Name: James A. Syvertsen
Email: syvertsj.jhu@gmail.com
Course: 605.782 – Java Web Development JHU
Assignment: HW1
URL: http://web1.apl.jhu.edu:8080/syvertsj-forms/index.html

Resources:
Source code from “forms” project handout.
Primarily the following:

- ServletUtilities.java
- ShowParameters.java
- Three-params-form.html
- Index.html

Project Source code:

1. PartOne.java
2. PartTwo.java
3. PartThree.java
4. part-one.html
5. part-two.html
6. part-three.html
7. Index.html

```
// PartOne.java //

package coreservlets;

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

/** Shows all the parameters sent to the servlet via either
 *  GET or POST. Specially marks parameters that have
 *  no values or multiple values.
 *  <p>
 *  From <a href="http://courses.coreservlets.com/Course-Materials/">the
 *  coreservlets.com tutorials on servlets, JSP, Struts, JSF, Ajax, GWT, and Java</a>.
 */

@WebServlet("/part-one")
public class PartOne extends HttpServlet {
  @Override
  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
      throws ServletException, IOException {
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String docType =
      "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 " +
      "Transitional//EN\">\n";
    String title = "Registration Form Entries";
    out.println(docType +
                "<HTML>\n" +
                "<HEAD><TITLE>" + title + "</TITLE></HEAD>\n" +
                "<BODY BGCOLOR=\"#FDF5E6\">\n" +
                "<H1 ALIGN=CENTER>" + title + "</H1>\n" +
                "<TABLE BORDER=1 ALIGN=CENTER>\n" +
                "<TR BGCOLOR=\"#FFAD00\">\n" +
                "<TH>Property<TH>Property Value(s)");
    Enumeration<String> paramNames = request.getParameterNames();
    while(paramNames.hasMoreElements()) {
      String paramName = paramNames.nextElement();
      out.print("<TR><TD>" + paramName + "\n<TD>");
```

```
    String[] paramValues =
      request.getParameterValues(paramName);
    if (paramValues.length == 1) {
      String paramValue = paramValues[0];
      if (paramValue.length() == 0)
        out.println("<I>No Value</I>");
      else
        out.println(paramValue);
    } else {
      out.println("<UL>");
      for(int i=0; i<paramValues.length; i++) {
        out.println("<LI>" + paramValues[i]);
      }
      out.println("</UL>");
    }
   }
   out.println("</TABLE>\n</BODY></HTML>");
  }

  @Override
  public void doPost(HttpServletRequest request,
                     HttpServletResponse response)
     throws ServletException, IOException {
   doGet(request, response);
  }
}


// PartTwo.java //


package coreservlets;

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

/** Shows all the parameters sent to the servlet via either
 *  GET or POST. Specially marks parameters that have
 *  no values or multiple values.
```

```
 *  <p>
 *  From <a href="http://courses.coreservlets.com/Course-Materials/">the
 *  coreservlets.com tutorials on servlets, JSP, Struts, JSF, Ajax, GWT, and Java</a>.
 */

@WebServlet("/part-two")
public class PartTwo extends HttpServlet {
  @Override
  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
      throws ServletException, IOException {
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String docType =
      "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 " +
      "Transitional//EN\">\n";
    String title = "Registration Form Entries";
    docType +=  "<HTML>\n" +
                "<HEAD><TITLE>" + title + "</TITLE></HEAD>\n" +
                "<BODY BGCOLOR=\"#FDF5E6\">\n" +
                "<H1 ALIGN=CENTER>" + title + "</H1>\n" +
                "<TABLE BORDER=1 ALIGN=CENTER>\n" +
                "<TR BGCOLOR=\"#FFAD00\">\n" +
                "<TH>Property<TH>Property Value(s)";
    Enumeration<String> paramNames = request.getParameterNames();
    while(paramNames.hasMoreElements()) {
      String paramName = paramNames.nextElement();
      docType += "<TR><TD>" + paramName + "\n<TD>";
      String[] paramValues =
        request.getParameterValues(paramName);
      if (paramValues.length == 1) {
        String paramValue = paramValues[0];
        if (paramValue.length() == 0)
          docType += "<I>No Value</I>";
        else
          docType += this.filter(paramValue);
      } else {
        docType += "<UL>";
        for(int i=0; i<paramValues.length; i++) {
          docType += "<LI>" + this.filter(paramValues[i]);
        }
        docType += "</UL>";
```

```
    }
  }
  docType += "</TABLE>\n</BODY></HTML>";
  out.println(docType);
}

@Override
public void doPost(HttpServletRequest request,
                   HttpServletResponse response)
    throws ServletException, IOException {
  doGet(request, response);
}


/** Replaces characters that have special HTML meanings
 *  with their corresponding HTML character entities.
 *  Specifically, given a string, this method replaces all
 *  occurrences of
 *  {@literal
 *  '<' with '&lt;', all occurrences of '>' with
 *  '&gt;', and (to handle cases that occur inside attribute
 *  values), all occurrences of double quotes with
 *  '&quot;' and all occurrences of '&' with '&amp;'.
 *  Without such filtering, an arbitrary string
 *  could not safely be inserted in a Web page.
 *  }
 */

public static String filter(String input) {
  if (!hasSpecialChars(input)) {
       return(input);
  }
  StringBuilder filtered = new StringBuilder(input.length());
  char c;
  for(int i=0; i<input.length(); i++) {
    c = input.charAt(i);
    switch(c) {
      case '<': filtered.append("&lt;"); break;
      case '>': filtered.append("&gt;"); break;
      case '"': filtered.append("&quot;"); break;
      case '&': filtered.append("&amp;"); break;
      default: filtered.append(c);
```

```
    }
   }
   return(filtered.toString());
  }

  private static boolean hasSpecialChars(String input) {
   boolean flag = false;
   if ((input != null) && (input.length() > 0)) {
    char c;
    for(int i=0; i<input.length(); i++) {
     c = input.charAt(i);
     switch(c) {
       case '<': flag = true; break;
       case '>': flag = true; break;
       case '"': flag = true; break;
       case '&': flag = true; break;
     }
    }
   }
   return(flag);
  }

}

// PartThree.java //

package coreservlets;

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

/** Shows all the parameters sent to the servlet via either
 *  GET or POST. Specially marks parameters that have
 *  no values or multiple values.
 *  <p>
 *  From <a href="http://courses.coreservlets.com/Course-Materials/">the
 *  coreservlets.com tutorials on servlets, JSP, Struts, JSF, Ajax, GWT, and Java</a>.
 */
```

```
@WebServlet("/part-three")
public class PartThree extends HttpServlet {
  @Override
  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
     throws ServletException, IOException {
   response.setContentType("text/html");
   PrintWriter out = response.getWriter();
   String docType =
    "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 " +
    "Transitional//EN\">\n";
   String title = "Registration Form Entries";
   docType +=  "<HTML>\n" +
            "<HEAD><TITLE>" + title + "</TITLE></HEAD>\n" +
            "<BODY BGCOLOR=\"#FDF5E6\">\n" +
            "<H1 ALIGN=CENTER>" + title + "</H1>\n" +
            "<TABLE BORDER=1 ALIGN=CENTER>\n" +
            "<TR BGCOLOR=\"#FFAD00\">\n" +
            "<TH>Property<TH>Property Value(s)";
   Enumeration<String> paramNames = request.getParameterNames();
   boolean incompleteForm = false;
   while(paramNames.hasMoreElements()) {
    String paramName = paramNames.nextElement();
    docType += "<TR><TD>" + paramName + "\n<TD>";
    String[] paramValues = request.getParameterValues(paramName);
    if (paramValues.length == 1) {
     String paramValue = paramValues[0];
     if (paramValue.length() == 0) {
      incompleteForm = true;
      docType += "<I>No Value</I>";
     } else {
      docType += this.filter(paramValue);
     }
    } else {
     docType += "<UL>";
     for(int i=0; i<paramValues.length; i++) {
      docType += "<LI>" + this.filter(paramValues[i]);
     }
     docType += "</UL>";
    }
   }
   docType += "</TABLE>\n</BODY></HTML>";
```

```
  if (incompleteForm) {
    rewriteForm(request, response);
  } else {
    out.println(docType);
  }

}

@Override
public void doPost(HttpServletRequest request,
                   HttpServletResponse response)
    throws ServletException, IOException {
  doGet(request, response);
}


/** Replaces characters that have special HTML meanings
 *  with their corresponding HTML character entities.
 *  Specifically, given a string, this method replaces all
 *  occurrences of
 *  {@literal
 *  '<' with '&lt;', all occurrences of '>' with
 *  '&gt;', and (to handle cases that occur inside attribute
 *  values), all occurrences of double quotes with
 *  '&quot;' and all occurrences of '&' with '&amp;'.
 *  Without such filtering, an arbitrary string
 *  could not safely be inserted in a Web page.
 *  }
 */

public static String filter(String input) {
  if (!hasSpecialChars(input)) {
        return(input);
  }
  StringBuilder filtered = new StringBuilder(input.length());
  char c;
  for(int i=0; i<input.length(); i++) {
    c = input.charAt(i);
    switch(c) {
      case '<': filtered.append("&lt;"); break;
      case '>': filtered.append("&gt;"); break;
```

```
      case '"': filtered.append("&quot;"); break;
      case '&': filtered.append("&amp;"); break;
      default: filtered.append(c);
    }
  }
  return(filtered.toString());
}

private static boolean hasSpecialChars(String input) {
  boolean flag = false;
  if ((input != null) && (input.length() > 0)) {
    char c;
    for(int i=0; i<input.length(); i++) {
      c = input.charAt(i);
      switch(c) {
        case '<': flag = true; break;
        case '>': flag = true; break;
        case '"': flag = true; break;
        case '&': flag = true; break;
      }
    }
  }
  return(flag);
}

public void rewriteForm(HttpServletRequest request,
                HttpServletResponse response)
        throws ServletException, IOException {
  Enumeration<String> paramNames = request.getParameterNames();
  String[] formValue = new String[4];
  for (int c=0; paramNames.hasMoreElements(); c++) {
    String paramName = paramNames.nextElement();
    String[] paramValues = request.getParameterValues(paramName);
    if (paramValues.length == 1) {
      String paramValue = paramValues[0];
      if ( paramValue.isEmpty() ) {
        formValue[c] = "No Value";
      } else {
        formValue[c] = this.filter(paramValue);
      }
    } else {
      for(int i=0; i < paramValues.length; i++) {
```

```
        if ( paramValues[i].isEmpty() )
          formValue[c] = "No Value";
        else
          formValue[c] = this.filter(paramValues[i]);
       }
     }
   }

   response.setContentType("text/html");
   PrintWriter out = response.getWriter();
   String formType =
            "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 " +
            "Transitional//EN\">\n";
   String formtitle = "Registration Form";
   formType +=  "<HTML><HEAD><TITLE>" + formtitle + "</TITLE></HEAD>\n" +
            "<BODY BGCOLOR=\"#FDF5E6\">\n" +
            "<H1 ALIGN=CENTER>" + formtitle + "</H1>\n\n" +
            "<FORM ACTION=\"part-three\" METHOD=\"POST\">\n" +
            "  First Name: <INPUT TYPE=\"TEXT\" NAME=\"firstName\" VALUE=\"" + formValue[0] +
"\"><BR>\n" +
            "  Last Name:  <INPUT TYPE=\"TEXT\" NAME=\"lastName\"  VALUE=\"" + formValue[1]  +
"\"><BR>\n" +
            "  Phone Number: <INPUT TYPE=\"TEXT\" NAME=\"phoneNumber\" VALUE=\"" +
formValue[2] + "\"><BR>\n" +
            "  Shoe Size:  <INPUT TYPE=\"TEXT\" NAME=\"shoeSize\" VALUE=\"" + formValue[3] +
"\"><BR>\n" +
            "<CENTER><INPUT TYPE=\"SUBMIT\"></CENTER>\n" +
            "</FORM>\n" +
            "</BODY></HTML>\n";
   out.println(formType);
  }

}
```

// part-one.html //

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML><HEAD><TITLE>Registration Form</TITLE></HEAD>
<BODY BGCOLOR="#FDF5E6">
<H1 ALIGN="CENTER">Registration Form</H1>

<FORM ACTION="part-one" METHOD="POST">
```

```
  First Name: <INPUT TYPE="TEXT" NAME="firstName" VALUE="(first name)"><BR>
  Last Name:  <INPUT TYPE="TEXT" NAME="lastName"  VALUE="(last name)"><BR>
  Phone Number: <INPUT TYPE="TEXT" NAME="phoneNumber" VALUE="(phone #)"><BR>
  Shoe Size:  <INPUT TYPE="TEXT" NAME="shoeSize" VALUE="(shoe size)"><BR>
  <CENTER><INPUT TYPE="SUBMIT"></CENTER>
</FORM>

</BODY></HTML>

// part-two.html //

(same as part-one.html)

// part-three.html //

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML><HEAD><TITLE>Registration Form</TITLE></HEAD>
<BODY BGCOLOR="#FDF5E6">
<H1 ALIGN="CENTER">Registration Form</H1>

<FORM ACTION="part-three" METHOD="POST">
  First Name: <INPUT TYPE="TEXT" NAME="firstName"><BR>
  Last Name:  <INPUT TYPE="TEXT" NAME="lastName"><BR>
  Phone Number: <INPUT TYPE="TEXT" NAME="phoneNumber"><BR>
  Shoe Size:  <INPUT TYPE="TEXT" NAME="shoeSize"><BR>
  <CENTER><INPUT TYPE="SUBMIT"></CENTER>
</FORM>

</BODY></HTML>


// index.html //

<!DOCTYPE html>
<html>
<head><title>Form processing</title>
<link rel="stylesheet"
      href="./css/styles.css"
      type="text/css"/>
</head>
<body>
<table class="title">
  <tr><th>Form processing</th></tr>
</table>
<p>
Reminder: these apps run only on Tomcat 7 or other servers
that support servlets 3.0 and later. The course Web site has
an alternative version of this project that runs on Tomcat 6
and other servers that support servlets 2.4 or 2.5.
</p>
<fieldset>
<legend>Links</legend>
<ul>
```

```
  <li><a href="part-one.html">part-one</a></li>
  <li><a href="part-two.html">part-two</a></li>
  <li><a href="part-three.html">part-three</a></li>
</ul>
</fieldset>
<p/>

</font>
</body></html>
```