Config database:

Bé con truy cập vào file:

**./src/main/resources/application.properties**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Tạo 1 đối tượng mới**

1. **Tạo model trong**

**./src/main/java/com/base/main/models**

Thêm 2 cái này để định nghĩa nó là model và khi chạy sẽ tạo bảng trong SQL luôn



Mấy cái định nghĩa id, các cột em nhìn sẽ hiểu

Để có relationship như laravel cũng đơn giản

A screenshot of a computer code

Description automatically generated with medium confidence

Như thế này kho có 1 sinh viên sv, thì sv.getThongTinSinhVien() sẽ là 1 đối tượng được nối 1-1 với sinh viên đó,… tương tự

1. Tạo repository

src\main\java\com\base\main\repositories

Mục đích: để sự dụng JPA hibernate (ORM) để truy vấn database

Cái này em chỉ cần tạo interface thế này thôi



Như thế đã sử dụng được hầu như các hàm trong ORM, nếu cần viết các câu lệnh truy vấn mà ORM không có thể viết vào đó.

1. Tao vO

VO là value object, cần tạo để sự dụng trong service và controller. Dùng để truyền tải 1 đối tượng, (liên hệ Hải để hiểu rõ hơn)

1. Tạo service

\src\main\java\com\base\main\services

Trong này chủ yếu viết các hàm CRUD đến DB

1. Tạo controller

Như route trong laravel thôi

Tạo view ở \src\main\webapp\WEB-INF\views

Tạo file jsp nha

Để controller trả về 1 view thì chỉ cần return về 1 chuỗi chưa đường dẫn của file đó ở trong folder views

Ví dụ ta có filde index.jsp ở trong fordel sinhvien

src\main\webapp\WEB-INF\views\sinhvien\index.jsp

Để trả về view khi xử lí xong ở controller ta chỉ cần **return** "sinhvien/index";

@GetMapping

**public** String index(ModelMap modelMap) {

modelMap.addAttribute("sinhviens", sinhvienService.findAll());

**return** "sinhvien/index";

}

Mọi thắc mắc xin liên hệ người yêu của bạn