



FORM TAGS & DATABINDING

ThS. Nguyễn Nghiệm
0913.745.789 - NghiemN@fpt.edu.vn



NỘI DUNG

- Giới thiệu databinding
- Databinding đơn giản
- Form tags
- Hiển thị và buộc dữ liệu cho các List Control
 - ✱ ComboBox
 - ✱ ListBox
 - ✱ Nhóm CheckBox
 - ✱ Nhóm Radio





DATABINDING LÀ GÌ?

- Databinding là sự kết nối dữ liệu của bean trong model đến các thành phần giao diện
- Khi thay đổi dữ liệu trong bean thì dữ liệu trên các thành phần giao diện cũng thay đổi theo.
- Phương pháp thông thường

```
<form action="login.htm" method="post">
  <div>User Name:</div>
  <input name="id" value="${user.id}">

  <div>Password:</div>
  <input name="password" value="${user.password}">

  <hr>
  <button>Login</button>
</form>
```



HIỂN THỊ DỮ LIỆU LÊN FORM

- Hạn chế đối với hiển thị dữ liệu thông thường

- ✱ Đổ dữ liệu và chọn các mục trên các List Control

- Combox
- Listbox
- Radiobuttons
- Checkboxes

Rõ ràng bạn cần phải sử dụng các vòng lặp để sinh các list control trên => phức tạp khi viết mã

- ✱ Kiểm và thông báo lỗi



SPRING FORM

- Spring MVC cung cấp thư viện thẻ giúp kết nối dữ liệu lên form dễ dàng hơn
- Form sau sẽ thực hiện buộc dữ liệu của bean user lên form tương tự như trên

```
<%@ taglib uri="http://www.springframework.org/tags/form" prefix="form" %>
```

```
<form:form action="login.htm" modelAttribute="user">
```

```
    <div>User Name:</div>
```

```
    <form:input path="id"/>
```

```
    <div>Password:</div>
```

```
    <form:input path="password"/>
```

```
    <hr>
```

```
    <button>Login</button>
```

```
</form:form>
```



DATABINDING LÀ GÌ?

● Với Spring Form

- ✱ Khi thay đổi dữ liệu trong bean thì dữ liệu trên các thành phần giao diện cũng thay đổi theo.
- ✱ Kiểm và hiển thị lỗi một cách dễ dàng

● Thực hiện

- ✱ Model: lớp bean mô tả thông tin form
- ✱ Action: Tạo bean và đặt vào model
- ✱ View: tạo spring form kết nối với bean



SPRING FORM

```
@RequestMapping("spring-form")
public String databinding(ModelMap model) {
    Employee employee = new Employee();
    employee.setEmail("nghiemn@fpt.edu.vn");
    employee.setName("Nguyễn Nghiệm");
    employee.setAge(44);

    model.addAttribute("employee", employee);
    return "spring-form";
}
```

```
public class Employee {
    String name;
    String email;
    Integer age;
    getters/setters
}
```

1

2

```
<form:form action="spring-form.htm"
            modelAttribute="employee">
    <div>Full Name:</div>
    <form:input path="name"/>
    <div>Age:</div>
    <form:input path="age"/>
    <div>Email Address:</div>
    <form:input path="email"/>
    <hr>
    <button>Save</button>
</form:form>
```

3

Spring MVC x

localhost:8080

SPRING FORM

Full Name:

Age:

Email Address:



FORM TAGS

<%@ taglib uri="http://www.springframework.org/tags/form" prefix="form" %>

- <form:form>
 - <form:input/>
 - <form:textarea/>
 - <form:checkbox/>
 - <form:radio/>
 - <form:hidden/>
 - <form:password/>
 - <form:button/>
 - <form:select/>
 - <form:radio buttons/>
 - <form:checkboxes/>
- <form>
 - <input type="text"/>
 - <textarea/>
 - <input type='checkbox'/>
 - <input type='radio'/>
 - <input type='hidden'/>
 - <input type='password'/>
 - <button/>
 - <select/>
 - Nhóm radio
 - Nhóm checkbox



ĐỔ DỮ LIỆU VÀO LIST CONTROL

Spring MVC - List Control x

localhost:8080/Blank/list-contrc

ĐỔ DỮ LIỆU VÀO LIST CONTROL

Tên nhân viên:

Quốc tịch:

Trình độ:
☐ Doctor ☒ Master ☐ University ☐ Other

Chức danh:

Sở thích:
☒ Travelling ☐ Music ☒ Sports ☐ Reading

<form:select>

<form:checkboxes>

<form:select multiple>

<form:checkboxes>

List<Country>

String[]



LIST-CONTROL.JSP

```
<form:form action="list-control.htm" modelAttribute="employee">
  <div>Tên nhân viên:</div>
  <form:input path="name"/>

  <div>Quốc tịch:</div>
  <form:select path="country"
    items="${countries}" itemValue="id" itemLabel="name"/>

  <div>Trình độ:</div>
  <form:radiobuttons path="degree" items="${degrees}"/>

  <div>Chức danh:</div>
  <form:select path="positions" multiple="true" size="3"
    items="${positions}"/>

  <div>Sở thích:</div>
  <form:checkboxes path="hobbies" items="${hobbies}"/>

  <hr>
  <button>Chấp nhận</button>
</form:form>
```



ĐỒ DỮ LIỆU VÀO COMBOBOX

```
@ModelAttribute("countries")  
public List<Country> getCountries() {  
    Country c1 = new Country();  
    c1.setId("VN");  
    c1.setName("Việt Nam");
```

```
    Country c2 = new Country();  
    c2.setId("US");  
    c2.setName("United States of America");
```

```
    Country c3 = new Country();  
    c3.setId("UK");  
    c3.setName("United Kindom");
```

```
    List<Country> list = new ArrayList<Country>();  
    list.add(c1);  
    list.add(c2);  
    list.add(c3);
```

```
    return list;
```

```
}
```

```
public class Country {  
    String id;  
    String name;  
  
    getters/setters  
}
```

United States of America ▼

```
<form:select path="country"  
    items="${countries}" itemValue="id" itemLabel="name"/>
```



ĐỔ DỮ LIỆU VÀO NHÓM CHECKBOX

☒ Travelling ☐ Music ☒ Sports ☐ Reading

`<form:checkboxes path="hobbies" items="${hobbies}"/>`

```
@ModelAttribute("hobbies")
public String[] getHobbies() {
    String[] hobbies = {"Travelling", "Music",
                        "Sports", "Reading"};
    return hobbies;
}
```



ĐỔ DỮ LIỆU VÀO NHÓM RADIO

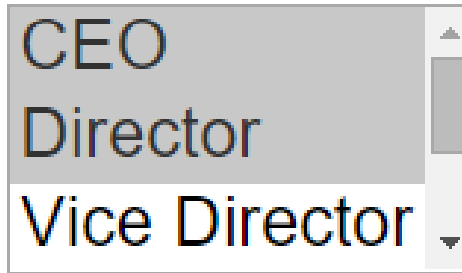
☐ Doctor ☒ Master ☐ University ☐ Other

`<form:radiobuttons path="degree" items="${degrees}"/>`

```
@ModelAttribute("degrees")
public String[] getDegrees() {
    String[] degrees = {"Doctor", "Master",
                        "University", "Other"};
    return degrees;
}
```



ĐỔ DỮ LIỆU VÀO LISTBOX



`<form:select path="positions" multiple="true" size="3"
items="${positions}"/>`

```
@ModelAttribute("positions")  
public String[] getPositions() {  
    String[] positions = {"CEO", "Director",  
        "Vice Director", "Assitance", "Employee"};  
    return positions;  
}
```



BUỘC DỮ LIỆU VỚI LIST CONTROL

```
@RequestMapping("/list-control")
```

```
public String listControl(ModelMap model) {  
    Employee emp = new Employee();  
    emp.setName("Nguyễn Văn Tèo");  
    emp.setCountry("US");  
    emp.setDegree("Master");  
    emp.setPositions(new String[]{"CEO", "Director"});  
    emp.setHobbies(new String[]{"Travelling", "Sports"});  
  
    model.addAttribute("employee", emp);  
  
    return "list-control";  
}
```

United States of America ▾

☐ Doctor ☒ Master ☐ University ☐ Other

CEO
Director
Vice Director ▾

☒ Travelling ☐ Music ☒ Sports ☐ Reading

```
public class Employee {  
    String name;  
    String country;  
    String degree;  
    String[] positions;  
    String[] hobbies;  
    getters/setters  
}
```



- Giới thiệu databinding
- Databinding đơn giản
- Form tags
- Hiển thị và buộc dữ liệu cho các List Control
 - ✱ ComboBox
 - ✱ ListBox
 - ✱ Nhóm CheckBox
 - ✱ Nhóm Radio

