1. **Vẽ và trình bày cách Spring MVC xử lý request**

**2 Trình bày cơ chế Dependency Injection trong Spring**

- Là một cơ chế sử dụng đễ loại bỏ sự phụ thuộc giữa các mã của chương trình, giúp cho việc quản lý mã chương trình ít bị phụ thuộc vào nhau

**3 Có bao nhiêu cách để thực hiện Dependency Injection? Trình bày?**

**-** Bean được Denpendency Injection phải được tạo trược (mannual hoặc tự động)

- Dùng annotation @Autowired để chỉ định chỗ nào được DI

Có 4 cách để thực hiện DI nhưng thường sử dụng 2 cách:

+ Setter injection

+ Contructor injection

**4 Framework là gì ? Framework khác Library chỗ nào ?**

Là một khung chương trình có tính trừu tượng cung cấp các tính năng chung và thông dụng để tạo ra các ứng dụng khác. Người dụng bổ sung code và tuân theo các quy tắc để sử dụng

Library chỉ cung cấp các chức năng tiện ích hay class để người dùng gọi trong quá trình sử dụng

FrameWord cung cấp cơ thế IoC điều khiển ngược cho phép gọi chính code của mình

**5 Spring Framework là gì ?**

- Là một Framework giúp lập trình viên java phát triển java backend, java web một cách dễ dàng, nhanh chóng, hỗ trợ lập trình viên java có các thư viện, công cụ để lập trình với java. Java web mà không phải mất thời gian xây dựng ban đầu.

**6 Lợi ích của Spring Framework?**

- Giúp giải quyết các vấn đề thông dụng giúp lập trình viên tập trung xử lý nghiệp vụ.

-Tăng tốc độ phát triển

- Cung cấp môi trường làm việc tiêu chuẩn, giúp dễ dàng giao tiếp giữa các bên cùng tham gia phát triển

- Các framework thường có cộng đồng lớn, các giải pháp đã được đánh giá và thử nghiệm, hệ sinh thái đầy đủ.

**7 Nguyên lý đảo ngược quyền điều khiển (Inversion of Control) là gì ?**

-(Inversion of control) Là một nguyên lý thiết kế trong công nghệ phần mềm trong đó có các thành phần nó dựa vào để làm việc bị đảo ngược quyền điều khiển khi so sánh với lập trình hướng thủ tục truyền thống.

**8 Bean là gì?**

-là những module chính của chương trình được tao và quản lý bởi spring ioc container

**9 Trong Spring có bao nhiêu Bean Scope?**

**10 @Autowire là gì?**

- đánh dấu cho spring biết rằng sẽ tự động inject bean tương ứng vào vị trí được đánh dấu.

**11 @Component có ý nghĩa gì?**

- Là một annotation đánh dấu trên các class để giúp spring biết nó là một bean

**12 Trình bày ý nghĩa của Controller**

- Là một thành phần mvc được sử dụng để điều hướng cho request controller nhận dữ liệu từ request , gọi model để sử lý và trả dữ liệu về cho view

**13 Trình bày ý nghĩa của ModelAndView Interface**

- Chúng ta sử dụng Interface Model để truyền dữ liệu từ Controller sang view để hiển thị Spring cho phép chúng ta sử dụng model như là một tham số trong method của controller nên chúng ta dễ dàng lấy, chỉnh sửa dũ liệu để truyền qua view.

**14 Trình bày ý nghĩa của ModelMap Interface**

- ModelMap cũng tương tự như Model. Chúng ta có thể sử dụng ModelMap như một tham số trong method của Controller.

**15 Trình bày ý nghĩa của ViewResolver Interface**

Controller do người dùng định nghĩa trả về tên của view và các thông tin chi tiết khác tới FrontController. Tên của view được trả về có kiểu là String , nên cần xử lý tiếp bởi ViewResolver.

**16 Phân biệt POST với PUT thường sử dụng để làm gì?**

Post dùng để tạo dữ liệu còn Put dùng cập nhật dữ liệu.

**17 @RequestMapping làm gì?**

Được sử dụng để ánh xạ các request tới các action tương ứng của controller

Bao gồm các thuộc tính ánh xạ đến

+ url + http + các tham số + các headers + media types

Có thể sử dụng cho class hoặc phương thức

Gồm : + GetMapping + Post + Put + Patch+ DeleteMapping

**18 Trình bày cơ chế Data Binding trong Spring**

Là cơ chế liên kết dữ liệu đầu vào hoặc đầu ra với model.

Giúp cho việc tương tác dữ liệu dễ dàng hơn vì các form sẽ được liên kết với đối tượng biểu diễn dữ liệu ở phía sau. Khi tương tác sẽ tự động chuyển đổi qua lại cho phụ hợp, dựa trên dataBinder

Cơ chế: Hoạt động dựa trên data Binder

- Bean: ModelAtribute

-BindingResult: Hiển thị dữ liệu lỗi

DataBinder

( Gồm Formatter, Validator, Converter, PropertyEditor)

Dữ liệu input

từ form

* 🡪

**19 Thuộc tính consumes trong các Request Mapping là gì ?**

Thuộc tính consumes ánh xạ đến Content Type của request

Để quy định định dạng dữ liệu mà một request có thể process khi người dùng request tới, ví dụ như định dạng dữ liệu JSON,

// produce ánh xạ đén accept // params ánh xạ đến tham số của đường dẫn // headers ánh xạ đến header của request

**20 Thymeleaf là gì?**

Là một bộ xử lý view được sử dụng cho các web và các ứng dụng độc lập. Cho phép xử lý 6 loại template mode: hthl, xml, text, javascrips, css, raw

**21 Sử dụng lặp trong Thymeleaf như thế nào? Sử dụng điều kiện trong Thymeleaf như thế nào?**

**-** Vòng lặp: Cú pháp th:each = “object, interStat: ${list} ”

Trong đó object là các đối tượng trong danh sách

interStat cung cấp các:

index: chỉ số vòng lặp hiện tại, bắt đầu bằng 0

count: thứ tự của vòng lặp hiện tại

size tổng p tử

even hoặc odd: kiểm tra phần tử chẳn lẻ

first hoặc last

current : trả về p tử object hiện tại

- Biểu thức điều kiện:

Th:if = “${condition}” trả về true trong các trường hợp

Một giá trị boolean true; một số khác 0 ; 1 ký tự non-zero; một chuỗi khác false off no

Một giá trị không phải số ký tự boolean hoặc chuỗi

**22 Formatter là gì ? Converter là gì ?**

Fornarrer Converter dùng để chuyển đổi dữ liệu kiểu này sang kiểu khác

Kiểu dữ liệu ban đầu của Fortmatter là String còn Converter type nào cũng được

Fortmatter phù hợp tầng web, còn converter có thể dùng ở bất cư tầng nào

Trường hợp chuyển đổi dữ liệu từ form input thì nên dùng Formatter

**23 ORM là gì ?**

Opject Ralational mapping là kỹ thuật liên kết, truy suất giữa các đối tượng lập trình với các đối tượng trong DB.

Cho phép lt viên tập trung thao tác với các đối tượng mà không cần quá quan tâm tới BD thực tế đang sử dụng

**24 JPA là gì? Spring Data JPA là gì ?**

- JPA java persistence API cung cấp các đặc tả để duy trì đọc quản lí dữ liệu từ đối tượng java đến cấc quan hệ trong CSDL. Cung cấp 1 mô hình POJO persistence ánh xạ các table và mối quan hệ trong database sang classs/mqh trong java.

- Spring Data Java giúp cho việc sử dụng jpa trong spring 1 cách dễ dàng hơn có thể sử dụng với các framework khác OpenJpa hibernate,..

Cải tiến JPA tiêu chuẩn đơn giản hóa tầng truy xuất tự tạo các repository , tạo các truy xuất thông qua tên phương thức, ghi log, phân trang,…

**25 Hibernate là gì ? Hibernate làm gì với database**

Hibernate là một ORM frameword triển khai các interface JPA để thực hiện các lệnh truy vấn với DB

**26 Liệt kê một số annotation của hibernate?**

Entity Controller Service Repository

**27 Entity là gì?**

- Là đối tượng đại diện cho dữ liệu trong ứng dụng

- thường được gọi là POJO plain old java opject

- Sẽ được ánh xạ tới 1 bảng trong table

**28 Persistence Context & Entity Manager là gì?**

- Persistence Context: Là tập hợp các thể hiện của entity được quản lí, tồn tại, trong 1 kho dư liệu

- Entity Manager là

+ Khai báo các phương thức để thương tác với persistence Context

+ Tạo hoặc xóa các thể hiện của entity

+TÌm kiếm entity theo khóa chính

+ thực thi các câu lệnh truy vấn entity

**29 Câu lệnh truy vấn động là gì? Câu lệnh truy vấn tĩnh là gì?**

- phuong thức CreateQuery giúp tạo truy vấn dồng

- CreateQuertName tạo truy vấn tính

Các truyền tham số

C1: dùng :param và method setParameter(“param”, value)

C2: dùng ?index và method setParameter(index,value)

**30 Trạng thái của Entity bao gồm những gì?**

**31 Có bao nhiêu loại mapping trong Hibernate?**

|  |  |
| --- | --- |
| [Many-to-One](https://viettuts.vn/hibernate/hibernate-many-to-one-relationship-voi-mysql-tren-eclipse) | Mapping mối quan hệ many-to-one trong Hibernate |
| [One-to-One](https://viettuts.vn/hibernate/hibernate-one-to-one-relationship-voi-mysql-tren-eclipse) | Mapping mối quan hệ one-to-one trong Hibernate |
| [One-to-Many](https://viettuts.vn/hibernate/hibernate-one-to-many-relationship-voi-mysql-tren-eclipse) | Mapping one-to-many relationship using Hibernate |
| [Many-to-Many](https://viettuts.vn/hibernate/hibernate-many-to-many-relationship-voi-mysql-tren-eclipse) | Mapping mối quan hệ many-to-many trong Hibernate |

**32 SessionFactory là gì?**

Là một interface giúp tạo ra session kết nối đến database bằng cách đọc các cấu hình trong Hibernate configuration. Mỗi một database phải có một session factory.

**33 Validation dữ liệu là gì?**

- Là thực hiện đánh giá xác minh tính hợp lệ của dữ liệu đầu vào

- Các tầng valid

+ UI( client, FE)

+ Business Layer ( BE, Service) : Spring hỗ trợ

+ Datalayer( BD)

**34 Trình bày cách triển khai validate dữ liệu trong Spring**

- Thêm thư viện validate vào file grandle

- Khai báo anoutation validate trong entity

- Khai báo tham sosoo valid/validation và bingdingResult trong Controller

- Implement hiển thị lỗi trong file html tương ứng

**35 Binding Result là gì?**

BindingResult là nơi Spring chứa kết quả của việc xác thực dữ liệu. Nếu như có lỗi trong quá trình xác thực đầu vào thì chúng ta có thể kiểm tra bằng BindingResult.

**36 AOP là gì? AOP gồm những yếu tố nào?**

Aspect orient programing lâp trình khía cạnh nó là 1 kỹ thuật lập trình nhằm phân tách chương trình thành các module riêng lẻ phân biệt và không phụ thuộc vào nhau. Khi thi thực hiện chương trình thì các module sẽ tập hợp lại với nhau nhưng khi sửa đổi chức năng thì chỉ cần sửa đổi 1 module

**37** **Các loại Advice**

Có 3 loại advice trong Spring AOP framework đó là: before + after-finally, after-returning, after-throwing + around advice

**38 JoinPoint là**

Joinpoint: là một điểm trong chương trình, là những nơi có thể được chèn những xử lí phụ:

PointCut:Là 1 cách xác định joinpoint, Nó là các biểu thức được sử dụng để kiểm tra nó có khớp với các Jointpoint để xác định xem Advice có cần được thực hiện hay không.

**39 Nêu cách triển khai AOP mà bạn dùng trong dự án**

**40 Xử lý ngoại lệ trong Spring Web MVC như thế nào?**

**41 Cookie là gì?**

Là tệp trang web người dùng truy cập tạo ra, giúp người dùng trải nghiệm tốt hơn bắng cách lưu thông tin duyệt web, có thể duy trì trạng thái đang nhập. Ghi nhớ các tùy chọn, cung cấp các phù hợp với người dùng

**42 Phân biệt Session và Cookie**

Giống: đều tạo ra dữ liệu để sử dụng trong quá trình tương tác

|  |  |
| --- | --- |
| Khác: Cookie | Session |
| * Lưu ở máy client * Có thể điều chỉnh thời gian sống cho từng cookie * Type: String | * Lưu ở server * Thời gian sống session đều giống nhau phụ thuộc vào cookie * Nhiều loại kể cả object |

43 Nêu cách triển khai Session mà bạn dùng trong dự án

44 Web Service là gì? Lấy ví dụ về web service

Là thành phần ứng dụng được hiển thị dưới các dạng dịch vụ trên www

Có thể sử dụng tích hợp với các ứng dụng được viết bằng các ngôn ngữ khác nhau chạy trên các nền tảng khác nhau

Webservice hoạt động như 1 server trong mô hình client server sử dụng giao thức http/https mà chỉ thực hiện 1 tác vụ cụ thể

45 Phân biệt Web Service và Web truyền thông

46 SOAP là gì? RESTful là gì?

- SOAP: dùng xml để định nghĩa dữ liệu từ văn bản và truyền thông qua http

- Rest:  (**RE**presentational **S**tate **T**ransfer) là một dạng chuyển đổi cấu trúc dữ liệu, định nghĩa dữ liệu dưới dạng xml và json truyền thông qua http

-RESTful là các web service xây dựng dựa trên REST xử lí các hoạt động CRUD

**47 jQuery là gì? Ajax là gì?**

Ajax là 1 nhóm các công nghệ phát triển web được sử dụng để tạo ra các trang web giàu tính tương tác nhanh hơn mượt hơn với xxml, css,html,js

Cho phép trang web cập nhật bất đồng bộ bằng cách trao đổi các đối trượng dữ liệu nhỏ nhới server mà không cần tải lại toàn bộ trang

**48 Nêu ví dụ về cách bạn triển khai Ajax trong dự án của bạn**

**Sử dụng REST**

**49 @RequestHeader và @ResponseHeader có ý nghĩa gì**

**50 I18n và L10n là gì?**

**51 Nêu cách triển khai I18N và I10N trong dự án Spring**

**52 Interceptor là gì?**

**53 Bảo mật là gì? Cơ chế bảo mật trong Spring như thế nào?**

**54 Authentication là gì ? Authorization là gì ?**

Authentiction là một hành động nhằm thiết lập hoặc chứng thực một thông điệp hoặc một đối tượng nào đó đang tin cậy

Cũng có nghĩa công nhận nguồn gốc của đối tượng để xác thực thì thông thường người ta sẽ dùng nhiều nhân tố khác nhau để chứng minh

**55 Liệt kê các cơ chế xác thực**

**56 CSRF là gì**

**CSRF** là một kiểu tấn công gây sự nhầm lẫn tăng tính xác thực và cấp quyền của nạn nhân khi gửi một request giả mạo đến máy chủ. Vì thế một lỗ hổng CSRF ảnh hưởng đến các quyền của người dùng ví dụ như quản trị viên, kết quả là chúng truy cập được đầy đủ quyền.

**57 CORS là gì**

**CORS** là một cơ chế cho phép nhiều tài nguyên khác nhau (fonts, Javascript, v.v…) của một trang web có thể được truy vấn từ domain khác với domain của trang đó. **CORS** là viết tắt của từ Cross-origin resource sharing.

58 Spring Boot là gì?

- Là 1 module của spring frame word cung cấp tính năng rapid applycation development có thể chạy 1 ứng dụng độc lập mà cần rất ít confix

59 Sự khác nhau giữa SpringBoot và SpringMVC là gì?

| **Spring Boot** | **Spring MVC** |
| --- | --- |
| **Spring Boot** là một module của Spring để đóng gói ứng dụng dựa trên Spring với các giá trị mặc định hợp lý. | **Spring MVC** model view controller-based web framework dựa trên Spring. |
| Nó cung cấp các cấu hình mặc định để xây dựng **Spring-powered framework**. | Nó cung cấp các tính năng sẵn sàng sử dụng để xây dựng một ứng dụng web. |
| Không có yêu cầu đối với các file mô tả khi triển khai. | Cần có nhiều file mô tả triển khai. |
| Nó tránh mã viết sẵn và kết hợp các dependencies lại với nhau trong một đơn vị duy nhất. | Nó chỉ định từng dependency riêng biệt. |
| Nó làm giảm thời gian phát triển và tăng năng suất. | Cần nhiều thời gian hơn để đạt được điều tương tự. |

60 Cấu hình cho Spring Boot Tìm kiếm các Bean ở nhiều package khác nhau bằng cách nào?

@ComponentScan({"me.loda.spring.componentscan.others2","me.loda.spring.componentscan.others"})

61 Trình bày Spring Security? @EnableWebSecurity làm gì?

Spring Security là một dự án nổi bật trong hệ sinh thái Spring. Spring Security cung cấp các dịch vụ bảo mật toàn diện cho các ứng dụng doanh nghiệp có nền tảng Java EE.

* **Authentication** (xác thực): là tiến trình thiết lập một principal. Principal có thể hiểu là một người, hoặc một thiết bị, hoặc một hệ thống nào đó *có thể* thực hiện một hành động trong ứng dụng của bạn.
* **Authorization** (phân quyền) hay **Access-control**: là tiến trình quyết định xem một principal *có được phép* thực hiện một hành động trong ứng dụng của bạn hay không. Trước khi diễn tiến tới Authorization, principal cần phải được thiết lập bởi Authentication.

62 Trình bày Spring Boot JPA? Cài đặt?

63 Để sử dụng interface JpaRepository cần cung cấp những thông tin nào?

64 Bạn có biết annotation @ConfigurationProperties? Trình bày?

@ConfigurationProperties annotation cho phép truy xuất các thuộc tính được đặt trong các tệp cấu hình một cách tự động và mạnh mẽ