

```
1 import java.io.*;
2 import java.sql.*;
3 import javax.servlet.*;
4 import javax.servlet.http.*;
5
6 public class OmatRyhmat extends HttpServlet {
7
8     final String dbDriver="org.postgresql.Driver";
9     final String dbServer ="jdbc:postgresql://localhost:10388/tsoha";
10    final String dbUser= "avtanska";           // replace with your db user account
11    final String dbPassword ="postgres"; // replace with your password
12
13    public void service(HttpServletRequest req, HttpServletResponse res)
14        throws ServletException, IOException {
15        ServletOutputStream out;
16        res.setContentType("text/html");
17        out= res.getOutputStream();
18
19        HttpSession session = req.getSession(false);
20        String submitLisays = req.getParameter("submitLisays");
21        String submitPoisto = req.getParameter("submitPoisto");
22        String submitPoistaRyhma = req.getParameter("submitPoistaRyhma");
23        String sessionTunnus = (String)session.getValue("sessionTunnus");
24        String[] jaset = req.getParameterValues("jaset");
25        String[] henkilot = req.getParameterValues("henkilot");
26        String ryhmä = req.getParameter("ryhma");
27        String omatryhmat = req.getParameter("omatryhmat");
28        String virhe;
29
30        out.println("<html><head><title></title>" +
31            "<link rel='stylesheet' type='text/css' href='http://" +
32            "db.cs.helsinki.fi/u/avtanska/tsoha/perus.css' /><body>" +
33            "<form method='post' action='http://db.cs.helsinki.fi/" +
34            "s/avtanska/OmatRyhmat'">");
35
36        if (omatryhmat.equals("tyhjä")) {
37            out.println("<h3>Valitse ryhmäsi</h3>");
38        }
39        else {
40            Connection con = null;
41
42            /*
43             * Jos lisätään henkilöitä ryhmään
44             */
45
46            if (submitLisays != null && henkilot != null) {
47
48                con= createDbConnection(dbDriver,dbServer,dbUser,dbPassword,out);
49
50                if (con==null) {
51                    out.println("</body></html>");
52                    return;
53                }
54
55                Statement stmt = null;
56                ResultSet rs = null;
57                try {
58                    stmt = con.createStatement();
59
60                    for (int i = 0; i < henkilot.length; i++) {
61                        String insertJäsenyys =
62                            "INSERT INTO jäsenyys VALUES ('" + henkilot[i] +
63                            "','" + omatryhmat + "')";
64                        int count = stmt.executeUpdate(insertJäsenyys);
65                    }
66
67                } catch (SQLException ee) {
68                    out.println("Tietokantavirhe "+ee.getMessage());
69                }
```

```
70         } finally {
71             try {
72                 if (rs!=null) rs.close();
73                 if (stmt!=null) stmt.close();
74                 con.close();
75             } catch(SQLException e) {
76                 out.println("An SQL Exception was thrown.");
77             }
78         }
79     }
80 } // loppu if submitLisays ei null
81
82
83 /*
84  * Jos poistetaan jäseniä ryhmästä
85  */
86
87 if (submitPoisto != null && jaset != null) {
88     con= createDbConnection(dbDriver,dbServer,dbUser,dbPassword,out);
89
90     if (con==null) {
91         out.println("</body></html>");
92         return;
93     }
94
95     Statement stmt = null;
96     ResultSet rs = null;
97     try {
98         stmt = con.createStatement();
99
100         for (int i = 0; i < jaset.length; i++) {
101             String deleteJäsenyys =
102                 "DELETE FROM jäsenyys where " + "tunnus='" + jaset[i] +
103                 "' and ryhmä='" + omatryhmat + "'";
104             int count = stmt.executeUpdate(deleteJäsenyys);
105         }
106     }
107     catch (SQLException ee) {
108         out.println("Tietokantavirhe "+ee.getMessage());
109     } finally {
110         try {
111             if (rs!=null) rs.close();
112             if (stmt!=null) stmt.close();
113             con.close();
114         } catch(SQLException e) {
115             out.println("An SQL Exception was thrown.");
116         }
117     }
118 }
119
120 } // loppu if submitPoisto ei null
121
122
123 /*
124  * Jos koko ryhmä poistetaan
125  */
126
127 if (submitPoistaRyhma != null) {
128     con= createDbConnection(dbDriver,dbServer,dbUser,dbPassword,out);
129
130     if (con==null) {
131         out.println("</body></html>");
132         return;
133     }
134
135     Statement stmt = null;
136     ResultSet rs = null;
137     try {
138         stmt = con.createStatement();
```

```
139         int count = 0;
140
141
142         /*
143          * Poistetaan ensin jäsenyys ryhmään kuuluvilta
144          */
145
146         String deleteJäsenyys = "DELETE FROM jäsenyys where " +
147                                 "ryhmä='" + omatryhmat + "'";
148         count = stmt.executeUpdate(deleteJäsenyys);
149
150
151         /*
152          * Lopuksi poistetaan itse ryhmä
153          */
154
155         String deleteRyhmä = "DELETE FROM ryhmä where nimi='" +
156                               omatryhmat + "'";
157         count = stmt.executeUpdate(deleteRyhmä);
158
159         out.println("<h3>Ryhmän poisto onnistui.</h3>" +
160                    "<p>Ryhmä näkyy vielä omien ryhmiesi listassa,<br />" +
161                    "mutta se on poistettu tietokannasta. Seuraavan<br />" +
162                    "kerran kun kirjoittaudut sisään, sitä ei enää<br />" +
163                    "listassakaan näy.</p></body></html>");
164
165         } catch (SQLException ee) {
166             out.println("Tietokantavirhe "+ee.getMessage());
167         } finally {
168             try {
169                 if (rs!=null) rs.close();
170                 if (stmt!=null) stmt.close();
171                 con.close();
172             } catch (SQLException e) {
173                 out.println("An SQL Exception was thrown.");
174             }
175         }
176
177     } // loppu if submitRyhmanPoisto ei null
178
179
180     /*
181     * Jos tullaan 'Omat ryhmät'-valintalaatikon kautta
182     */
183
184     else {
185
186         out.println("<h3>Ryhmän hallinta</h3>");
187
188         out.println("<table border='0'><tr><td align='left' valign='center' " +
189                    "colspan='3'>" +
190                    "Ryhmän nimi:&nbsp;&nbsp;&nbsp;<b>" + omatryhmat + "</b>" +
191                    "</td></tr>");
192
193         con=null;
194         con= createDbConnection(dbDriver,dbServer,dbUser,dbPassword,out);
195         if (con==null) {
196             out.println("</body></html>");
197             return;
198         }
199
200         Statement stmt = null;
201         ResultSet rs = null;
202         try {
203             stmt = con.createStatement();
204
205             String selectOmistaja =
206                 "SELECT omistaja FROM ryhmä WHERE nimi='" +
207                 omatryhmat + "'";
```



```

277
278     out.println("</td>");
279
280     out.println("<td align='center' valign='bottom'>" +
281         "<input type='submit' name='submitLisays' " +
282         "value='&nbsp;&lt;&lt;&nbsp;&nbsp;&nbsp;'>" +
283         "<br /><br />" +
284         "<input type='submit' name='submitPoisto' " +
285         "value='&nbsp;&gt;&gt;&nbsp;&nbsp;&nbsp;'>" +
286         "<input type='hidden' name='omategyhmat' " +
287         "value='" + omategyhmat + "'>" +
288         "</td>");
289
290     /*
291     * Haetaan muut kuin ryhmään kuuluvat henkilöt tietokannasta
292     */
293
294     out.println("<td valign='top' width='120' align='center'>" +
295         "<p>Loput henkilöt:</p><select multiple name='henkilot'>");
296
297     stmt = null;
298     rs = null;
299
300     try {
301         stmt = con.createStatement();
302
303         String selectLoput =
304             "SELECT * FROM henkilö WHERE tunnus IN " +
305             "(SELECT tunnus FROM henkilö WHERE tunnus NOT IN " +
306             "(SELECT tunnus FROM jäsenyys WHERE ryhmä='" +
307             omategyhmat + "') " +
308             ") ORDER BY tunnus";
309
310         rs = stmt.executeQuery(selectLoput);
311
312         while(rs.next()) {
313             if (!(rs.getString("tunnus").equals("admin"))) {
314                 out.println("<option value='" + rs.getString("tunnus") +
315                     "'>" + rs.getString("etunimi") + " " +
316                     rs.getString("sukunimi"));
317             }
318         }
319
320     } catch (SQLException ee) {
321         out.println("Tietokantavirhe "+ee.getMessage());
322     } finally {
323         try {
324             if (rs!=null) rs.close();
325             if (stmt!=null) stmt.close();
326             con.close();
327         } catch (SQLException e) {
328             out.println("An SQL Exception was thrown.");
329         }
330     }
331
332     out.println("</select></td>");
333
334     out.println("</tr><tr><td colspan='3' height='40' align='center'>" +
335         "<input type='reset' value='Tyhjennä'>" +
336         "</td></table>");
337
338     out.println("<table><tr><td colspan='2'>" +
339         "<input type='submit' name='submitPoistaRyhma' " +
340         "value='Poista ryhmä'>" +
341         "</td></tr></table></form>");
342
343     out.println("<form method='post' action='RyhmanAjat'>" +
344         "<table><tr><td colspan='2'>" +
345         "<input type='submit' name='submit' value='Ryhmän ajat'>" +

```

```
346         "<input type='hidden' name='omatryhmat' value='" +
347         omatryhmat + "'" +
348         "</td></tr></table></form></body></html>");
349
350     } // loppu else ryhmän hallinnan perussivu
351
352     } // loppu else omatryhmat ei tyhjä
353 }
354
355
356 private Connection createDbConnection(
357     String dbDriver, String dbServer, String dbUser, String dbPassword,
358     ServletOutputStream out) throws IOException {
359
360     // establish a database connection
361     try{
362         Class.forName(dbDriver);                // load driver
363     } catch (ClassNotFoundException e) {
364         out.println("Couldn't find driver "+dbDriver);
365         return null;
366     }
367     Connection con=null;
368     try {
369         con = DriverManager.getConnection(dbServer,dbUser,dbPassword);
370     } catch (SQLException se) {
371         out.println("Couldn't get connection to "+dbServer+
372             " for "+ dbUser+"<br>");
373         out.println(se.getMessage());
374     }
375     return con;
376 }
377
378 }
```