

@Requestbody: CLIENT -> SERVER (parse JSON -> DTO)

@Responsebody (parse DTO -> JSON): từ server trả ra client dưới dạng json

@Controller: định nghĩa 1 class là API

@ResController = @Controller + @Responsebody

@Postmapping("/....") = @RequestMapping + method(POST, PUT, GET, DELETE)

@RequestMapping: ánh xạ url lên class hay một phương thức xử lý riêng

@Data:

@NoArgsConstructor:

@AllArgsConstructor:

@Entity: định nghĩa 1 class là 1 entity

@Modifying: cho phép thực hiện nhiều truy vấn, để cập nhật nhiều truy vấn

@Query: cung cấp để viết 1 câu SQL

@Service: đánh dấu các class thuộc tầng service để code xử lý trong đó

@Scope: được định nghĩa phạm vi một @Component class hay một @Bean method.

@Repository: dùng để đánh dấu các class thuộc tầng giao tiếp với database

@Transactional:

@Autowired: tự động tìm kiếm và tiêm các bean tương ứng mà chúng ta đã khai báo class

@Component (viết chú thích trên class): Đánh dấu 1 class là bean. Sử dụng để định nghĩa một khuôn mẫu chung cho toàn bộ dự án.

@Bean (viết chú thích trên method): thực chất là 1 object java và được container tạo ra và quản lý.

Spring bean: chính là những Java Object mà từ đó tạo nên khung sườn của một Spring application. Chúng được cài đặt, lắp ráp và quản lý bởi Spring IoC container.

Ví dụ: chúng ta có 1 class A bên trong chưa rất nhiều properties là các object của các class B, C, D,… và nếu khỏi tạo ra 1 object A thì trước đó phải khỏi tạo ra được các object B, C, D mà biết đâu B, C, D lại phụ thuộc vào nhiều object khác nữa. Như vậy để tạo ra 1 object hoàn chỉnh thì rất khó kiểm soát. Đối với bean thì chúng ta chỉ cần gửi yêu cầu đến container sẽ tự động tìm trong context xem có bean nào trong đó hay không nếu thấy thì nó sẽ tạo ra 1 object tương ứng và trả về .

Spring Data JPA không phụ thuộc vào DB mà nó sẽ phụ thuộc vào Entity

JDBC -> JPA -> Spring Data JPA

JDBC: Statement và PreparedStatement

JPA: Entity Manager

Spring Data JPA: viết sẵn các hàm

HQL (Hibernate query language): kiểu như khi đổi DB từ MySQL sang SQL Server thì chúng ta không cần phải thay đổi qua nhiều, đỡ tốn công sức

@Table : map với các thuộc tính trong DB

@OneToMany

@ManyToOne

@ManyToMany