1 Trình bày cách Spring MVC xử lý request

?

2 Trình bày cơ chế Dependency Injection trong Spring

DI là cơ chế tiêm phụ thuộc để giảm sự phụ thuộc

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

1. **Constructor injection**: các dependency đc cung cấp thông qua constructor của class.
2. **Setter injection**: client tạo ra một setter method để các class khác có thể sử dụng chúng để cấp dependency.
3. **Interface injection**: dependency sẽ cung cấp một hàm injector để inject nó vào bất kì client nào đc truyền vào. Các client phải implement một interface mà có một setter method dành cho việc nhận dependency.

4 Framework là gì ? Framework khác Library chỗ nào ?

**Framework**là bộ khung cung cấp các tính năng chung và thông dụng để hỗ trợ phát triển ứng dụng một cách nhanh chóng.

5 Spring Framework là gì ?

**Spring Framework** là một framework ứng dụng vào bộ chứa đảo ngược điều khiển cho nền tảng java hoạt động dựa trên 2 nguyên tắc design chính là DI và AOP

6 Lợi ích của Spring Framework?

* Giúp bạn tạo ứng dụng độc lập, có thể bắt đầu bằng java.jar.
* Nó cung cấp cấu hình POM được xác định chính xác cho Maven.
* Cho phép bạn nhúng trực tiếp Undertow, Tomcat hoặc Jetty.
* Giúp bạn tự động cấu hình bất cứ khi nào có thể.

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

8 Bean là gì?

bean là những module chính của chương trình, được tạo ra và quản lý bởi Spring IoC container.

9 Trong Spring có bao nhiêu Bean Scope?

* **Singleton:**Chỉ duy nhất một thể hiện của bean sẽ được tạo cho mỗi container. Đây là scope mặc định cho spring bean. Khi sử dụng scope này cần chắc chắn rằng các bean không có các biến/thuộc tính được share.
* Prototype: Một thể hiện của bean sẽ được tạo cho mỗi lần được yêu cầu(request)
* Request: giống với prototype scope, tuy nhiên nó dùng cho ứng dụng web, một thể hiện của bean sẽ được tạo cho mỗi HTTP request.
* Session: Mỗi thể hiện của bean sẽ được tạo cho mỗi HTTP Session
* Global-Session: Được sử dụng để tạo global sesion bean cho các ứng dụng Portlet.

10 @Autowire là gì?

Autowired **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?

@Component: **là một annotation của class.** **Nó dùng để đánh dấu class Java là một bean**.

12 Trình bày ý nghĩa của Controller

13 Trình bày ý nghĩa của ModelAndView Interface

14 Trình bày ý nghĩa của ModelMap

15 Trình bày ý nghĩa của ViewResolver Interface

Các ViewResolver (trình phân giải view) **là các đối tượng chịu trách nhiệm lấy các đối tượng View cho một hành động và vị trí cụ thể**

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

Theo tiêu chuẩn RFC 2616, **POST method được dùng để gửi các request kèm theo một entity đến server yêu cầu tạo một tài nguyên mới dựa trên entity được cung cấp**. Mặt khác, PUT method nên kèm theo một định danh (thường là ID) cùng với một entity.

17 @RequestMapping làm gì?

**ánh xạ các HTTP request tới các phương thức xử lý của MVC và REST controller**

18 Thuộc tính consumes trong các Request Mapping là gì ?

19 Thuộc tính produces trong các Request Mapping là gì ?

20 Trình bày cơ chế Data Binding trong Spring

Data Binding **là cơ chế liên kết dữ liệu đầu vào hoặc đầu ra với các đối tượng model**

21 So sánh RequestParam và PathVariable ?

22 Thymeleaf là gì?

Thymeleaf là một Java template engine dùng để xử lý và tạo HTML, XML, Javascript, CSS và text.

23 Sử dụng lặp trong Thymeleaf như thế nào? Sử dụng điều kiện trong Thymeleaf như thế nào?

24 Formatter là gì ? Converter là gì ?

Formatter cũng hoạt động giống như converter, tức là **chuyển đổi một kiểu dữ liệu sang kiểu dữ liệu khác**. Tuy nhiên, kiểu dữ liệu nguồn của Formatter là String, trong khi đó converter có thể làm việc với bất cứ kiểu dữ liệu nguồn nào

25 ORM là gì ?

Là kỹ thuật liên kế các đối tượng trong lập trình với các table trong database

26 JPA là gì? Spring Data JPA là gì ?

**JPA** là viết tắt của **Java Persistence API**, nó là một đặc tả Java cho việc ánh xạ giữa các đối tượng Java với cơ sở dữ liệu quan hệ sử dụng công nghệ phổ biến là **ORM**

**Spring Data** là một module của **Spring Framework**. Mục đích của **Spring Data** **JPA**làgiảm thiểu việc thực hiện quá nhiều bước để có thể implement được JPA.

27 Hibernate là gì ? Hibernate làm gì với database

**Hibernate là** một thư viện ORM (Object Relational Mapping) mã nguồn mở giúp lập trình viên viết ứng dụng Java có thể map các objects (pojo) **với** hệ quản trị cơ sở dữ liệu quan hệ, và hỗ trợ thực hiện các khái niệm lập trình hướng đối tượng **với** cớ dữ liệu quan hệ.

28 Liệt kê một số annotation của hibernate?

@Entity

## @Table

## @Column

## @Id

29 Entity là gì?

30 Persistence Context & Entity Manager là gì?

Persistence context có thể được coi như **một container hoặc là cache cấp đầu tiên cho tất cả các đối tượng mà chúng ta đã load hoặc lưu vào cơ sở dữ liệu trong một session, logic nghiệp vụ sẽ quyết định ranh giới các session**.

EntityManager. **Là một interface cung cáp các API cho việc tương tác giữa ứng dụng với database**.

31 Câu lệnh truy vấn động là gì? Câu lệnh truy vấn tĩnh là gì?

32 Trạng thái của Entity bao gồm những gì?

### New (Mới tạo ra, chưa được lưu xuống CSDL):

### Persistent (được ghi xuống CSDL và quản lý trong Persistence Context):

### Detached (tạm thời tách ra khỏi Persistence Context)

### Removed:

33 Có bao nhiêu loại mapping trong Hibernate?

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

### Class name and data type ID

35 Validation dữ liệu là gì?

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

36 Trình bày cách triển khai validate dữ liệu trong Spring

37 Binding Result là gì?

BindingResult **là nơi Spring chứa kết quả của việc xác thực dữ liệu**

38 AOP là gì? AOP gồm những yếu tố nào?

Là kỹ thuật lập trình phân tách chương trình thành nhiều module riêng biệt không phụ thuộc lẫn nhau

Trong **AOP**, chương trình của chúng ta được chia thành 2 loại concern: Core concern/ Primary concern: **là** requirement, logic xử lý chính của chương trình.

39 Các loại Advice?

**before, after, after-returning, after-throwing, and around advice**.

40 JoinPoint là gì?

Là những điểm trong chương trình, nới có thể chèn cross cutting, như ghi log sau khi chạy method bất kỳ thì vị trí sau method đó gọi là join point

41 Nêu cách triển khai AOP mà bạn dùng trong dự án

42 Mục đích xử lý ngoại lệ trong Spring Web MVC ?

43 Các cách xử lý ngoại lệ trong Spring Web MVC ?

2.1. Sử dụng try-catch. Phương pháp này thường để xử lý các ngoại lệ tại request (Controller).

2.2. Sử dụng @ExceptionHandler để ủy quyền cho front controller xử lý ngoại lệ.

2.3. Sử dụng HandlerExceptionResolver. Phương pháp này được dùng để xử lý ngoại lệ tại servlet.

44 Cookie là gì?

Cookies là **các tệp** được **trang web người dùng truy cập**tạo ra. Cookie giúp trải nghiệm trực tuyến của bạn dễ dàng hơn bằng cách **lưu thông tin duyệt web**. Với Cookies, các trang web có thể **duy trì trạng thái đăng nhập của bạn**, **ghi nhớ tùy chọn** trang web và cung cấp nội dung phù hợp với vị trí của người dùng.

45 Phân biệt Session và Cookie

Giống nhau : tạo ra dữ liệu trong quá trình sử dụng ứng dụng

Khác nhau:

Cookie lưu trữ dữ liệu trên máy client, session lưu trên sever.

Cookie có thể set thời gian sống, session tồn tại trong một phiên làm việc

Cookie chỉ lưu kiểu string session lưu được nhiều kiểu dữ liệu.

Session hoạt động dựa trên cookie nhờ vào session Id

46 Session là gì ? Cho ví dụ về session ?

Là một phiên làm việc giữa client và sever

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

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

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

Facebook

49 Phân biệt Web Service và Web truyền thông

|  |  |
| --- | --- |
| Website | WebService |
| Có giao diện | Không có giao diện |
| dùng bởi con người | Dung bởi ứng dụng khác |
| Hoat động trên đa nền tảng vì được tinh chỉnh để phù hợp với nhiều trình duyệt và hệ điều hành khác nhau | Độc lập về nền tảng, all phải sử dụng giao thức chuẩn. |
| Là ứng dụng đầu cuối, truy cập qua URL qua đó nhận được những dữu liệu text, hình ảnh, âm thanh có thể dễ dàng hiểu được | Dữ liệu trả về webservice người dung khó hiểu được, dữ liệu được ứng dụng khác sử dụng và chế biến thành dữ liệu người dung có thể hiểu được. |
| Truy cập thông qua thành phần của giao diện như button, form | Truy cập thông qua các phương thức http request  Get post put delete |

50 SOAP là gì? RESTful là gì?

[**SOAP**](https://bizflycloud.vn/tin-tuc/soap-la-gi-20181117112550075.htm)**(Simple Object Access Protocol - Giao thức truy cập đối tượng đơn giản)** là giao thức nhắn tin cho phép các chương trình chạy trên các hệ điều hành khác nhau (như Windows và Linux) giao tiếp được với nhau thông qua Giao thức HTTP và Ngôn ngữ XML.

51 jQuery là gì? Ajax là gì?

**JAX là chữ viết tắt của Asynchronous JavaScript and XML, AJAX = Asynchronous JavaScript and XML.** Đây là một công nghệ giúp chung ta tạo ra những Web động mà hoàn toàn không reload lại trang

 jQuery là thư viện được viết từ JavaScript, jQuery giúp xây dựng các chức năng bằng Javascript dễ dàng

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

53 @RequestHeader và @ResponseHeader có ý nghĩa gì

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

58 Authentication là gì ? Authorization là gì ?

**Authentication (xác thực) có nghĩa là xác nhận danh tính của riêng bạn, trong khi authorization (ủy quyền) có nghĩa là cấp quyền truy cập vào hệ thống**. Nói một cách đơn giản, authentication là quá trình xác minh bạn là ai, trong khi authorization là quá trình xác minh những gì bạn có quyền truy cập.

59 Các cơ chế xác thực ?

OTP

Oauth2

JWT

HTTP basic

cookie

60 CSRF là gì

Là kỹ thuật tấn công = cách sử dụng quyền xác thực của người dung đối với website

61 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.

62 Spring Boot là gì?

Spring Boot **là một dự án phát triển bởi JAV (ngôn ngữ java) trong hệ sinh thái Spring framework**. Nó giúp cho các lập trình viên chúng ta đơn giản hóa quá trình lập trình một ứng dụng với Spring, chỉ tập trung vào việc phát triển business cho ứng dụng.

63 Sự khác nhau giữa SpringBoot và SpringMVC là gì?

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

65 Trình bày Spring Security? @EnableWebSecurity làm gì?

66 Trình bày Spring Boot JPA? Cài đặt?