

# Hướng dẫn xây dựng ứng dụng ReactJS Flask API SQLite QLHS - I

## Giáo viên: Nguyễn Hùng Cường

### Bước 1: Xây dựng ứng dụng Flask Backend

Mở một trình soạn thảo, sau đó xây dựng ứng dụng Flask (ở trong tài liệu này ta sử dụng PyCharm IDE).

Tiếp theo, ta tạo file config.py, rồi khai báo đường dẫn đến CSDL SQLite như sau.

```
config.py x
1  SQLAlchemy_DATABASE_URI = 'sqlite:///quanlynhanvien.db'
2
```

Sau đó, ta tạo file models.py, rồi khởi tạo class NhanVien, class này sẽ được mapping với bảng nhanvien trong CSDL. Class gồm 5 thuộc tính sẽ tương ứng với 5 cột trong bảng như sau.

```
models.py x
1  from flask_sqlalchemy import SQLAlchemy
2
3  db = SQLAlchemy()
4
5  # định nghĩa class để mapping với bảng nhanvien trong CSDL
6  class NhanVien(db.Model):
7      ma = db.Column(db.Integer, primary_key=True)
8      # khai báo nullable=False để cột bắt buộc phải nhập
9      hoten = db.Column(db.String(50), nullable=False)
10     tuoi = db.Column(db.Integer)
11     diachi = db.Column(db.String(50))
12     anh = db.Column(db.String(200))
13
14     def __repr__(self):
15         # Method này cho phép hiển thị thông tin chi tiết về đối tượng
16         return f'Task {self.ma} : {self.hoten} : {self.tuoi} : {self.diachi}'
```

Sau đó, ta tạo file services.py trong đó có 2 method get() và post() để thêm mới và lấy về tất cả các bản ghi trong CSDL.

```

services.py
1  from flask_restful import Resource
2  from flask import request
3  from models import NhanVien, db
4  from flask_cors import CORS, cross_origin
5
6
7  # định nghĩa class cung cấp các method cho phép gọi API từ frontend
8  class NhanVienManager(Resource):
9      @cross_origin()
10     def get(self):
11         # lấy về tất cả nhân viên từ database.
12         employees = NhanVien.query.all()
13         # Chuyển đổi danh sách nhân viên thành JSON và trả về response.
14         listemployee = [{'ma': employee.ma, 'hoten': employee.hoten, 'tuoi': employee.tuoi,
15                          'diachi': employee.diachi, 'anh': employee.anh}
16                          for employee in employees]
17         #return {'listemployee': listemployee}
18         return listemployee

```

Method post() cho phép thêm mới thông tin nhân viên qua API.

```

services.py
19
20     def post(self):
21         # Lấy JSON data từ request.
22         response_data = request.get_json()
23         # Kiểm tra xem dữ liệu có hợp lệ hay không.
24         if not response_data:
25             return {'message': 'No input data provided'}, 400
26         hoten = response_data.get('hoten')
27         if not hoten:
28             return {'message': 'Họ tên nhân viên không được bỏ trống !'}, 400
29         # Thêm nhân viên mới vào database.
30         new_employee = NhanVien(hoten=response_data.get('hoten'), tuoi=response_data.get('tuoi'),
31                                diachi=response_data.get('diachi'), anh=response_data.get('anh'))
32         db.session.add(new_employee)
33
34         # Ghi dữ liệu vào database.
35         db.session.commit()
36         # Trả về một message cho người dùng.
37         return {'message': 'Đã thêm nhân viên mới', 'nhanvien': {'ma': new_employee.ma,
38                                                                    'hoten': new_employee.hoten,
39                                                                    'tuoi': new_employee.tuoi,
40                                                                    'diachi': new_employee.diachi,
41                                                                    'anh': new_employee.anh}}

```

Sau đó, trong file app.py, ta viết mã để thực thi ứng dụng, đồng thời có cài đặt CORS để cho phép client có thể kết nối.

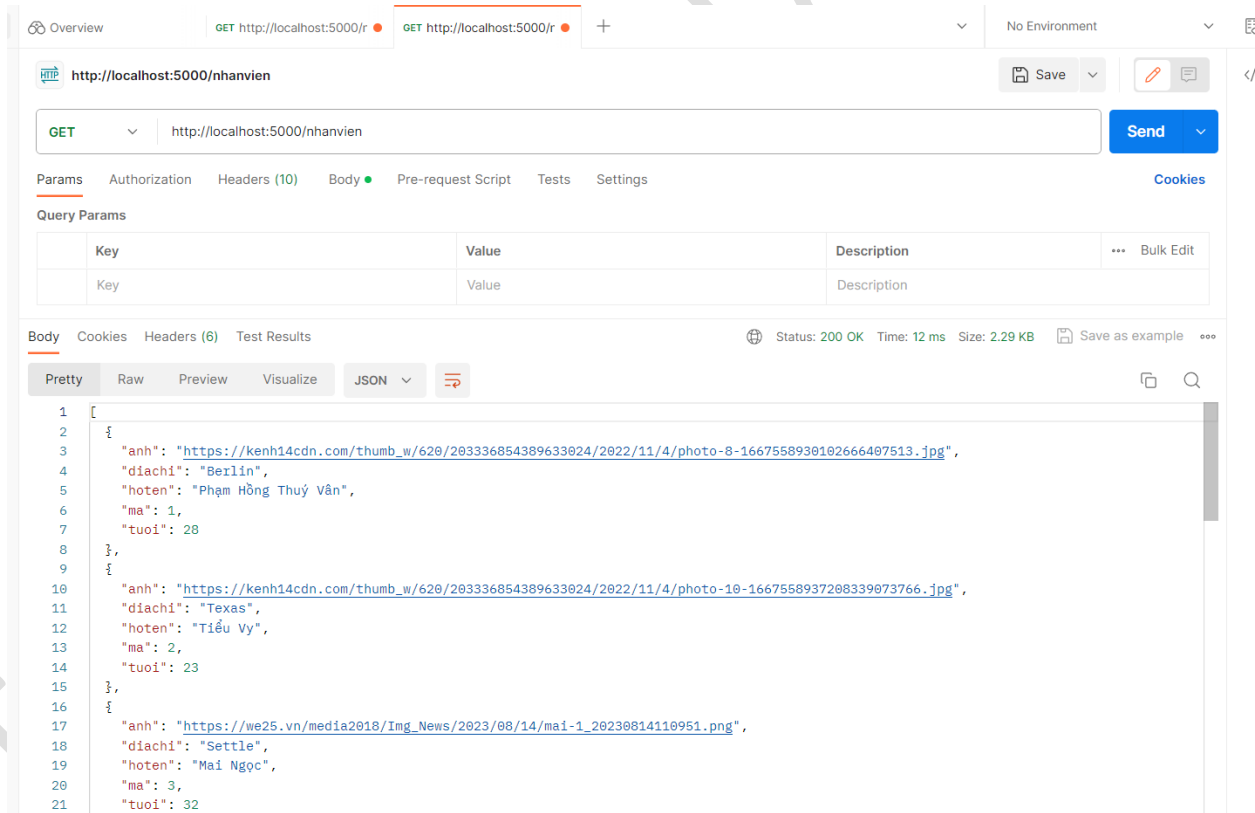
```
app.py x
1  from flask import Flask
2      from flask_restful import Api
3      from models import db
4      import config
5      from services import NhanVienManager
6  from flask_cors import CORS
7
8      # Khoi tao Flask application va Flask-RESTful API manager.
9      app = Flask(__name__)
10     cors = CORS(app, resources={r"/api/*": {"origins": "*"}})
11
12
13     app.config.from_object(config)
14     # Khoi tao doi tuong Flask-SQLAlchemy.
15     db.init_app(app)
16     # Tao doi tuong Flask-RESTful API manager.
17     api = Api(app)
18     # Tao endpoints.
19     api.add_resource(NhanVienManager, '/nhanvien')
20
21  if __name__ == '__main__':
22     # Tao database tables.
23     with app.app_context():
24         db.create_all()
25     # Start Flask development web server.
26     app.run(debug=True)
```

## Bước 2: Thực thi ứng dụng backend và test API

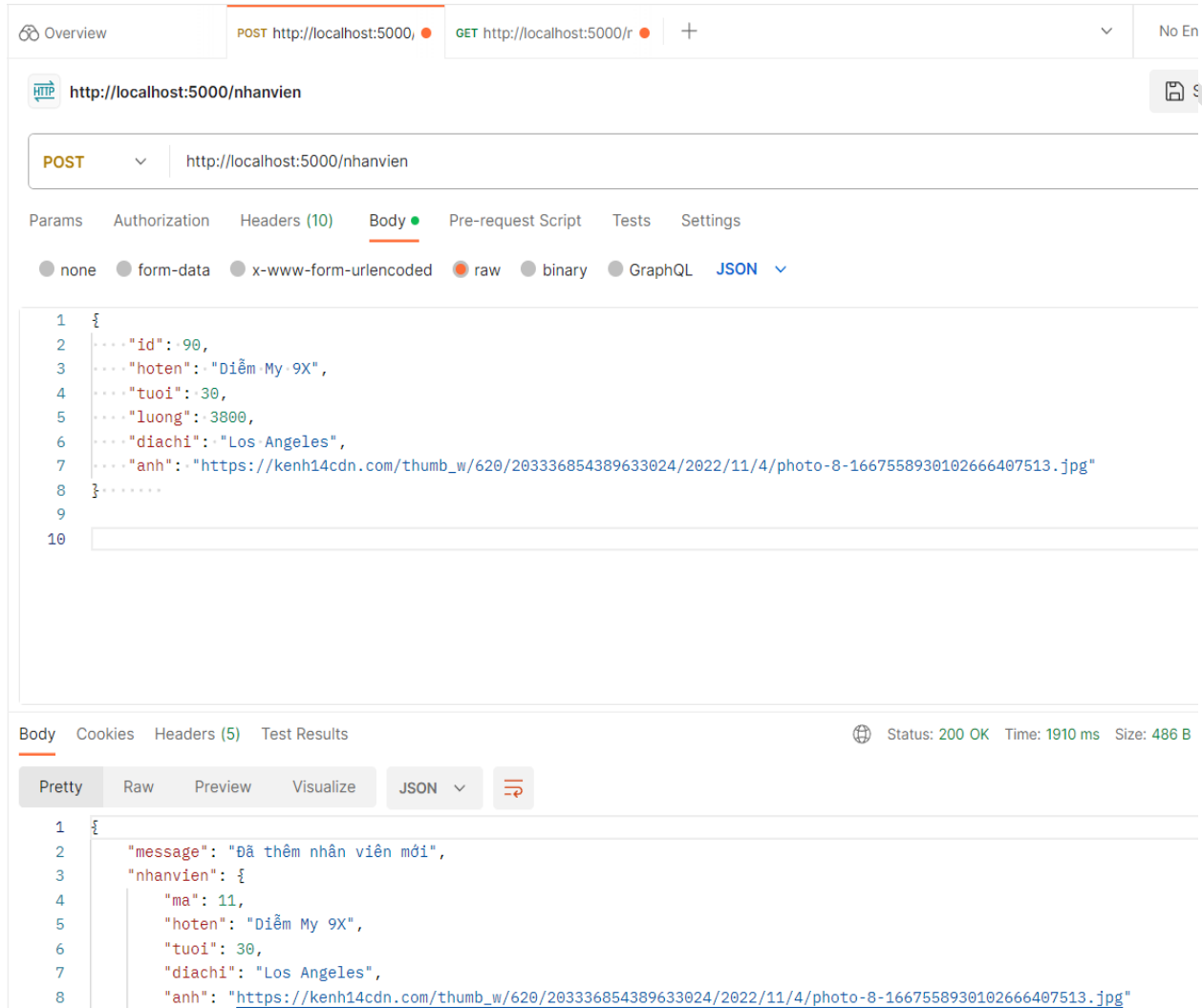
Sau khi đã viết mã xong, ta thực thi ứng dụng Flask.

```
Run: app
D:\ademobaigiang2023\demoFlask\Flaskquanlynhanvien\venv\Scripts\python.exe D:\ademobaigiang2023\demoFlask\Flaskquanlynhanvien/app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 119-297-043
127.0.0.1 - - [23/Dec/2023 15:20:09] "GET /nhanvien HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2023 15:28:38] "GET /nhanvien HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2023 15:29:49] "GET /nhanvien HTTP/1.1" 200 -
* Detected change in 'D:\ademobaigiang2023\demoFlask\Flaskquanlynhanvien\models.py', reloading
* Restarting with stat
* Debugger is active!
* Debugger PIN: 119-297-043
```

Sau đó, ta mở công cụ Postman, rồi gõ URL vào, sau đó chọn method GET để truy vấn dữ liệu từ CSDL. Ta thấy dữ liệu trong bảng nhanvien đã được liệt kê đúng như mong muốn. Ở đây ta đã chèn sẵn một số bản ghi vào bảng nhanvien trong CSDL.



Tiếp theo, ta test method POST để thêm mới dữ liệu vào bảng. Ta gửi dữ liệu nhanvien lên API và sẽ thấy kết quả hiển thị thành công.



The screenshot shows a web browser interface for testing HTTP requests. The URL bar shows `http://localhost:5000/nhanvien`. The request method is `POST`. The request body is a JSON object:

```
1 {
2   "id": 90,
3   "hoten": "Diễm My 9X",
4   "tuoi": 30,
5   "luong": 3800,
6   "diachi": "Los Angeles",
7   "anh": "https://kenh14cdn.com/thumb_w/620/203336854389633024/2022/11/4/photo-8-1667558930102666407513.jpg"
8 }
```

The response body is a JSON object:

```
1 {
2   "message": "Đã thêm nhân viên mới",
3   "nhanvien": {
4     "ma": 11,
5     "hoten": "Diễm My 9X",
6     "tuoi": 30,
7     "diachi": "Los Angeles",
8     "anh": "https://kenh14cdn.com/thumb_w/620/203336854389633024/2022/11/4/photo-8-1667558930102666407513.jpg"
9   }
10 }
```

Sau đó, ta kiểm tra dữ liệu bằng cách mở bảng trong công cụ DB Browser ra, ta sẽ thấy bảng nhanvien có chứa dữ liệu như hình bên dưới.

DB Browser for SQLite - D:\ademobaigiang2023\demoflask\flaskquanlynhanvien\instance\quanlynhanvien.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

Table: nhan\_vien Filter in any column

	ma	hoten	tuoi	diachi	anh
	Filter	Filter	Filter	Filter	Filter
1	1	Phạm Hồng Thuý Vân	28	Berlin	<a href="https://kenh14cdn.com/thumb_w/620/203336854389633024/2022/11/4/photo-8-1667558930102666407513.jpg">https://kenh14cdn.com/thumb_w/620/203336854389633024/2022/11/4/photo-8-1667558930102666407513.jpg</a>
2	2	Tiểu Vy	23	Texas	<a href="https://kenh14cdn.com/thumb_w/620/203336854389633024/2022/11/4/photo-10-1667558937208339073766.jpg">https://kenh14cdn.com/thumb_w/620/203336854389633024/2022/11/4/photo-10-1667558937208339073766.jpg</a>
3	3	Mai Ngọc	32	Settle	<a href="https://we25.vn/media2018/img_News/2023/08/14/mai-1_20230814110951.png">https://we25.vn/media2018/img_News/2023/08/14/mai-1_20230814110951.png</a>
4	4	Khánh Trang	35	Paris	<a href="https://static1.dienanh.net/upload/202202/d0526ee9-84bc-45c9-be37-71247c74e8ec.jpeg">https://static1.dienanh.net/upload/202202/d0526ee9-84bc-45c9-be37-71247c74e8ec.jpeg</a>
5	5	Đỗ Thị Hà	24	Berlin	<a href="https://i.vietgiaitri.com/2021/11/8/gan-dat-guong-mat-ty-le-vang-tieu-vy-hay-do-thi-ha-an-tuong-hon-afa-6142005.jpg">https://i.vietgiaitri.com/2021/11/8/gan-dat-guong-mat-ty-le-vang-tieu-vy-hay-do-thi-ha-an-tuong-hon-afa-6142005.jpg</a>
6	6	Phương Nhi	23	Zurich	<a href="https://static-images.vnncdn.net/files/publish/2022/8/21/phuong-nhi-1-560.jpg">https://static-images.vnncdn.net/files/publish/2022/8/21/phuong-nhi-1-560.jpg</a>
7	7	Hà Kiều Anh	45	Cần Thơ	<a href="https://toquoc.mediacdn.vn/thumb_w/...">https://toquoc.mediacdn.vn/thumb_w/...</a>
8	8	Thuý Ngân	28	Montreal	<a href="https://kenh14cdn.com/thumb_w/620/203336854389633024/2022/11/4/photo-8-1667558930102666407513.jpg">https://kenh14cdn.com/thumb_w/620/203336854389633024/2022/11/4/photo-8-1667558930102666407513.jpg</a>
9	9	Thuý Ngân	28	Montreal	<a href="https://media-cdn-v2.laodong.vn/Storage/NewsPortal/2020/12/28/865890/Thuy-Ngan-4.JPG">https://media-cdn-v2.laodong.vn/Storage/NewsPortal/2020/12/28/865890/Thuy-Ngan-4.JPG</a>
10	10	Trương Ngọc Ảnh	45	Đà Nẵng	<a href="https://images2.thanhnien.vn/thumb_w/640/528068263637045248/2023/3/4/truong-ngoc-...">https://images2.thanhnien.vn/thumb_w/640/528068263637045248/2023/3/4/truong-ngoc-...</a>

Dưới đây là cấu trúc thư mục của ứng dụng Backend viết bằng Flask

