Tất cả nằm trong project loind

1. Tạo Project

<https://laptrinhjavaweb.com/huong-dan-tao-project-spring-boot-co-ban-10>

1. Restful API:

* Tạo DTO mới
* Một số annotation: (cách cũ)
* @Controller: đánh dấu đây là controller
* @RequestMapping(value =”/new”, method = …): tên api, phải có method đấy nhá
* @ReponseBody: xác định kiểu trả về
* @RequestBody: kiểu nhận đc từ request



* Cách mới:
* @RestController
* Không cần dùng @ResponseBody
* Thay @RequestMapping bằng các HTTPMethod mình dùng (POST, PUT, GET, DELETE)



1. Data JPA

* Vào pom.xml, thêm

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>${mysql.version}</version>

</dependency>

<dependency>

<groupId>jakarta.xml.bind</groupId>

<artifactId>jakarta.xml.bind-api</artifactId>

<version>2.3.2</version>

</dependency>

<!-- Runtime, com.sun.xml.bind module -->

<dependency>

<groupId>org.glassfish.jaxb</groupId>

<artifactId>jaxb-runtime</artifactId>

<version>2.3.2</version>

</dependency>

* Vào Appication.java thêm: @EntityScan("com.laptrinhjavaweb.entity")
* Vào application.properties, thêm:

spring.datasource.url = jdbc:mysql://localhost:3306/newspringboot

spring.datasource.username = root

spring.datasource.password = mothernumber1@

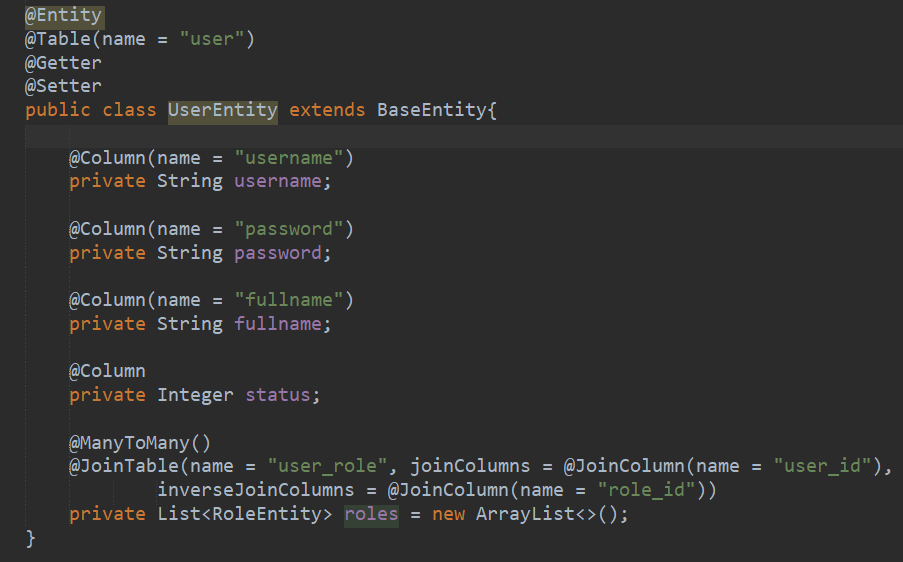
#spring.jpa.hibernate.ddl-auto = none

spring.jpa.hibernate.ddl-auto = create-drop

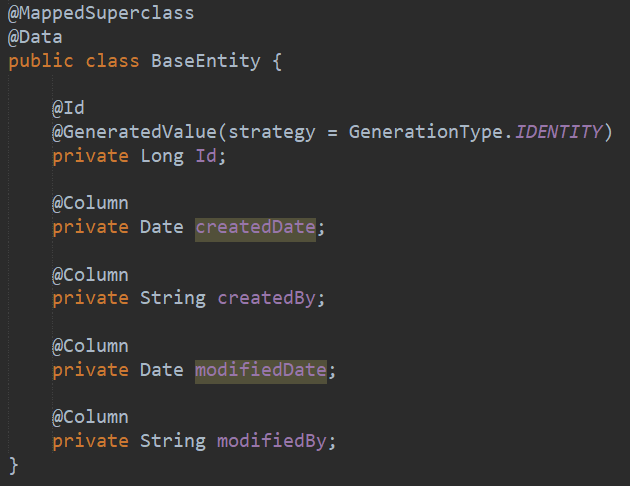
#spring.jpa.hibernate.ddl-auto = create

spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.MySQL5Dialect

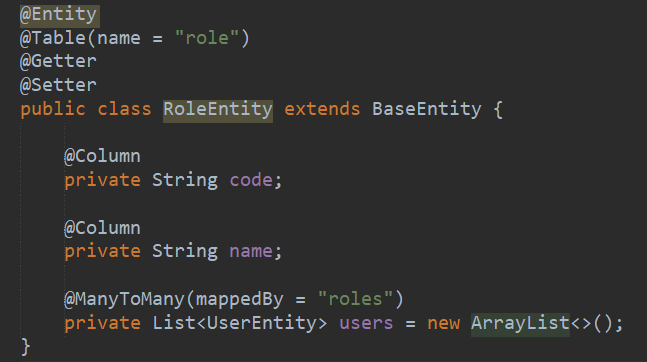
* Tạo package entity, tạo class bất kì, vd: USerEntity
* @Table: tên bảng trong mysql
* BaseEntity : tự tạo, có chứa ID, …
* @ManytoMany: quan hệ nhiều nhiều, vd: 1 user có nhiều role, 1 role có nhiều user. Cần có mấy cái joinColumns, … Bên RoleEntity cũng cần khia báo 1 @ManyToMany
* Mình đang đứng ở UserEntity nên joinColumns = @JoinColumn(name = "user\_id"), cái name nó là user\_id



* BaseEntity: cần có @MappedSuperClass

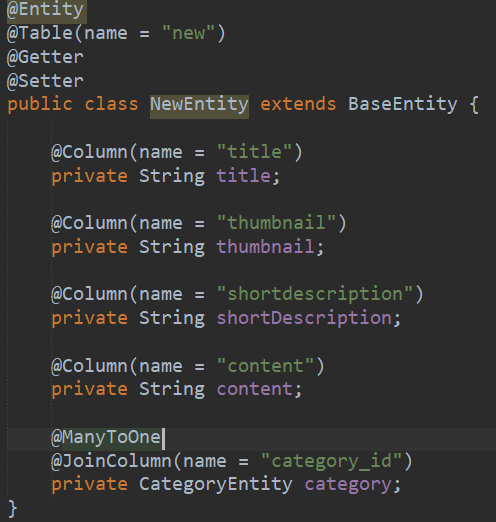


* RoleEntity: cần mappedBy vs biến roles mk khai báo bên UserEntity

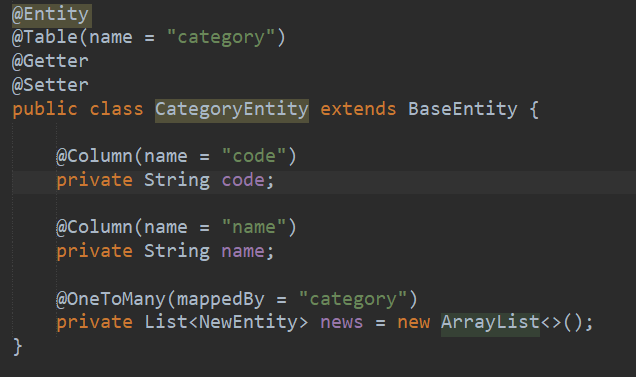


@OneToMany và @ManyToOne

* Vd như: 1 category có nhiều bài viết, 1 bài viết thì chỉ thuộc về 1 cate duy nhất
* NewEntity: cần có @JoinColumn

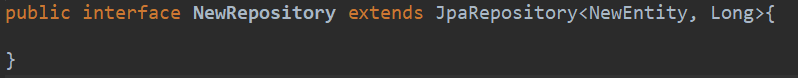


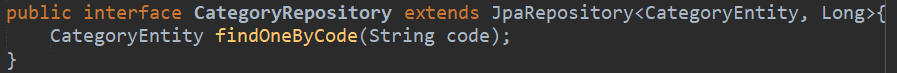
* CategoryEntity: chỉ cần mappedBy cái category mk khai báo bên NewEntity là đc



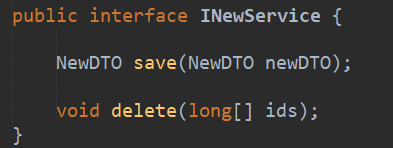
1. Thêm, sửa, xóa:

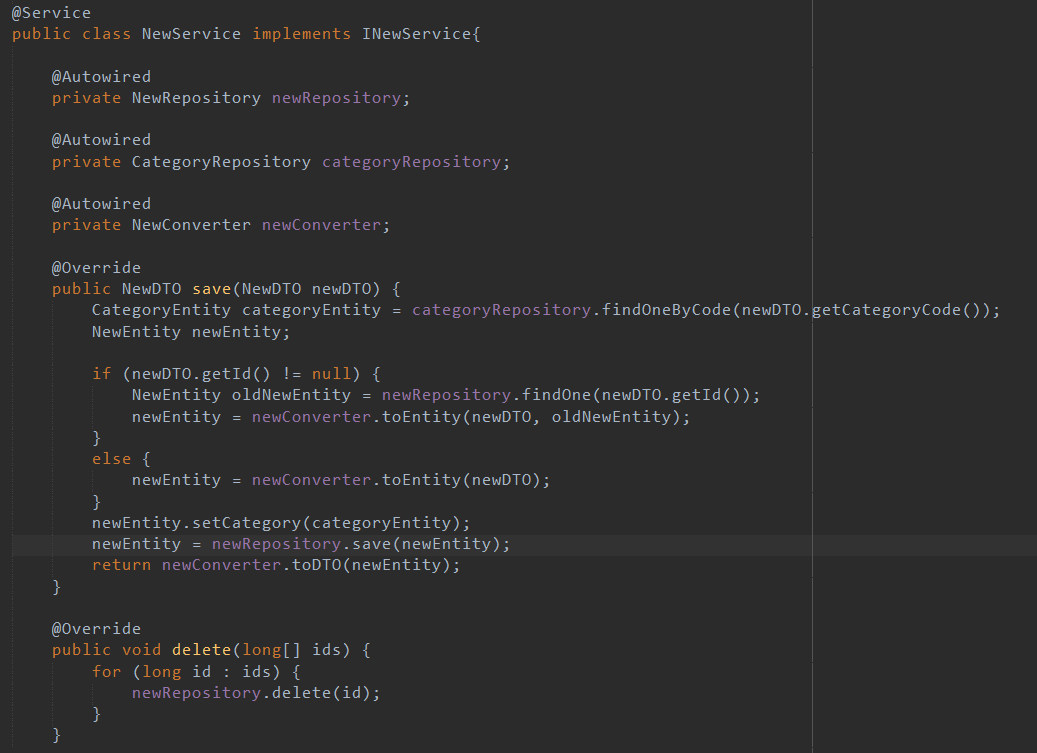
* Tạo repository (interface nhá)





* Tạo service và serviceimpl (cái này cần @Service)



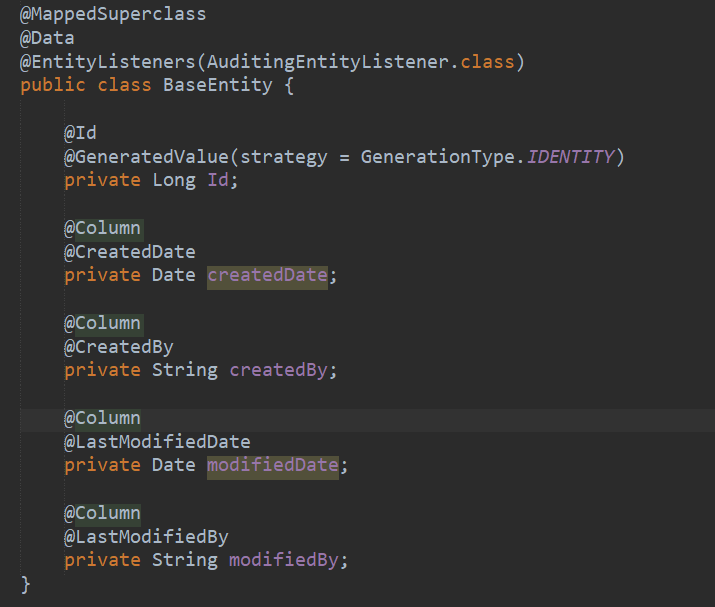


* Bên API thì đơn giản rồi:

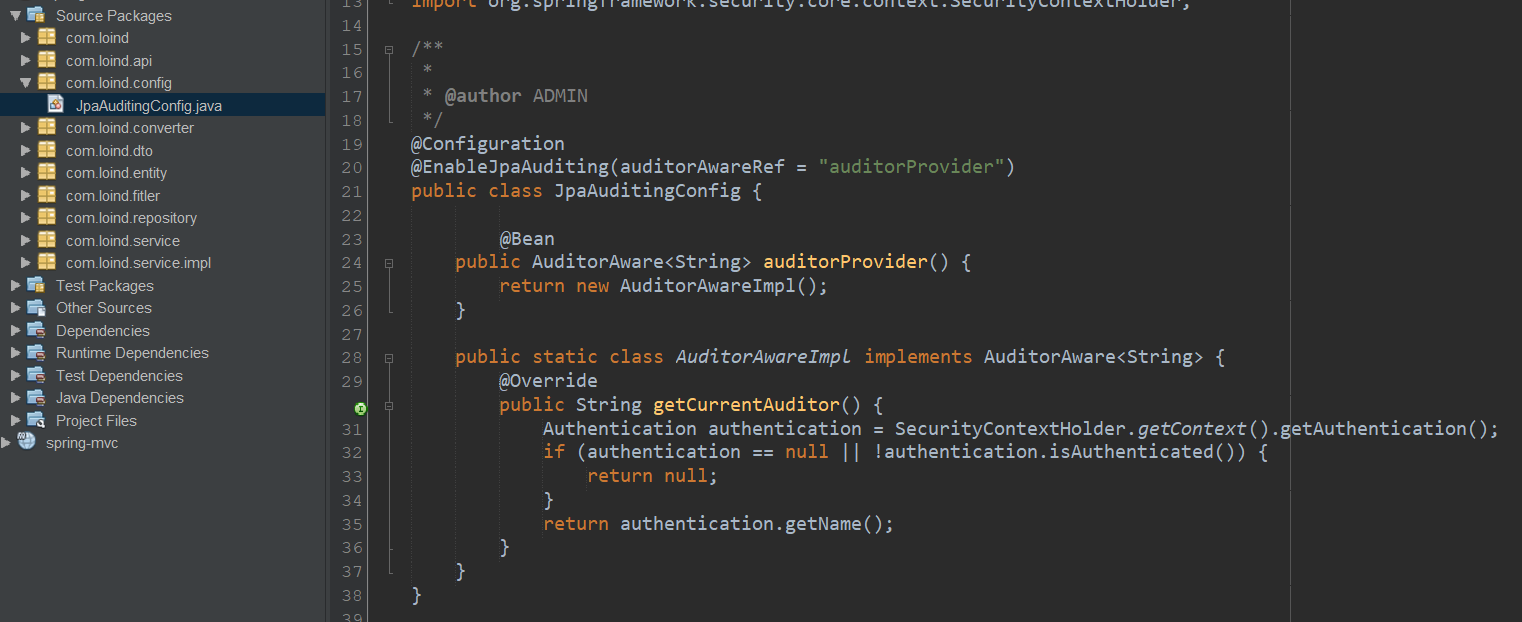


1. Entity nâng cao một tý:

* Bên BaseEntity, thêm @CreatedDate, … và trên đầu class thêm @entityListener



* Để mà dung đc 4 cái @ kia á, thì ccafn tạo 1 package mới, tên là config, chứa 1 class JpaAuditing.java:



* và bên pom.xml cần thêm

<dependency>

<groupId>org.springframework.boot</groupId>

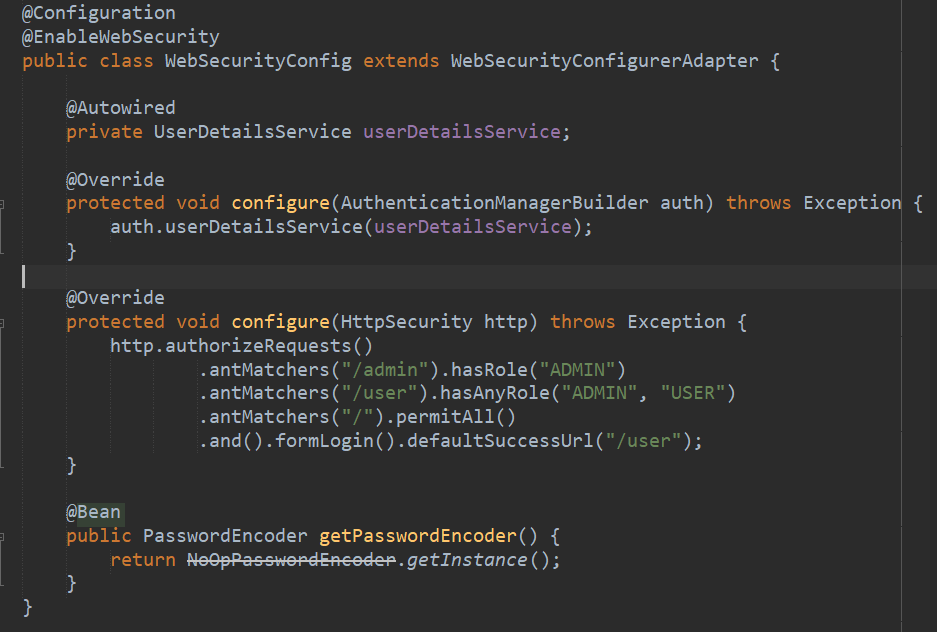
<artifactId>spring-boot-starter-security</artifactId>

</dependency>

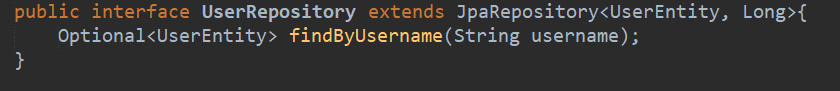
* Lúc chạy sẽ bị báo lỗi: default password gì đó thì chỉ cần vào file Application.java (file chạy), sửa: @SpringBootApplication(exclude = {SecurityAutoConfiguration.class })

1. JPA Security

* Trong pomxml, add security vào
* Vào package config, add file WebSecurityConfig.java



* Tạo UserRepository



* Tạo UserDetailsServiceImpl.java



* Vào package utils, tạo MyUserDetails



* Vào file chạy thêm @EnableJpaRepositories(basePackageClasses = UserRepository.class) là xong 😊