

## Lecture Note 7 Database Programming I

By A/Prof Heng Tao SHEN  
School of Information Technology and Electrical Engineering  
The University Of Queensland

P. 2

### Outline

- Major database management systems:
  - MySQL (Next lecture)
  - Oracle (Next lecture)
- Important operations:
  - Uploading Files (this lecture)

### Introduction

- When we are talking about “database”, what will you think of immediately?
  - Oracle? MySQL? DB2? Sybase? ...
- Strictly speaking, each of the above is a DBMS – Database Management System.
  - A database is simply a place where you store a lot of data.
- In Web Information Systems, sometimes, we may need to create our own “database”
  - In fact, file/directory structure

P. 3

### How to Upload Files?

- Uploading files is very simple, but very tricky!
- No matter how expert you are, you will always encountered many problems...
- We will discuss two methods for uploading files:
  - Using PHP
  - Using JSP

P. 4

### Upload File – Preparation

- In order to upload a file, you must prepare a form first:
 

```
<form method="POST" enctype="multipart/form-data" action="...">
  <p>File: <input type="file" name="userFile" size="20">
  <input type="submit" value="Submit" name="uploadBtn"></p>
</form>
```



P. 5

### Upload File – PHP (1/6)

- In the previous form, set the action to the correct location.
  - E.g. fileHandler.php
- Mechanism of PHP uploading file:
  - When you submit a form, the file will be uploaded to a tmp folder at the server side.
    - This tmp directory is defined by the server admin.
    - Can changed in the configuration file: php.ini
  - You need to copy the uploaded file from the tmp folder to your destination folder.
- Simple? Yes! But in reality, you will always encounter many troubles...

P. 6

## Upload File – PHP (2/6)

- The code for fileHandler.php:

```
<?php
$dir = 'you_target_directory';
$uploadfile = $dir . "/" .
    basename($_FILES['userfile']['name']);
if(move_uploaded_file($_FILES['userfile']['tmp_name'], $upload
file) ){
    echo "File is valid, and was successfully uploaded.";
}
else {
    echo "Cannot upload the file.";
    echo 'Here is some more debugging info: ' .
    print_r($_FILES);
}
?>
```

- Note :**

- If no file is selected for upload, then `$_FILES['userfile']['size']` is 0, and `$_FILES['userfile']['tmp_name']` is null.
- The file will be deleted from the temporary directory at the end of the request if it has not been moved away or renamed.

P.7

## Upload File – PHP (3/6)

- All parameters:

- `$_FILES['userfile']['name']`
  - The original name of the file on the client machine.
- `$_FILES['userfile']['type']`
  - The mime type of the file, if the browser provided this information. An example would be "image/gif". This mime type is however not checked on the PHP side and therefore don't take its value for granted.
- `$_FILES['userfile']['size']`
  - The size, in bytes, of the uploaded file.
- `$_FILES['userfile']['tmp_name']`
  - The temporary filename of the file in which the uploaded file was stored on the server.
- `$_FILES['userfile']['error']`
  - The [error code](#) associated with this file upload. This element was added in PHP 4.2.0

P.8

## Upload File – PHP (4/6)

- Possible reason why you cannot upload a file:

- The destination directory does not exist
  - Make sure you create the directory before uploading the file!
- The directory permission is not correct (most difficult to solve!)
  - You need to set the permission to "write".
    - If the host OS is Unix/Linux, then you can use the command `chmod`.
      - `chmod(0777)`
    - Remember to change back to 0755 or even 0700. Yet, not all hosting agent provide this command. Another way is to change via FTP software.
  - If the host OS is Windows, then...
    - Much more difficult, because Windows does not allow the command `chmod`. Solve it case by case.
    - One solution (usually works, but not guarantee):
      - <http://www.howtocode.info/index.php/how-to-fix-windows-server-upload-file-inherit-permissions-error/>

P.9

## Upload File – PHP (5/6)

- The max file size exists the limit
  - This can be set in the configuration file "php.ini"
    - Locate in the php installation directory.
    - After you change it, restart the php server.
  - Always check the max size (`upload_max_filesize`) in the `php.ini()`
  - Or use to control the size before upload
    - `<!--MAX_FILE_SIZE` in bytes must precede the file input field -->
    - `<input type="hidden" name="MAX_FILE_SIZE" value="30000" />`
- The syntax of your target directory ('you\_target\_directory') is not correct.
  - It can be [www.YourHost.com/YourTarget/](http://www.YourHost.com/YourTarget/)
  - It can also be [someLocation/anotherLocation/](http://someLocation/anotherLocation/)
    - But for the second case, it is more tricky. You sometimes may need to figure out it by trial-and-error...

P.10

## Upload File – PHP (6/6)

- Major reference:
  - <http://au.php.net/manual/en/features.file-upload.post-method.php>
  - How to upload multiple files?
- Change file execution mode (chmod):
  - PHP reference:
    - <http://au.php.net/manual/en/function.chmod.php>
  - Complete tutorial of chmod:
    - <http://catcode.com/teachmod/>

P.11

## Upload File – JSP (1/12)

- Upload File using JSP is...
  - Very complex.
  - Very difficult.
  - Not intuitive.
  - Need to understand Java Bean and very good in Java Programming!
- But... Wait...
  - If it is really that complex... then there must be some packages that already implemented something to help us accomplish the task more easily, right?
    - Luckily... YES!

P.12

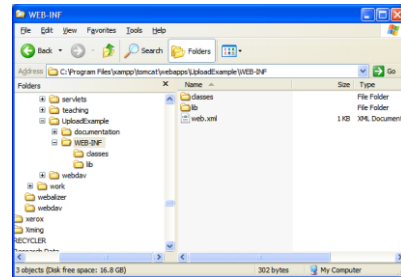
## Upload File – JSP (2/12)

- Package:
  - We do it with the help of a package call UploadBean from JavaZoom:
    - <http://www.javazoom.net/zservlets/uploadbean/uploadbean.html> download it!
- Installation:
  - Assume you have a directory called "UploadExample" under your Tomcat "webapps" directory, then:
    - Unzip the content of the downloaded file into this directory.
    - Hence, you should got something similar to the next page...

P. 13

## Upload File – JSP (3/12)

- Note that your WEB-INF should contains "web.xml" and a folder "lib". Inside "lib" should contains 4 files.



P. 14

## Upload File – JSP (4/12)

- Open "SimpleUpload.jsp". Modify the uploaded directory:

```
<html>
<%% page language="java" import="javazoom.upload.*,java.util.*" %>
<%% page errorPage="ExceptionHandler.jsp" %>

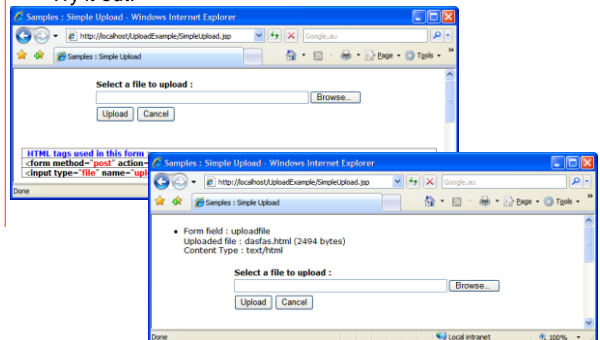
<jsp:useBean id="upBean" scope="page" class="javazoom.upload.UploadBean" >
  <jsp:setProperty name="upBean" property="folderstore" value="c:/xxxx" />
</jsp:useBean>

<head>
...
...
  - You need to modify the bold text in the above code only.
```

P. 15

## Upload File – JSP (5/12)

- Try it out!



## Upload File – JSP (6/12)

- We now explain how we do the file upload.
- Write the required headings/parameters/imports:
 

```
<%%page contentType="text/html;"%>

<%%page import="java.util.*, javazoom.upload.*" %>

<jsp:useBean id="upBean" scope="page" class="javazoom.upload.UploadBean" >
  <jsp:setProperty name="upBean" property="folderstore" value="c://XXXX" />
</jsp:useBean>
:
:
```
- Note:
  - You can put the uploaded file anywhere in the computer.
  - Remember to create a folder before you can upload anything.

P. 17

## Upload File – JSP (7/12)

- The actual upload (simplified from the original) process:

```
<%
if (multipartFormRequest.isMultipartFormRequest()) {
    MultipartFormRequest data = newMultipartFormRequest(request);

    //get files uploaded
    Hashtable files = data.GetFiles();

    if (files != null) {
        if (files.isEmpty()) {
            UploadFile file = (UploadFile)files.get("userFile");
            if (file != null) {
                // 1. do sth here
                upBean.store(data, "userFile");
            }
        }
    }
    else {
        // 2. no file upload
    }
}
%>
```

- The parameter "userFile" in the above code is the name of the upload element in the form (read P.5 again).

P. 18

## Upload File – JSP (8/12)

- In the code segment 1 in the previous page, you can write:

```
file.getFileName();
file.getFileSize();
file.getFileType();
```

- The meaning of them are self-explanatory.

- Moreover, you can simplify the code in segment 1 by simply writing:

```
upBean.store(data);
```

P. 19

## Upload File – JSP (9/12)

- How to handle multiple files upload?

```
– The code:
upBean.store(data);
already saved all files!!!
```

- A simple way to get all file data is to iterate via it (java 1.6 or later):

```
Hashtable files = data.GetFiles();
for(Collection<UploadFile> file : files.values()){
    // Do sth here. E.g.
    out.print("File name is: " + file.getFileName());
    out.print("<br>");
}
```

P. 20

## Upload File – JSP (10/12)

- You may need to store session information for the uploaded data. Then you need to modify the Java Bean declaration:

```
<jsp:useBean id="upBean" scope="session" class="javazoom.upload.UploadBean" >
<jsp:setProperty name="upBean" property="folderstore" value="c:/XXXX" />
</jsp:useBean>
```

- You can also get the history of the bean by:

```
Vector history = upBean.getHistory();
for(int i=0; i<history.size(); i++){
    UploadParameters up = (UploadParameters) history.elementAt(i);
    out.println("Uploaded file : " + up.getFilename() + " " +
        "Size: " + up.getFilesize() + " bytes " + " " +
        "Content Type: " + up.getContentType() + " " +
        "StoreModel: " + up.getStoremodelName() + " " +
        up.getStoreinfo());
}
```

P. 21

## Upload File – JSP (11/12)

- There are many other options as well (e.g. save into a zip file):

```
<jsp:useBean id="upBean" scope="session" ... >
<jsp:setProperty name="upBean" property="parser"
value="<%= MultipartFormDataRequest.CFUPARSER %>" />
<jsp:setProperty name="upBean" property="parsertmpdir"
value="c:/xxxxx" />
<jsp:setProperty name="upBean" property="maxfiles"
value="10" />
<jsp:setProperty name="upBean" property="overwrite"
value="true" />
<jsp:setProperty name="upBean" property="filesizeLimit"
value="1000000" />
<jsp:setProperty name="upBean" property="blacklist"
value="*.zip,*.rar,*.exe" />
<jsp:setProperty name="upBean" property="zipfilestore"
value="c:/xxx/uploads.zip" />
</jsp:useBean>
```

P. 22

## Upload File – JSP (12/12)

- One guide

– <http://www.javazoom.net/jzservlets/uploadbean/documentation/developerguide.html>

- Try to read all other examples to understand all of the options!

- Note:

– Always remember to copy all .jar files from the directory "WEB-INF/lib" to the new location when you write your own JSP.  
– You can read the user input in the form by simply using this statement if you have included Javazoom:

```
MultipartFormDataRequest data = new
MultipartFormDataRequest(request);
String input = data.getParameter("name_of_the_element");
```

P. 23

## Appendix

- Sending email using PHP:

– <http://au.php.net/manual/en/function.mail.php>

- Sending email using JSP:

– <http://www.stardeveloper.com/articles/display.html?article=2001101101&page=1>

- What do you feel about these two issues?

P. 24

## Summary

- In this lecture, we demonstrated:
  - Introduce the idea of database programming.
  - How to upload files using PHP and JSP.
  - Understand more about the strengths and weaknesses of two different server-side scripts.
- Next week
  - Database Programming II