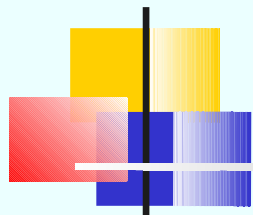




Lập trình mạng

Lập trình web với Spring

Giảng viên: TS. Nguyễn Mạnh Hùng
Học viện Công nghệ Bưu chính Viễn thông (PTIT)



Nội dung

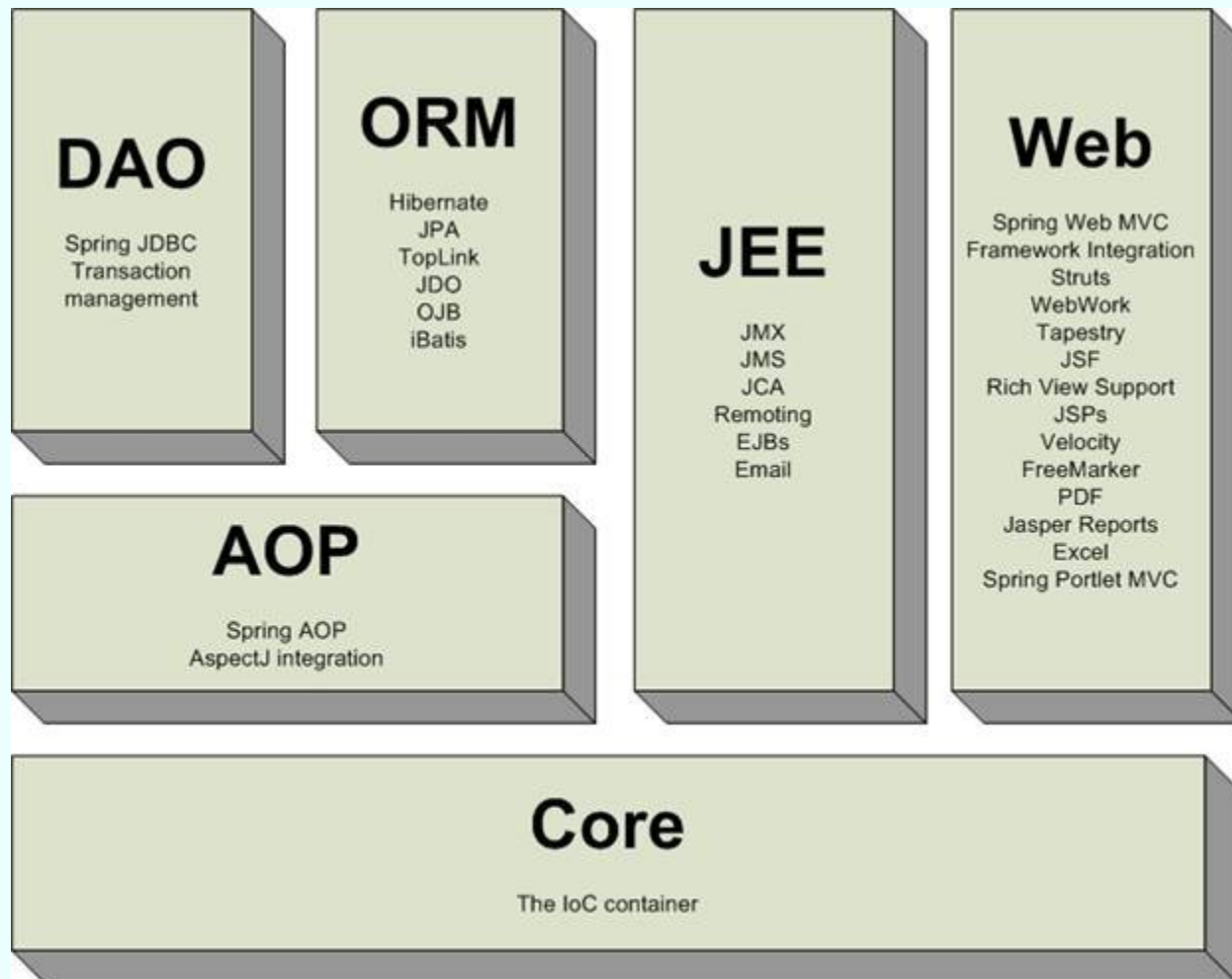
- Giới thiệu về Spring framework
- Lập trình web với Spring
- Bài tập



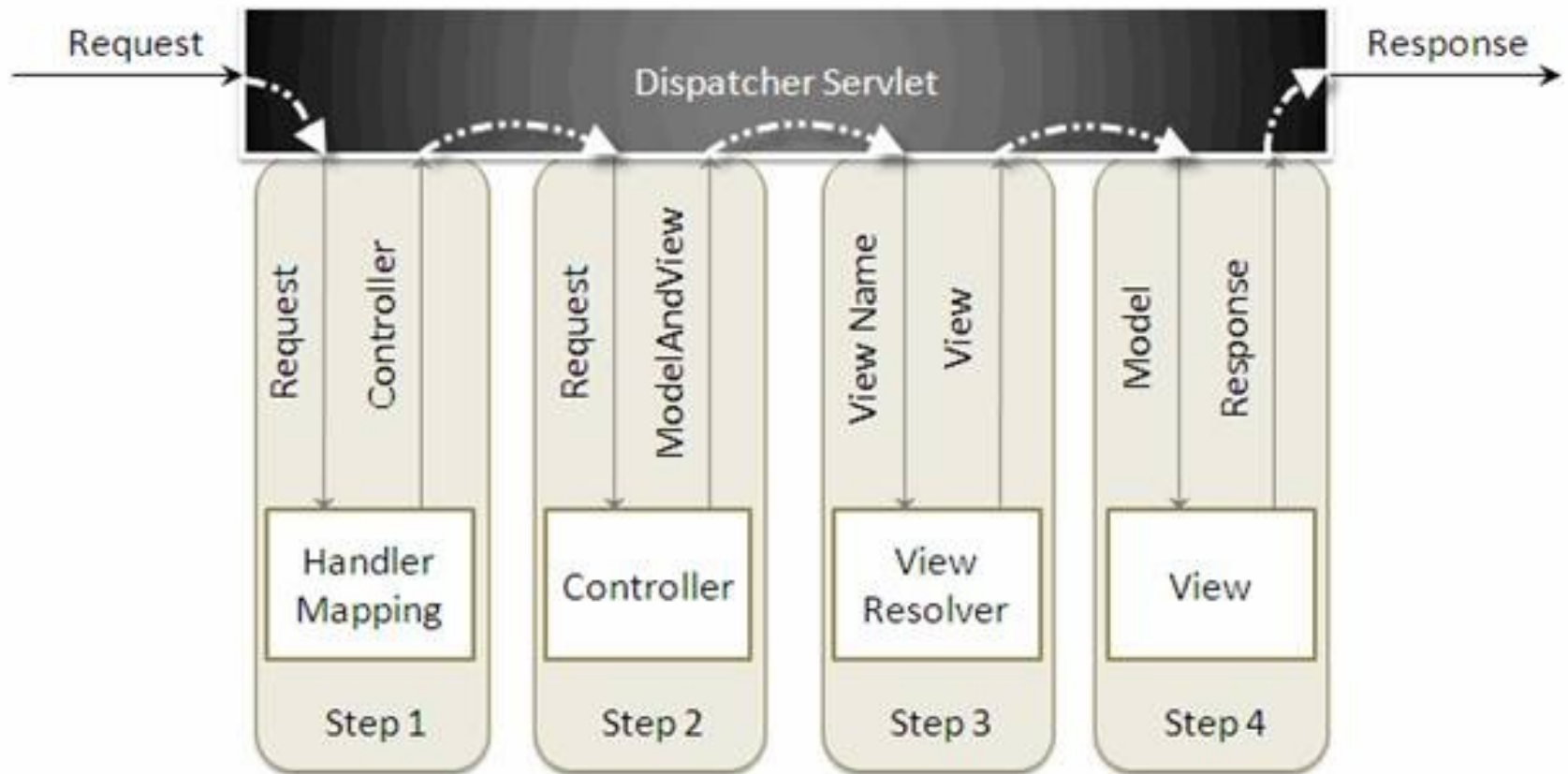
Spring Framework



Component view

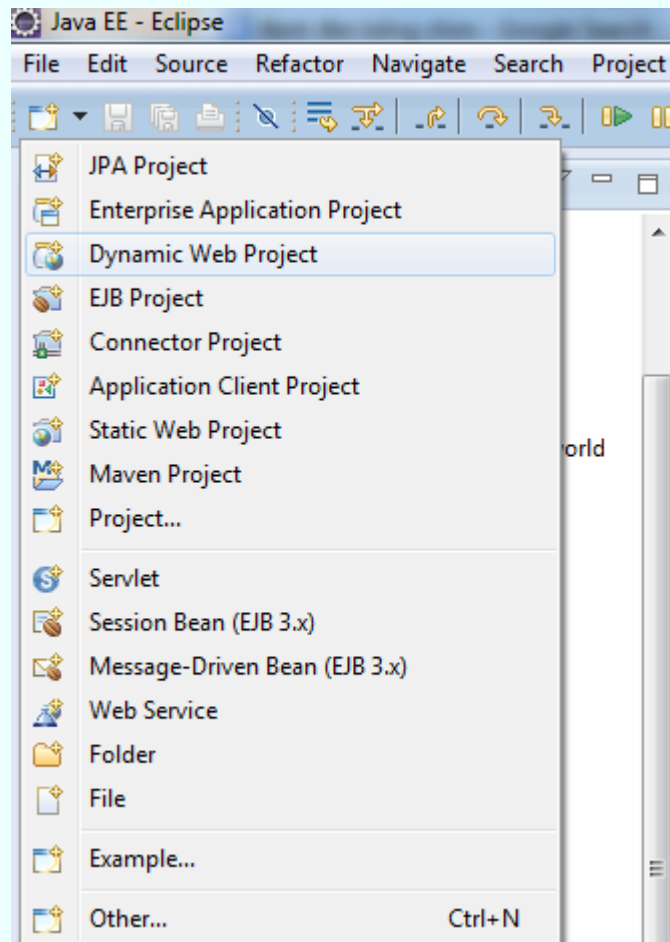


Event handling view



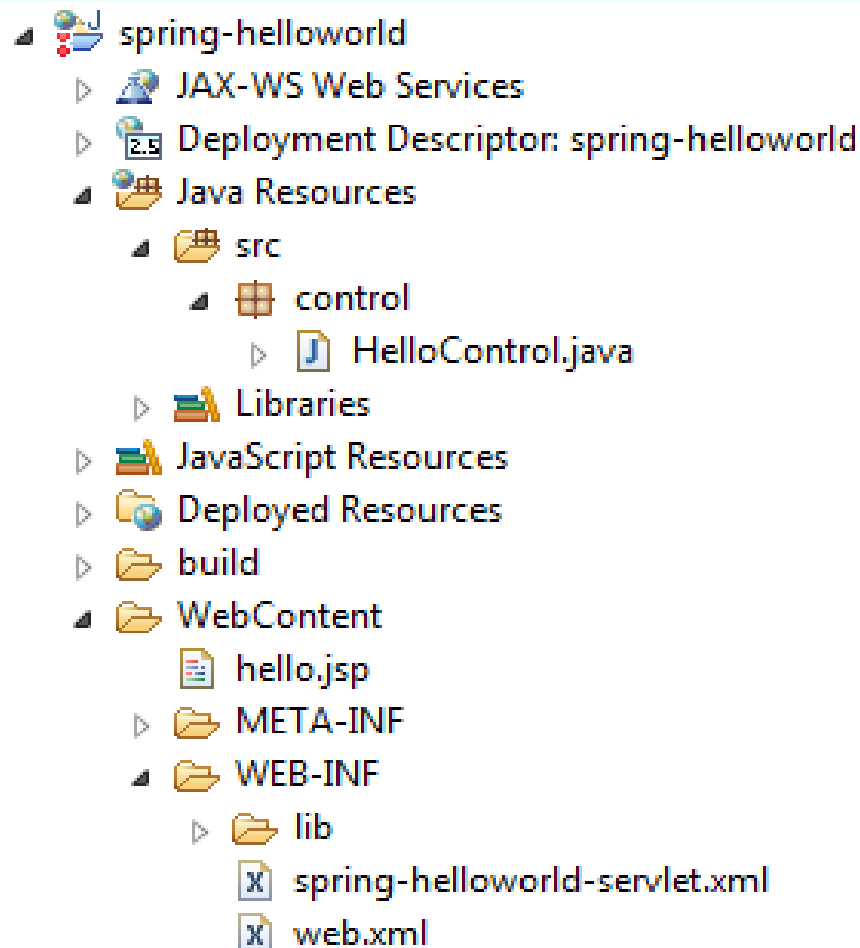
Spring với Eclipse (1)

- New project → Dynamic web project



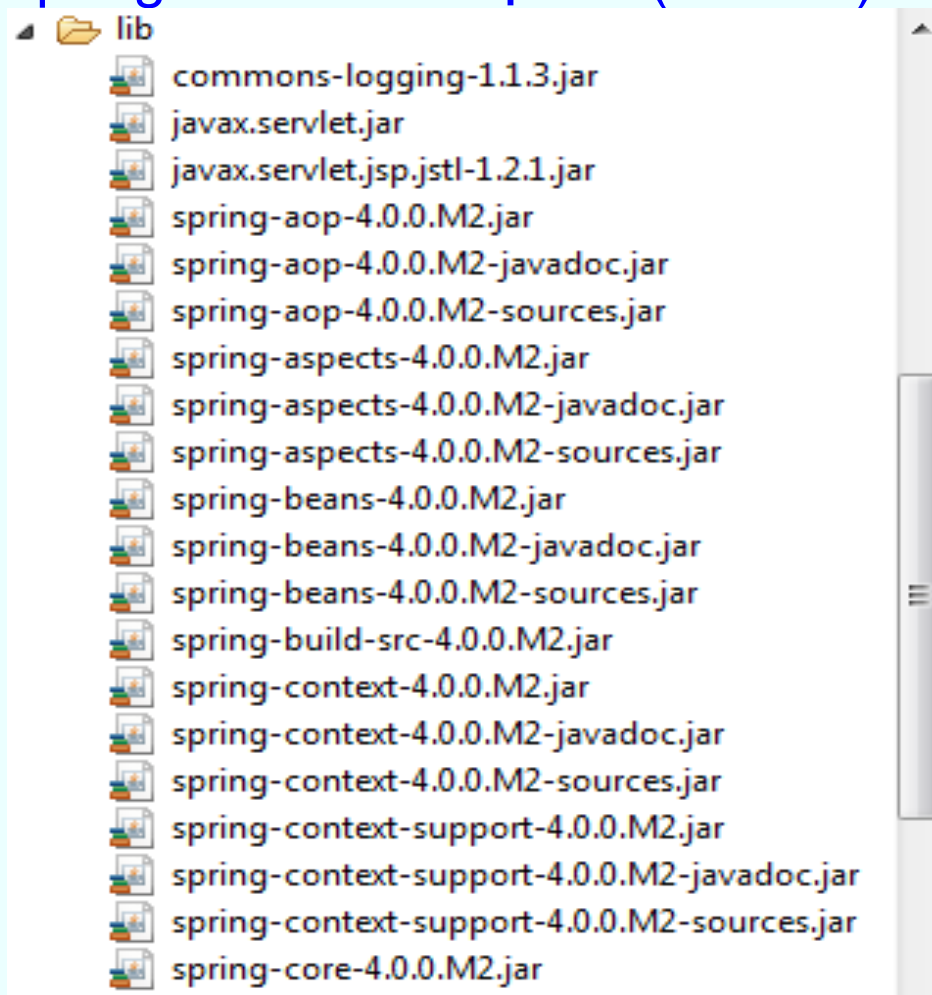
Spring với Eclipse (2)

- Chọn tên project: spring-helloworld, click finish
- Tạo các thư mục và các file tương ứng:



Spring với Eclipse (3)

- Add các file thư viện vào thư mục WEB-INF/lib: copy hết các file .jar của Spring 4 vào thư mục lib (61 files)





HelloControl.java

```
package control;
```

```
import org.springframework.stereotype.Controller;
```

```
import org.springframework.web.bind.annotation.RequestMapping;
```

```
import org.springframework.web.bind.annotation.RequestMethod;
```

```
import org.springframework.ui.ModelMap;
```

```
@Controller
```

```
public class HelloControl{
```

```
    @RequestMapping(value = "/hello", method = RequestMethod.GET)
```

```
    public String printHello(ModelMap model) {
```

```
        model.addAttribute("message",  
                            "Hello Spring MVC Web Framework!");
```

```
        return "hello";
```

```
    }
```

```
}
```



web.xml

```
<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>Spring MVC Web Application</display-name>

  <servlet>
    <servlet-name>spring-helloworld</servlet-name>
    <servlet-class>
      org.springframework.web.servlet.DispatcherServlet
    </servlet-class>
    <init-param>
      <param-name>contextConfigLocation</param-name>
      <param-value>/WEB-INF/spring-helloworld-
servlet.xml</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>

  <servlet-mapping>
    <servlet-name>spring-helloworld</servlet-name>
    <url-pattern>/</url-pattern>
  </servlet-mapping>
</web-app>
```



spring-helloworld-servlet.xml

```
<beans xmlns="http://www.springframework.org/schema/beans"
  xmlns:context="http://www.springframework.org/schema/context"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="
    http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans-3.0.xsd
    http://www.springframework.org/schema/context
    http://www.springframework.org/schema/context/spring-context-3.0.xsd">

  <context:component-scan base-package="control" />

  <bean class="org.springframework.web.servlet.view.
        InternalResourceViewResolver">
    <property name="prefix" value="/" />
    <property name="suffix" value=".jsp" />
  </bean>

</beans>
```

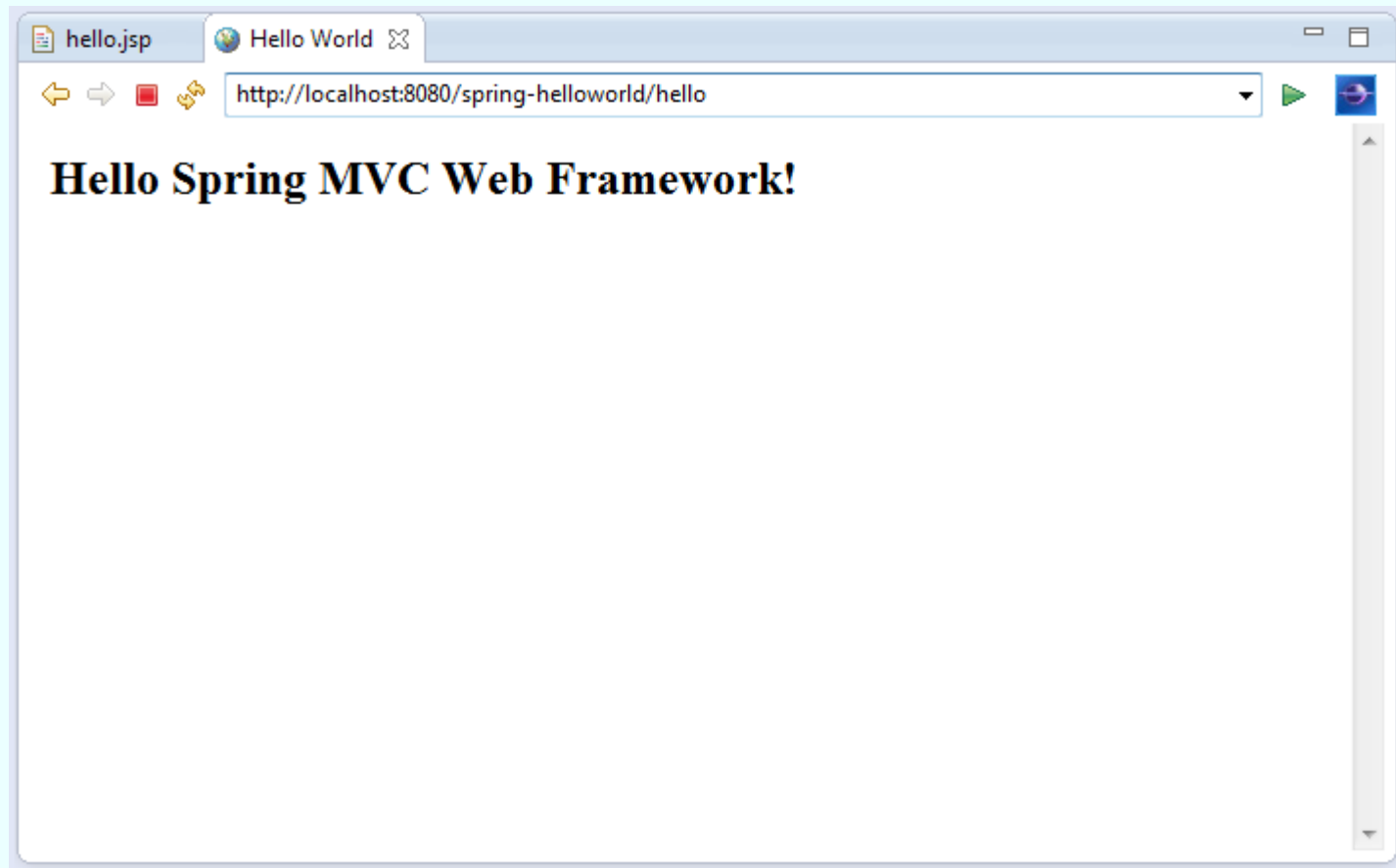


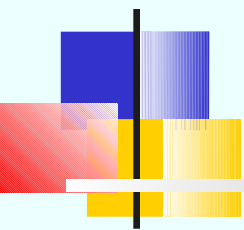
hello.jsp (1)

```
<%@ page contentType="text/html; charset=UTF-8" %>
<html>
<head>
<title>Hello World</title>
</head>
<body>
    <h2>${message}</h2>
</body>
</html>
```



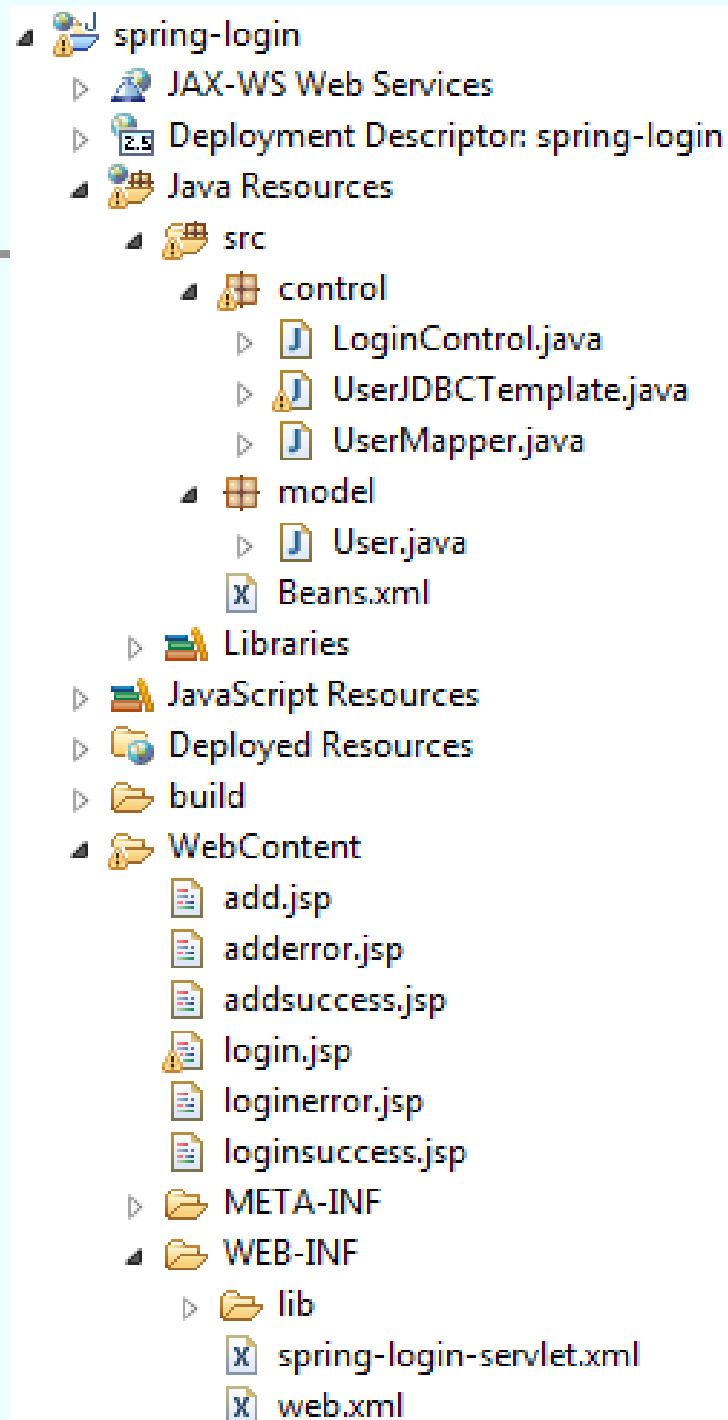
hello.jsp (2)














Ví dụ 1: Login dùng Spring

Tạo project



Bảng User

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
 id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
 username	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 password	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 fullName	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 idCardNumber	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 idCardType	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 address	VARCHAR(250)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 description	VARCHAR(500)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 address	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL



User.java

```
package model;
```

```
public class User {  
    private Integer id;  
    private String username;  
    private String password;  
    private String fullName;  
    private String idCardNumber;  
    private String idCardType;  
    private String address;  
    private String description;  
  
    public Integer getId() {  
        return id;  
    }  
    public void setId(Integer id) {  
        this.id = id;  
    }  
}
```

//getter và setter các thuộc tính còn lại



Beans.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-3.0.xsd ">

    <!-- Initialization for data source -->
    <bean id="dataSource"
class="org.springframework.jdbc.datasource.DriverManagerDataSource">
        <property name="driverClassName" value="com.mysql.jdbc.Driver"/>
        <property name="url"
value="jdbc:mysql://localhost:3306/hotelmanagement"/>
        <property name="username" value="root"/>
        <property name="password" value="12345678"/>
    </bean>

    <!-- Definition for userJdbcTemplate bean -->
    <bean id="userJdbcTemplate" class="control.UserJdbcTemplate">
        <property name="dataSource" ref="dataSource" />
    </bean>
</beans>
```



UserJdbcTemplate.java

```
package control;
import java.util.List;
import javax.sql.DataSource;
import model.User;
import org.springframework.jdbc.core.JdbcTemplate;

public class UserJdbcTemplate {
    private DataSource dataSource;
    private JdbcTemplate jdbcTemplateObject;

    public void setDataSource(DataSource dataSource) {
        this.dataSource = dataSource;
        this.jdbcTemplateObject = new JdbcTemplate(dataSource);
    }

    public boolean checkLogin(User user) {
        String SQL = "select * from tblUser where username = ?
                        and password = ?";

        List<User> result = jdbcTemplateObject.query(SQL,
new Object[]{user.getUsername(),user.getPassword()}, new UserMapper());
        if(result.size()>0)
            return true;
        return false;
    }
}
```



UserMapper.java

```
package control;
```

```
import java.sql.ResultSet;
```

```
import java.sql.SQLException;
```

```
import org.springframework.jdbc.core.RowMapper;
```

```
import model.User;
```

```
public class UserMapper implements RowMapper<User> {
```

```
    public User mapRow(ResultSet rs, int rowNum) throws SQLException {
```

```
        User user = new User();
```

```
        user.setId(rs.getInt("id"));
```

```
        user.setUsername(rs.getString("username"));
```

```
        user.setPassword(rs.getString("password"));
```

```
        user.setFullName(rs.getString("fullName"));
```

```
        user.setIdCardNumber(rs.getString("idCardNumber"));
```

```
        user.setIdCardType(rs.getString("idCardType"));
```

```
        user.setAddress(rs.getString("address"));
```

```
        user.setDescription(rs.getString("description"));
```

```
        return user;
```

```
    }
```

```
}
```



LoginControl.java (1)

```
package control;
import model.User;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.ui.ModelMap;
```

```
@Controller
```

```
public class LoginControl{
    private ApplicationContext context = null;
    private UserJDBCTemplate userJDBCTemplate = null;
    public LoginControl(){
        context = new ClassPathXmlApplicationContext("Beans.xml");
        userJDBCTemplate =
            (UserJDBCTemplate)context.getBean("userJDBCTemplate");
    }
}
```

```
@RequestMapping(value = "/login", method = RequestMethod.GET)
public ModelAndView userLogin() {
    return new ModelAndView("login", "command", new User());
}
```



LoginControl.java (2)

```
@RequestMapping(value = "/loginCheck", method = RequestMethod.POST)
public String checkUser(@ModelAttribute("SpringWeb")User user,
    ModelMap model) {
    model.addAttribute("username", user.getUsername());

    if(userJdbcTemplate.checkLogin(user)){
        return "loginsuccess";
    }
    return "loginerror";
}
```



web.xml

```
<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>Spring MVC Web Application</display-name>

  <servlet>
    <servlet-name>spring-login</servlet-name>
    <servlet-class> org.springframework.web.servlet.DispatcherServlet
    </servlet-class>
    <init-param>
      <param-name>contextConfigLocation</param-name>
      <param-value>/WEB-INF/spring-login-servlet.xml</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>

  <servlet-mapping>
    <servlet-name>spring-login</servlet-name>
    <url-pattern>/</url-pattern>
  </servlet-mapping>

</web-app>
```



spring-login-servlet.xml

```
<beans xmlns="http://www.springframework.org/schema/beans"
  xmlns:context="http://www.springframework.org/schema/context"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:mvc="http://www.springframework.org/schema/mvc"

  xsi:schemaLocation="
    http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans-3.0.xsd
    http://www.springframework.org/schema/context
    http://www.springframework.org/schema/context/spring-context-3.0.xsd
    http://www.springframework.org/schema/mvc
    http://www.springframework.org/schema/mvc/spring-mvc-3.0.xsd">

  <mvc:annotation-driven />

  <context:component-scan base-package="control" />

  <bean id="viewResolver"
class="org.springframework.web.servlet.view.InternalResourceViewResolver">
    <property name="prefix" value="/" />
    <property name="suffix" value=".jsp" />
  </bean>

</beans>
```




Trang login.jsp (1)

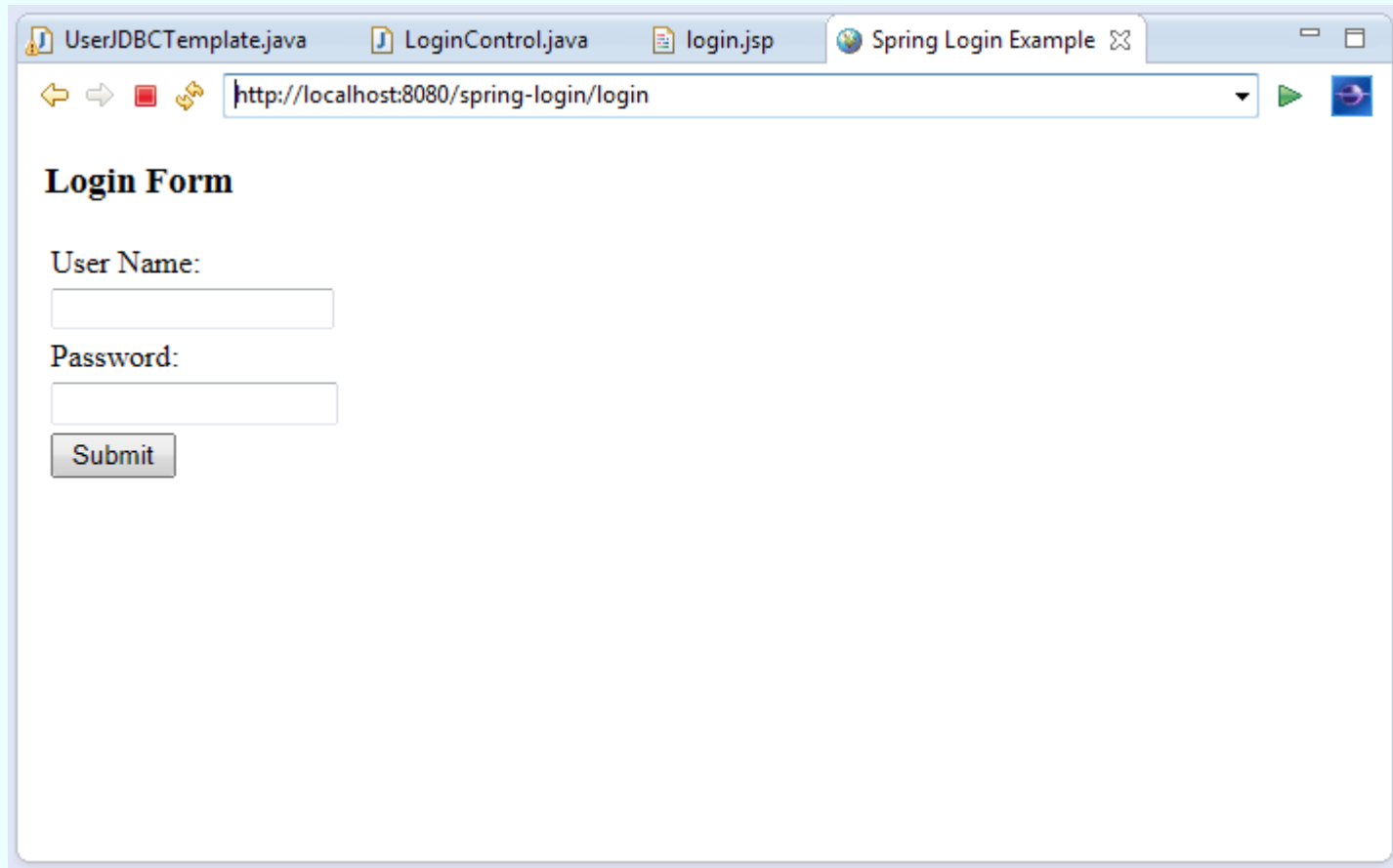
```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Spring Login Example</title>
</head>

<body>
<h3>Login Form</h3>
<form:form method="POST" action="/spring-login/LoginCheck">
<table>
  <tr><td>User Name:</td></tr>
  <tr><td><form:input path="username" /></td></tr>
  <tr><td>Password:</td></tr>
  <tr><td><form:password path="password" /></td></tr>
  <tr><td><input type="submit" value="Submit" /></td></tr>
</table>
</form:form>
</body>
</html>
```



Trang login.jsp (2)



UserJDBCTemplate.java LoginControl.java login.jsp Spring Login Example

http://localhost:8080/spring-login/login

Login Form

User Name:

Password:

Submit



Trang loginsuccess.jsp (1)

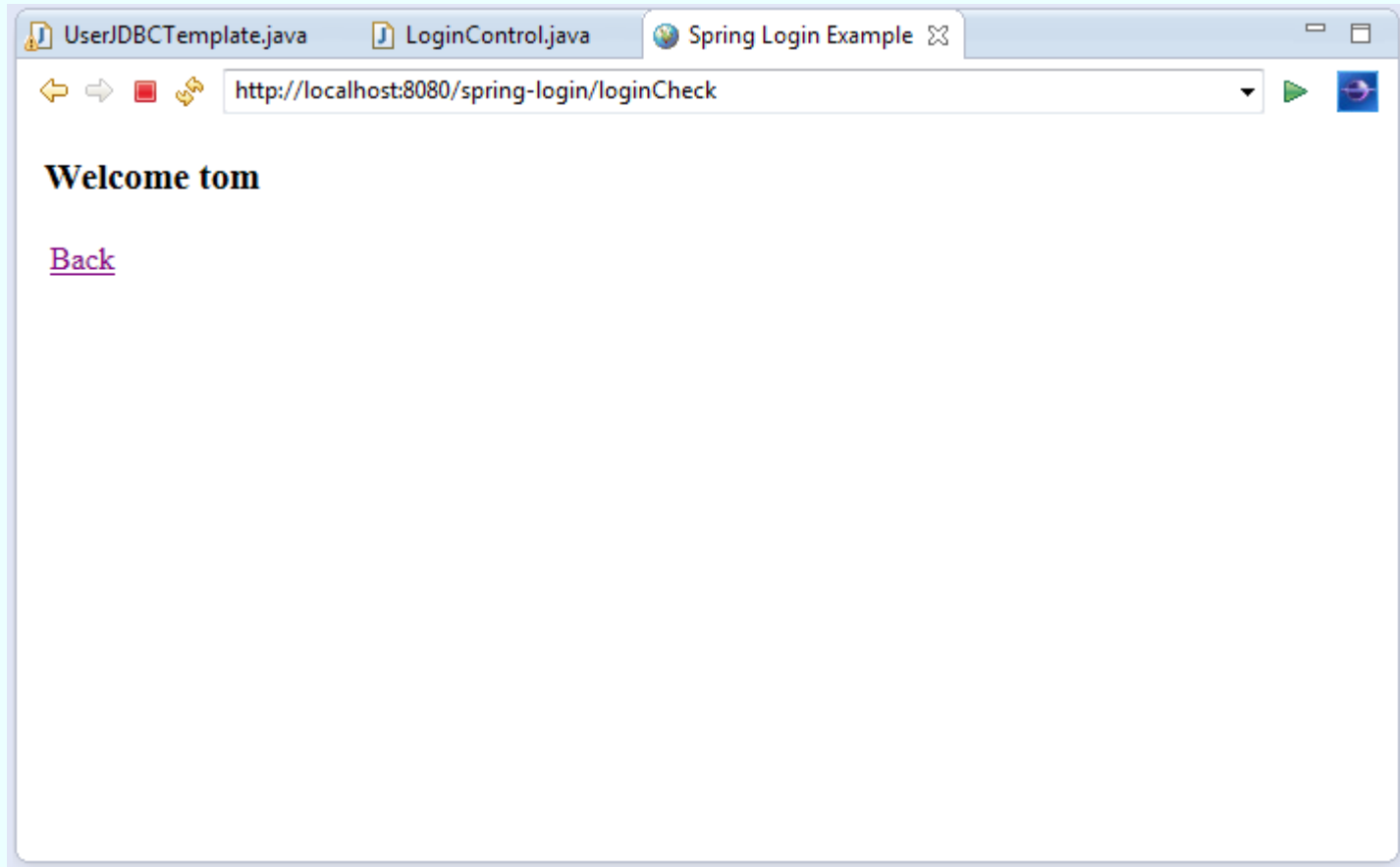
```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="core" uri="http://java.sun.com/jsp/jstl/core"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Spring Login Example</title>
</head>

<body>
<h3>Welcome <core:out value="${username}" /></h3>
<table>
  <tr>
    <td><a href="Login">Back</a></td>
  </tr>
</table>
</body>
</html>
```



Trang loginsuccess.jsp (2)





Trang loginerror.jsp (1)

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
```

```
<title>Spring Login Example</title>
```

```
</head>
```

```
<body>
```

```
<h3>Login Error !!! Click below to login again</h3>
```

```
<table>
```

```
    <tr>
```

```
        <td><a href="Login">Retry</a></td>
```

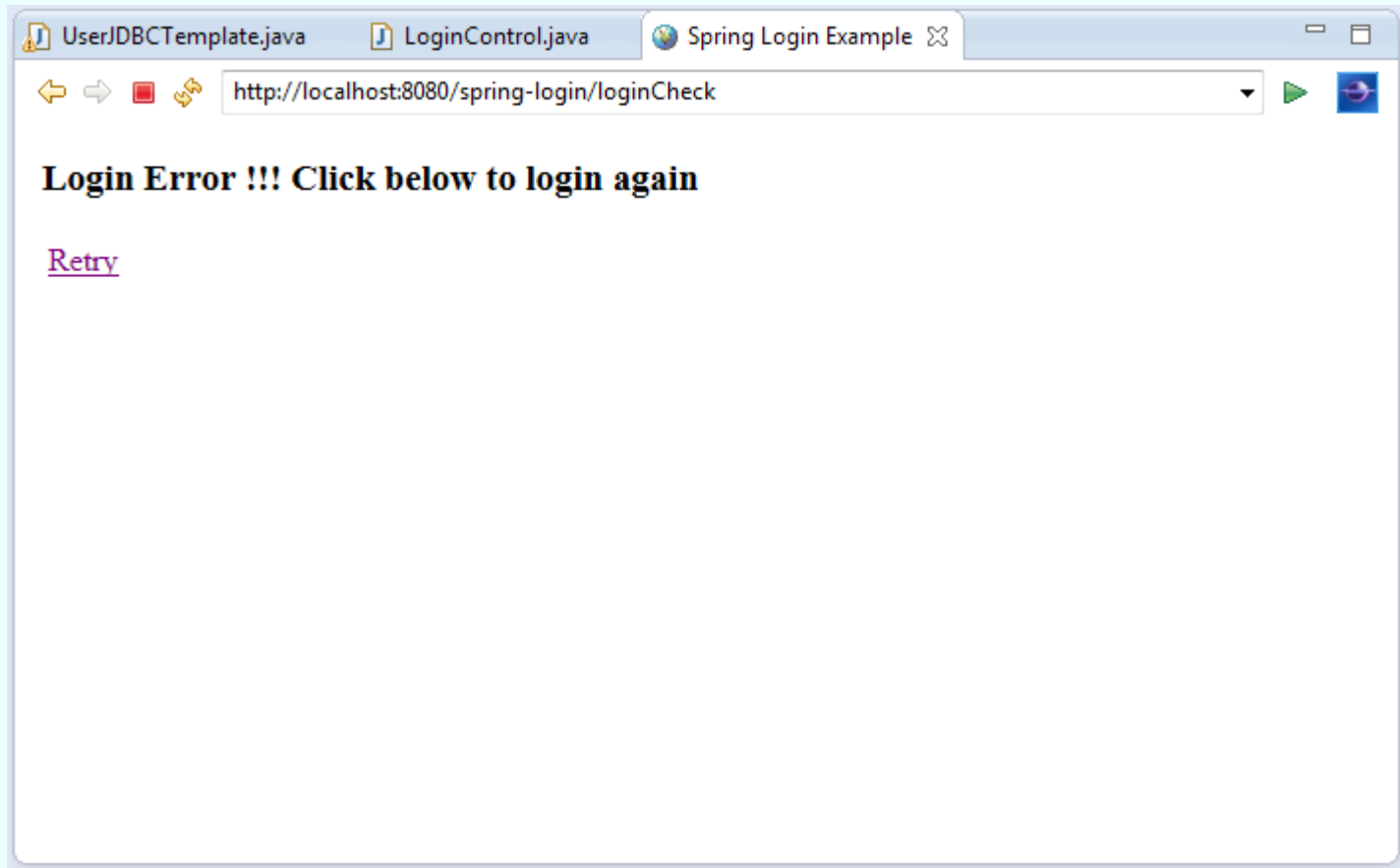
```
    </tr>
```

```
</table>
```

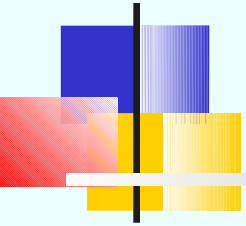
```
</body>
```

```
</html>
```

Trang loginerror.jsp (2)



Bổ sung chức năng: Đăng kí user mới





Thêm vào lớp UserJdbcTemplate.java

```
public boolean create(User user) {
    String SQL = "select * from tblUser where username = ?";
    List<User> result = jdbcTemplateObject.query(SQL,
        new Object[]{user.getUsername()}, new UserMapper());
    if(result.size()>0)
        return false;

    SQL = "insert into tblUser (username, password, fullName, idCardNumber,
        idCardType, address, description) values (?, ?, ?, ?, ?, ?, ?)";

    jdbcTemplateObject.update( SQL, user.getUsername(), user.getPassword(),
        user.getFullName(), user.getIdCardNumber(), user.getIdCardType(),
        user.getAddress(), user.getDescription());
    return true;
}
```




Thêm vào lớp LoginControl.java

```
@RequestMapping(value = "/add", method = RequestMethod.GET)
public ModelAndView userAdd() {
    return new ModelAndView("add", "command", new User());
}

@RequestMapping(value = "/addUser", method = RequestMethod.POST)
public String addUser(@ModelAttribute("SpringWeb")User user,
    ModelMap model) {
    model.addAttribute("username", user.getUsername());

    if(userJdbcTemplate.create(user)){
        return "addsuccess";
    }

    return "adderror";
}
```



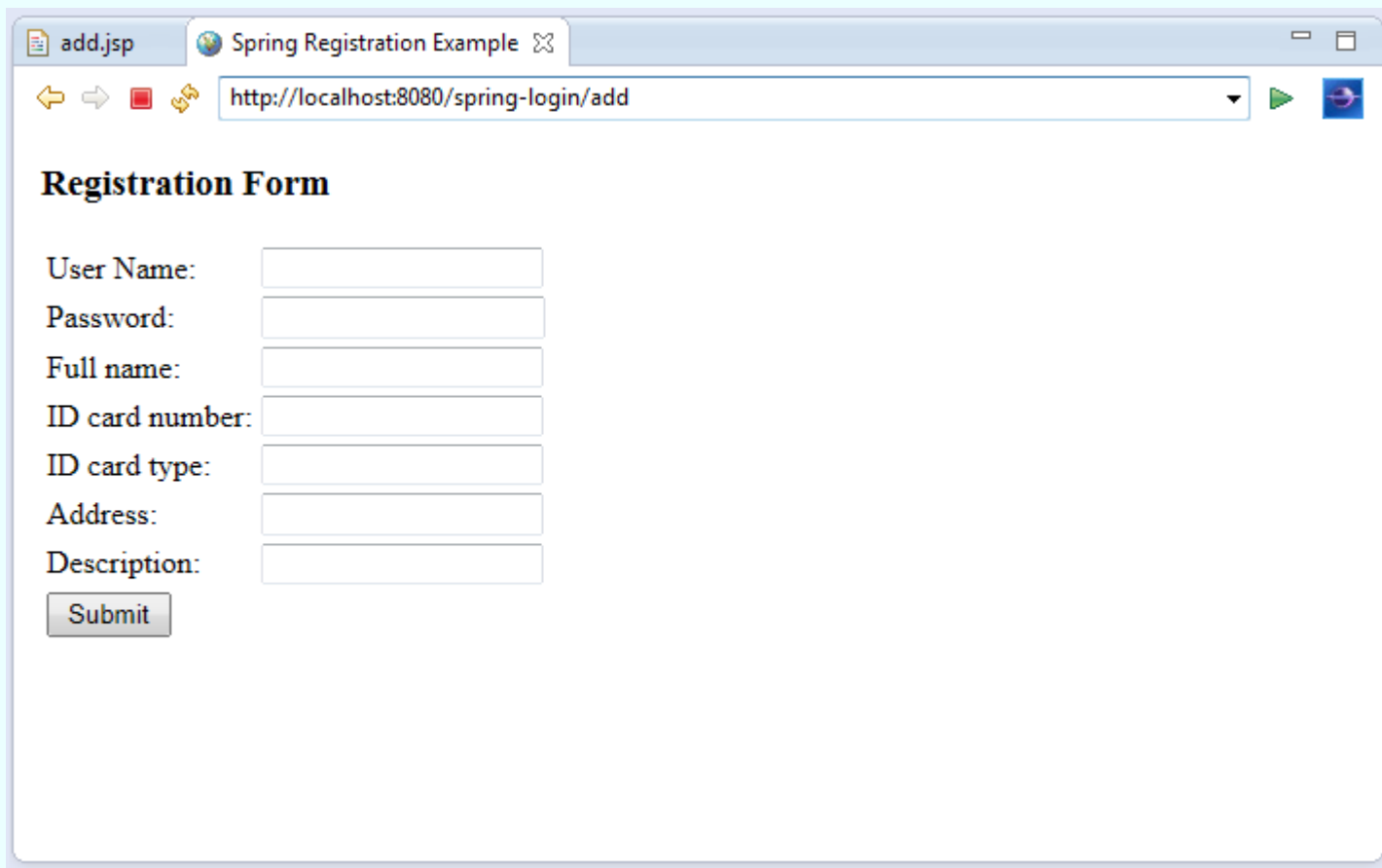
Trang add.jsp (1)

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Spring Registration Example</title>
</head>

<body><h3>Registration Form</h3>
<form:form method="POST" action="/spring-login/addUser">
<table>
  <tr><td>User Name:</td><td><form:input path="username" /></td></tr>
  <tr><td>Password:</td><td><form:password path="password" /></td></tr>
  <tr><td>Full name:</td><td><form:input path="fullName" /></td></tr>
  <tr><td>ID card number:</td><td><form:input path="idCardNumber" /></td></tr>
  <tr><td>ID card type:</td><td><form:input path="idCardType" /></td></tr>
  <tr><td>Address:</td><td><form:input path="address" /></td></tr>
  <tr><td>Description:</td><td><form:input path="description" /></td></tr>
  <tr><td><input type="submit" value="Submit" /></td></tr>
</table>
</form:form>
</body>
</html>
```

Trang add.jsp (2)



The screenshot shows a web browser window with the title 'Spring Registration Example'. The address bar displays 'http://localhost:8080/spring-login/add'. The page content is titled 'Registration Form' and contains several input fields and a submit button.

Registration Form

User Name:

Password:

Full name:

ID card number:

ID card type:

Address:

Description:



Trang addsuccess.jsp (1)

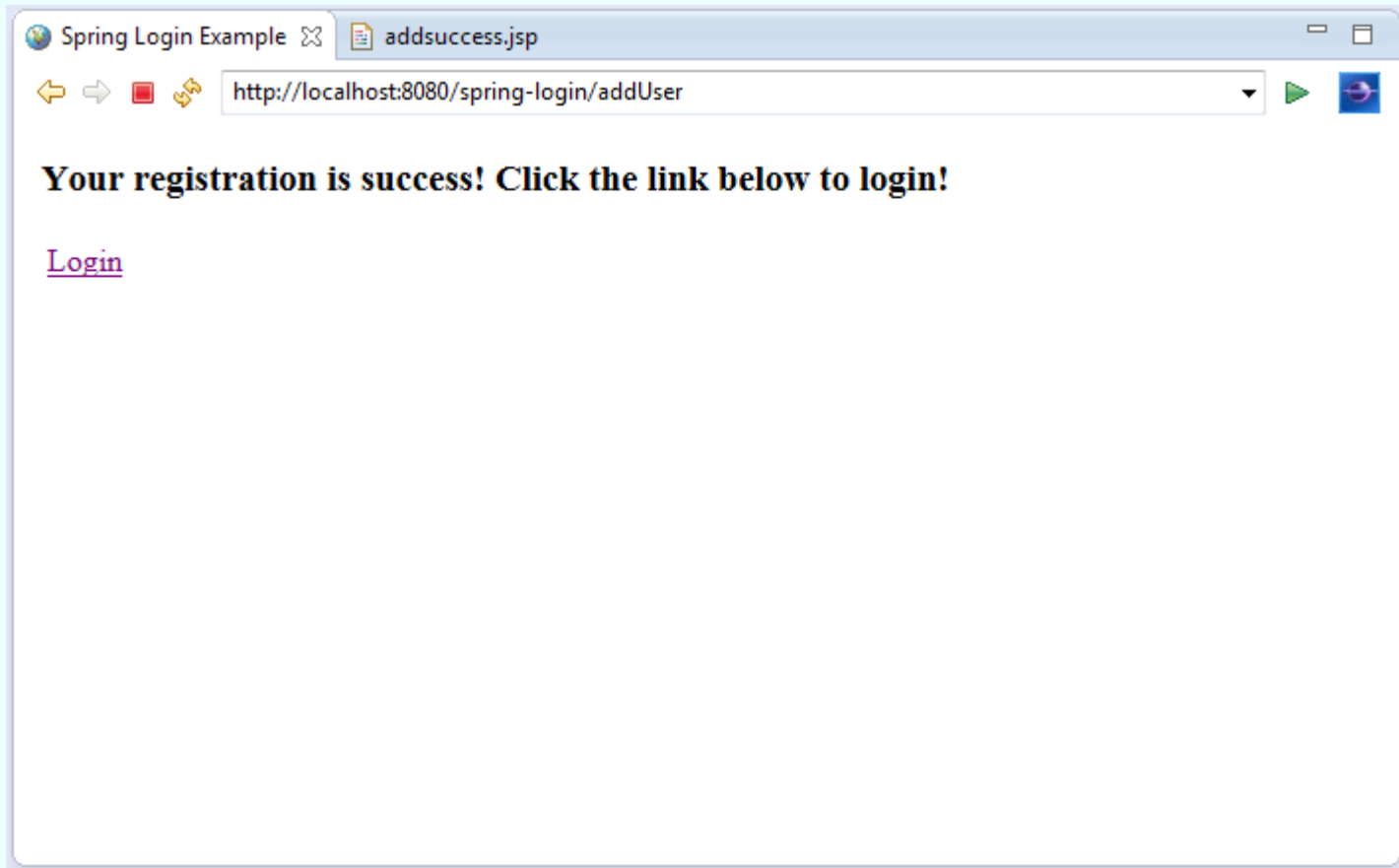
```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="core" uri="http://java.sun.com/jsp/jstl/core"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Spring Login Example</title>
</head>

<body>
<h3>Your registration is success! Click the link below to login!</h3>
<table>
    <tr>
        <td><a href="login">Login</a></td>
    </tr>
</table>
</body>
</html>
```



Trang addsuccess.jsp (2)





Trang adderror.jsp (1)

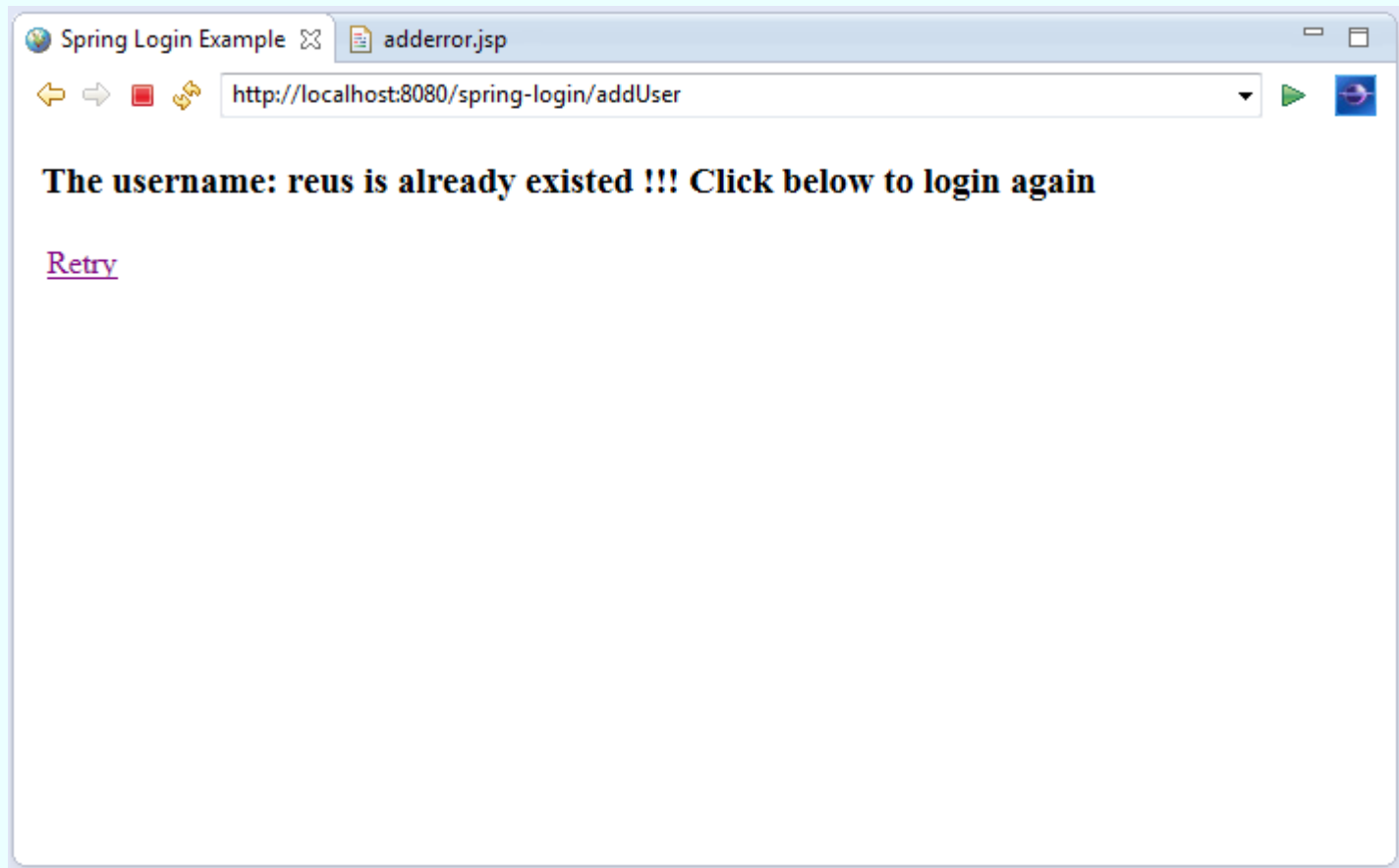
```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="core" uri="http://java.sun.com/jsp/jstl/core"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Spring Login Example</title>
</head>

<body>
<h3>The username: <core:out value="${username}" /> is already
existed !!! Click below to login again</h3>
<table>
    <tr>
        <td><a href="add">Retry</a></td>
    </tr>
</table>
</body>
</html>
```



Trang adderror.jsp (2)





Bài tập (1)

Thiết kế CSDL cho phần mềm quản lí bán vé cho một chuỗi rạp chiếu phim với mô tả như sau:

- Hãng có một chuỗi rạp chiếu phim (Mã rạp, tên rạp, địa chỉ, giới thiệu).
- Mỗi rạp chiếu phim có nhiều phòng chiếu khác nhau (Mã phòng chiếu, số lượng ghế, đặc điểm phòng chiếu)
- Mỗi phim (Mã phim, tên phim, loại phim, năm sản xuất, mô tả) có thể được chiếu tại nhiều phòng chiếu khác nhau vào nhiều thời điểm khác nhau
- Mỗi phòng chiếu có thể chiếu nhiều phim khác nhau tại nhiều thời điểm khác nhau
- Mỗi một thời điểm nhất định, trong một phòng chiếu chỉ có duy nhất một phim được chiếu



Bài tập (2)

Thiết kế CSDL cho phần mềm quản lí bán vé cho một chuỗi rạp chiếu phim như trước, có bổ sung phần bán vé như sau:

- Cùng một phim, chiếu tại cùng 1 phòng chiếu nhưng nếu ở các khung giờ và ngày khác nhau có thể có giá vé khác nhau.
- Nhân viên chỉ bán vé cho khách hàng khi phòng chiếu tại giờ chiếu mà khách hàng yêu cầu vẫn còn đủ số lượng ghế trống cho khách hàng.
- Khách hàng có thể trả lại vé sau khi đã mua, và có thể phải chịu tiền phạt: trả trước 48h thì miễn phí, trả trước 24h thì mất phí 20%, trả trước 12h thì mất phí 40%, trả trước 6h thì mất phí 60%, trả sau 6h thì mất phí 100%, tính từ giờ khởi chiếu



Bài tập (3)

Thiết kế CSDL cho phần mềm quản lí bán vé cho một chuỗi rạp chiếu phim như trước, có bổ sung phần khách hàng thân thiết như sau:

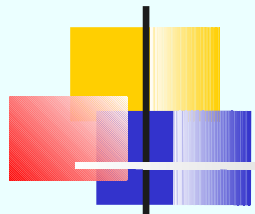
- Mỗi khách hàng có một thẻ khách hàng thân thiết có thể tích điểm.
- Mỗi lần mua vé có xuất thẻ thì khách hàng được cộng điểm theo tỉ lệ: cứ 10000VND thì được cộng thêm 1 điểm. Ví dụ nếu hóa đơn thanh toán 116000VND thì được cộng 11 điểm.
- Khi số điểm đạt ngưỡng nào đấy thì có thể đổi vé xem phim miễn phí. Ví dụ, cứ 200 điểm được đổi 1 vé xem phim 2D, 400 điểm được đổi 1 vé xem phim 3D. Nếu khách hàng có 280 điểm và mua 2 vé xem phim với tổng hóa đơn là 200000VND, nếu khách hàng thanh toán hết thì sẽ được cộng 20 điểm vào thẻ. Nếu khách hàng muốn đổi vé miễn phí thì sẽ được đổi 1 vé, khách hàng chỉ phải trả 100000VND cho vé còn lại, trong thẻ còn 80 điểm, và sau giao dịch này khách hàng chỉ được cộng số điểm bằng tỉ lệ phần tiền thanh toán: 10 điểm ứng với 100000VND.



Bài tập (4)

Với CSDL cho phần mềm quản lí bán vé cho một chuỗi rạp chiếu phim như trước, cài đặt các chức năng sau theo springs:

- Thêm một phim mới vào CSDL
- Lên lịch chiếu phim trong một ngày cho một phòng chiếu
- Bán một số vé của một phim trong một phòng chiếu ở một khung giờ nào đó
- Tìm kiếm các phim + phòng chiếu + giờ chiếu trong một khoảng thời gian nào đó mà đang còn ghế trống
- Thống kê các phim có doanh thu cao nhất đến thấp nhất trong một khoảng thời gian nào đó
- Thống kê các khung giờ chiếu có đông khách hàng xem phim từ cao nhất đến thấp nhất
- Thống kê các khách hàng mua nhiều vé nhất đến ít vé nhất trong một khoảng thời gian nào đó



Bài tập (5)

- Cài đặt bài toán quản lí khách sạn theo mô hình spring



Questions?
