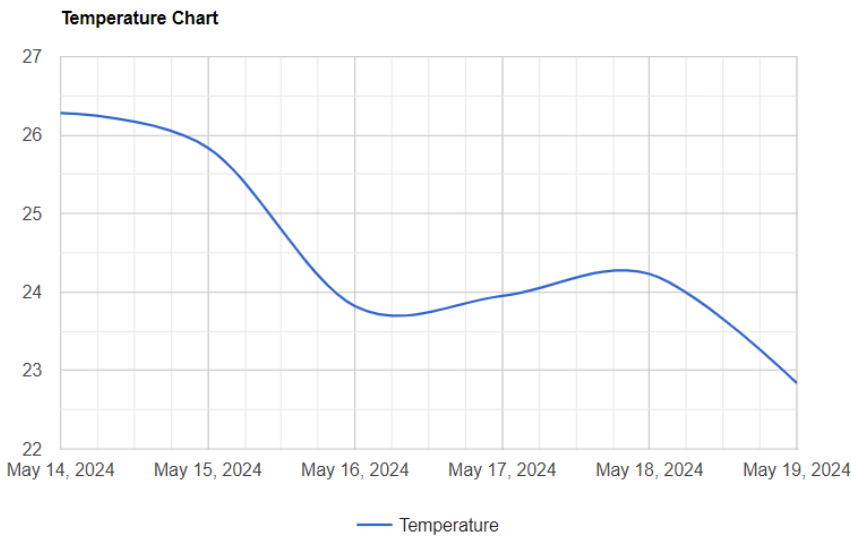
```

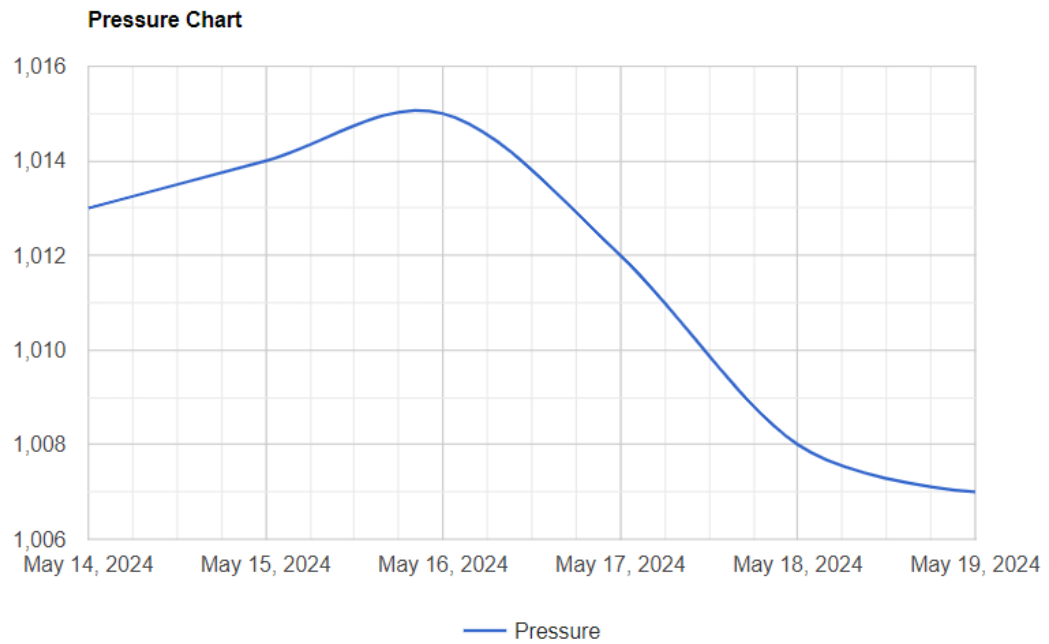
3. Node-red



4. Kết quả

Vẽ biểu đồ thời tiết





5. Kết luận

- Học được cách triển khai node-red, sử dụng json từ fastapi
- Học được kỹ năng: Chia thời gian chạy deadline, sửa code với chatGPT