|  |  |
| --- | --- |
| 1 | Vẽ và trình bày cách Spring MVC xử lý request     * Khi có một request đến thì DispatcherServlet (Front controller) sẽ đón nhận. * DispatcherServlet sẽ sử dụng HandlerMapping để xác định gửi requesr đến controller tương ứng. Sau đó controller sẽ làm việc với model để kiểm tra và lấy dữ liệu. * Sauk hi có kết quả thì controller sẽ gửi trả dữ liệu cho Front controller và dữ liệu sẽ được render đến View Template tương ứng. Và View Template sẽ được gửi trả về Front Controller, rồi gửi về Client. |
| 2 | Trình bày cơ chế Dependency Injection trong Spring  Dependencyinjectionlàmột cơchếđểtriểnkhai IoC.  Là cơ chế giảm sự phụ thuộc giữa 2 Module (ex: SumController, SumService), 2 module này sẽ không phụ thuộc trực tiếp vào nhau mà thông qua abstract (ISumService) |
| 3 | Có bao nhiêu cách để thực hiện Dependency Injection? Trình bày? |
| 4 | Framework là gì ? Framework khác Library chỗ nào ?  - Frame work là các ứng dụng phần mềm có tính trừu tượng (bộ khung) mà có các quy tắc có sẵn lập trình viên sẽ dựa vào để phát triển web của mình.  - Diểm khác biệt lớn nhất giữa Framework và Library đó chính là cơ chế “Inversion of Control”  • Với Library: Ứng dụng nắm quyền điều khiển (control)  • VớiFramework: Framework nắm quyền điều khiển |
| 5 | Spring Framework là gì ? |
| 6 | Lợi ích của Spring Framework? |
| 7 | Nguyên lý đảo ngược quyền điều khiển (Inversion of Control) là gì ?  IoClàmộtnguyênlýtrongpháttriểnphầnmềm,trongđóviệcđiều khiểncác  đốitượnghoặccácthànhphầncủahệthốngđượcthực hiệnbởiframework  hoặc các container  • IoCchophépframeworknắmgiữquyềnđiềukhiểnluồngthựcthi củahệ  thốngvàgọicácmãnguồnkhác  • Lợi ích:  • Táchrờiviệcthựcthi(execution)vàviệctriểnkhai(implementation)  • Dễ chuyển đổi giữa các implementation  • Dễ phân tách module hơn  • Dễkiểmthửhơn, bằngcáchtáchrời các thành phầnriênglẻ |
| 8 | Bean là gì? |
| 9 | Trong Spring có bao nhiêu Bean Scope? |
| 10 | @Autowire là gì? |
| 11 | @Component có ý nghĩa gì? |
| 12 | Trình bày ý nghĩa của Controller  Spring controller xử lý các request để thực hiện các logic nghiệp vụ  Spring controller nhận ủy quyền xử lý các request từ phía front controller. Công việc cơ bản của nó là nhận dữ liệu cần thiết từ front controller, xác định view cần sử dụng, triệu gọi lớp xử lý nghiệp vụ cần thiết, sắp xếp dữ liệu mà view cần đến vào model, gửi thông tin về view cùng model lại cho front controller. |
| 13 | Trình bày ý nghĩa của ModelAndView Interface  • ModelAndView đại diện cho mộ tview cùng với các dữ liệu sử dụng trong view đó |
| 14 | Trình bày ý nghĩa của ModelMap Interface |
| 15 | **Trình bày ý nghĩa của ViewResolver Interface**  ViewResolver là cơ chế để xử lý tầng view của Spring MVC  ViewResolver ánh xạ tên của view sang đối tượng view tương ứng. Được xác định dựa trên prefix, suffix  Có nhiều implementation khác nhau của ViewResolver:  • XmlViewResolver: Giúp xem file viết bằng XML  • UrlBasedViewResolver: Giúp ánh xạ trực tiếp URL tới tên logic của view. |
| 16 | Phân biệt POST với PUT thường sử dụng để làm gì? |
| 17 | @RequestMapping làm gì?  Annotation @RequestMapping ánh xạ một URL đến một một controller hoặc phương thức của controller  • Annotation @RequestMapping được sử dụng để ánh xạ các request  tới các action tương ứng của controller |
| 18 | Trình bày cơ chế Data Binding trong Spring |
| 19 | Thuộc tính consumes trong các Request Mapping là gì ? |
| 20 | Thymeleaf là gì? |
| 21 | Sử dụng lặp trong Thymeleaf như thế nào? Sử dụng điều kiện trong Thymeleaf như thế nào? |
| 22 | Formatter là gì ? Converter là gì ? |
| 23 | ORM là gì ? |
| 24 | JPA là gì? Spring Data JPA là gì ? |
| 25 | Hibernate là gì ? Hibernate làm gì với database |
| 26 | Liệt kê một số annotation của hibernate? |
| 27 | Entity là gì? |
| 28 | Persistence Context & Entity Manager là gì? |
| 29 | Câu lệnh truy vấn động là gì? Câu lệnh truy vấn tĩnh là gì? |
| 30 | Trạng thái của Entity bao gồm những gì? |
| 31 | Có bao nhiêu loại mapping trong Hibernate? |
| 32 | SessionFactory là gì? |
| 33 | Validation dữ liệu là gì? |
| 34 | Trình bày cách triển khai validate dữ liệu trong Spring |
| 35 | Binding Result là gì? |
| 36 | AOP là gì? AOP gồm những yếu tố nào? |
| 37 | Các loại Advice? |
| 38 | JoinPoint là gì? |
| 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? |
| 42 | Phân biệt Session và Cookie |
| 43 | Nêu cách triển khai Session mà bạn dùng trong dự án |
| 44 | Web Service là gì? Lấy ví dụ về web service |
| 45 | Phân biệt Web Service và Web truyền thông |
| 46 | SOAP là gì? RESTful là gì? |
| 47 | jQuery là gì? Ajax là gì? |
| 48 | Nêu ví dụ về cách bạn triển khai Ajax trong dự án của bạn |
| 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 ? |
| 55 | Liệt kê các cơ chế xác thực |
| 56 | CSRF là gì |
| 57 | CORS là gì |
| 58 | Spring Boot là gì? |
| 59 | Sự khác nhau giữa SpringBoot và SpringMVC là gì? |
| 60 | Cấu hình cho Spring Boot Tìm kiếm các Bean ở nhiều package khác nhau bằng cách nào? |
| 61 | Trình bày Spring Security? @EnableWebSecurity làm gì? |
| 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? |