

Forms Processing with Servlets

Vittayasak Rujivorakul

Lecture 2

Forms Processing with Servlets

Vittayasak Rujivorakul

Lecture 2

HTML Form

Enter your Login and Password	
Login:	
Passwd:	
Submit Reset	

Enter your Login and Password	
Login:	Vittayasak
Passwd:	password
Submit Reset	

HTML Form Code :

```
<html>
<head>
<title>Form Test</title>
<meta http-equiv="Content-Type"
      content="text/html; charset=TIS-620">
</head>

<body bgcolor="#FFFFFF" text="#0000FF">
<center>
<br><br>
<form name="FormLogin" method="GET"
      action="/servlet/LoginServlet">
<table width="300" border="0" cellpadding="1"
      cellspacing="1">
<tr bgcolor="#000099">
<td colspan="2">
<div align="center"><font color="#FFFFFF">
<b>Enter your Login and Password</b></font>
</div></td>
</tr>
<tr bgcolor="#FFFFCC">
<td width="126"> Login:</td>
<td width="251">
<input type="text" name="login">
</td>
</tr>
<tr bgcolor="#FFFFCC">
<td width="126"> Passwd:</td>
<td width="251">
<input type="password" name="passwd">
</td>
</tr>
<tr bgcolor="#FFFFCC">
<td width="126">&nbsp;&nbsp;&nbsp;</td>
<td width="251">
<input type="submit" name="Submit"
value="Submit">
<input type="reset" name="Submit2"
value="Reset">
</td>
</tr>
</table>
</form>
</center>
</body>
</html>
```

The (FORM) tag

- <FORM> tag parameter ประกอบด้วย
 - NAME : ใช้ระบุชื่อ form
 - METHOD : สำหรับระบุวิธีการในการส่งข้อมูล เช่น GET, POST
 - ACTION : สำหรับระบุ Servlet ปลายทางที่จะรับ form ไป process
 - ENCTYPE : ระบุวิธีการในการ Encryption Type
- ภายใน FORM จะประกอบไปด้วย Input box , Select box, Radio button, Checkbox ..etc

METHOD

- GET Data มองเห็นที่ URL ได้ไม่เกิน 256 char
- POST ไม่ show Data ใช้ Data ได้มากกว่า 65535 char และส่ง object ได้ด้วย
- HEAD ใช้กับ Header บอกเกี่ยวกับการ Access
- PUT การส่งข้อมูลไปให้ server
- DELETE การลบข้อมูลบน server
- TRACE show request
- OPTION ใช้ในการติดต่อระหว่าง Browser กับ web server

Action and ENCTYPE

- Action ใช้ในการกำหนด ปลายทางของ form ว่าเมื่อมีการ submit แล้วจะไปไหนต่อ
- ENCTYPE ใช้ในการเข้ารหัสข้อมูล หรืออาจใช้ในการส่งข้อมูลรูปแบบอื่นที่ไม่ได้ใช้ text data เช่น file binary (ใช้ได้กับ method Post เท่านั้น)

Form Process Script

```
#!/usr/local/bin/perl
# LoginPerl.pl

use CGI;
$obj = new CGI;

print "Content-type: text/html\n\n";
$login = $obj->param(login);
$password = $obj->param(password);
$hostip = $ENV{'REMOTE_ADDR'};
print "<html><head><title>Login Perl</title></head>\n";
print "<body><center>\n";
print "Your Login = $login<br>\n";
print "Your Password = $password<br>\n";
print "You are Login from [$hostip]\n";

print "</center></body></html>";
```

การทำงานของ Form Process Script (ต่อ)

- สิ่งที่ browser ส่งมาจะมองเห็นดังนี้
`http://localhost:8009/cgi-bin/LoginPerl.pl?login=Vittayasak&password=&Submit=Submit`
- การรับ parameter : ใช้ `$obj->param(parametername)` ในการรับ ค่าจาก form ที่ส่งมา และ `$ENV{'system parameter'}` ในการรับ ค่าที่ system ส่งมา กับ request
- การ set content type : ทำได้โดยการสั่ง print “Content-type: text/html\n\n” เป็นการบอกกับ browser ว่าให้แสดงผลเป็น text data ใน format HTML
- การแสดงผล : ทำได้โดยใช้คำสั่ง print “response message”

Forms - Processing Servlet

```
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class LoginServlet extends HttpServlet{
    public void doGet (HttpServletRequest req,
        HttpServletResponse res) throws
        ServletException , IOException{
        Enumeration keys;
        String key,login,passwd;
        String value;
        String title;
        res.setContentType("text/html");
        PrintWriter out = res.getWriter();
        out.println("<head><title>Login
        Servlet</title></head>");

        login = req.getParameter("login");
        passwd = req.getParameter("passwd");
        out.println("Login: " + login + " , " +
        "Passwd: " + passwd + "<br>");

        keys = req.getParameterNames( );
        out.println("All Parameter = ");
        while (keys.hasMoreElements( )) {
            key = (String) keys.nextElement( );
            value = req.getParameter(key);
            out.print(" " + key + " : " + value + " ,
            ");
        }

        public String getServletInfo( ){
            return "A Login Servlet";
        }
    }
}
```

Http Servlet Service Method

- doGet ใช้รับ Method GET
- doPost ใช้รับ Method POST
- doPut ใช้รับ Method PUT
- doTrace ใช้รับ Method Trace
- doOptions ใช้ระบุ option ที่รองรับระหว่าง server กับ browser
- service

โดยจะต้องมี HTTP Servlet Request และ HTTP Servlet-Response อยู่ด้วยเสมอ

Parsing the Request

- สำหรับ all request ใช้
 - getParameterName จะ return เป็น enumerate
 - getParameterValues return เป็น สำหรับ parameter ที่มีค่า ได้มากกว่า 1 ค่า จึงจะ return Array of value มา
 - getParameter จะ return ค่าของ name parameter
- สำหรับ HTTP GET ใช้
 - getQueryString รับ Data หลังเครื่องหมาย ? ทั้งหมด
 - parseQueryString

Parsing the Request (ต่อ)

- สำหรับ HTTP POST, PUT และ DELETE request ใช้
 - getReader จะ return BufferedReader ใช้ในการอ่าน raw data เมื่อคาดว่าจะรับ text data
 - getInputStream จะ return ServletInputStream ใช้ในการอ่าน raw data เมื่อคาดว่าจะรับ binary data

Example Passing the Request

```
int bufsize = 1024*8;

ServletOutputStream out = res.getOutputStream();

ServletInputStream in = req.getInputStream();

int contentLength = req.getContentLength();
byte[] content = new byte[ contentLength ];
int result;
int totalRead = 0;

try{
    result = in.readLine(content, 0, content.length);
    while(result !=1) {
        totalRead += result;
        result = in.readLine(content, totalRead, bufsize);
    }
} catch(IOException ioe) {
    getServletContext().log(ioe, "Error in EchoServlet");
}
out.write(content, 0, totalRead);
out.flush();
```

Returning a Response

HttpServletResponse Object จะมีวิธีการ return data ไปยัง user 2 ทางด้วยกัน
คือ

- getPrintWriter method จะ return Writer ใช้ในการส่ง text data
- getOutputStream method จะ return ServletOutputStream ใช้ในการส่งข้อมูลที่เป็น binary

HTTP Header Data

- ในการทำงาน เราจะต้อง set HTTP header data ก่อนที่จะทำงานกับ Writer หรือ Output Stream โดยใช้ method มีชื่อว่า set Content Type

```
response.setContentType ("text / html");
```

Form Process LAB

1. สร้าง HTML Form สำหรับ กรอก ชื่อ , สกุล , Email และมีชื่อแต่ละ element ดังนี้

Name :	name
Surname :	surname
Email :	email
	Submit Clear

2. เขียน Servlets เพื่อรับ parameter จาก Form และแสดงผล ชื่อ, สกุล และ Email โดยรองรับทั้ง POST และ GET method พร้อมติดตั้งและทดสอบ