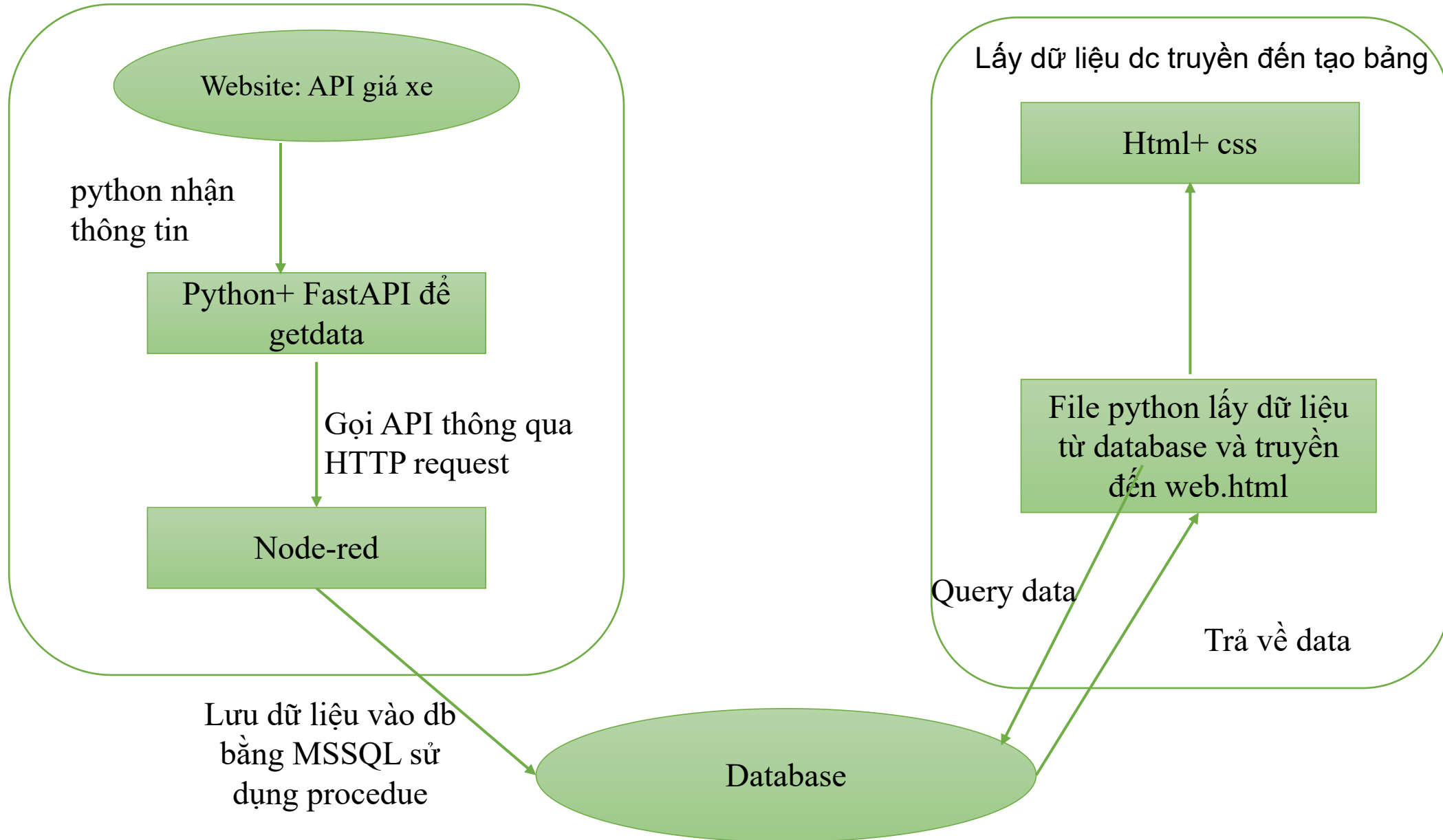


Sơ Đồ Tổng Quát



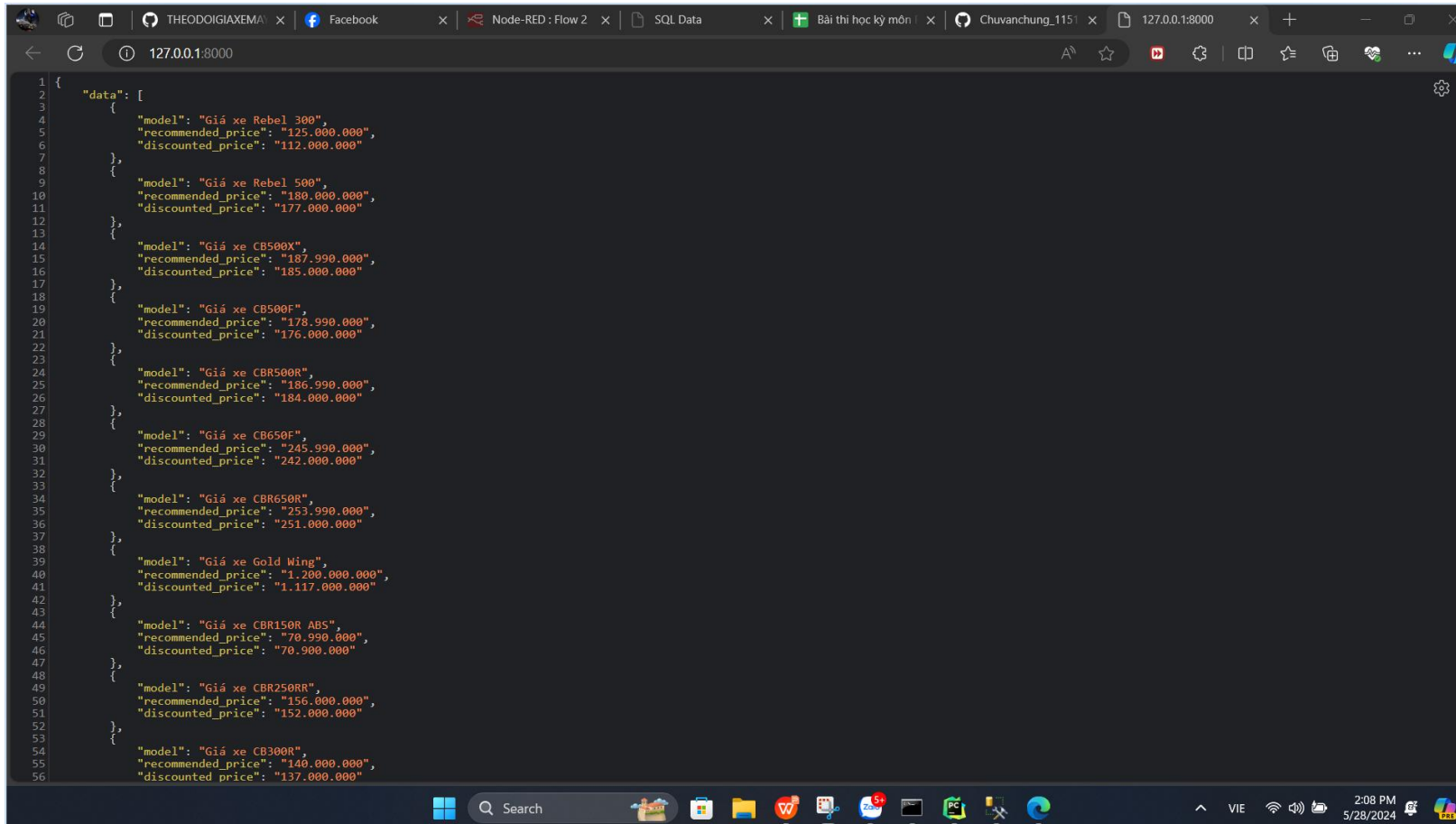
Bước 1: Viết 1 pyhton lấy dữ liệu từ API giá xe , tạo fastAPI gửi dữ liệu đi với địa chỉ localhost

```

12
13 @app.get('/')
14 def get_moto_prices():
15     try:
16         url = 'https://giaxe.2banh.vn/bang-gia-xe/bang-gia-xe-moto-119.html'
17         response = requests.get(url)
18         response.raise_for_status() # Kiểm tra xem yêu cầu có thành công không
19         soup = BeautifulSoup(response.content, features='html.parser')
20
21         # Trích xuất dữ liệu từ trang web
22         table = soup.find('table') # Giả sử bảng đầu tiên chứa dữ liệu mong muốn
23         if not table:
24             raise ValueError("Không tìm thấy bảng nào trong trang web.")
25
26         data = []
27         rows = table.find_all('tr')
28         for row in rows[1:]: # Bỏ qua hàng tiêu đề
29             cols = row.find_all('td')
30             if len(cols) >= 3: # Kiểm tra nếu hàng có ít nhất 3 cột
31                 model = cols[0].get_text(strip=True)
32                 if model in desired_models: # Chỉ lấy các mô hình xe trong danh sách bạn quan tâm
33                     item = {
34                         "model": model,
35                         "recommended_price": cols[1].get_text(strip=True),
36                         "discounted_price": cols[2].get_text(strip=True)
37                     }
38                 data.append(item)

```

Bước 2: FastAPI trả về dữ liệu là chuỗi json



The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000`. The browser tabs include `THEODOIGIAXEMA`, `Facebook`, `Node-RED: Flow 2`, `SQL Data`, `Bài thi học kỳ môn`, `Chuvanchung_1151`, and `127.0.0.1:8000`. The main content area displays a JSON response from a FastAPI endpoint, which is a list of car models with their recommended and discounted prices. The JSON is formatted with syntax highlighting and line numbers on the left side of the editor.

```
1 {  
2   "data": [  
3     {  
4       "model": "Giá xe Rebel 300",  
5       "recommended_price": "125.000.000",  
6       "discounted_price": "112.000.000"  
7     },  
8     {  
9       "model": "Giá xe Rebel 500",  
10      "recommended_price": "180.000.000",  
11      "discounted_price": "177.000.000"  
12     },  
13     {  
14       "model": "Giá xe CB500X",  
15       "recommended_price": "187.990.000",  
16       "discounted_price": "185.000.000"  
17     },  
18     {  
19       "model": "Giá xe CB500F",  
20       "recommended_price": "178.990.000",  
21       "discounted_price": "176.000.000"  
22     },  
23     {  
24       "model": "Giá xe CBR500R",  
25       "recommended_price": "186.990.000",  
26       "discounted_price": "184.000.000"  
27     },  
28     {  
29       "model": "Giá xe CB650F",  
30       "recommended_price": "245.990.000",  
31       "discounted_price": "242.000.000"  
32     },  
33     {  
34       "model": "Giá xe CBR650R",  
35       "recommended_price": "253.990.000",  
36       "discounted_price": "251.000.000"  
37     },  
38     {  
39       "model": "Giá xe Gold Wing",  
40       "recommended_price": "1.200.000.000",  
41       "discounted_price": "1.117.000.000"  
42     },  
43     {  
44       "model": "Giá xe CBR150R ABS",  
45       "recommended_price": "70.990.000",  
46       "discounted_price": "70.900.000"  
47     },  
48     {  
49       "model": "Giá xe CBR250RR",  
50       "recommended_price": "156.000.000",  
51       "discounted_price": "152.000.000"  
52     },  
53     {  
54       "model": "Giá xe CB300R",  
55       "recommended_price": "140.000.000",  
56       "discounted_price": "137.000.000"  
57     }  
58   ]  
59 }
```

Tạo node-red bằng cách sử dụng câu lệnh node-red chạy trên CMD

```
node-red
C:\Users\ngoct>node-red
28 May 18:41:32 - [info]

Welcome to Node-RED
=====

28 May 18:41:32 - [info] Node-RED version: v3.1.9
28 May 18:41:32 - [info] Node.js version: v20.13.1
28 May 18:41:32 - [info] Windows_NT 10.0.22631 x64 LE
28 May 18:41:32 - [info] Loading palette nodes
28 May 18:41:35 - [info] Settings file : C:\Users\ngoct\.node-red\settings.js
28 May 18:41:35 - [info] Context store : 'default' [module=memory]
28 May 18:41:35 - [info] User directory : \Users\ngoct\.node-red
28 May 18:41:35 - [warn] Projects disabled : editorTheme.projects.enabled=false
28 May 18:41:35 - [info] Flows file : \Users\ngoct\.node-red\flows.json
28 May 18:41:35 - [warn]

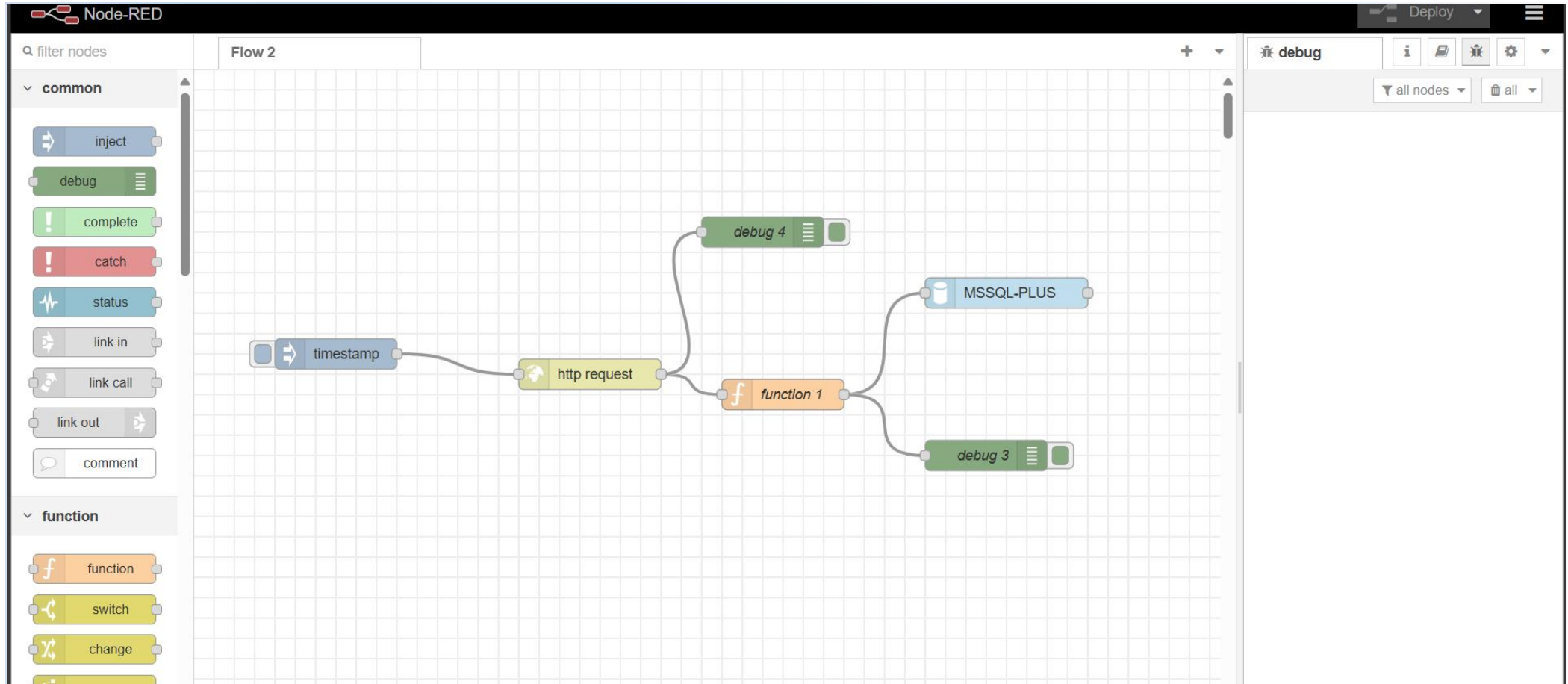
-----
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

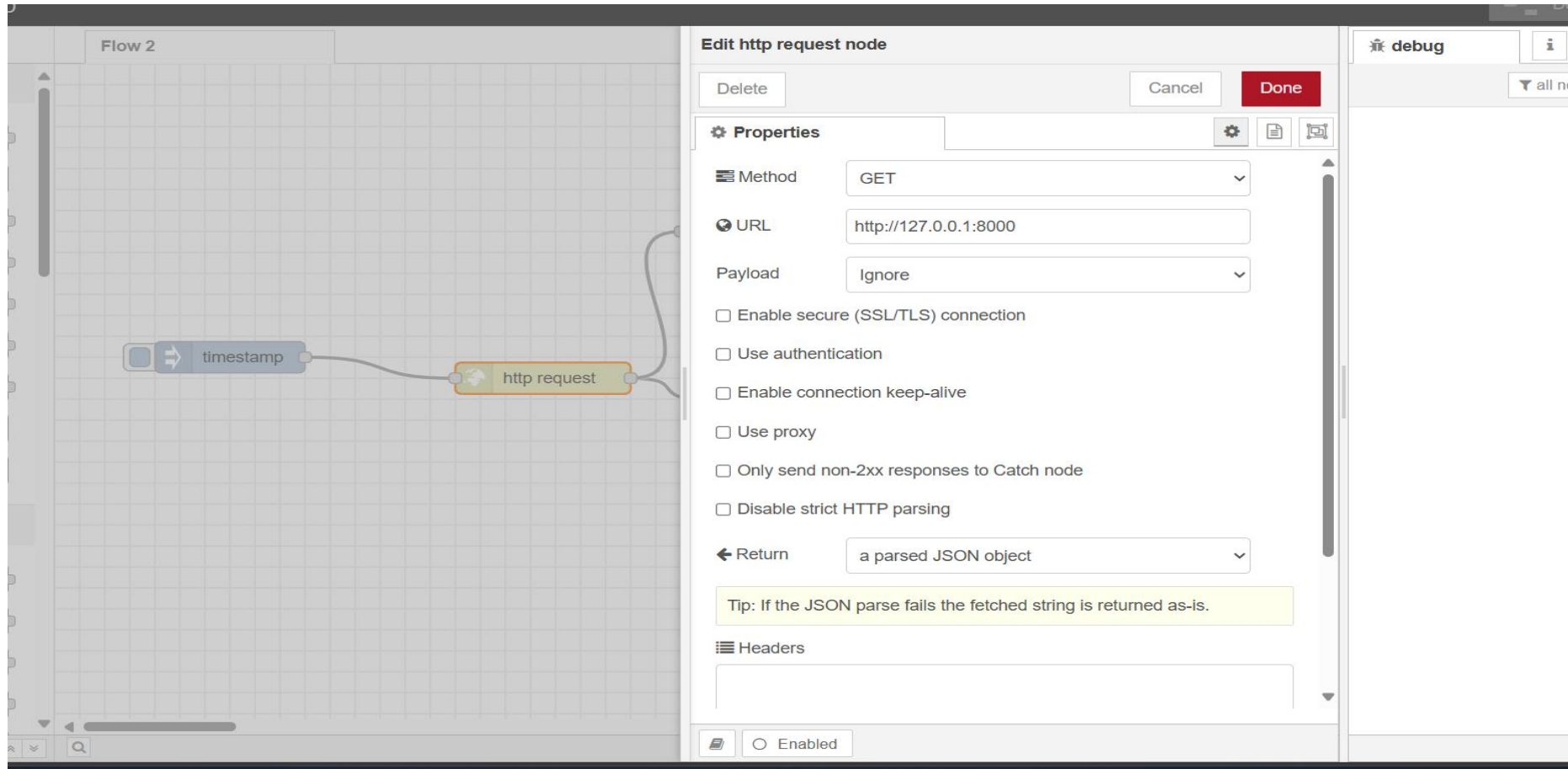
You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----

28 May 18:41:35 - [info] Server now running at http://127.0.0.1:1880/
28 May 18:41:35 - [info] Starting flows
28 May 18:41:35 - [info] Started flows
```

Sơ đồ tổng quan liên kết trên node-red



Sử dụng HTTPs request: để nhận dữ liệu từ fastAPI



Do dữ liệu có nhiều sự trùng lặp về cấu trúc nên ta cần tạo 1 hàm function để trả về lần lượt dữ liệu -> lưu vào database

Name: function 1

Setup On Start On Message On Stop

```
1 function processVehicleData(vehicleData) {
2   var records = [];
3   var data = vehicleData.data; // Truy cập mảng dữ liệu bên trong
4
5   // Chuẩn bị dữ liệu cho mỗi mẫu xe
6   for (var i = 0; i < data.length; i++) {
7     var model = data[i].model;
8     var recommended_price = data[i].recommended_price;
9     var discounted_price = data[i].discounted_price;
10
11     // Kiểm tra dữ liệu giá có null không
12     if (recommended_price !== null && discounted_price !== null) {
13       var record = {
14         model: model,
15         recommended_price: String(recommended_price), //
16         discounted_price: String(discounted_price) //
17       };
18       records.push(record);
19     }
20   }
21
22   var index = 0;
```

Bước 3: Thực hiện procedure để thêm dữ liệu vào sql

The screenshot displays the Node-RED web interface. On the left, the 'common' nodes palette is visible, containing nodes like inject, debug, complete, catch, status, link in, link call, link out, and comment. The central workspace, labeled 'Flow 2', shows a 'timestamp' node connected to an 'http request' node. On the right, the 'Edit MSSQL node' configuration panel is open. It includes a 'Delete' button, 'Cancel', and 'Done' buttons. The 'Properties' section shows the 'Connection' set to 'NNTHAI\SQLEXPRESS'. The 'Query mode' is set to 'Query' and the 'Query' editor contains the following SQL statement:

```
1 EXEC SP_giaban @mauxe = @mauxe , @giadexuat= @param2 ,
```

The 'Parameters' section is set to 'Editor' and lists three input parameters:

Name	Type	Value
mauxe	NVarChar	msg.payload.model
param2	NVarChar	msg.payload.recommend
param3	NVarChar	msg.payload.discounted

The 'Parse Mustache' checkbox is checked. The 'Output property' is set to 'msg.payload'. At the bottom, there is an 'Enabled' checkbox.

SP Thêm dữ liệu vào database

```
2.0.20
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

ALTER PROCEDURE [dbo].[SP_giaban]
-- Add the parameters for the stored procedure here
    @action nvarchar(50) = null ,
    @mauxe nvarchar(255) = null,
    @giadexuat nvarchar (255)= null,
    @giabanuudai nvarchar (255) = null
AS
BEGIN
    DECLARE @json nvarchar(max) = '';

    BEGIN
        INSERT INTO dbo.bang1(mauxe, giadexuat, giabanuudai) VALUES (@mauxe , @giadexuat ,@giabanuudai)

        SELECT @json+=FORMATMESSAGE(N'{Thêm dữ liệu thành công}')
        IF((@json is null)or(@json=''))
            SELECT N'{"ok":0,"msg":"không có dữ liệu","data":[]}' as json;
        ELSE
            BEGIN
                SELECT @json=REPLACE(@json,'(null)','null')
                SELECT N'{"ok":1,"msg":"ok","data":['+left(@json,len(@json)-1)+']}' as json;
            END
    END
END
```

Dữ liệu đã có trong database.

	id	mauxe	giadexuat	giabanuudai
	14	Giá xe CB50...	187.990.000	185.000.000
	15	Giá xe CB50...	178.990.000	176.000.000
	16	Giá xe CBR5...	186.990.000	184.000.000
	17	Giá xe CB65...	245.990.000	242.000.000
	18	Giá xe CBR6...	253.990.000	251.000.000
	19	Giá xe Gold ...	1.200.000.000	1.117.000.000
	20	Giá xe CBR1...	70.990.000	70.900.000
	21	Giá xe CBR2...	156.000.000	152.000.000
	22	Giá xe CB30...	140.000.000	137.000.000
	23	Giá xe CB35...	129.990.000	129.000.000
	24	Giá xe CB10...	468.000.000	465.000.000
	25	Giá xe CBR1...	949.000.000	946.000.000
	26	Giá xe CBR1...	1.049.000.000	1.046.000.000
	27	Giá xe Gold ...	1.230.000.000	1.225.000.000
	28	Giá xe Afric...	689.990.000	686.000.000
	29	Giá xe CB15...	105.000.000	103.000.000
	30	Giá xe Rebe...	125.000.000	112.000.000
	31	Giá xe Rebe...	180.000.000	177.000.000
	32	Giá xe CB50...	187.990.000	185.000.000
	33	Giá xe CB50...	178.990.000	176.000.000
	34	Giá xe CBR5...	186.990.000	184.000.000
	35	Giá xe CB65...	245.990.000	242.000.000
	36	Giá xe CBR6...	253.990.000	251.000.000
	37	Giá xe Gold ...	1.200.000.000	1.117.000.000
	38	Giá xe CBR1...	70.990.000	70.900.000
	39	Giá xe CBR2...	156.000.000	152.000.000
	40	Giá xe CB30...	140.000.000	137.000.000
	41	Giá xe CB35...	129.990.000	129.000.000
	42	Giá xe CB10...	468.000.000	465.000.000
	43	Giá xe CBR1...	949.000.000	946.000.000
	44	Giá xe CBR1...	1.049.000.000	1.046.000.000
	45	Giá xe Gold ...	1.230.000.000	1.225.000.000
	46	Giá xe Afric...	689.990.000	686.000.000
	47	Giá xe CB15...	105.000.000	103.000.000
	48	Giá xe Rebe...	125.000.000	112.000.000
	49	Giá xe Rebe...	180.000.000	177.000.000
	50	Giá xe CB50...	187.990.000	185.000.000
	51	Giá xe CB50...	178.990.000	176.000.000
	52	Giá xe CBR5...	186.990.000	184.000.000
	53	Giá xe CB65...	245.990.000	242.000.000
	54	Giá xe CBR6...	253.990.000	251.000.000
	55	Giá xe Gold ...	1.200.000.000	1.117.000.000
	56	Giá xe CBR1...	70.990.000	70.900.000
	57	Giá xe CBR2...	156.000.000	152.000.000
	58	Giá xe CB30...	140.000.000	137.000.000
	59	Giá xe CB35...	129.990.000	129.000.000
	60	Giá xe CB10...	468.000.000	465.000.000
	61	Giá xe CBR1...	949.000.000	946.000.000
	62	Giá xe CBR1...	1.049.000.000	1.046.000.000
	63	Giá xe Gold ...	1.230.000.000	1.225.000.000
	64	Giá xe Afric...	689.990.000	686.000.000
	65	Giá xe CB15...	105.000.000	103.000.000
	66	Giá xe Rebe...	125.000.000	112.000.000
	67	Giá xe Rebe...	180.000.000	177.000.000
	68	Giá xe CB50...	187.990.000	185.000.000
	69	Giá xe CB50...	178.990.000	176.000.000
	70	Giá xe CBR5...	186.990.000	184.000.000
	71	Giá xe CB65...	245.990.000	242.000.000
	72	Giá xe CBR6...	253.990.000	251.000.000
	73	Giá xe Gold ...	1.200.000.000	1.117.000.000
	74	Giá xe CBR1...	70.990.000	70.900.000
	75	Giá xe CBR2...	156.000.000	152.000.000
	76	Giá xe CB30...	140.000.000	137.000.000
	77	Giá xe CB35...	129.990.000	129.000.000
	78	Giá xe CB10...	468.000.000	465.000.000
	79	Giá xe CBR1...	949.000.000	946.000.000
	80	Giá xe CBR1...	1.049.000.000	1.046.000.000
	81	Giá xe Gold ...	1.230.000.000	1.225.000.000
	82	Giá xe Afric...	689.990.000	686.000.000
	83	Giá xe CB15...	105.000.000	103.000.000
	84	Giá xe Rebe...	125.000.000	112.000.000
	85	Giá xe Rebe...	180.000.000	177.000.000
	86	Giá xe CB50...	187.990.000	185.000.000
	87	Giá xe CB50...	178.990.000	176.000.000
	88	Giá xe CBR5...	186.990.000	184.000.000
	89	Giá xe CB65...	245.990.000	242.000.000
	90	Giá xe CBR6...	253.990.000	251.000.000
	91	Giá xe Gold ...	1.200.000.000	1.117.000.000
	92	Giá xe CBR1...	70.990.000	70.900.000
	93	Giá xe CBR2...	156.000.000	152.000.000
	94	Giá xe CB30...	140.000.000	137.000.000
	95	Giá xe CB35...	129.990.000	129.000.000
	96	Giá xe CB10...	468.000.000	465.000.000
	97	Giá xe CBR1...	949.000.000	946.000.000
	98	Giá xe CBR1...	1.049.000.000	1.046.000.000
	99	Giá xe Gold ...	1.230.000.000	1.225.000.000
	100	Giá xe Afric...	689.990.000	686.000.000

Tạo 1 file giaodien(python) để lấy dữ liệu từ SQL SEVER
Sử dụng truy vấn `SELECT id, mauxe, giadexuat, giabanuudai` FROM bang1 lấy dữ liệu từ database

```
1 usage (1 dynamic)  ngocthai
28 @app.get(path: "/", response_class=HTMLResponse)
29 async def get_data(request: Request):
30     connection = None
31     try:
32         connection = pyodbc.connect(connection_string)
33         cursor = connection.cursor()
34         cursor.execute("SELECT id, mauxe, giadexuat, giabanuudai FROM bang1:")
35         rows = cursor.fetchall()
36
37         data = []
38         for row in rows:
39             id_value = row[0]
40             mauxe_value = row[1]
41             giadexuat_value = row[2]
42             giabanuudai_value = row[3]
43
44             # Kiểm tra giá trị của mauxe
45             if mauxe_value is not None:
46                 # Tạo đối tượng InputData chỉ khi mauxe không phải là null
47                 data.append(InputData(id=id_value, mauxe=mauxe_value, giadexuat=giadexuat_value, giabanuudai=giabanuudai_value))
48             else:
49                 # Xử lý trường hợp mauxe là null (ví dụ: bỏ qua hoặc gán giá trị mặc định)
50                 # Ví dụ: bỏ qua dòng dữ liệu này
51                 pass
52
53     return templates.TemplateResponse("web.html", {"request": request, "data": data})
```

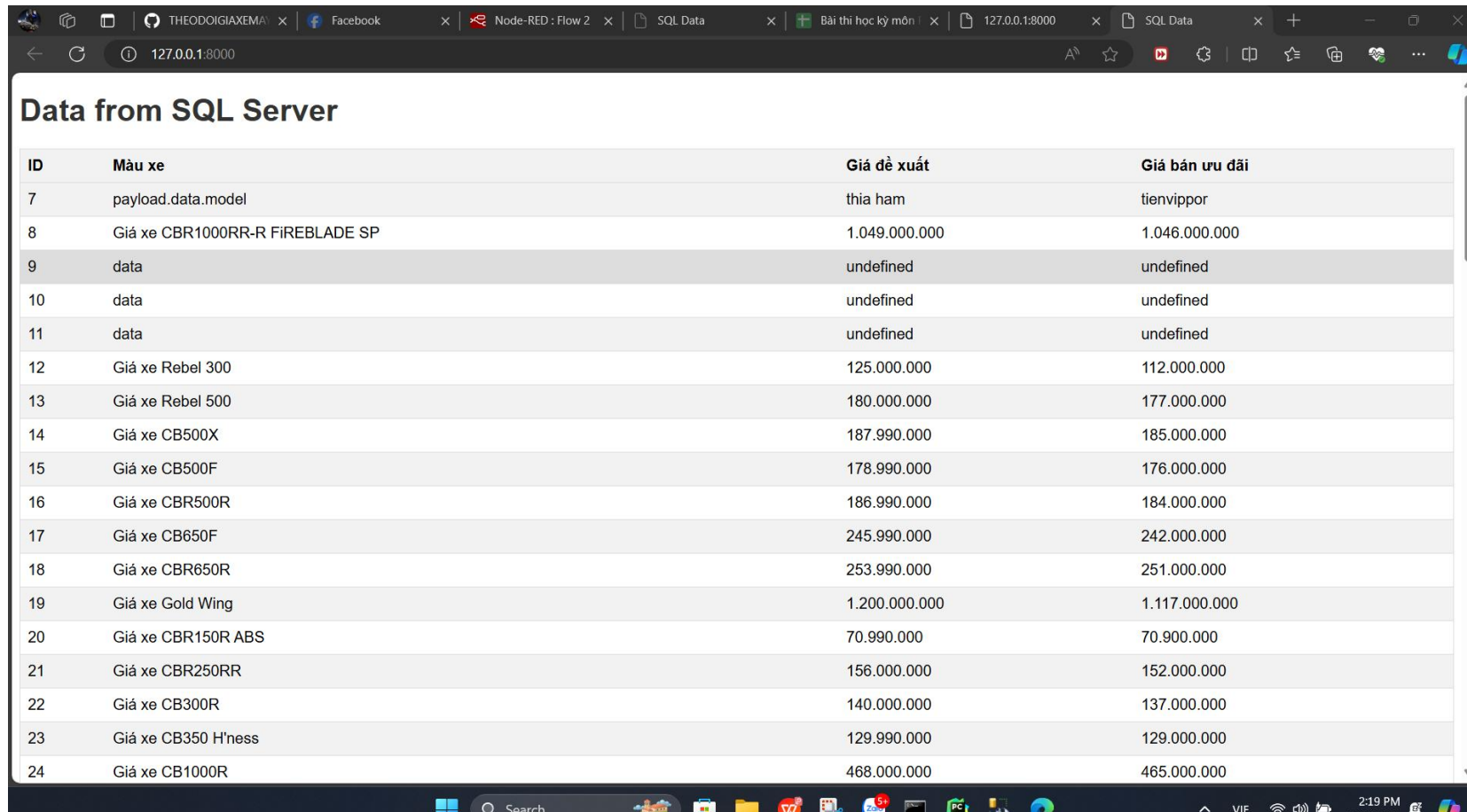
Sau khi có dữ liệu thì truyền dữ liệu đó đến file web.html nằm trong thư mục **templates**

```
return templates.TemplateResponse("web.html", {"request": request, "data": data})
```

Trên file web.html thì xử lý dữ liệu nhận được và tạo thành bảng

```
        </style>
    </head>
    <body>
        <h1>Data from SQL Server</h1>
        <table>
            <tr>
                <th>ID</th>
                <th>Màu xe</th>
                <th>Giá để xuất</th>
                <th>Giá bán ưu đãi</th>
            </tr>
            {% for item in data %}
                <tr>
                    <td>{{ item.id }}</td>
                    <td>{{ item.mauxe }}</td>
                    <td>{{ item.giadexuat }}</td>
                    <td>{{ item.giabanuudai }}</td>
                </tr>
            {% endfor %}
```

Kết quả sau khi hoàn thành .



Data from SQL Server

ID	Màu xe	Giá đề xuất	Giá bán ưu đãi
7	payload.data.model	thia ham	tienvippor
8	Giá xe CBR1000RR-R FIREBLADE SP	1.049.000.000	1.046.000.000
9	data	undefined	undefined
10	data	undefined	undefined
11	data	undefined	undefined
12	Giá xe Rebel 300	125.000.000	112.000.000
13	Giá xe Rebel 500	180.000.000	177.000.000
14	Giá xe CB500X	187.990.000	185.000.000
15	Giá xe CB500F	178.990.000	176.000.000
16	Giá xe CBR500R	186.990.000	184.000.000
17	Giá xe CB650F	245.990.000	242.000.000
18	Giá xe CBR650R	253.990.000	251.000.000
19	Giá xe Gold Wing	1.200.000.000	1.117.000.000
20	Giá xe CBR150R ABS	70.990.000	70.900.000
21	Giá xe CBR250RR	156.000.000	152.000.000
22	Giá xe CB300R	140.000.000	137.000.000
23	Giá xe CB350 H'ness	129.990.000	129.000.000
24	Giá xe CB1000R	468.000.000	465.000.000

Kết luận

Việc Kết hợp ba công nghệ này có thể tạo ra các giải pháp mạnh mẽ và linh hoạt cho nhiều loại dự án. Python có thể được sử dụng để xây dựng các ứng dụng và phân tích dữ liệu phức tạp, trong khi Node-RED giúp đơn giản hóa việc tích hợp các hệ thống và tự động hóa. SQL, với khả năng quản lý dữ liệu mạnh mẽ, sẽ đảm bảo rằng dữ liệu được tổ chức và truy vấn một cách hiệu quả. Sự kết hợp này không chỉ tăng cường khả năng phát triển mà còn mở ra nhiều cơ hội cho sự sáng tạo và tối ưu hóa quy trình làm việc trong các dự án công nghệ hiện đại.