

Nguyễn Thị Mỹ Duyên

K2005480106033

Bài tập về nhà

Yêu cầu :

1. gen scripts sql: struct + data.
2. code py fastAPI
3. flow.json export từ node-red
4. full code project web (asp dot net) : html + js + css + aspx + aspx.cs (api): nêu dùng git trên vs2022 để upload.
5. pdf (save as word) : mô tả lại quá trình làm. nhớ chụp ảnh màn hình những bước quan trọng.

Quá trình làm bài:

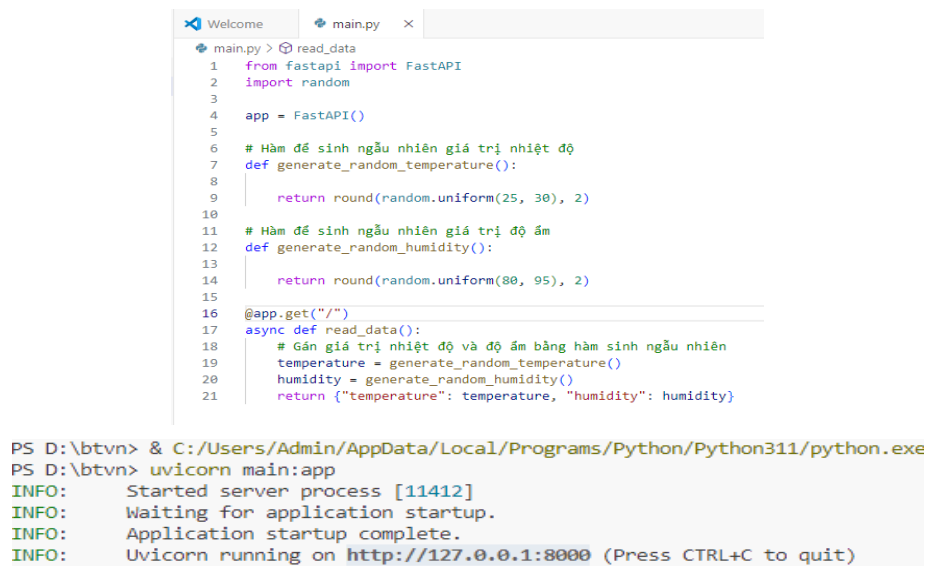
1. Thiết lập môi trường

- Cài đặt fastapi :
 - Mở terminal thực hiện hai câu lệnh

`pip install fastapi`

`pip install uvicorn`

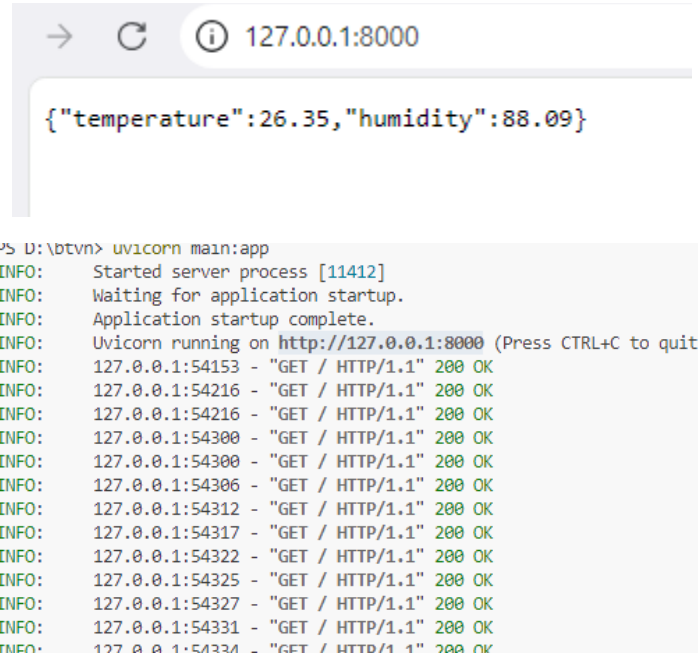
- Khởi chạy fastapi :



```
main.py > read_data
1 from fastapi import FastAPI
2 import random
3
4 app = FastAPI()
5
6 # Hàm để sinh ngẫu nhiên giá trị nhiệt độ
7 def generate_random_temperature():
8     return round(random.uniform(25, 30), 2)
9
10 # Hàm để sinh ngẫu nhiên giá trị độ ẩm
11 def generate_random_humidity():
12     return round(random.uniform(80, 95), 2)
13
14
15
16 @app.get("/")
17 async def read_data():
18     # Gán giá trị nhiệt độ và độ ẩm bằng hàm sinh ngẫu nhiên
19     temperature = generate_random_temperature()
20     humidity = generate_random_humidity()
21     return {"temperature": temperature, "humidity": humidity}
```

```
PS D:\btvn> & C:/Users/Admin/AppData/Local/Programs/Python/Python311/python.exe
PS D:\btvn> uvicorn main:app
INFO: Started server process [11412]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
```

- Sau khi chạy sẽ trả về 1 chuỗi dạng json

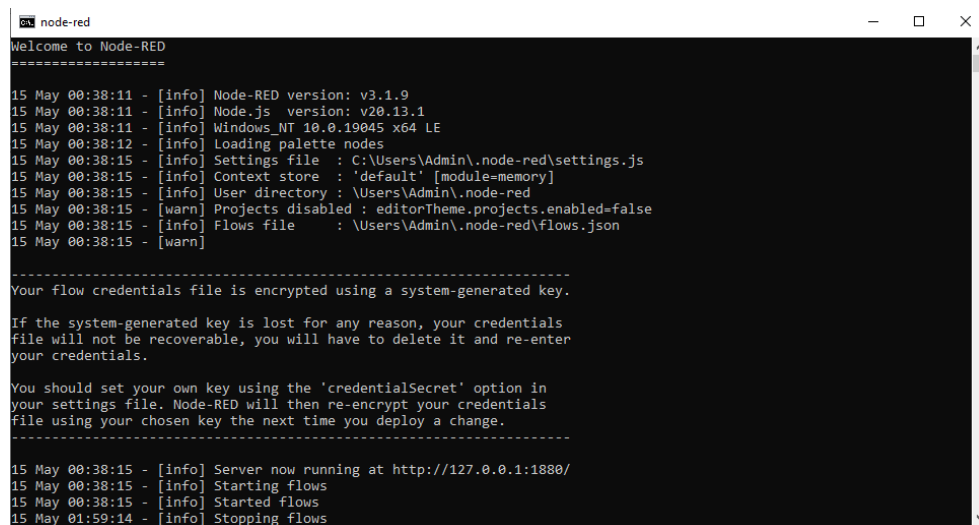


```
→ ↻ ⓘ 127.0.0.1:8000
{"temperature":26.35,"humidity":88.09}

PS D:\bvtvn> uvicorn main:app
INFO: Started server process [11412]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO: 127.0.0.1:54153 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54216 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54216 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54300 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54300 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54306 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54312 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54317 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54322 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54325 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54327 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54331 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:54334 - "GET / HTTP/1.1" 200 OK
```

- Cài đặt node-red: cần tải nodejs trước sau đó mới cài đặt được node-red bằng câu lệnh

npm install -g --unsafe-perm node-red



```
node-red
Welcome to Node-RED
=====
15 May 00:38:11 - [info] Node-RED version: v3.1.9
15 May 00:38:11 - [info] Node.js version: v20.13.1
15 May 00:38:11 - [info] Windows_NT 10.0.19045 x64 LE
15 May 00:38:12 - [info] Loading palette nodes
15 May 00:38:15 - [info] Settings file : C:\Users\Admin\.node-red\settings.js
15 May 00:38:15 - [info] Context store : 'default' [module=memory]
15 May 00:38:15 - [info] User directory : \Users\Admin\.node-red
15 May 00:38:15 - [warn] Projects disabled : editorTheme.projects.enabled=false
15 May 00:38:15 - [info] Flows file : \Users\Admin\.node-red\flows.json
15 May 00:38:15 - [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.
-----
15 May 00:38:15 - [info] Server now running at http://127.0.0.1:1880/
15 May 00:38:15 - [info] Starting flows
15 May 00:38:15 - [info] Started flows
15 May 01:59:14 - [info] Stopping flows
```

Sau đó dán đường link <http://127.0.0.1:1880> để mở node-red.

- Kết nối node-red với sql

Cài đặt **node-red-contrib-mssql-plus** sau đó kết nối với sql

Properties

- Name: btvn
- Server: DESKTOP-PED1FFB
- Port: 1433
- Username: sa
- Password:
- Domain:
- Database: btvn
- TDS Version: 7_4 (SQL Server 2012 ~ 2022)

- Sử dụng node-red lấy dữ liệu:

➤ Dán đường link vào http request

http request

Properties

- Method: GET
- URL: http://127.0.0.1:8000/

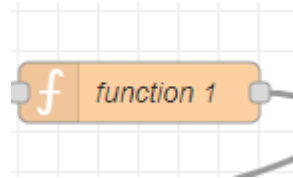
➤ Set thời gian lấy dữ liệu và lưu vào database trong timestamp

timestamp

Properties

- ☒ Inject once after 0.1 seconds, then
- Repeat: interval
- every 30 seconds

- Sau đó viết function để lưu dữ liệu vào database



```
msg.payload = `insert into history(sid, value) values
(1,${msg.payload.temperature}); insert into history(sid, value)
values (2,${msg.payload.humidity})`
return msg;
```

- Dữ liệu được lưu vào database

	id	sid	value	time
16	20	2	93	2024-05-14 22:42:35.950
17	21	1	25	2024-05-14 22:43:06.100
18	22	2	85	2024-05-14 22:43:06.103
19	23	1	26	2024-05-14 22:43:36.730
20	24	2	90	2024-05-14 22:43:36.737
21	25	1	30	2024-05-14 22:44:06.110
22	26	2	88	2024-05-14 22:44:06.110
23	27	1	26	2024-05-14 22:44:36.063
24	28	2	88	2024-05-14 22:44:36.067
25	29	1	27	2024-05-14 22:45:06.043
26	30	2	80	2024-05-14 22:45:06.043
27	31	1	28	2024-05-14 22:45:36.053
28	32	2	88	2024-05-14 22:45:36.053
29	33	1	28	2024-05-14 22:46:06.110
30	34	2	82	2024-05-14 22:46:06.113
31	35	1	26	2024-05-14 22:46:36.090
32	36	2	85	2024-05-14 22:46:36.117
33	37	1	27	2024-05-14 22:47:06.090
34	38	2	81	2024-05-14 22:47:06.090
35	39	1	27	2024-05-14 22:47:36.117
36	40	2	85	2024-05-14 22:47:36.120
37	41	1	29	2024-05-14 22:48:06.140
38	42	2	87	2024-05-14 22:48:06.140
39	43	1	28	2024-05-14 22:48:36.150
40	44	2	87	2024-05-14 22:48:36.150
41	45	1	30	2024-05-14 22:49:06.147
42	46	2	89	2024-05-14 22:49:06.150