

12장. 파일 업로드, 다국어 처리



Security - login



파일 업로드

글쓰기

제목	<input type="text" value="title"/>
작성자	<input type="text" value="cloud"/>
내용	<div><div>content</div><div></div></div>
업로드	<div><div>파일 선택</div><div>선택된 파일 없음</div></div>
<div><div>등록</div><div>목록</div></div>	



◆ MultipartFile 객체 필드 추가

```
@Setter
@Getter
public class BoardVO implements Serializable {

    private static final long serialVersionUID = 30L;

    private int bno;
    private String title;
    private String writer;
    private String content;
    @DateTimeFormat(pattern="yyyy-MM-dd")
    private Date regDate;
    private int cnt;

    private MultipartFile uploadFile; //파일 업로드
}
```



◆ FileUpload 라이브러리 추가

```
<dependency>
  <groupId>commons-fileupload</groupId>
  <artifactId>commons-fileupload</artifactId>
  <version>1.3.3</version>
</dependency>
```

```
> commons-io-2.2.jar - C:\Users\Wkiyon\m2\repository\commons-io\commons-
> hamcrest-core-1.3.jar - C:\Users\Wkiyon\m2\repository\org\hamcrest\hamcre
```



파일 업로드

◆ MultipartResolver 빈 등록

```
<!-- 파일 업로드 : 최대 파일 크기: 1MB -->  
<bean id="multipartResolver"  
      class="org.springframework.web.multipart.commons.CommonsMultipartResolver">  
    <property name="maxUploadSize" value="1000000" />  
</bean>
```



파일 업로드

◆ 파일 업로드 처리

```
@PostMapping("/insertBoard")
@PreAuthorize("isAuthenticated()")
public String insertBoard(BoardVO vo) throws IllegalStateException, IOException{
    //파일 업로드 처리
    MultipartFile uploadFile = vo.getUploadFile();
    if(!uploadFile.isEmpty()) {
        String fileName = uploadFile.getOriginalFilename();
        String filePath = "C:/springwork/SpringBoard2/src/main/webapp/upload/";
        uploadFile.transferTo(new File(filePath + fileName));
    }
    service.insert(vo);
    return "redirect:/board/boardList";
}
```

```
src
├── main
│   ├── webapp
│   │   ├── resources
│   │   └── upload
│   │       └── dish1.jpg
│   └── WEB-INF
│       ├── classes
│       └── spring
│           ├── appServlet
│           ├── message-context.xml
│           ├── root-context.xml
│           └── security-context.xml
```



파일 업로드

◆ 글쓰기 페이지

- 업로드 항목 추가
- form 속성 추가 : `enctype="multipart/form-data"`

```
<form action="/board/insertBoard" method="post" enctype="multipart/form-data"
      onsubmit="return checkForm()" name="newWrite">
  <table class="tbl_reg">
    <tr>
      <td width="100">제목</td>
      <td align="left"><input type="text" name="title" placeholder="title">
    </tr>
```

```
    <tr>
      <td>업로드</td>
      <td><input type="file" name="uploadFile"></td>
    </tr>
    <tr>
      <td colspan="2" align="center">
        <input type="submit" value="등록">
      </td>
    </tr>
```



다국어 처리

◆ 다국어 처리

다국어 지원이란 '국제화'라고도 하며 하나의 JSP 페이지를 다양한 언어로 서비스하는 것을 의미한다.

LOGIN

Korean | English

ID	<input type="text"/>
PASSWORD	<input type="password"/>
<input type="button" value="LOG-IN"/>	



다국어 처리

◆ 메시지 파일 작성하기

메시지 파일의 확장자는 .properties이며, 파일명은 언어에 해당하는 Locale정보를 결합하여 작성한다.

messageSource_en.properties

```
#customLogin.jsp  
  
message.user.login.title=LOGIN  
message.user.login.id=ID  
message.user.login.password=PASSWORD  
message.user.login.loginBtn=LOG-IN  
  
message.user.login.language.en=English  
message.user.login.language.ko=Korean
```

```
▼ src/main/resources  
  > com  
  ▼ message  
    messageSource_en.properties  
    messageSource_ko.properties  
    messageSource.txt  
    META-INF  
    log4j.xml
```



다국어 처리

◆ 메시지 파일 작성하기

messageSource.txt

```
message.user.login.title=로그인  
message.user.login.id=아이디  
message.user.login.password=비밀번호  
message.user.login.loginBtn=로그인
```

```
message.user.login.language.en=영어  
message.user.login.language.ko=한글
```

messageSource_ko.properties

```
message.user.login.title=\uB85C\uADF8\uC778  
message.user.login.id=\uC544\uC774\uB514  
message.user.login.password=\uBE44\uBC00\uBC88\uD638  
message.user.login.loginBtn=\uB85C\uADF8\uC778
```

```
message.user.login.language.en=\uC601\uC5B4  
message.user.login.language.ko=\uD55C\uAE00
```



다국어 처리

◆ MessageSource 등록

message-context.xml

```
<!-- 다국어 관련 설정 -->
<!-- MessageSource 등록 -->
<bean id="messageSource"
      class="org.springframework.context.support.ResourceBundleMessageSource">
    <property name="basenames">
        <list>
            <value>message.messageSource</value>
        </list>
    </property>
</bean>

<!-- LocaleResolver 등록 -->
<bean id="LocaleResolver"
      class="org.springframework.web.servlet.i18n.SessionLocaleResolver"/>

<!-- LocaleChangeInteerceptor 등록 -->
<mvc:interceptors>
    <bean class="org.springframework.web.servlet.i18n.LocaleChangeInterceptor">
        <property name="paramName" value="lang" />
    </bean>
</mvc:interceptors>
```


















다국어 처리

◆ MessageSource 등록

Configure Namespaces

Select XSD namespaces to use in the configuration file

<input type="checkbox"/>		aop - http://www.springframework.org/schema/aop
<input checked="" type="checkbox"/>		beans - http://www.springframework.org/schema/beans
<input type="checkbox"/>		c - http://www.springframework.org/schema/c
<input type="checkbox"/>		cache - http://www.springframework.org/schema/cache
<input type="checkbox"/>		context - http://www.springframework.org/schema/context
<input type="checkbox"/>		jdbc - http://www.springframework.org/schema/jdbc
<input type="checkbox"/>		jee - http://www.springframework.org/schema/jee
<input type="checkbox"/>		lang - http://www.springframework.org/schema/lang
<input checked="" type="checkbox"/>		mvc - http://www.springframework.org/schema/mvc
<input type="checkbox"/>		mybatis-spring - http://mybatis.org/schema/mybatis-spring
<input type="checkbox"/>		p - http://www.springframework.org/schema/p
<input type="checkbox"/>		security - http://www.springframework.org/schema/security
<input type="checkbox"/>		task - http://www.springframework.org/schema/task
<input type="checkbox"/>		tx - http://www.springframework.org/schema/tx
<input type="checkbox"/>		util - http://www.springframework.org/schema/util



다국어 처리

◆ web.xml에 message-context.xml 등록

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



다국어 처리

◆ 로그인 화면 처리

태그 라이브러리 지시자 선언

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

```
<spring:message code= " 메시지 키값" />
```

```
<%@ taglib uri="http://www.springframework.org/tags" prefix="spring" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title><spring:message code="message.user.login.title" /></title>
<link rel="stylesheet" href="/resources/css/style.css">
</head>
<body>
    <jsp:include page="./menu.jsp" />
    <div id="container">
        <section id="login">
            <div class="title">
                <h2><spring:message code="message.user.login.title" /></h2>
            </div>
```

customLogin.jsp



다국어 처리

```
<p>
  <a href="customLogin?lang=en">
    <spring:message code="message.user.Login.Language.en" />
  </a>
  <a href="customLogin?lang=ko">
    <spring:message code="message.user.Login.Language.ko" />
  </a>
</p>
<h3 class="error"><c:out value="${error}" /></h3>
<form action="/login" method="post">
  <table class="tbl_login">
    <tr>
      <td><spring:message code="message.user.Login.id" /></td>
      <td><input type="text" name="username"></td>
    </tr>
    <tr>
      <td><spring:message code="message.user.Login.password" /></td>
      <td><input type="password" name="password"></td>
    </tr>
    <tr>
      <td colspan="2">
        <input type="submit" value="<spring:message code="message.user.Login.LoginBtn" />
      </td>
    </tr>
  </table>
  <input type="hidden" name="${_csrf.parameterName}" value="${_csrf.token}" />
</form>
```

customLogin.jsp

