



<http://www.lycato.com>



Spring Boot

Form & Validation

Nguyễn Nghiệm





- ✓ FORM & DATABINDING
- ✓ FORM VALIDATION





Form & Databinding



DATABINDING LÀ GÌ?

Contact Form

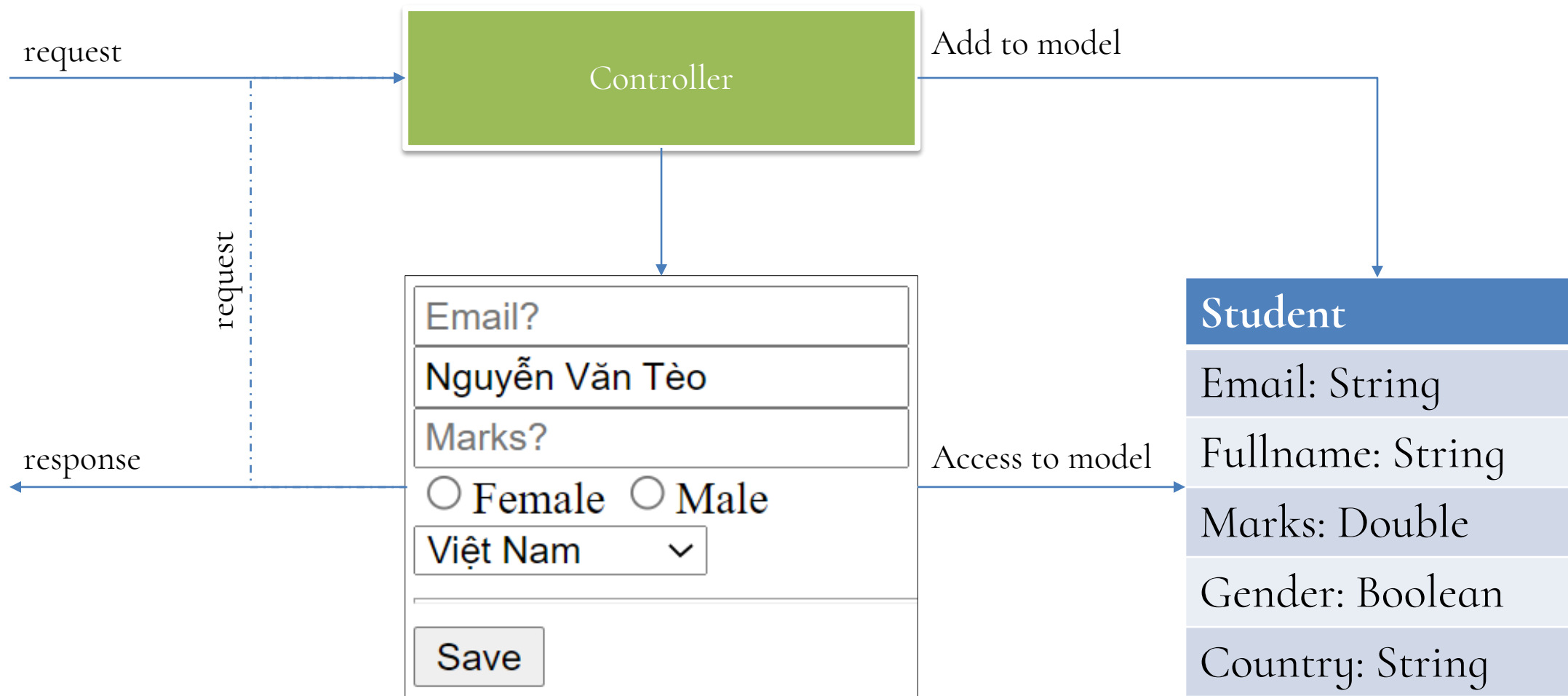
<input type="text" value="First name"/>	<input type="text" value="Last name"/>
<input type="text" value="Email address"/>	<input type="text" value="Telephone number"/>
<input type="text" value="Website url"/>	<input type="text" value="Select department"/>
no file selected View <input type="button" value="Choose File"/>	
Upload JPG, PNG, DOC, DOCX, ... (max 3MB)	
<input type="text" value="Enter message or comment"/>	
Hint: Don't be negative or off topic	
<input type="text" value="Enter captcha"/>	37PBAF
<input type="button" value="Cancel"/>	<input type="button" value="Submit Form"/>



Databinding là sự kết nối dữ liệu từ model với các điều khiển trên giao diện và ngược lại (two-way databinding) \Leftrightarrow Thay đổi dữ liệu trong model thì dữ liệu trên view sẽ thay đổi theo và ngược lại.



DATABINDING LÀ GÌ?





DATABINDING

```
@GetMapping("/form")
public String form(@ModelAttribute("student") Student std) {
    std.setFullname("Nguyễn Văn Tèo");
    return "form";
}

@PostMapping("/save")
public String form(@ModelAttribute("student") Student std) {
    ...
    return "form";
}
```

Controller

```
@Data
public class Student {
    String email;
    String fullname;
    ...
}
```

Model

```
<form th:object="${student}" action="/save" method="post">
    <input th:field="*{email}" placeholder="Email?"><br>
    <input th:field="*{fullname}" placeholder="Full Name?">
    ...
    <button>Save</button>
</form>
```

View



FORM & STUDENT

```
<form th:object="${student}" action="/save" method="post">
  <input th:field="*{email}"><br>
  <input th:field="*{fullname}"><br>
  <input th:field="*{marks}"><br>
  <select th:field="*{country}">
    <option value="VN">Việt Nam</option>
    <option value="US">United States</option>
  </select><br>
  <input th:field="*{gender}" th:value="false" type="radio">Female
  <input th:field="*{gender}" th:value="true" type="radio">Male
  <hr>
  <button>Save</button>
</form>
```

```
@Data
public class Student {
    String email;
    String fullname;
    Double marks;
    Boolean gender;
    String country;
}
```

```
<input th:field="*{email}">
```



```
<input name="email" th:value="*{email}">
```



❑ Phần tử form và thuộc tính Thymeleaf

- ❖ `<form th:action="@{url}" th:object="${bean}">`
- ❖ `<input th:field="*{property}">`
- ❖ `<select th:field="*{property}">`
- ❖ `<textarea th:field="*{property}">`
- ❖ `<any th:errors="*{property}">`

❑ Một số thẻ và thuộc tính bổ trợ

- ❖ `th:selected`
- ❖ `th:checked`
- ❖ `th:disabled`
- ❖ `th:readonly`



Form Validation



Email Address

Không để trống email

Full Name

Không để trống họ và tên

Average Marks

Không để trống điểm

Gender

☐ Male ☐ Female

Chưa chọn giới tính

Quốc tịch

Chưa chọn quốc tịch

Validate

❑ Model

- ❖ @NotBlank
- ❖ @Email
- ❖ @Min
- ❖ ...

❑ Controller

- ❖ @Validated/@Valid
- ❖ BindingResult/Errors

❑ View

- ❖ th:errors



FORM VALIDATION

```
@PostMapping("/save")
public String form(Model model,
    @Validated @ModelAttribute("student") Student student,
    BindingResult errors) {
    if(errors.hasErrors()){
        model.addAttribute("message", "Please fix the errors!")
    }
    return "form";
}
```

Controller

```
${message}
<form th:object="${student}" action="/save" method="post">
    <input th:field="*{email}" placeholder="Email?">
    <i th:errors="*{email}"></i><br>
    <input th:field="*{fullname}" placeholder="Full Name?">
    <button>Save</button>
</form>
```

View

```
@Data
public class Student {
    @Email
    @NotBlank
    String email;
    String fullname;
    ...
}
```

Model



VALIDATION ANNOTATION

@Data

```
public class Student {  
    @Email @NotBlank  
    String email;  
    @NotBlank  
    String fullname;  
    @NotNull @Min(0) @Max(10)  
    Double marks;  
    @NotNull  
    Boolean gender;  
    @NotEmpty  
    String country;  
}
```

```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-validation</artifactId>  
</dependency>
```



☐ Numbers

- ❖ *@Min, @Max, @DecimalMin, @DecimalMax*
- ❖ *@Positive, @PositiveOrZero, @Negative, @NegativeOrZero*
- ❖ *@Digits*

☐ String

- ❖ *@Email, @Pattern*
- ❖ *@NotBlank*

☐ Collection

- ❖ *@NotEmpty, @Size*

☐ Date & Time

- ❖ *@Past, @PastOrPresent, @Future, @FutureOrPresent*

☐ Others

- ❖ *@NotNull, @Null, @AssertFalse, @AssertTrue*



@Data

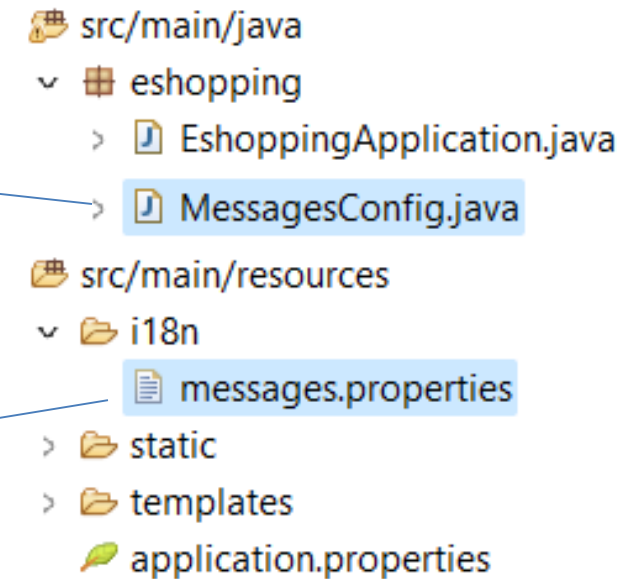
```
public class Student {  
    @Email(message = "Sai định dạng email")  
    @NotBlank(message = "Vui lòng không để trống email")  
    String email;  
    @NotBlank  
    String fullname;  
    @NotNull @Max(10)  
    @Min(value = 0, message = "Điểm phải >= 0")  
    Double marks;  
    @NotNull  
    Boolean gender;  
    @NotEmpty  
    String country;  
}
```



TÙY BIẾN THÔNG BÁO LỖI

@Configuration

```
public class MessagesConfig {  
    @Bean("messageSource")  
    public MessageSource getMessageSource() {  
        ReloadableResourceBundleMessageSource ms  
            = new ReloadableResourceBundleMessageSource();  
        ms.setBasenames("classpath:i18n/messages");  
        ms.setDefaultEncoding("utf-8");  
        return ms;  
    }  
}
```





TÙY BIẾN THÔNG BÁO LỖI

```
@Data
public class Student {
    @NotBlank(message = "Không để trống email")
    @Email(message = "Không đúng định dạng email")
    String email;
```

<form th:object="\${student}">...</form>

Key: Annotation.bean.property

NotBlank.student.email=Email is required
Email.student.email=Email is not correct format
...
typeMismatch=Data type is not matched

student_en.properties

NotBlank.student.email=Không để trống email
Email.student.email=Không đúng định dạng email
NotBlank.student.fullname=Không để trống họ và tên
...
typeMismatch=Sai kiểu dữ liệu

student_vi.properties



✓ Databinding

- ✓ th:object

- ✓ th:field

✓ Form & Validation

- ✓ Validation Annotation

- ✓ @Validated/@Valid,

- ✓ BindingResult

- ✓ errors.hasErrors()

- ✓ th:errors

