|  |  |
| --- | --- |
| 1.Data binding là gì? Vì sao phải sử dụng Data binding? |  **Định nghĩa:** Data Binding là quá trình **tự động ánh xạ (mapping)** dữ liệu từ request (form, query parameter) vào các đối tượng Java (model/bean) trong ứng dụng Spring MVC.   **Ví dụ:** Nếu form HTML gửi lên name=Nguyen&age=20, Spring MVC sẽ tự động gán vào đối tượng Student:  public class Student {  private String name;  private int age;  // getter, setter  }  Khi ta khai báo trong Controller:  @PostMapping("/save")  public String save(@ModelAttribute Student student) {  // student.getName() = "Nguyen"  // student.getAge() = 20  }   **Tại sao cần dùng?**   1. Giảm code lặp khi lấy dữ liệu từ request (request.getParameter(...)). 2. Tự động chuyển đổi kiểu dữ liệu (String → int, Date, boolean, …). 3. Giúp code dễ đọc, ngắn gọn, tách biệt Controller và View. |
| 2. Cơ chế hoạt động của Data binding trong Spring | Cơ chế hoạt động:   1. Người dùng nhập dữ liệu vào form HTML và submit. 2. Spring DispatcherServlet nhận request, tìm đúng Controller. 3. Spring sử dụng **DataBinder** để:    * Đọc dữ liệu từ request (parameter, path, query string).    * Ánh xạ (bind) giá trị này vào thuộc tính của **command object / model object** (POJO).    * Nếu có converter hoặc property editor → nó thực hiện ép kiểu. 4. Sau khi binding xong → Controller nhận được đối tượng đã chứa dữ liệu. |
| 3. Thuộc tính modelAttribute trong thẻ <form:form> dùng để làm gì? Cách sử dụng? |  **Ý nghĩa:** modelAttribute cho Spring biết đối tượng nào trong Model sẽ được dùng để binding dữ liệu từ form. Khi submit form → dữ liệu trong form sẽ được map ngược lại vào đối tượng này.   **Cách dùng:**  <%@ taglib uri="http://www.springframework.org/tags/form" prefix="form"%>  <form:form action="save" method="post" modelAttribute="student">  Name: <form:input path="name"/> <br/>  Age: <form:input path="age"/> <br/>  <input type="submit" value="Save"/>  </form:form>   **Controller:**  @GetMapping("/form")  public String showForm(Model model) {  model.addAttribute("student", new Student());  return "studentForm";  }  @PostMapping("/save")  public String save(@ModelAttribute("student") Student student) {  // student chứa dữ liệu từ form  return "result";  } |
| 4. Các bước thực hiện data binding trong Spring | 1. **Tạo model class (POJO):** Ví dụ: Student.java với các thuộc tính name, age, .... 2. **Khai báo modelAttribute trong Controller:**    * @ModelAttribute trong tham số method, hoặc    * model.addAttribute("student", new Student()); 3. **Tạo form sử dụng Spring Form Tag Library:**    * Dùng <form:form modelAttribute="student"> và các tag <form:input path="..."/>. 4. **Submit form → Spring DataBinder ánh xạ dữ liệu → Controller nhận object đã binding.** 5. **(Tùy chọn)** Nếu có lỗi binding (sai kiểu dữ liệu), có thể xử lý bằng BindingResult.   👉 Tóm lại:   * **Data binding** giúp tự động map dữ liệu từ request vào object. * **Cơ chế**: DispatcherServlet → DataBinder → Model object. * **modelAttribute**: chỉ định object nào dùng để binding trong form. * **Các bước**: Tạo POJO → Khai báo modelAttribute → Form binding → Controller nhận dữ liệu. |