**Mục tiêu**

Luyện tập phát triển phần mềm web backend, sử dụng Spring MVC.

* Mục tiêu quan trọng của bài thực hành này giúp Học Viên có thể convert một dự án JAVA Servlet sang Spring MVC

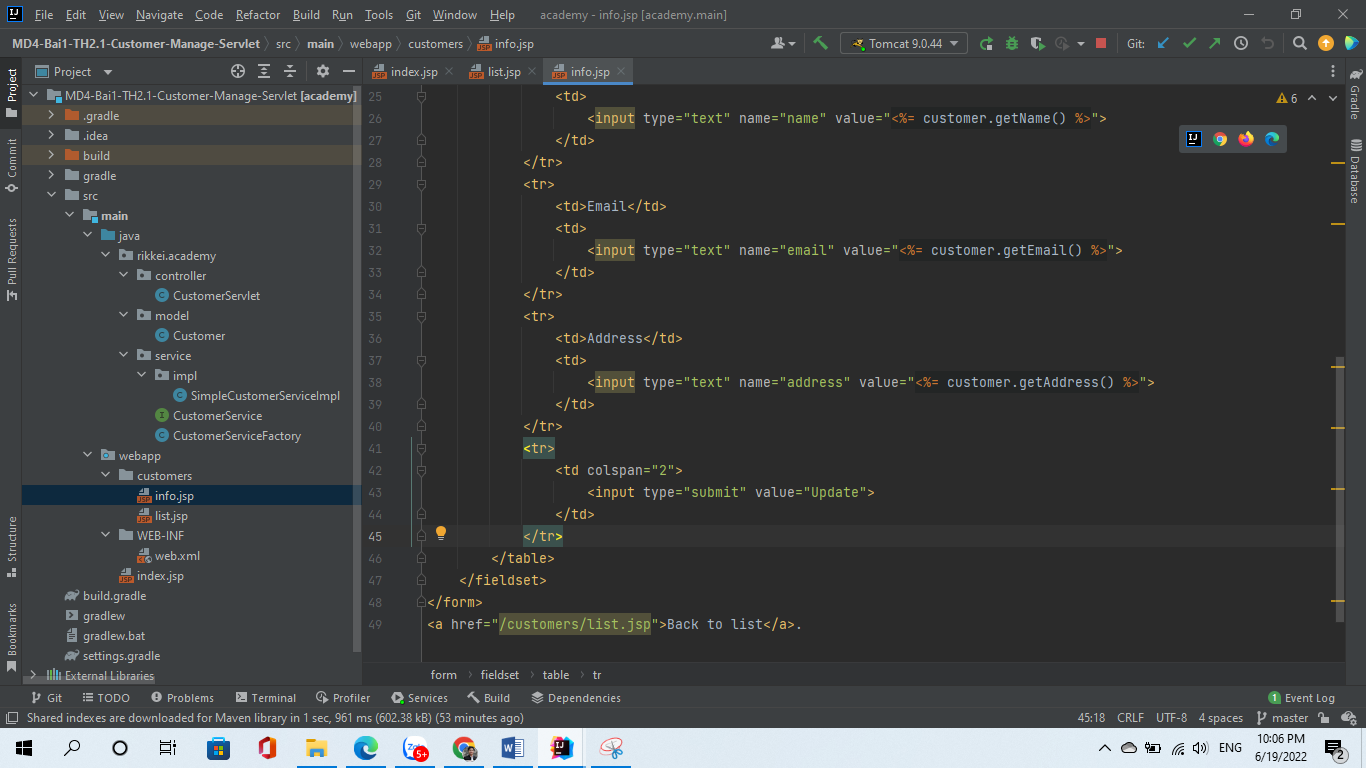
**Hướng dẫn**

* **tiếp nhận một dự án web backend sử dụng Java Servlet**

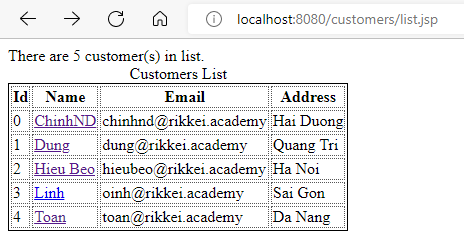
Download dự án JAVA Servlet MD3 :

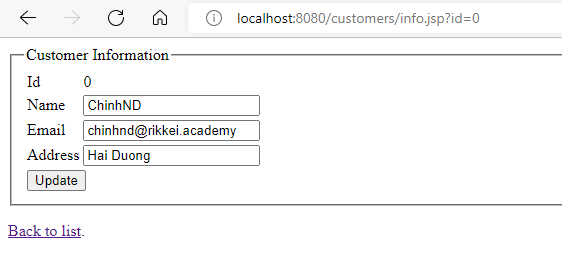
<https://github.com/rikkei-academy/MD4-Bai1-TH2.1-Customer-Manage-Servlet.git>

Mở dự án bằng Intellij :



* Add Tomcat và chạy dự án:





* Bây giờ chúng ta sẽ thực hiện lần lượt các bước để Convert 1 dự án Java Servlet sang Spring MVC
* Trước tiên thêm thư viện vào build gradle:

dependencies **{** compileOnly('javax.servlet:javax.servlet-api:4.0.1')  
  
 testImplementation("org.junit.jupiter:junit-jupiter-api:$**{**junitVersion**}**")  
 testRuntimeOnly("org.junit.jupiter:junit-jupiter-engine:$**{**junitVersion**}**")  
 implementation group: 'org.springframework', name: 'spring-webmvc', version: '5.3.2'  
 implementation group: 'javax.servlet', name: 'jstl', version: '1.2'  
**}**

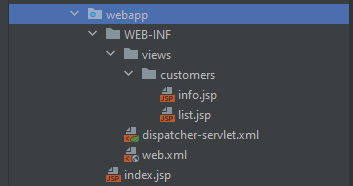
* Build thư viện => Tạo file: **dispatcher-servlet.xml** ở trong thư mục: **WEB-INF** và thêm config vào:

<?xml version="1.0" encoding="UTF-8"?>  
<beans xmlns="http://www.springframework.org/schema/beans"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xmlns:p="http://www.springframework.org/schema/p"  
 xmlns:mvc="http://www.springframework.org/schema/mvc"  
 xsi:schemaLocation="  
 http://www.springframework.org/schema/beans  
 http://www.springframework.org/schema/beans/spring-beans.xsd  
 http://www.springframework.org/schema/context  
 http://www.springframework.org/schema/context/spring-context.xsd  
 http://www.springframework.org/schema/mvc  
 http://www.springframework.org/schema/mvc/spring-mvc.xsd">  
  
 <!-- Spring MVC Annotation -->  
 <context:annotation-config />  
 <mvc:annotation-driven/>  
  
 <!--Controller -->  
 <context:component-scan base-package="rikkei.academy.controller"/>  
 <!--View -->  
 <bean id="viewResolver"  
 p:prefix="/WEB-INF/views/" p:suffix=".jsp" class="org.springframework.web.servlet.view.InternalResourceViewResolver"/>  
</beans>

* Quan sát đoạn config trong **dispatcher-servlet.xml**

<bean id="viewResolver"  
 p:prefix="/WEB-INF/views/" p:suffix=".jsp" class="org.springframework.web.servlet.view.InternalResourceViewResolver"/>

Phần này ta đang config đường dẫn các file .jsp qua thư muc views nên ta cần tạo 1 thư mục views => và kéo các file jsp vào trong views => Lúc này dự án có cấu trúc như sau:



* Sửa lại config trong file: **web.xml:**

<?xml version="1.0" encoding="UTF-8"?>  
  
<web-app id="WebApp\_ID" version="2.4"  
 xmlns="http://java.sun.com/xml/ns/j2ee"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee  
 http://java.sun.com/xml/ns/j2ee/web-app\_2\_4.xsd">  
 <display-name>example</display-name>  
 <servlet>  
 <servlet-name>spring</servlet-name>  
 <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>  
 <init-param>  
 <param-name>contextConfigLocation</param-name>  
 <param-value>/WEB-INF/dispatcher-servlet.xml</param-value>  
 </init-param>  
 <load-on-startup>1</load-on-startup>  
 </servlet>  
 <servlet-mapping>  
 <servlet-name>spring</servlet-name>  
 <url-pattern>/</url-pattern>  
 </servlet-mapping>  
</web-app>

* Sửa lại file: **list.jsp:**

<%@ **taglib** prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>  
<style>  
 table {  
 border: 1px solid #000;  
 }  
  
 th, td {  
 border: 1px dotted #555;  
 }  
</style>  
There are ${requestScope.customers.size()} customer(s) in list.  
<table>  
 <caption>Customers List</caption>  
 <thead>  
 <tr>  
 <th>Id</th>  
 <th>Name</th>  
 <th>Email</th>  
 <th>Address</th>  
 </tr>  
 </thead>  
 <tbody>  
 <**c:forEach** var="c" items="${requestScope.customers}">  
 <tr>  
 <td>  
 <**c:out** value="${c.id}"/>  
 </td>  
 <td><a href="detail/${c.getId()}">Detail</a></td>  
 <td>  
 <**c:out** value="${c.email}"/>  
 </td>  
 <td>  
 <**c:out** value="${c.address}"/>  
 </td>  
 </tr>  
 </**c:forEach**>  
 </tbody>  
</table>

* Sửa lại code Class: **CustomerServlet**

package rikkei.academy.controller;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.stereotype.Controller;  
import org.springframework.ui.Model;  
import org.springframework.web.bind.annotation.GetMapping;  
import rikkei.academy.model.Customer;  
import rikkei.academy.service.CustomerService;  
import java.util.List;  
@Controller  
public class CustomerServlet {  
 @Autowired  
 private CustomerService customerService;  
  
 @GetMapping("/customers")  
 public String showList(Model model) {  
 List<Customer> customers = customerService.findAll();  
 model.addAttribute("customers", customers);  
 return "customers/list";  
 }  
}

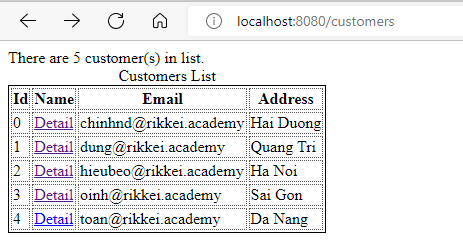
* Ta thấy rằng **customerService** đang báo lỗi đỏ do chưa được tiêm sự phụ thuộc => Ta cần tiêm sự phụ thuộc vào file: **dispatcher-servlet.xml** như sau:

<bean name="customerService" class="rikkei.academy.service.impl.SimpleCustomerServiceImpl"/>

* Kết quả file **dispatcher-servlet.xml** lúc này:

<?xml version="1.0" encoding="UTF-8"?>  
<beans xmlns="http://www.springframework.org/schema/beans"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xmlns:p="http://www.springframework.org/schema/p"  
 xmlns:mvc="http://www.springframework.org/schema/mvc"  
 xsi:schemaLocation="  
 http://www.springframework.org/schema/beans  
 http://www.springframework.org/schema/beans/spring-beans.xsd  
 http://www.springframework.org/schema/context  
 http://www.springframework.org/schema/context/spring-context.xsd  
 http://www.springframework.org/schema/mvc  
 http://www.springframework.org/schema/mvc/spring-mvc.xsd">  
  
 <!-- Spring MVC Annotation -->  
 <context:annotation-config />  
 <mvc:annotation-driven/>  
  
 <!--Controller -->  
 <context:component-scan base-package="rikkei.academy.controller"/>  
 <!--View -->  
 <bean id="viewResolver"  
 p:prefix="/WEB-INF/views/" p:suffix=".jsp"  
 class="org.springframework.web.servlet.view.InternalResourceViewResolver"/>  
 <bean name="customerService" class="rikkei.academy.service.impl.SimpleCustomerServiceImpl"/>  
</beans>

* Quay lại Class: **CustomerServlet =>** Ta thấy hết dòng báo lỗi đỏ**: customerService:**
* Chạy phần mềm và quan sát kết quả:



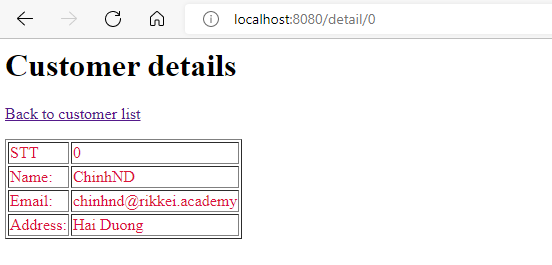
* **Ta thấy rằng vẫn lấy được list giống như JAVA SERVLET -> Nhưng đây là JAVA SPRING MVC -> Như vậy chúng ta đã bước đầu chuyển đổi được từ dự án JAVA SERVLET sang SPRING MVC**
* Với phần: **detail** ta sửa lại code ở file: **info.jsp**

<%@ **taglib** uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>  
<%@ **page** contentType="text/html;charset=UTF-8" language="java" %>  
<html>  
<head>  
 <title>Detail</title>  
</head>  
<body>  
<h1>Customer details</h1>  
<p>  
 <a href="/customers">Back to customer list</a>  
</p>  
  
<table border="1px" style="color: crimson">  
 <tr>  
 <td>STT</td>  
 <td>${requestScope["customer"].getId()}</td>  
 </tr>  
 <tr>  
 <td>Name: </td>  
 <td>${requestScope["customer"].getName()}</td>  
 </tr>  
 <tr>  
 <td>Email: </td>  
 <td>${requestScope["customer"].getEmail()}</td>  
 </tr>  
 <tr>  
 <td>Address: </td>  
 <td>${requestScope["customer"].getAddress()}</td>  
 </tr>  
</table>  
</body>  
</html>

* Thêm code vào **CustomerServlet:**

@GetMapping("/detail/{id}")  
public String detail(@PathVariable Long id, HttpServletRequest request) {  
 Customer customer = customerService.findOne(id);  
 request.setAttribute("customer", customer);  
 return "/customers/info";  
}

* Chạy và thử chức năng detail:



**Tổng kết**

Bạn đã bước đầu biết cách sử dụng dispatcher servlet, bài sau sẽ tiếp tục sử dụng những khả năng mà Spring MVC mang lại để hoàn toàn chuyển đổi dự án này thành sử dụng Spring MVC.

Mã nguồn tham khảo:

<https://github.com/rikkei-academy/MD4-Bai1-TH2-Convert-Servlet-To-Spring-MVC.git>

Hướng dẫn nộp bài:

Up code lên github.

Paste link github vào phần nộp bài.