DEZVOLTAREA APLICATIILOR WEB

LAB 5

Lect. Univ. Dr. Mihai Stancu

DEPARTAMENTUL DE INFORMATICĂ

Adding Cookies with JSP

```
<%
     // Create cookies for first and last names.
     Cookie firstName = new Cookie("first_name",
     request.getParameter("first_name"));
     Cookie lastName = new Cookie("last_name", request.getParameter("last_name"));
     // Set expiry date after 24 Hrs for both the cookies.
     firstName.setMaxAge(60*60*24);
     lastName.setMaxAge(60*60*24);
     // Add both the cookies in the response header.
     response.addCookie( firstName );
     response.addCookie( lastName );
%>
<html>
<head>
<title>Setting Cookies</title>
</head>
<body>
     <center>
          <h1>Setting Cookies</h1>
     </center>
     <ul>
          <li><p><b>First Name:</b> <%= request.getParameter("first_name")%>
          </p></li>
          <li><p><b>Last Name:</b> <%= request.getParameter("last_name")%>
          </p></li>
     </ul>
</body>
</html>
```

Reading Cookies with JSP

```
<html>
<head>
     <title>Reading Cookies</title>
</head>
<body>
     <center>
     <h1>Reading Cookies</h1>
     </center>
<%
     Cookie cookie = null;
     Cookie[] cookies = null;
     // Get an array of Cookies associated with this domain
     cookies = request.getCookies();
     if( cookies != null ) {
          out.println("<h2> Found Cookies Name and Value</h2>");
          for (int i = 0; i < cookies.length; i++) {
               cookie = cookies[i];
               out.print("Name : " + cookie.getName() + ", ");
               out.print("Value: " + cookie.getValue()+" <br/>");
          }
     } else {
          out.println("<h2>No cookies founds</h2>");
     }
%>
</body>
</html>
```

Delete Cookies with JSP

```
<html>
<head><title>Reading Cookies</title></head>
<body>
<center>
<h1>Reading Cookies</h1>
</center>
<%
   Cookie cookie = null;
   Cookie[] cookies = null;
   // Get an array of Cookies associated with this domain
   cookies = request.getCookies();
   if( cookies != null ){
      out.println("<h2> Found Cookies Name and Value</h2>");
      for (int i = 0; i < cookies.length; i++){
         cookie = cookies[i];
         if((cookie.getName( )).compareTo("first_name") == 0 ){
            cookie.setMaxAge(0);
            response.addCookie(cookie);
            out.print("Deleted cookie: " +
            cookie.getName( ) + "<br/>");
         }
         out.print("Name : " + cookie.getName( ) + ",  ");
         out.print("Value: " + cookie.getValue( )+" <br/>");
      }
  }else{
      out.println("<h2>No cookies founds</h2>");
  }
%>
</body>
</html>
```

JSP – Session Tracking

```
<%@ page import="java.io.*,java.util.*" %>
<%
   // Get session creation time.
   Date createTime = new Date(session.getCreationTime());
   // Get last access time of this web page.
   Date lastAccessTime = new Date(session.getLastAccessedTime());

   String title = "Welcome Back to my website";
   Integer visitCount = new Integer(0);
   String visitCountKey = new String("visitCount");
   String userIDKey = new String("userID");
   String userID = new String("ABCD");

   // Check if this is new comer on your web page.
   if (session.isNew()){
      title = "Welcome to my website";
      session.setAttribute(userIDKey, userID);
      session.setAttribute(visitCountKey, visitCount);
   }
   visitCount = (Integer)session.getAttribute(visitCountKey);
   visitCount = visitCount + 1;
   userID = (String)session.getAttribute(userIDKey);
   session.setAttribute(visitCountKey, visitCount);
%>
<html>
<head><title>Session Tracking</title></head>
<body>
      <center><h1>Session Tracking</h1></center>
      <table border="1" align="center">
      <tr bgcolor="#949494"><th>Session info</th><th>Value</th></tr>
      <tr><td>id</td><td><% out.print( session.getId()); %></td></tr>
      <tr><td>Creation Time</td><td><% out.print(createTime); %></td></tr>
      <tr><td>Time of Last Access</td><td><% out.print(lastAccessTime); %></td></tr>
      <tr><td>User ID</td><td><% out.print(userID); %></td></tr>
      <tr><td>Number of visits</td><td><% out.print(visitCount); %></td></tr>
      </table>
</body>
</html>
```

File Upload Form

```
<html>
<head>
<title>File Uploading Form</title>
</head>
<body>
<h3>File Upload:</h3>
Select a file to upload: <br />
<form action="UploadServlet" method="post"
                        enctype="multipart/form-data">
<input type="file" name="file" size="50" />
<br />
<input type="submit" value="Upload File" />
</form>
</body>
</html>
```

JSP – Page Redirecting

```
<%@ page import="java.io.*,java.util.*" %>
<html>
<head>
<title>Page Redirection</title>
</head>
<body>
<center>
<h1>Page Redirection</h1>
</center>
<%
   // New location to be redirected
   String site = new String("http://www.photofuntoos.com");
   response.setStatus(response.SC_MOVED_TEMPORARILY);
   response.setHeader("Location", site);
%>
</body>
</html>
```

JSP – Hits Counter

```
<%@ page import="java.io.*,java.util.*" %>

<html>
<head>
<title>Applcation object in JSP</title>
</head>
<body>
<%
    Integer hitsCount = (Integer)application.getAttribute("hitCounter");
    if( hitsCount == null || hitsCount == 0 ){
       /* First visit */
       out.println("Welcome to my website!");
       hitsCount = 1;
    } else {
       /* return visit */
       out.println("Welcome back to my website!");
       hitsCount += 1;
    }
    application.setAttribute("hitCounter", hitsCount);
%>
<center>
<p>Total number of visits: <%= hitsCount%></p>
</center>
</body>
</html>
```

JSP – Auto Refresh

```
<%@ page import="java.io.*,java.util.*" %>
<html>
<head>
<title>Auto Refresh Header Example</title>
</head>
<body>
<center>
<h2>Auto Refresh Header Example</h2>
<%
   // Set refresh, autoload time as 5 seconds
   response.setIntHeader("Refresh", 5);
   // Get current time
   Calendar calendar = new GregorianCalendar();
   String am_pm;
   int hour = calendar.get(Calendar.HOUR);
   int minute = calendar.get(Calendar.MINUTE);
   int second = calendar.get(Calendar.SECOND);
   if(calendar.get(Calendar.AM_PM) == 0)
      am_pm = "AM";
   else
      am_pm = "PM";
   String CT = hour+":"+ minute +":"+ second +" "+ am_pm;
   out.println("Crrent Time: " + CT + "\n");
%>
</center>
</body>
</html>
```

JSP – Standard Tag Library (JSTL)

- Core tags, Formatting tags, SQL tags, XML tags, JSTL Functions
- `<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>`

Tag	Description
<c:out >	Like <%= ... >, but for expressions.
<c:set >	Sets the result of an expression evaluation in a 'scope'
<c:remove >	Removes a scoped variable (from a particular scope, if specified).
<c:catch>	Catches any Throwable that occurs in its body and optionally exposes it.
<c:if>	Simple conditional tag which evalutes its body if the supplied condition is true.
<c:choose>	Simple conditional tag that establishes a context for mutually exclusive conditional operations, marked by <when> and <otherwise>
<c:when>	Subtag of <choose> that includes its body if its condition evalutes to 'true'.
<c:otherwise >	Subtag of <choose> that follows <when> tags and runs only if all of the prior conditions evaluated to 'false'.
<c:import>	Retrieves an absolute or relative URL and exposes its contents to either the page, a String in 'var', or a Reader in 'varReader'.
<c:forEach >	The basic iteration tag, accepting many different collection types and supporting subsetting and other functionality .
<c:forTokens>	Iterates over tokens, separated by the supplied delimeters.
<c:param>	Adds a parameter to a containing 'import' tag's URL.
<c:redirect >	Redirects to a new URL.
<c:url>	Creates a URL with optional query parameters

- Formatting tags
- `<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>`

Tag	Description
<fmt:formatNumber>	To render numerical value with specific precision or format.
<fmt:parseNumber>	Parses the string representation of a number, currency, or percentage.
<fmt:formatDate>	Formats a date and/or time using the supplied styles and pattern
<fmt:parseDate>	Parses the string representation of a date and/or time
<fmt:bundle>	Loads a resource bundle to be used by its tag body.
<fmt:setLocale>	Stores the given locale in the locale configuration variable.
<fmt:setBundle>	Loads a resource bundle and stores it in the named scoped variable or the bundle configuration variable.
<fmt:timeZone>	Specifies the time zone for any time formatting or parsing actions nested in its body.
<fmt:setTimeZone>	Stores the given time zone in the time zone configuration variable
<fmt:message>	To display an internationalized message.
<fmt:requestEncoding>	Sets the request character encoding

- SQL tags
- `<%@ taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql" %>`

Tag	Description
<sql:setDataSource>	Creates a simple DataSource suitable only for prototyping
<sql:query>	Executes the SQL query defined in its body or through the sql attribute.
<sql:update>	Executes the SQL update defined in its body or through the sql attribute.
<sql:param>	Sets a parameter in an SQL statement to the specified value.
<sql:dateParam>	Sets a parameter in an SQL statement to the specified java.util.Date value.
<sql:transaction>	Provides nested database action elements with a shared Connection, set up to execute all statements as one transaction.

JSP – Standard Tag Library (JSTL)

- XML tags
- `<%@ taglib prefix="xml" uri="http://java.sun.com/jsp/jstl/xml" %>`

Tag	Description
<x:out>	Like <%= ... >, but for XPath expressions.
<x:parse>	Use to parse XML data specified either via an attribute or in the tag body.
<x:set >	Sets a variable to the value of an XPath expression.
<x:if >	Evaluates a test XPath expression and if it is true, it processes its body. If the test condition is false, the body is ignored.
<x:forEach>	To loop over nodes in an XML document.
<x:choose>	Simple conditional tag that establishes a context for mutually exclusive conditional operations, marked by <when> and <otherwise>
<x:when>	Subtag of <choose> that includes its body if its expression evalutes to 'true'
<x:otherwise>	Subtag of <choose> that follows <when> tags and runs only if all of the prior conditions evaluated to 'false'
<x:transform>	Applies an XSL transformation on a XML document
<x:param>	Use along with the transform tag to set a parameter in the XSLT stylesheet

- JSTL Functions
- `<%@ taglib prefix="fn" uri="http://java.sun.com/jsp/jstl/functions"%>`

Tag	Description
fn:contains()	Tests if an input string contains the specified substring.
fn:containsIgnoreCase()	Tests if an input string contains the specified substring in a case insensitive way.
fn:endsWith()	Tests if an input string ends with the specified suffix.
fn:escapeXml()	Escapes characters that could be interpreted as XML markup.
fn:indexOf()	Returns the index withing a string of the first occurrence of a specified substring.
fn:join()	Joins all elements of an array into a string.
fn:length()	Returns the number of items in a collection, or the number of characters in a string.
fn:replace()	Returns a string resulting from replacing in an input string all occurrences with a given string.
fn:split()	Splits a string into an array of substrings.
fn:startsWith()	Tests if an input string starts with the specified prefix.
fn:substring()	Returns a subset of a string.
fn:substringAfter()	Returns a subset of a string following a specific substring.
fn:substringBefore()	Returns a subset of a string before a specific substring.
fn:toLowerCase()	Converts all of the characters of a string to lower case.
fn:toUpperCase()	Converts all of the characters of a string to upper case.
fn:trim()	Removes white spaces from both ends of a string.

Create Table

```
mysql> use TEST;
mysql> create table Employees
    (
     id int not null,
     age int not null,
     first varchar (255),
     last varchar (255)
    );
Query OK, 0 rows affected (0.08 sec)
```

Create Data Records

```
mysql> INSERT INTO Employees VALUES (100, 18, 'Zara', 'Ali');
Query OK, 1 row affected (0.05 sec)

mysql> INSERT INTO Employees VALUES (101, 25, 'Mahnaz', 'Fatma');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Employees VALUES (102, 30, 'Zaid', 'Khan');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Employees VALUES (103, 28, 'Sumit', 'Mittal');
Query OK, 1 row affected (0.00 sec)
```

SELECT operation

```
<%@ page import="java.io.*,java.util.*,java.sql.*"%>
<%@ page import="javax.servlet.http.*,javax.servlet.*" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>

<html>
<head><title>SELECT Operation</title></head>
<body>
     <sql:setDataSource var="snapshot" driver="com.mysql.jdbc.Driver"
          url="jdbc:mysql://localhost/TEST" user="root"  password="pass123"/>

     <sql:query dataSource="${snapshot}" var="result">
     SELECT * from Employees;
     </sql:query>

     <table border="1" width="100%">
     <tr>
        <th>Emp ID</th>
        <th>First Name</th>
        <th>Last Name</th>
        <th>Age</th>
     </tr>
     <c:forEach var="row" items="${result.rows}">
     <tr>
        <td><c:out value="${row.id}"/></td>
        <td><c:out value="${row.first}"/></td>
        <td><c:out value="${row.last}"/></td>
        <td><c:out value="${row.age}"/></td>
     </tr>
     </c:forEach>
     </table>
</body>
</html>
```

➢ INSERT operation

```
<%@ page import="java.io.*,java.util.*,java.sql.*"%>
<%@ page import="javax.servlet.http.*,javax.servlet.*" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>

<html>
<head><title>JINSERT Operation</title></head>
<body>
     <sql:setDataSource var="snapshot" driver="com.mysql.jdbc.Driver"
          url="jdbc:mysql://localhost/TEST" user="root"  password="pass123"/>

     <sql:update dataSource="${snapshot}" var="result">
     INSERT INTO Employees VALUES (104, 2, 'Nuha', 'Ali');
     </sql:update>

     <sql:query dataSource="${snapshot}" var="result">
     SELECT * from Employees;
     </sql:query>

     <table border="1" width="100%">
     <tr>
        <th>Emp ID</th>    <th>First Name</th>    <th>Last Name</th>    <th>Age</th>
     </tr>
     <c:forEach var="row" items="${result.rows}">
     <tr>
        <td><c:out value="${row.id}"/></td>    <td><c:out value="${row.first}"/></td>
        <td><c:out value="${row.last}"/></td>    <td><c:out value="${row.age}"/></td>
     </tr>
     </c:forEach>
     </table>
</body>
</html>
```

➢ DELETE operation

```
...

<html>
<head><title>DELETE Operation</title></head>
<body>
     <sql:setDataSource var="snapshot" driver="com.mysql.jdbc.Driver"
          url="jdbc:mysql://localhost/TEST" user="root"  password="pass123"/>

     <c:set var="empId" value="103"/>

     <sql:update dataSource="${snapshot}" var="count">
       DELETE FROM Employees WHERE Id = ?
       <sql:param value="${empId}" />
     </sql:update>

     <sql:query dataSource="${snapshot}" var="result">
        SELECT * from Employees;
     </sql:query>

     <table border="1" width="100%">
     <tr>
        <th>Emp ID</th>   <th>First Name</th>   <th>Last Name</th>   <th>Age</th>
     </tr>
     <c:forEach var="row" items="${result.rows}">
     <tr>
        <td><c:out value="${row.id}"/></td>   <td><c:out value="${row.first}"/></td>
        <td><c:out value="${row.last}"/></td>   <td><c:out value="${row.age}"/></td>
     </tr>
     </c:forEach>
     </table>
</body>
</html>
```

UPDATE operation

```
...

<html>
<head><title>DELETE Operation</title></head>
<body>
     <sql:setDataSource var="snapshot" driver="com.mysql.jdbc.Driver"
          url="jdbc:mysql://localhost/TEST" user="root"  password="pass123"/>

     <c:set var="empId" value="102"/>

     <sql:update dataSource="${snapshot}" var="count">
       UPDATE Employees SET last = 'Ali'
       <sql:param value="${empId}" />
     </sql:update>

     <sql:query dataSource="${snapshot}" var="result">
        SELECT * from Employees;
     </sql:query>

     <table border="1" width="100%">
     <tr>
        <th>Emp ID</th>   <th>First Name</th>   <th>Last Name</th>   <th>Age</th>
     </tr>
     <c:forEach var="row" items="${result.rows}">
     <tr>
        <td><c:out value="${row.id}"/></td>   <td><c:out value="${row.first}"/></td>
        <td><c:out value="${row.last}"/></td>   <td><c:out value="${row.age}"/></td>
     </tr>
     </c:forEach>
     </table>
</body>
</html>
```

Sending XML from a JSP

```
<%@ page contentType="text/xml" %>

<books>
   <book>
      <name>Padam History</name>
      <author>ZARA</author>
      <price>100</price>
   </book>
</books>
```

- Processing XML from a JSP
  - XercesImpl.jar: http://www.apache.org/dist/xerces/j/
  - xalan.jar: http://xml.apache.org/xalan-j/index.html

```
<books>
<book>
  <name>Padam History</name>
  <author>ZARA</author>
  <price>100</price>
</book>
<book>
  <name>Great Mistry</name>
  <author>NUHA</author>
  <price>2000</price>
</book>
</books>
```

Processing XML from a JSP

```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml" %>

<html>
<head>
  <title>JSTL x:parse Tags</title>
</head>
<body>
<h3>Books Info:</h3>
<c:import var="bookInfo" url="http://localhost:8080/books.xml"/>

<x:parse xml="${bookInfo}" var="output"/>
<b>The title of the first book is</b>:
<x:out select="$output/books/book[1]/name" />
<br>
<b>The price of the second book</b>:
<x:out select="$output/books/book[2]/price" />

</body>
</html>
```

Formatting XML from a JSP

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">

<xsl:output method="html" indent="yes"/>

<xsl:template match="/">
  <html>
  <body>
   <xsl:apply-templates/>
  </body>
  </html>
</xsl:template>

<xsl:template match="books">
  <table border="1" width="100%">
    <xsl:for-each select="book">
      <tr>
        <td><i><xsl:value-of select="name"/></i></td>
        <td><xsl:value-of select="author"/></td>
        <td><xsl:value-of select="price"/></td>
      </tr>
    </xsl:for-each>
  </table>
</xsl:template>
</xsl:stylesheet>
```

Formatting XML from a JSP

```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml" %>

<html>
<head>
  <title>JSTL x:transform Tags</title>
</head>
<body>
<h3>Books Info:</h3>
<c:set var="xmltext">
  <books>
    <book>
      <name>Padam History</name>
      <author>ZARA</author>
      <price>100</price>
    </book>
    <book>
      <name>Great Mistry</name>
      <author>NUHA</author>
      <price>2000</price>
    </book>
  </books>
</c:set>

<c:import url="http://localhost:8080/style.xsl" var="xslt"/>
<x:transform xml="${xmltext}" xslt="${xslt}"/>

</body>
</html>
```

JavaBean

```
public class StudentsBean implements java.io.Serializable {
   private String firstName = null;
   private String lastName = null;
   private int age = 0;

   public StudentsBean() {
   }
   public String getFirstName(){
      return firstName;
   }
   public String getLastName(){
      return lastName;
   }
   public int getAge(){
      return age;
   }
   public void setFirstName(String firstName){
      this.firstName = firstName;
   }
   public void setLastName(String lastName){
      this.lastName = lastName;
   }
   public void setAge(Integer age){
      this.age = age;
   }
}
```

Accessing JavaBeans Properties

```
<html>
<head>
<title>get and set properties Example</title>
</head>
<body>

<jsp:useBean id="students"
                    class="StudentsBean">
   <jsp:setProperty name="students" property="firstName"
                    value="Zara"/>
   <jsp:setProperty name="students" property="lastName"
                    value="Ali"/>
   <jsp:setProperty name="students" property="age"
                    value="10"/>
</jsp:useBean>

<p>Student First Name:
   <jsp:getProperty name="students" property="firstName"/>
</p>
<p>Student Last Name:
   <jsp:getProperty name="students" property="lastName"/>
</p>
<p>Student Age:
   <jsp:getProperty name="students" property="age"/>
</p>

</body>
</html>
```