Define custom annotation ‘@CourseCode’ to validate whether course code starts with a prefix

**CourseCode.java**

/\*

Form validation

this file has rules

Custom annotations, section 17 of udemy lec

this page is used to define annotation itself

\*/

package com.luv2code.springdemo.mvc.validation;

import java.lang.annotation.ElementType;

import java.lang.annotation.Retention;

import java.lang.annotation.RetentionPolicy;

import java.lang.annotation.Target;

import javax.validation.Constraint;

import javax.validation.Payload;

*@Constraint*(validatedBy = CourseCodeConstraintValidator.class)

*@Target*({*ElementType*.***METHOD***,*ElementType*.***FIELD***})

*@Retention*(*RetentionPolicy*.***RUNTIME***)

public @interface *CourseCode* {

// define default Course code

public String value() default "LUV";

// define default error msg

public String message() default "must start with LUV";

// define default grp

public Class<?>[] groups() default{};

// define default payloads

public Class<? extends Payload>[] payload() default {} ;

}

**CourseCodeConstraintValidator.java**

/\*

Form validation

this file has rules

Custom annotations, section 17 of udemy lec

\*/

package com.luv2code.springdemo.mvc.validation;

import javax.validation.ConstraintValidator;

import javax.validation.ConstraintValidatorContext;

public class CourseCodeConstraintValidator

implements ConstraintValidator<CourseCode, String>{

private String coursePrefix;

*@Override*

public void initialize(*CourseCode* theCourseCode) {

coursePrefix = theCourseCode.value();

}

*@Override*

public boolean isValid(String theCode, ConstraintValidatorContext arg1) {

boolean result=true;

if(theCode!=null)

result = theCode.startsWith(coursePrefix);

return result;

}

}

**Obj class(**customer.java**)**

/\*

Form validation

this file has rules

Custom annotations, section 17 of udemy lec

\*/

package com.luv2code.springdemo.mvc;

import com.luv2code.springdemo.mvc.validation.CourseCode;

public class Customer{

private String firstName;

private String lastName;

// @CourseCode(value="LAB",message="must start with LAB")

***@CourseCode*** //use default value and msg specified in CourseCode

**private String courseCode;**

**public String getCourseCode() {**

**return courseCode;**

**}**

**public void setCourseCode(String courseCode) {**

**this.courseCode = courseCode;**

**}**

public String getFirstName() {

return firstName;

}

public void setFirstName(String firstName) {

this.firstName = firstName;

}

public String getLastName() {

return lastName;

}

public void setLastName(String lastName) {

this.lastName = lastName;

}

}

**customer-form.jsp**

<!-- Form validation -->

<!-- only form:error is added in basic form -->

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

<!DOCTYPE html>

<html>

<head>

<title>Insert title here</title>

<style>

.error{color:*red*}

</style>

</head>

<body>

<form:form action=*"processForm"* modelAttribute=*"customer"*>

First name: <form:input path=*"firstName"* /> <br>

Last name(\*) : <form:input path=*"lastName"* /><br>

**Course code(\*) : <form:input path=*"courseCode"* />**

**<form:errors path=*"courseCode"* cssClass=*"error"* />** <br>

<br>

<input type=*"submit"* value=*"submit"*>

</form:form>

</body>

</html>

**customer-confirmation.jsp**

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

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

The customer is ${customer.firstName} ${customer.lastName} <br>

The course code is ${customer.courseCode}<br>

</body>

</html>