Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Introduction to Spring Framework in 10 Steps** Date: **11-06-2018**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **Create project with start.spring.io** | | * Create maven project with java, spring boot version >=2.0 * Group: default package, artifact: project name * Dependencies: add library * @SpringBootApplication : scan package and subpackage it in |
| **@Component và**  **@Autowire** | | * @Component: khai báo một thành phần để Spring container quản lý * @Autowire: Nối các dependency giữa các bean |
| **Dynamic Autowire** | | * Spring tự động scan và quản lý các bean. Sau đó nối chúng với nhau. * @primary dùng để khai báo dependency chính. |
| **Constructor**  **Setter injection** | | * Two way to inject dependency into another component * No setter or constructor will be use setter in the background |
| **Popularity of spring framework** | | * Enable Testable code: Because use DI so easy to write unit test * No plumbing code: không cần viết những code phải viết khi thực thi code chính * Flexible Architecture: * Stay current: create many project to support modern trend |
| **SUMMARY** | Tạo project với start.spring.io  Autowiring và dynamic autowire  Constructor and Setter injection- the best way is don’t use that two  Popularity of spring framework | |

Name: **IDE and Maven Project** Date: **11-06-2018**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **- Intellij**  **Key Shortcut** | | * Hint: Ctrl+ Space , Alt + enter; * Search: double+shift , * Goto declaration: ctrl + click * Type hierarchy: ctrl + H |
| **Maven là gì ?** | | * maven là một chương trình quản lý dự án cho phép các developers có thể quản lý về version, các dependencies ( các thư viện sử dụng trong dự án ) , quản lý build, tự động download javadoc & source, .... |
| **-POM.xml** | | * Project object model * Dependency: input library, and 1 library can call many dependencies ( others) |
| **Maven lifecycle** | | * Validate: xác nhận tất cả các thông tin của dự án là hợp lệ và chính xác. * Compile: biên dịch source code. * Test: chạy các unit test với một framework phù hợp. * Package: đóng gói các code đã được biên dịch và tổ chức các code đó đúng định dạng ví dụ như file jar hoặc file war đã cấu hình bên trong file pom. * Integration-test: thực hiện kiểm thử trong môi trường integration-test. * Verify: thực hiện kiểm tra việc đóng gói đã chính xác hay chưa. * Install: cài đặt các gói trong local repository. * Deploy: cài đặt package ở đây có thể là các file jar hoặc war lên một remote repository |
| **Maven working** | | * Read configuration file, check local repository, if exist it call from local   If not, It download from central. |
| **Maven command line** | | * In command dos, using mvn `command`, IDE just use `command` |
| **SUMMARY** | Tạo project và quản lý với maven | |

Name: **Spring in depth** Date: **13-06-2018**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **Autowiring**  **@Qualifier** | | -Tự động nối các instance (dependency) vào instance khác.  - @Qualifier(value= “”) cấu hình để spring biết dependency nào cần @Autowire. |
| **Dependency injection** | | * Chỉ việc tạo và nối các module cấp thấp (dependency) vào module cấp cao. |
| **Bean Scope** | | * Dependency : prototype & bean contain dependency: singleton * Dependency singleton. To make dependency become prototype => use proxyMode = “” |
| **Bean lifecycle** | | -Vòng đời của 1 bean  - trước khi khởi tạo : @PostConstruct  - khi thực hiện công việc  - sau khi phá hủy: @PreDestroy |
| **Xml configuration** | | * With annotation <context:component-scan base-package="package contain componet "/> * Without annotation:   + Define bean: <bean id=" name bean for get bean" class="class path "/>  + Inject dependency <property name=" dependency name field" ref=" bean id of dependency"/> setter injection ,  <constructor-arg ref=""/> contructor injection |
| **IoC container**  **Application context**  **Bean factory** | | * IoC: move the control out of controller or component which need dependency to framework. * IoC container: the program or framework provides that function * Application context and BeanFactory is 2 implement of spring IoC container |
| **@Component**  **@Repository**  **@Service**  **@Controller** | |  |
| **Read properties file** | | * With Annotation : @PropertySource("classpath: properties name") * With Xml +annotation: <context:property-placeholder location="classpath: properties name "/>   + @Value(“${key of properties}”)   * With pure xml: use <property> to inject |
| **SUMMARY** |  | |

Name: **Junit**  Date: **2018-06-14**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **JUNIT** | | - Junit dùng để test các method của một class nào đó  - sử dụng junit để dễ dàng hơn cho việc test vì junit hổ trợ các method dùng để test. |
| **Junit assert** | | * Method assert dùng để so sánh kết quả kỳ vọng và kết quả thực tế của method cần test |
| **Junit Annotation** | | -@Test : đánh dấu method dùng để test  -@Before , @After: đánh dấu method dùng chạy trước, sau mỗi method test  -@BeforeClass , @AfterClass: đánh dấu method dùng chạy trước, sau Class test |
| **SUMMARY** |  | |

Name: **Mockito** Date: **15-06-2018**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **Mockito framework** | | * Mockito là một framework thường dùng để giả lập kiểm thử, tạo giả một đối tượng cài đặt interface, giả lập lời gọi hàm đến đối tượng giả này và chỉ định kết quả trả lại cho lời gọi hàm giả lập đó. |
| **Stub là gì** | | * Stub là tạo ra một khai triển của interface dùng để đưa vào các dependency của class cần test. |
| **Mock là gi** | | * Mock Là một giả lập của dependency. |
| **Mock annotation** | | * @Mock tạo ra một dependency giả; * @InjectMocks: đưa mock vào instance của class cần dependency * -@RunWith: JUnit will invoke the class it references to run the tests in that class instead of the runner built into JUnit. |
| **SUMMARY** |  | |

Name: **Unit test in spring** Date: **2018-06-16**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **spring mvc**  Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **Dispatcher servlet** | | * Front controller |
| **Model – View - Controller** | | -Model : data container  -Controller: @Controller, @RequestMapping handle request and return view page  -View: html, jsp … |
| **Request parameter** | | * @RequestParam(“param name”) to binding data. * Instead of HttpServletRequest. |
|  | |  |
| **SUMMARY** |  | |

Name: **Spring mvc form tags** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **Text field**  **Using model to binding data** | | * Model: addAttribute(“key”,object); * Form modelAttribute=”key” to map object , path=”property of object” * Request submit handle method @ModelAttribute(“key”) binding data to object * Repond View: ${key.property} to access data |
| **Drop-down list** | | * Set cứng vào html   <**form:select path**="country">  <**form:option value**="Brazil" **label**="Brazil"/>  <**form:option value**="France" **label**="France"/>  <**form:option value**="Germany" **label**="Germany"/>  <**form:option value**="India" **label**="India"/> </**form:select**>   * Dùng java collection:  + tạo 1 property hashmap trong object và khởi tạo các giá trị   + tạo getter method cho hashmap đó  + <**form:options items**="**${key.**hashmap**}**" /> |
| **Drop-down list using properties file** | | * Tạo file properties; import util tag vào file cấu hình * Khai báo để load resource <util:properties id="countryOptions" location="classpath:../countries.properties" /> * Khai báo hashmap bên trong controller. @Value("#{countryOptions}")   private Map<String, String> countryOptions;   * Add hashmap vào model: model.AddAttribute(“hashmap key”, hashmap) * Khai báo form * <form:select path="country">   <form:options items="${“hashmap key”}" />  </form:select> |
| **Radio button** | | * Set cứng html   Favorite Language: Java <**form:radiobutton path**="favoriteLanguage" **value**="Java"/> C# <**form:radiobutton path**="favoriteLanguage" **value**="C#"/> PHP <**form:radiobutton path**="favoriteLanguage" **value**="PHP"/> Ruby <**form:radiobutton path**="favoriteLanguage" **value**="Ruby"/>   * Sử dụng file properties * Tạo file properties; import util tag vào file cấu hình, tạo hash map giống với dropdown list. HTML sử dụng radiobuttons   <**form:radiobuttons path**="favoriteLanguage" **items**="**${**favoriteLanguageOptions**}**"/> |
| **Check box** | | * Set cứng html   Linux <**form:checkbox path**="operatingSystems" **value**="linux"/> Mac OS <**form:checkbox path**="operatingSystems" **value**="Mac OS"/> MS Windows <**form:checkbox path**="operatingSystems" **value**="MS Windows"/>   * Sử dụng file properties**:** Tạo file properties; import util tag vào file cấu hình, tạo hash map giống với dropdown list. HTML sử dụng checkboxes   <**form:checkboxes path**="operatingSystems" **items**="**${**operatingSystemOptions**}**"/>   * Property trong object sử dụng collection để lưu trữ dữ liệu. |
| **SUMMARY** |  | |

Name: **Form Validate** Date: **2018-06-19**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **The need for validation** | |  |
| **Bean Validation** | |  |
| **Validation Annotation** | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Request Mapping** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **@RequestMapping** | | * Map request from view: also work fine with get and post if not limit or constrain * To limit use: method = RequestMethod.GET * method = RequestMethod.POST |
| **@GetMapping** | | * Only handle get method |
| **@PostMapping** | | * Only handle post mapping |
| **Which one** | |  |
| **Component (bean)** | |  |
| **SUMMARY** |  | |

Name: **Spring AOP**  Date: **2018-07-04**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
| **AOP terminology** | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |

Name: **Topic Title** Date: **Topic date**

|  |  |  |
| --- | --- | --- |
| **KEYWORD** | | **DETAIL** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **SUMMARY** |  | |