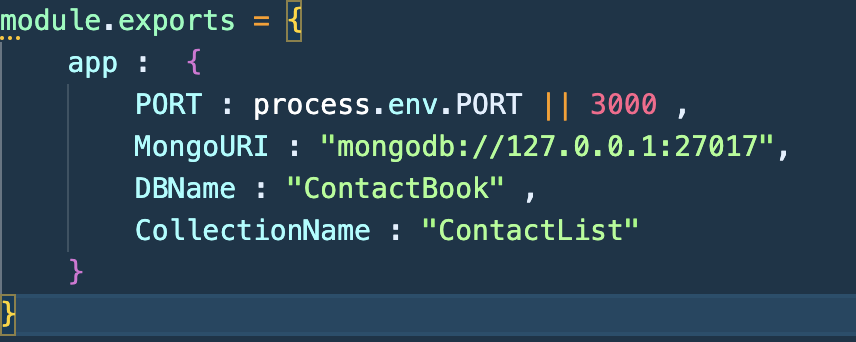
**Bài thực hành số 2**

**Kết nối và xây dựng với cơ sở dữ liệu MongoDB**

## Thiết lập các module cần thiết để có thể kết nối với Mongo

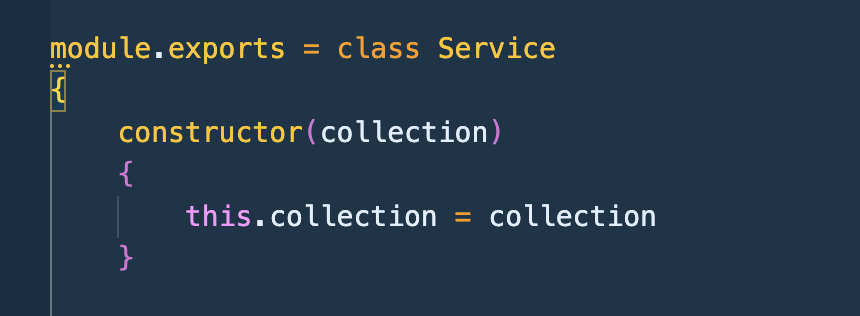
Đầu tiên thiết lập địa chỉ và DBName và CollectionName của database :



Thiết lập lớp trong file mongo.util.js như sau để có thể kết nối lần đầu và thiết lập và trả luôn về collection để tương tác ở các lần kết nối :



Tạo module contact.service.js và bắt đầu viết các hàm thao tác ở trong đây như sau :



Serivice sẽ nhận vào một collection và thao tác trực tiếp với collection đó qua các method ở bên trong , collection ở đây được lấy từ mongo.util.js với cú pháp MongDB.collection

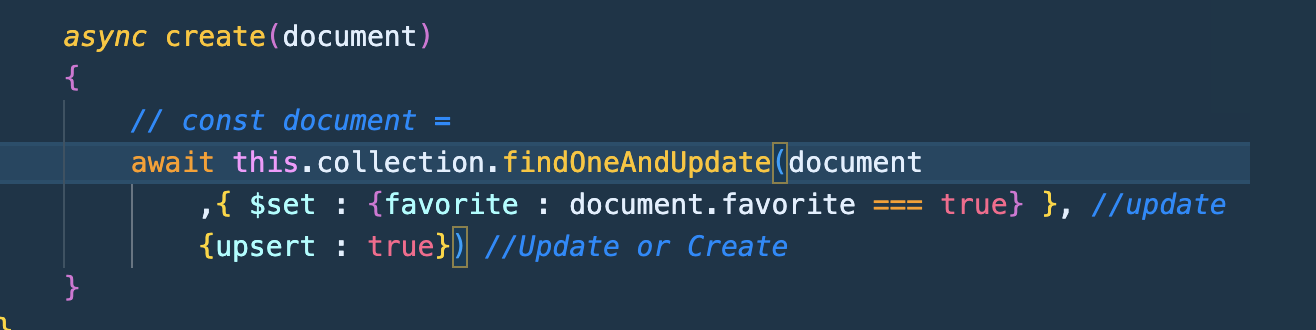
## Cài đặt các Controller xử lý

### 2.1 Controller Create

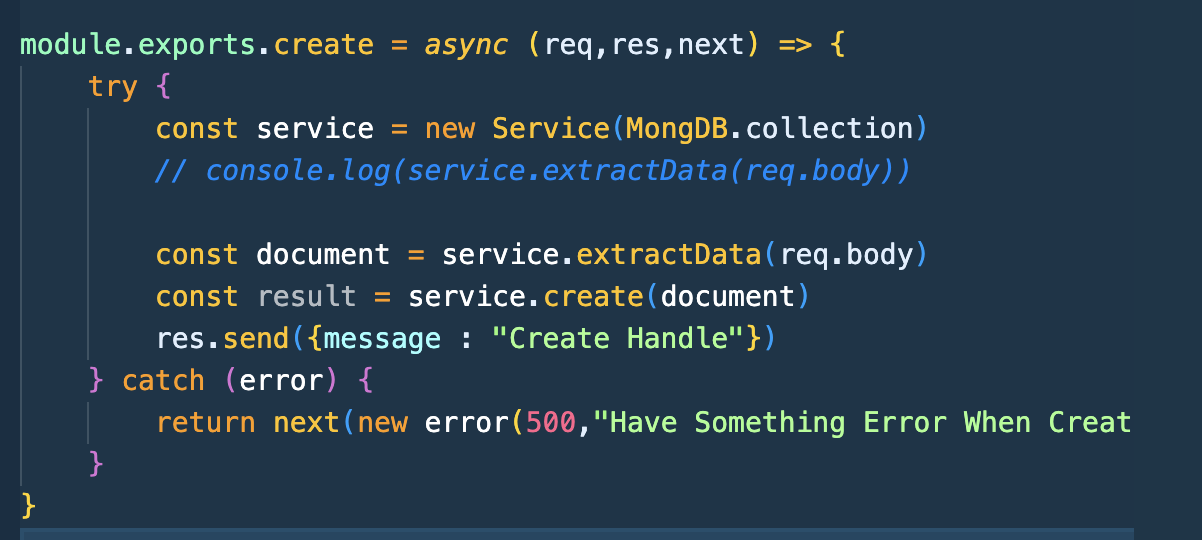
Để có thể thêm một document ứng với một user vào ContactBook ta cần chuyển đổi giá trị để có thể thêm vào cơ sở dữ liệu như sau : 

Dữ liệu vào chỉ bao gồm các field như : name , email , address , phone , favorite . Các thuộc tính còn lại và các thuộc tính trên nếu có undefined thì sẽ loại bỏ ra nhằm trắng lãng phí bộ nhớ

Tiếp theo đi cài đặt hàm khởi tạo cho service với tiêu chí như sau : nhận vào một document và tìm kiếm nếu đã tồn tại thì update lại còn không thì tạo mới một document mới :

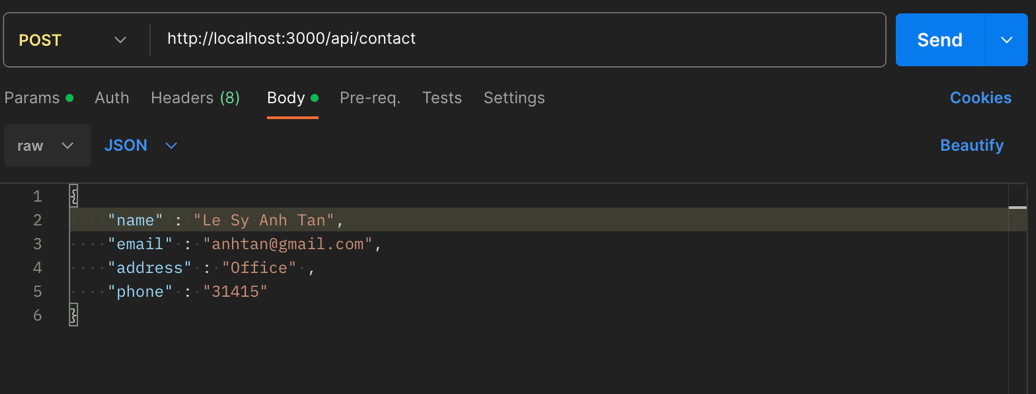


Và cuối dùng sử dụng các hàm đã cài đặt trong controller như sau :



Thử thêm một Document vào ta được như sau :

Thao tác bằng postman sử dụng phương thức và địa chỉ ứng với route như sau :



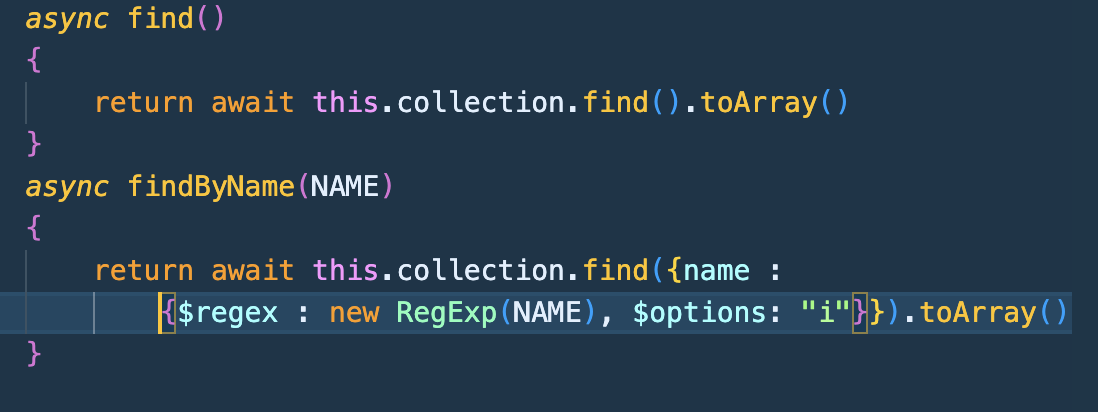
Trong MongoDB xuất hiện document :



### Controller findAll

Hàm sẽ hiển thị tất cả các thông tin của người dùng có trong ContactBook hoặc có thể truyền params vào để tìm kiếm theo tên tương tự

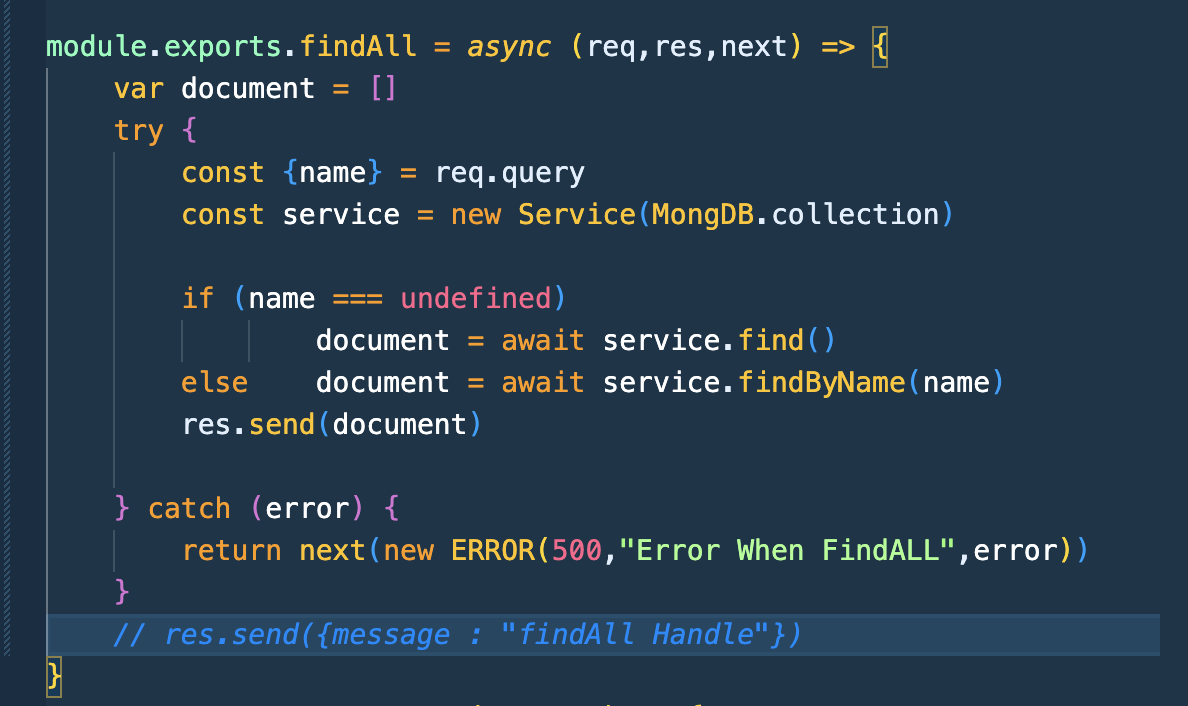
Cài đặt hàm find và findByName để truy vấn như sau :



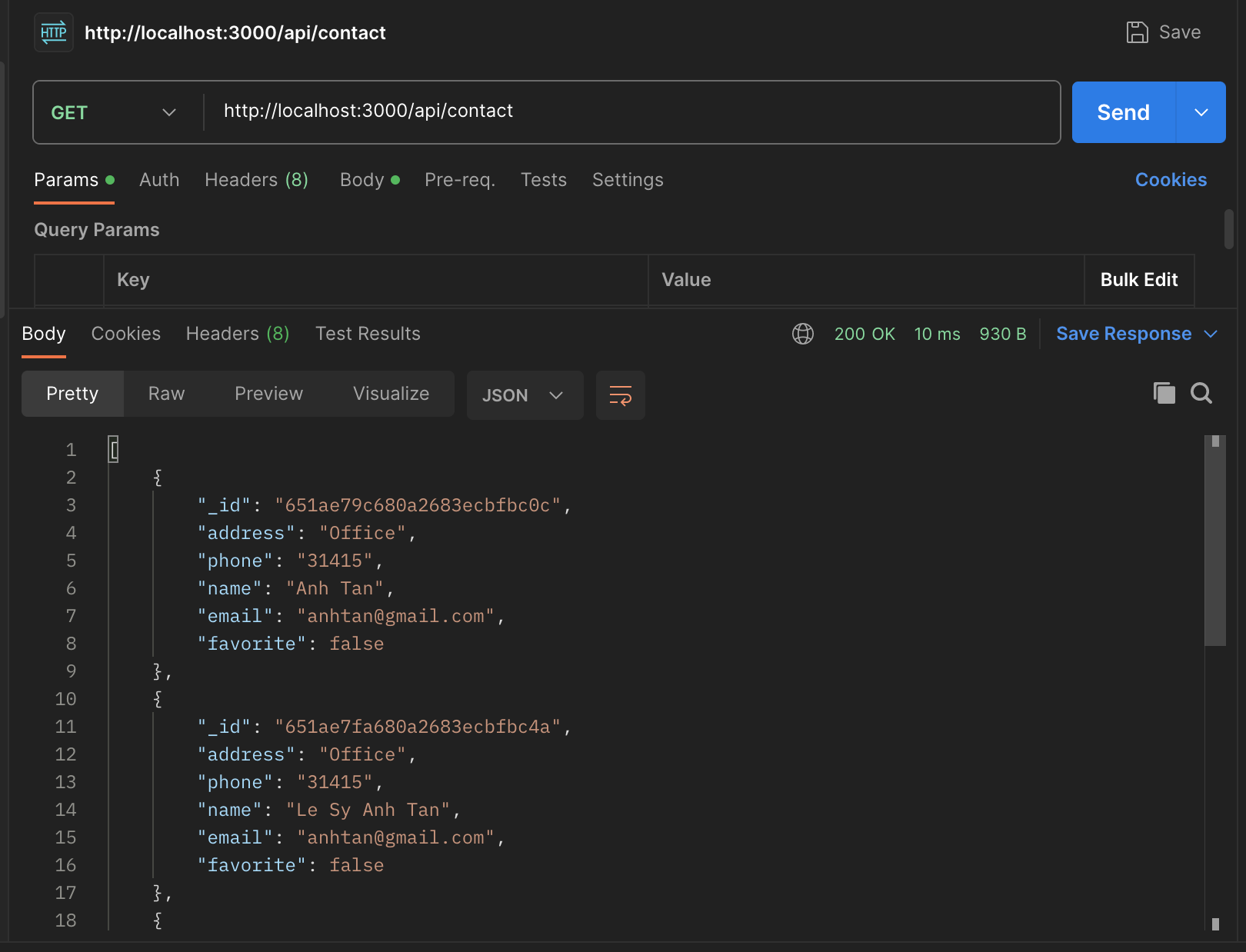
Hàm find và findByName đều trả về mảng các giá trị. cần tìm kiếm

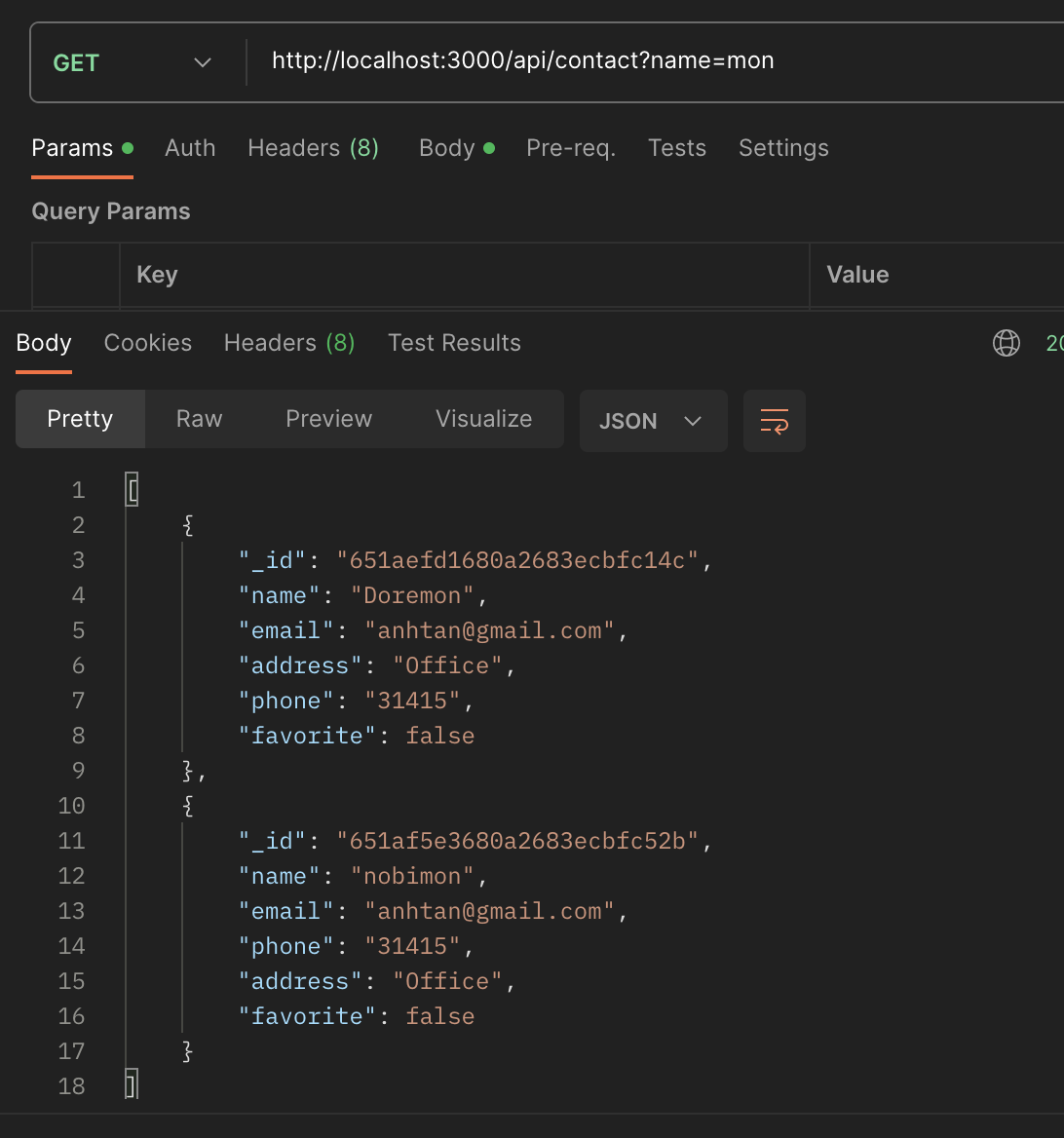
Hàm findByName sẽ lấy tất cả các tên ứng với các params được truy vấn

Lúc này cài đặt bên controller như sau :



Truy vấn bằng postman như sau :





### 2.3 Controller FindOne

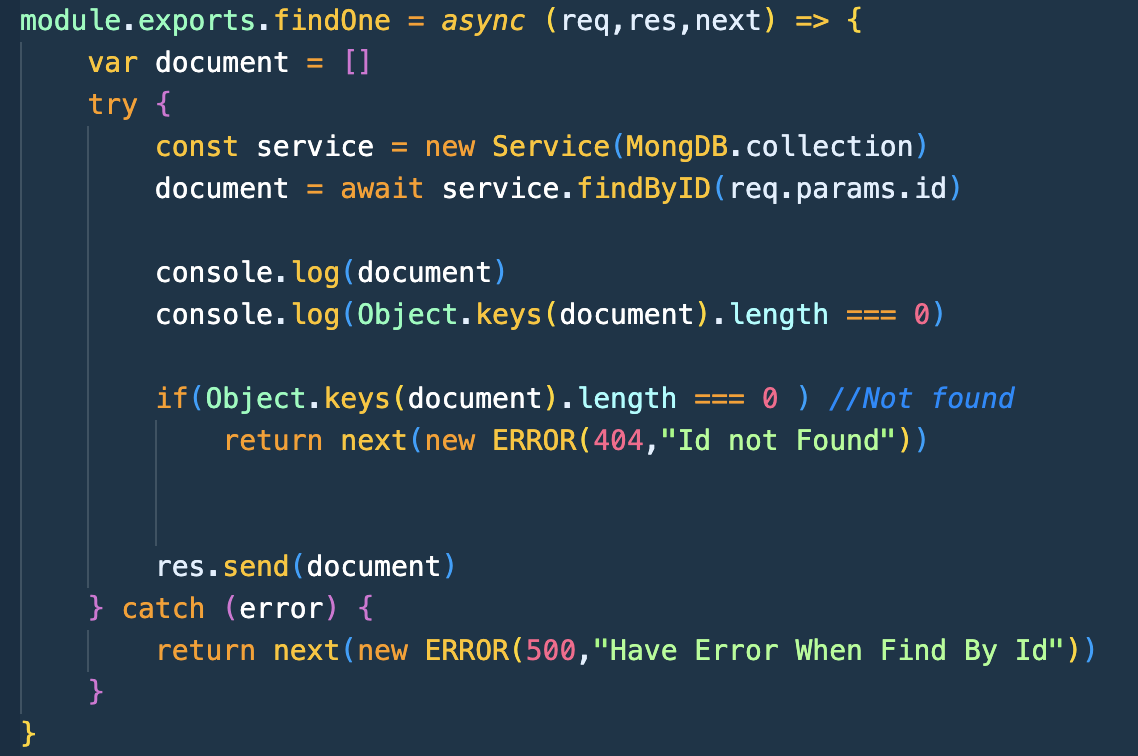
Tìm một đối tượng theo ID của đối tượng đó :

Thiết lập service hàm để tìm theo id đưa vào như sau :



Sử dụng. Class ObjectId có sẵn trong mongodb để chuyển đổi id nhập vào thành id Mongo có thể hiểu được để có thể truy vấn . Nếu id không

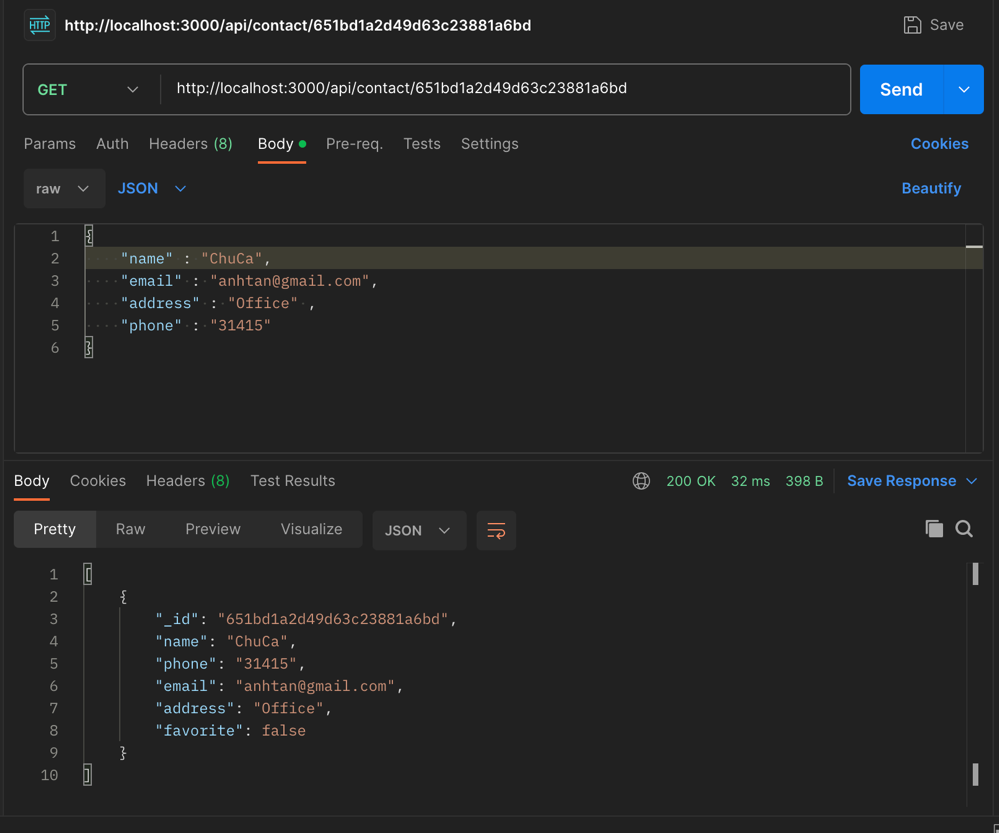
Cài đặt hàm FindOne như sau :



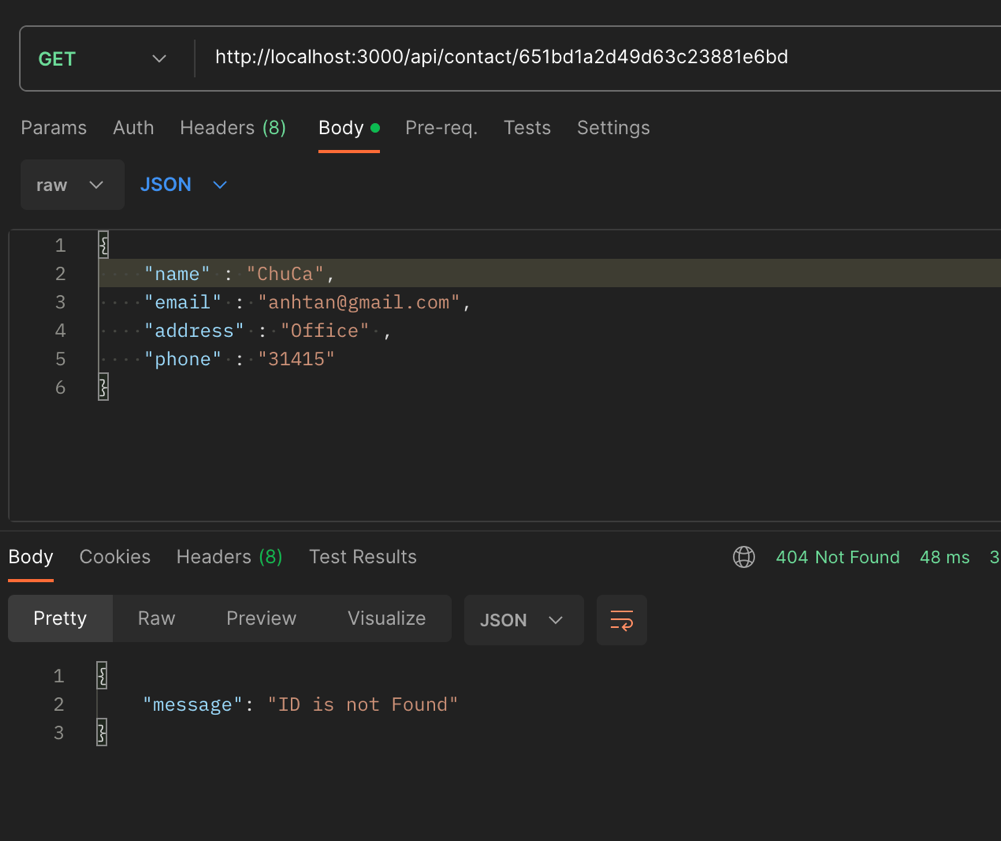
Gọi service và sử dụng menthod findByID vừa thiết kế ở trên để xuất ra được một mảng chứa đối tương tìm kiếm theo ID , nếu tìm không thấy thì sẽ trả về mảng rỗi khi đó ta sẽ bắt lỗi trả về client rằng id chưa tồn tại

Gọi hàm trên Postman như sau :

Đây là khi ID có tồn tại :



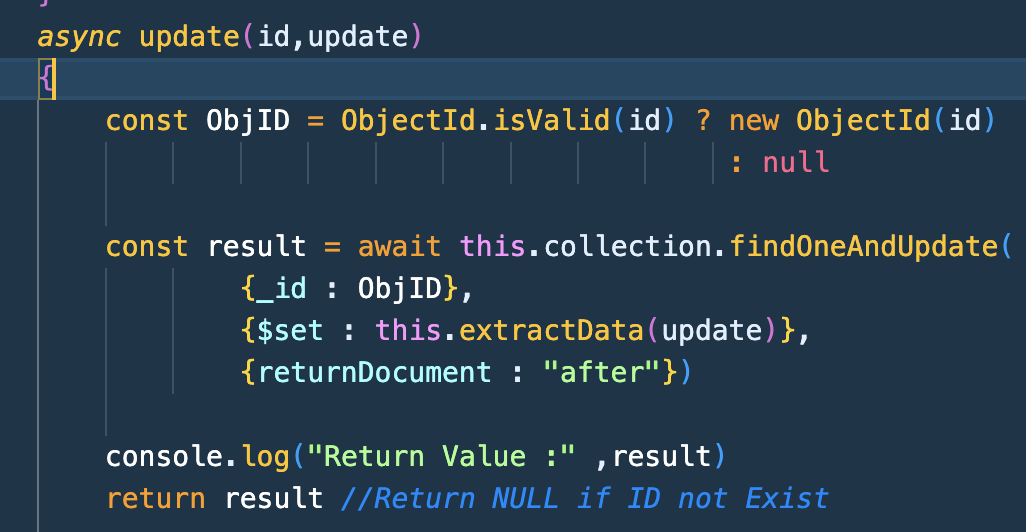
Khi tìm kiếm một id chưa có :



### Controller Update

Cập nhật các thuộc tính của một người dùng khi đã biết id của người dùng đó

Xây dựng một hàm của lớp Service để có thể cập nhật như sau :



Hàm nhận vào 2 đối số là ID và phần body cần cập nhật

Chuyển đổi và kiểm tra id nhận vào có hợp lệ để sử dụng

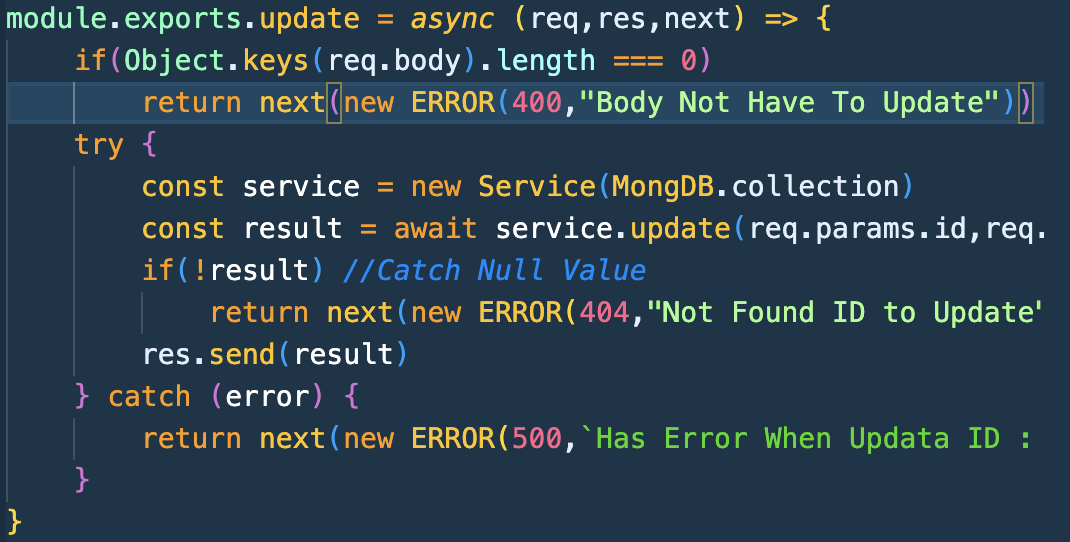
Sau đó khởi chạy hàm cập nhật của mongodb vào collection đang tương tác để thao tác cập nhật như sau :

Tìm kiếm theo ID đã xác định nếu id là null hoặc không có ID thì hàm sẽ trả về là null

Cập nhật các giá trị theo body update nhập vào và gọi hàm extractData để đảm bảo rằng dữ liệu là đúng theo khung

Cài đặt returnValue theo sau khi chỉnh sửa để có thể in ra ngoài object đã cập nhật

Tiếp theo sử dụng service vào cotroller như sau :

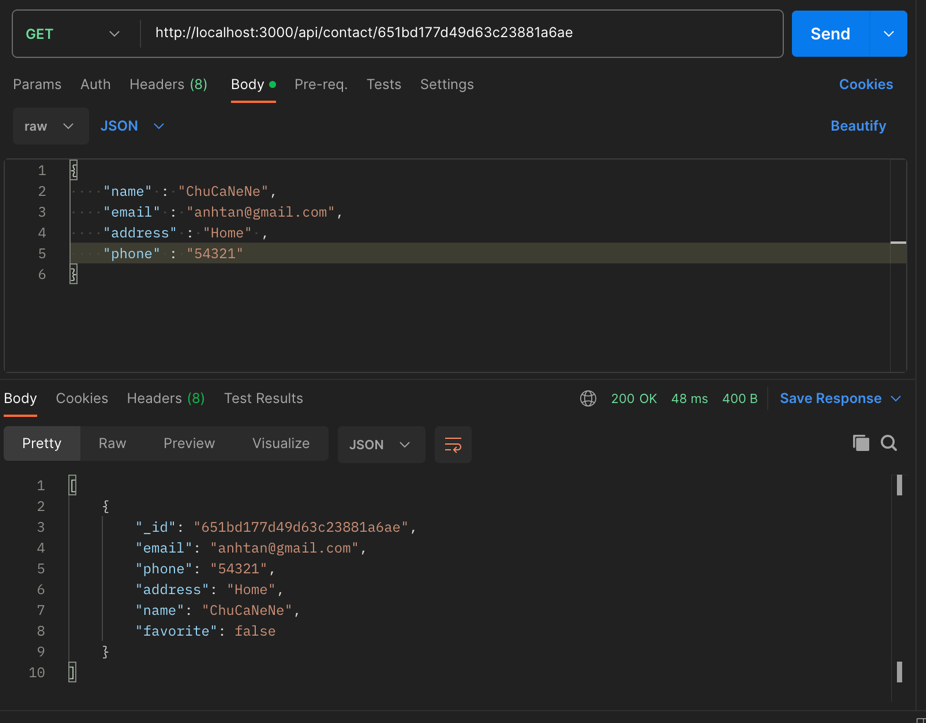


Khi nhận được request từ client thì đầu tiên sẽ kiểm tra phần body xem có yêu cầu update hay không nếu không thông báo lỗi cho client là phần thay đổi còn rỗng

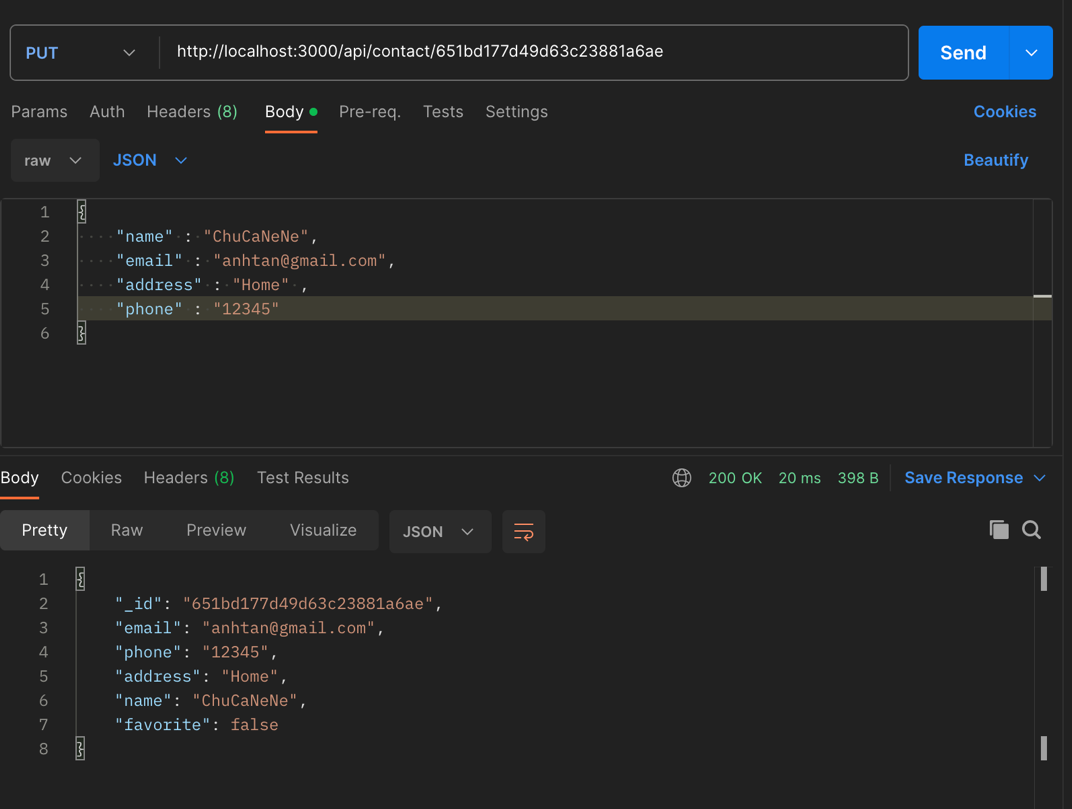
Tiếp đến khởi tạo một service theo collection và gọi service là update đã thiết lập ở trên và nếu trả về là null là ID nhập vào là chưa xác định được

Thực hiện chạy trên Postman ta có như sau :

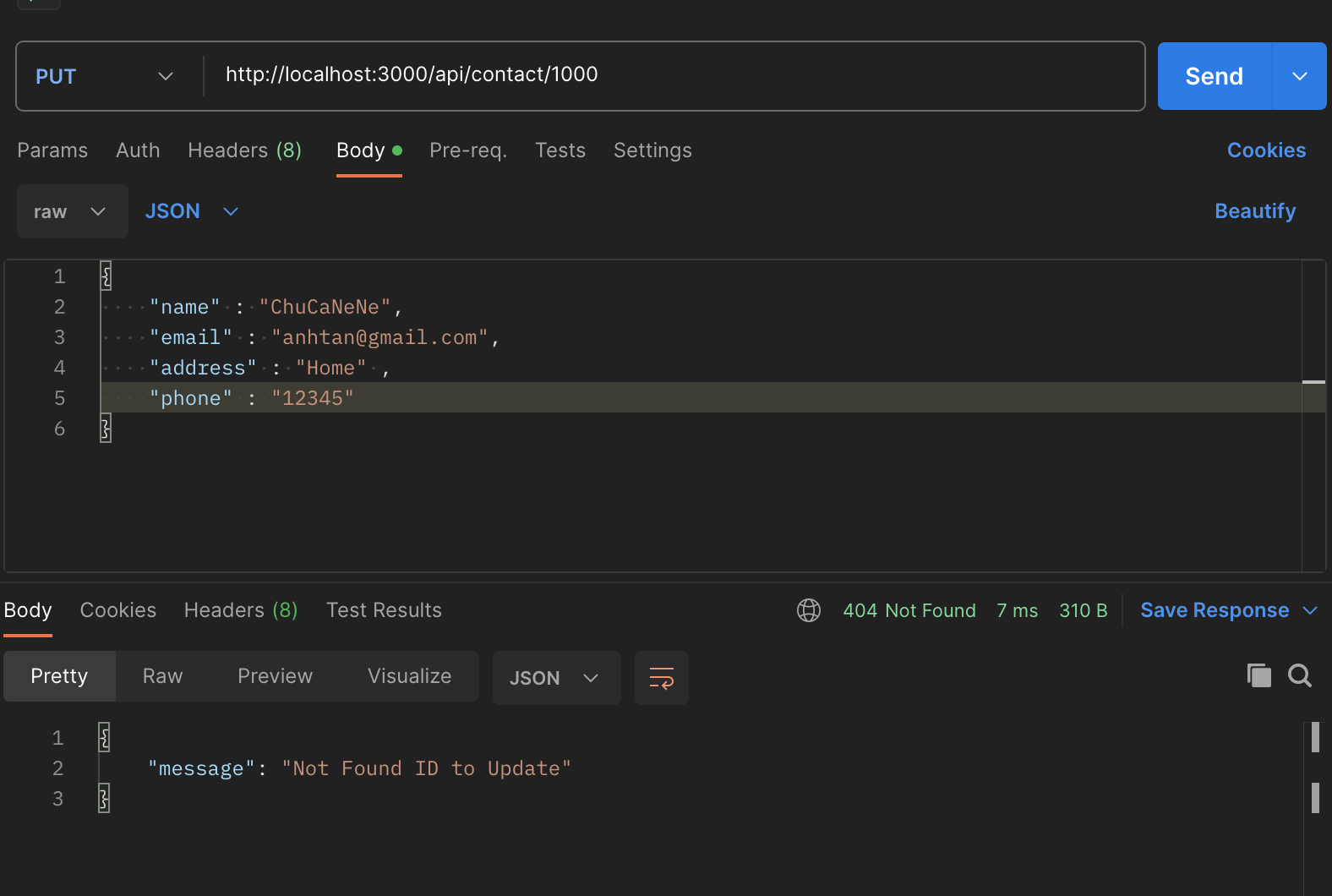
Đây là object ta sẽ thay đổi các thuộc tính



Thay đổi số điện thoaị của thuộc tính và giá trị được cập nhật thành công như sau :



Đây là trường hợp khi ID là không có

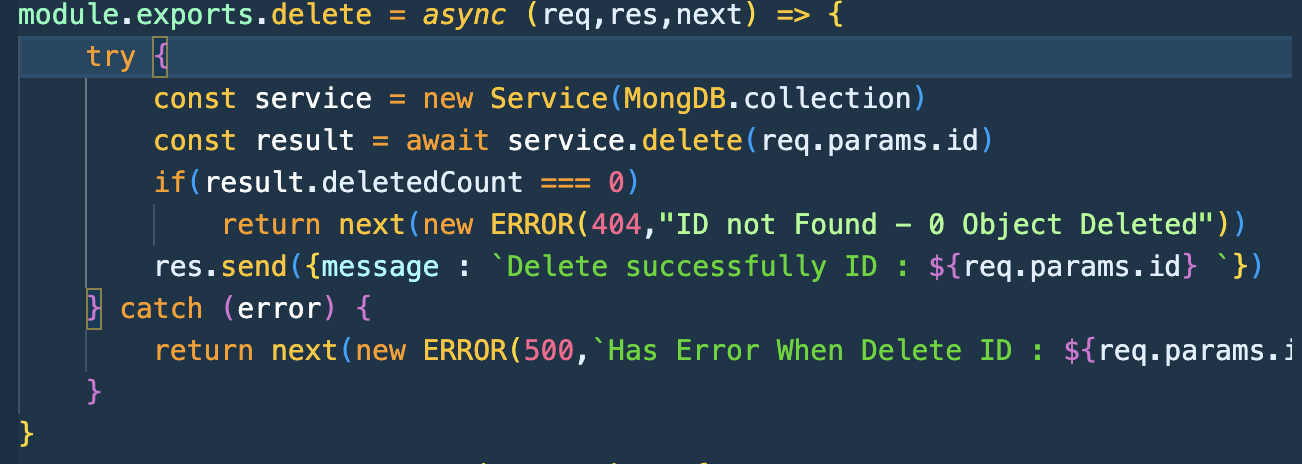


### Controller Delete

Cài đặt method delete trong Service và nhận một đối số là id để thực hiện xoá như sau :



Tiếp đến cài đặt và xét lỗi vào trong controller như sau :

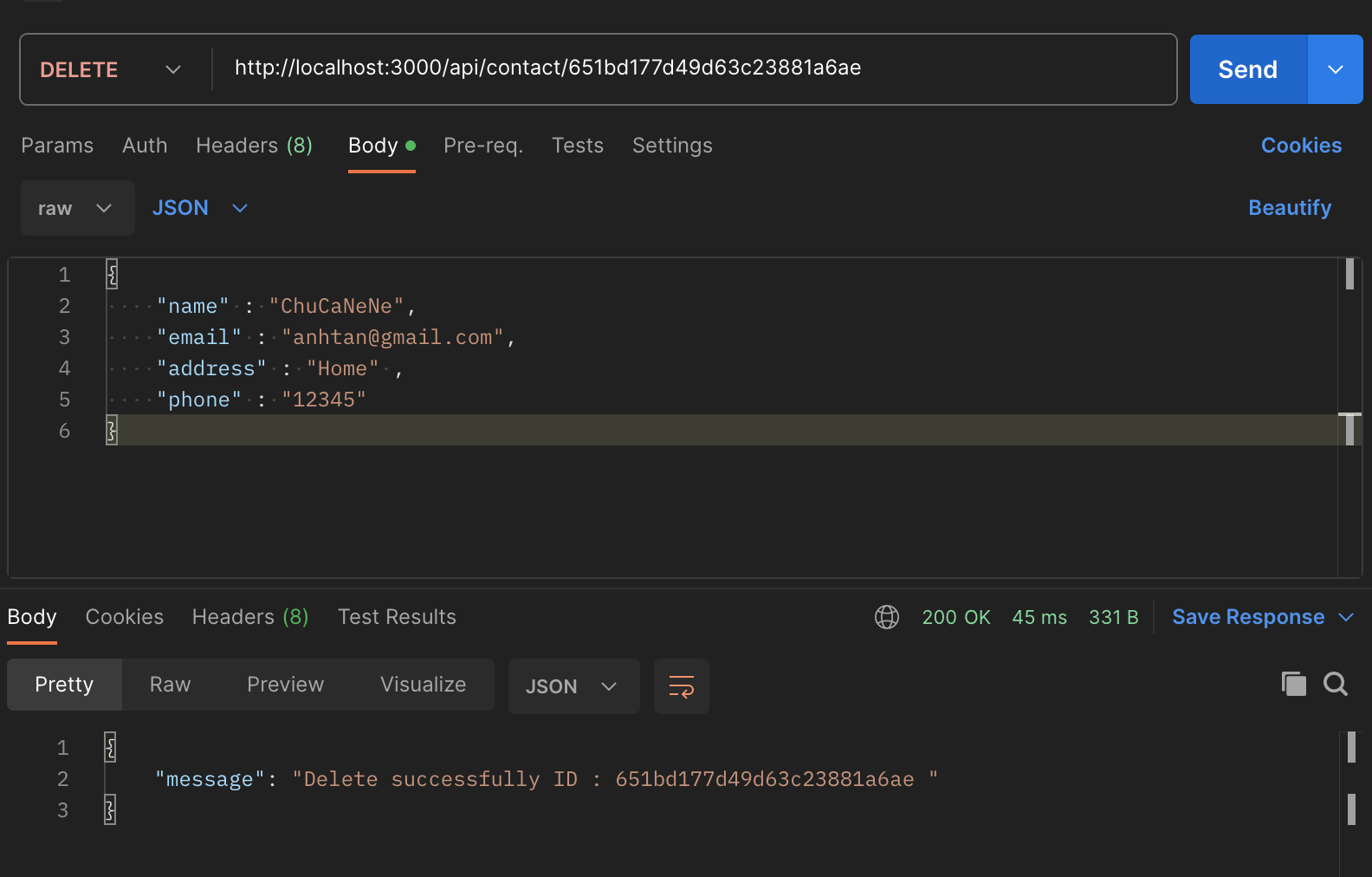


Thực hiện kiếm tra coltroller trong Postman như sau :

Xét object sau đây :

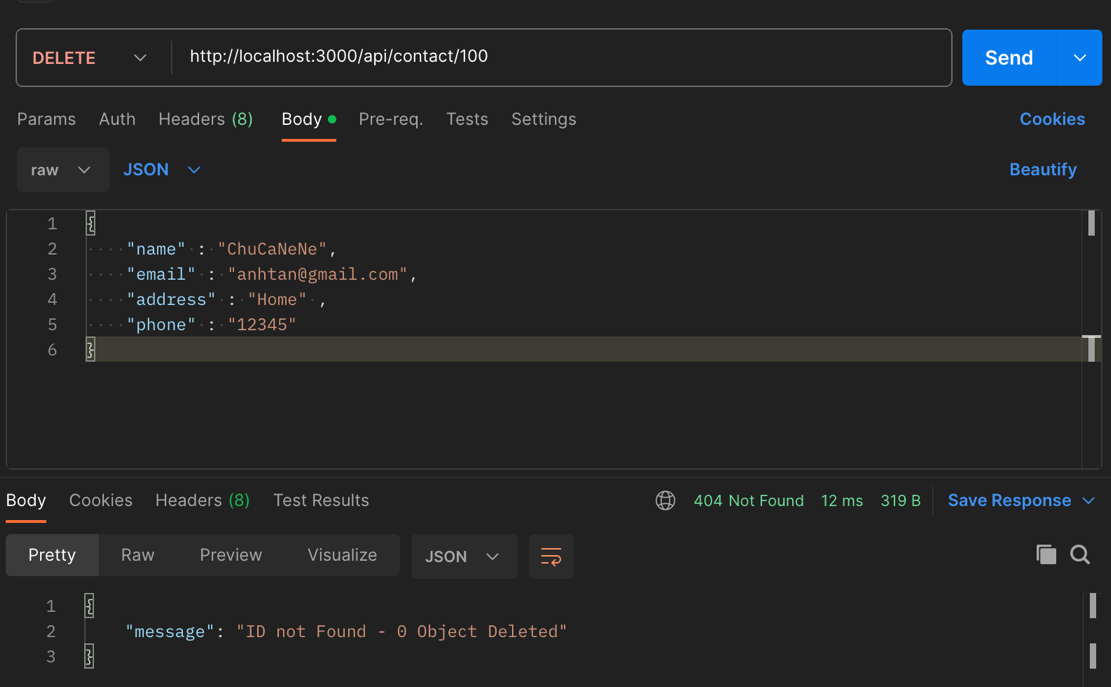


Thực hiện gọi đến controller với id như trên ta được :



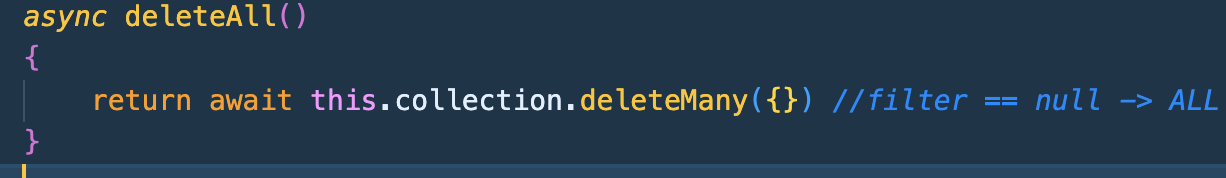
Lúc này đã xoá thành công đối tượng trong Collection

Nếu Nhập một id là không có hoặc sai :

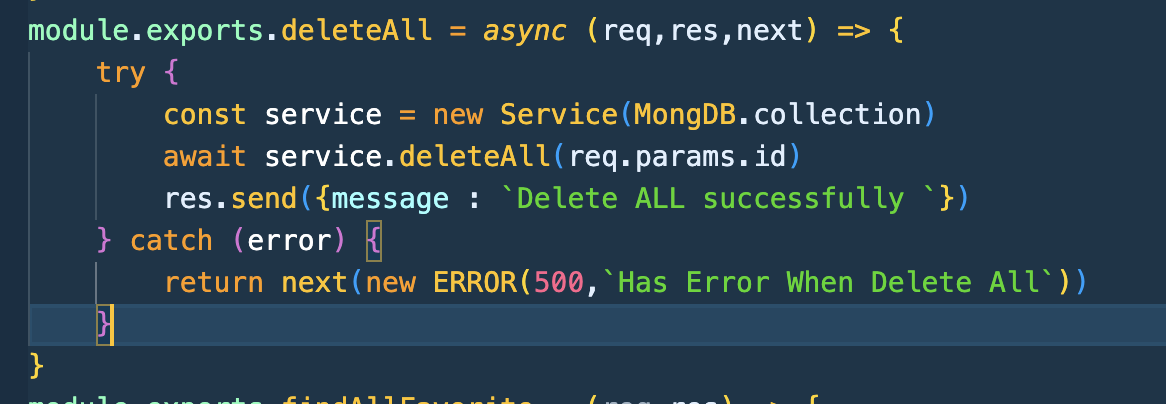


### Controller deleteAll

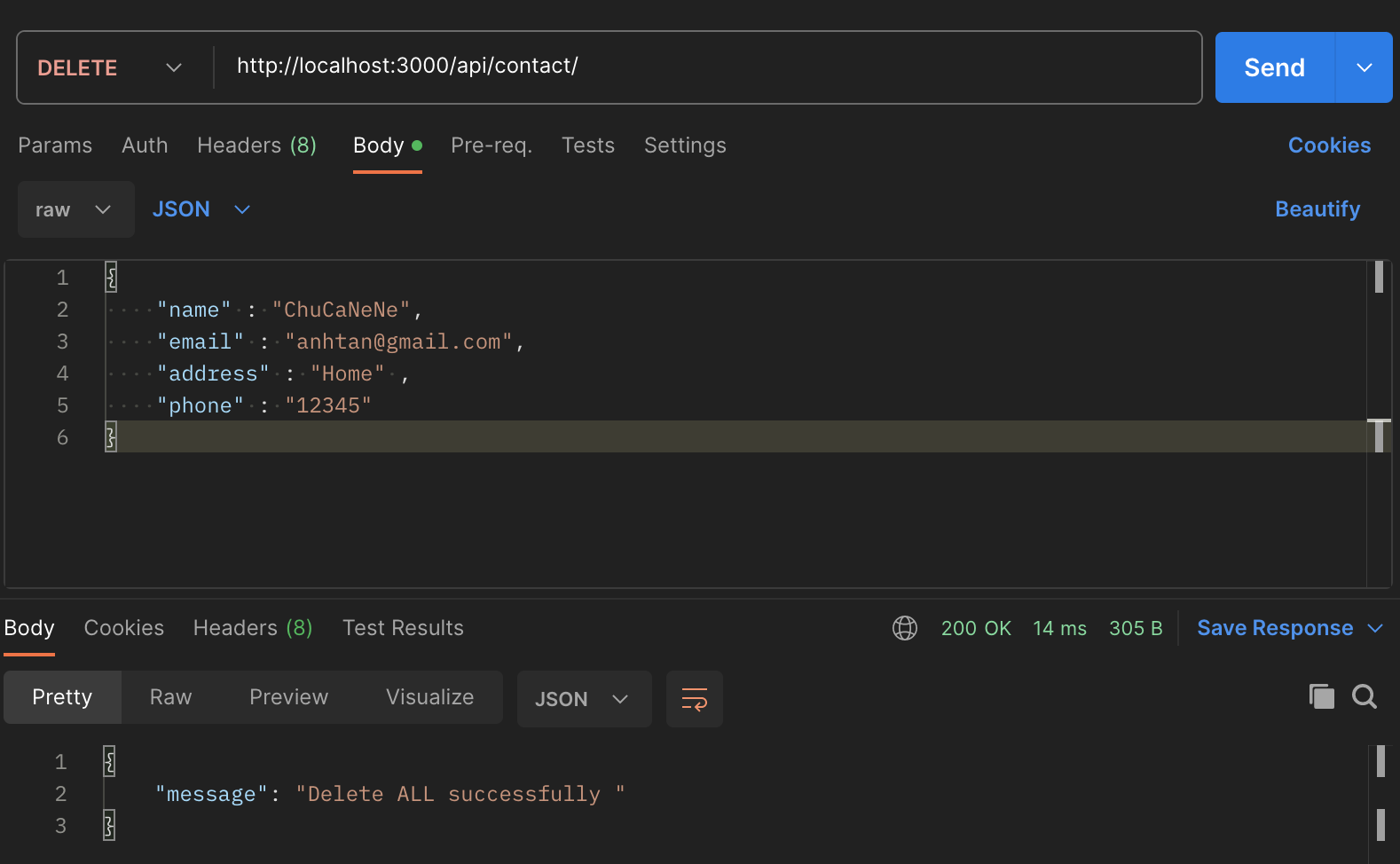
Xây dựng phương thức deleteAll trong service như sau :



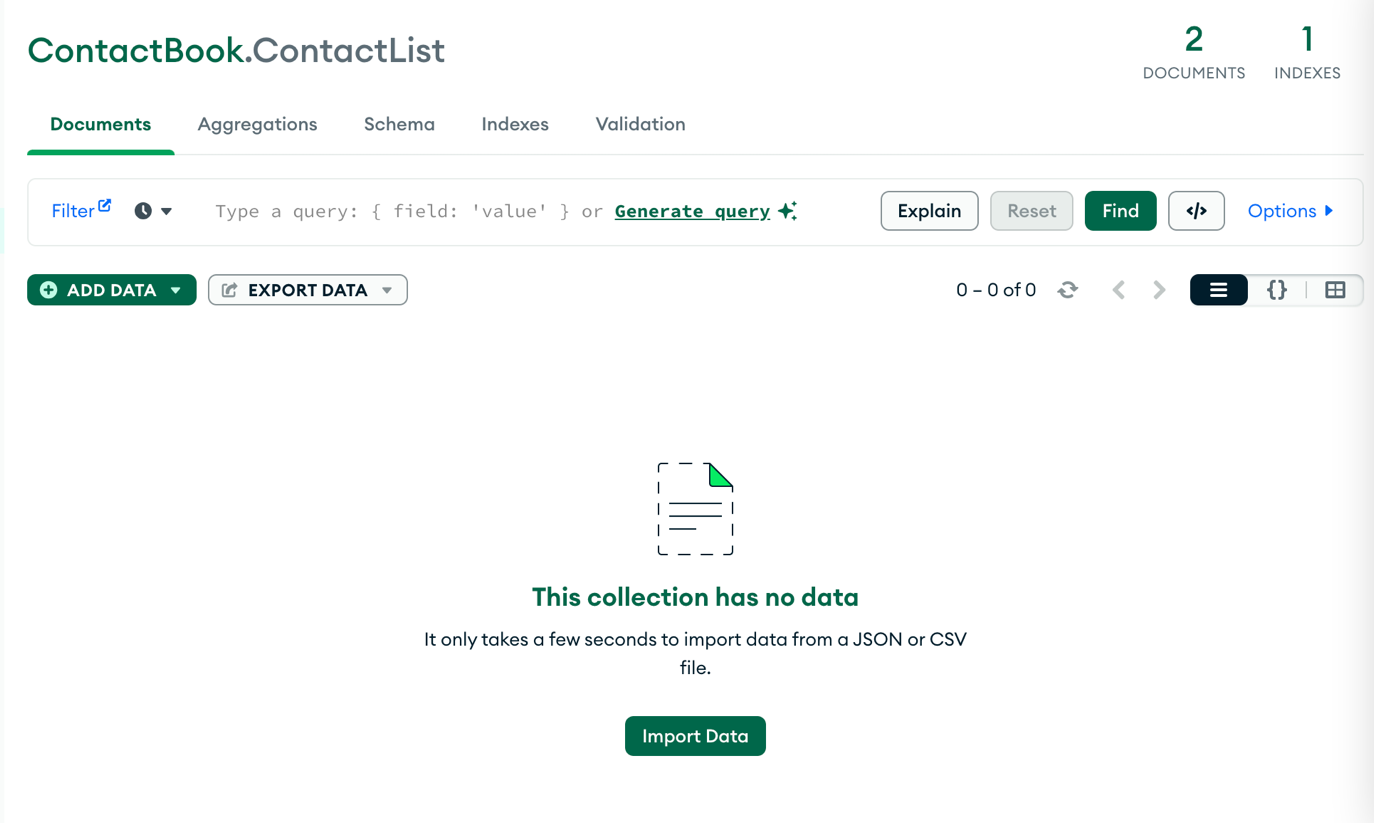
Thực hiện cài đặt vào controller và bắt lỗi khi có ngoại lệ như sau :



Thực hiện xoá tất cả các document trong Collection ta thực hiện trên Postman :

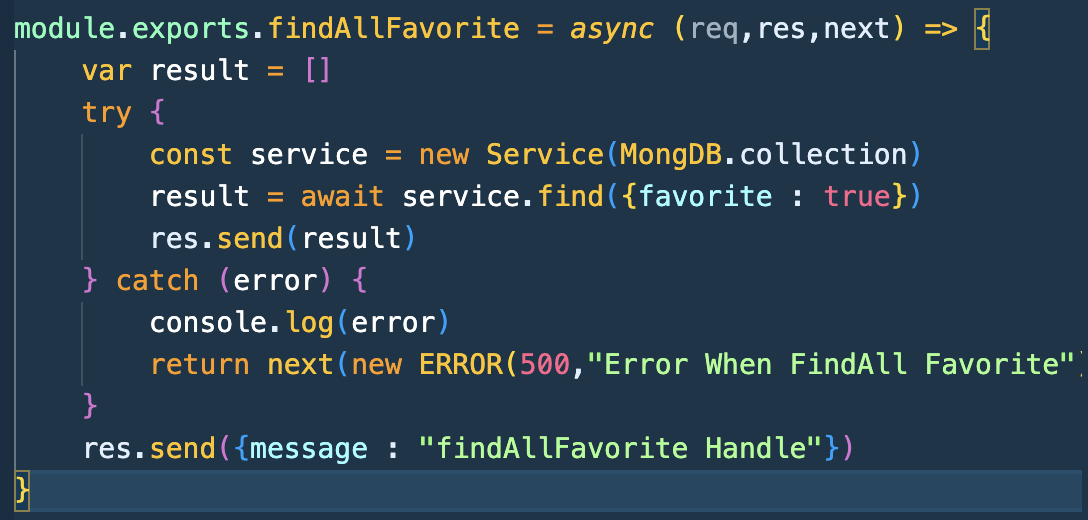


Collection ContactList lúc này :

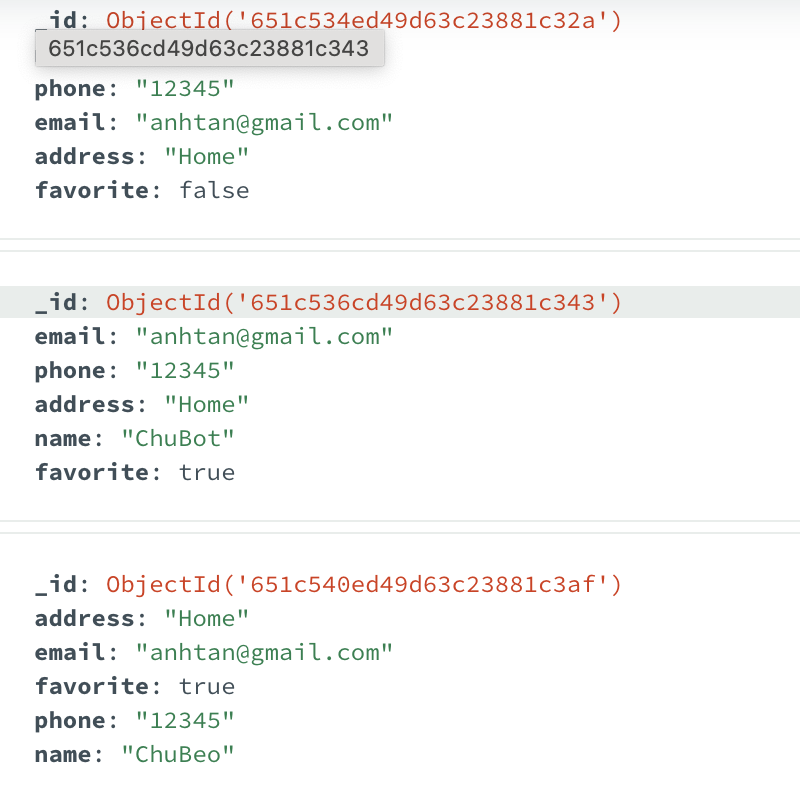


### Controller FindAllFavorite

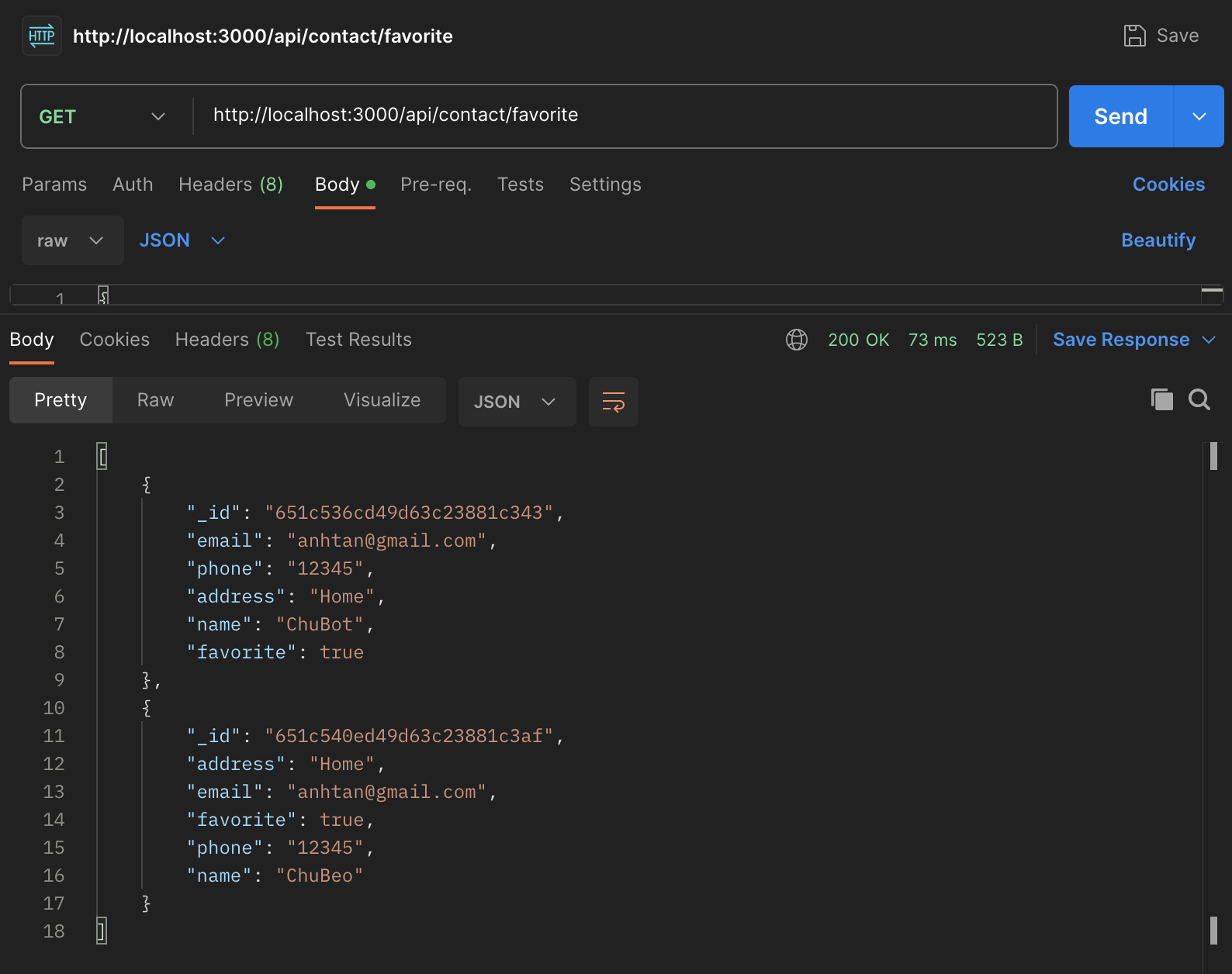
Tái sử dụng lại service find() với fillter {favorite : true } và cài đặt vào controller như sau :



Xét tập collection như sau :



Sử dụng Postman để truy vấn ta được :



## Up Dự Án Lên Github

Link github : <https://github.com/anhtan2223/B2113342_LeSyAnhTan_backend1>

Lúc này cấu trúc của thư mục sẽ như sau :

