

```
1 package net.board.db;
2
3 import java.sql.Connection;
4 import java.sql.PreparedStatement;
5 import java.sql.ResultSet;
6 import java.sql.SQLException;
7 import java.util.ArrayList;
8 import java.util.List;
9
10 import javax.naming.Context;
11 import javax.naming.InitialContext;
12 import javax.sql.DataSource;
13
14 public class BoardDAO {
15     Connection con;
16     PreparedStatement pstmt;
17     ResultSet rs;
18     DataSource ds;
19     public BoardDAO() {
20         try{
21             Context init = new InitialContext();
22             ds = (DataSource)
23                 init.lookup("java:comp/env/jdbc/OracleDB");
24
25         }catch(Exception ex){
26             System.out.println("DB 연결 실패 : " + ex);
27             return;
28         }
29     }
30
31     //글의 개수 구하기.
32     public int getListCount() {
33         int x= 0;
34
35         try{
36             con=ds.getConnection();
37             pstmt=con.prepareStatement("select count(*) from
38                 memberboard");
39             rs = pstmt.executeQuery();
40
41             if(rs.next()){
42                 x=rs.getInt(1);
43             }
44         }catch(Exception ex){
45             System.out.println("getListCount 에러: " + ex);
46
47         }finally{
48             if(rs!=null) try{rs.close();}catch(SQLException ex){}
49             if(pstmt!=null) try{pstmt.close();}catch(SQLException
50                 ex){}
51             if(con!=null) try{con.close();}catch(SQLException ex){}
```

```
49     }
50     return x;
51 }
52
53 //글 목록 보기.
54 public List getBoardList(int page,int limit){
55     String board_list_sql="select * from "+
56     "(select rownum rnum,BOARD_NUM,BOARD_ID,BOARD_SUBJECT,"+
57     "BOARD_CONTENT,BOARD_FILE,BOARD_RE_REF,BOARD_RE_LEV,"+
58     "BOARD_RE_SEQ,BOARD_READCOUNT,BOARD_DATE from "+
59     "(select * from memberboard order by "+
60     "BOARD_RE_REF desc,BOARD_RE_SEQ asc)) "+
61     "where rnum>=? and rnum<=?";
62
63     List list = new ArrayList();
64
65     int startrow=(page-1)*10+1; //읽기 시작할 row 번호.
66     int endrow=startrow+limit-1; //읽을 마지막 row 번호.
67     try{
68         con=ds.getConnection();
69         pstmt = con.prepareStatement(board_list_sql);
70         pstmt.setInt(1, startrow);
71         pstmt.setInt(2, endrow);
72         rs = pstmt.executeQuery();
73
74         while(rs.next()){
75             BoardBean board = new BoardBean();
76             board.setBOARD_NUM(rs.getInt("BOARD_NUM"));
77             board.setBOARD_ID(rs.getString("BOARD_ID"));
78
79             board.setBOARD_SUBJECT(rs.getString("BOARD_SUBJECT"));
80
81             board.setBOARD_CONTENT(rs.getString("BOARD_CONTENT"));
82             board.setBOARD_FILE(rs.getString("BOARD_FILE"));
83             board.setBOARD_RE_REF(rs.getInt("BOARD_RE_REF"));
84             board.setBOARD_RE_LEV(rs.getInt("BOARD_RE_LEV"));
85             board.setBOARD_RE_SEQ(rs.getInt("BOARD_RE_SEQ"));
86
87             board.setBOARD_READCOUNT(rs.getInt("BOARD_READCOUNT"));
88             board.setBOARD_DATE(rs.getDate("BOARD_DATE"));
89             list.add(board);
90         }
91
92         return list;
93     }catch(Exception ex){
94         System.out.println("getBoardList 에러 : " + ex);
95     }finally{
96         if(rs!=null) try{rs.close();}catch(SQLException ex){}
97         if(pstmt!=null) try{pstmt.close();}catch(SQLException
```

```
        ex){}
95         if(con!=null) try{con.close();}catch(SQLException ex){}
96     }
97     return null;
98 }
99
100 //글 내용 보기.
101 public BoardBean getDetail(int num) throws Exception{
102     BoardBean board = null;
103     try{
104         con = ds.getConnection();
105         pstmt = con.prepareStatement(
106             "select * from memberboard where BOARD_NUM =
107             ?");
108
109         pstmt.setInt(1, num);
110
111         rs= pstmt.executeQuery();
112
113         if(rs.next()){
114             board = new BoardBean();
115             board.setBOARD_NUM(rs.getInt("BOARD_NUM"));
116             board.setBOARD_ID(rs.getString("BOARD_ID"));
117
118             board.setBOARD_SUBJECT(rs.getString("BOARD_SUBJECT"));
119
120             board.setBOARD_CONTENT(rs.getString("BOARD_CONTENT"));
121
122             board.setBOARD_FILE(rs.getString("BOARD_FILE"));
123             board.setBOARD_RE_REF(rs.getInt("BOARD_RE_REF"));
124             board.setBOARD_RE_LEV(rs.getInt("BOARD_RE_LEV"));
125             board.setBOARD_RE_SEQ(rs.getInt("BOARD_RE_SEQ"));
126
127             board.setBOARD_READCOUNT(rs.getInt("BOARD_READCOUNT"));
128             board.setBOARD_DATE(rs.getDate("BOARD_DATE"));
129         }
130         return board;
131     }catch(Exception ex){
132         System.out.println("getDetail 에러 : " + ex);
133     }finally{
134         if(rs!=null)try{rs.close();}catch(SQLException ex){}
135         if(pstmt !=null)try{pstmt.close();}catch(SQLException
136             ex){}
137         if(con !=null)try{con.close();}catch(SQLException ex){}
138     }
139     return null;
140 }
141
142 //글 등록.
143 public boolean boardInsert(BoardBean board){
144     int num =0;
```

```
138         String sql="";
139
140         int result=0;
141
142         try{
143             con = ds.getConnection();
144             pstmt=con.prepareStatement(
145                 "select max(board_num) from memberboard");
146             rs = pstmt.executeQuery();
147
148             if(rs.next())
149                 num =rs.getInt(1)+1;
150             else
151                 num=1;
152
153             sql="insert into memberboard
154                 (BOARD_NUM,BOARD_ID,BOARD_SUBJECT,";
155             sql+="BOARD_CONTENT, BOARD_FILE,BOARD_RE_REF,"+
156                 "BOARD_RE_LEV,BOARD_RE_SEQ,BOARD_READCOUNT,"+
157                 "BOARD_DATE) values(?,?,?,?,?,?,?,sysdate)";
158
159             pstmt = con.prepareStatement(sql);
160             pstmt.setInt(1, num);
161             pstmt.setString(2, board.getBOARD_ID());
162             pstmt.setString(3, board.getBOARD_SUBJECT());
163             pstmt.setString(4, board.getBOARD_CONTENT());
164             pstmt.setString(5, board.getBOARD_FILE());
165             pstmt.setInt(6, num);
166             pstmt.setInt(7, 0);
167             pstmt.setInt(8, 0);
168             pstmt.setInt(9, 0);
169
170             result=pstmt.executeUpdate();
171             if(result==0)return false;
172
173             return true;
174         }catch(Exception ex){
175             System.out.println("boardInsert 에러 : "+ex);
176         }finally{
177             if(rs!=null) try{rs.close();}catch(SQLException ex){}
178             if(pstmt!=null) try{pstmt.close();}catch(SQLException
179                 ex){}
180             if(con!=null) try{con.close();}catch(SQLException ex){}
181         }
182         return false;
183     }
184
185     //글 답변.
186     public int boardReply(BoardBean board){
187         String board_max_sql="select max(board_num) from
188             memberboard";
189         String sql="";
```

```
187         int num=0;
188         int result=0;
189
190         int re_ref=board.getBOARD_RE_REF();
191         int re_lev=board.getBOARD_RE_LEV();
192         int re_seq=board.getBOARD_RE_SEQ();
193
194         try{
195             con = ds.getConnection();
196             pstmt=con.prepareStatement(board_max_sql);
197             rs = pstmt.executeQuery();
198             if(rs.next()) num =rs.getInt(1)+1;
199             else num=1;
200
201             sql="update memberboard set BOARD_RE_SEQ=BOARD_RE_SEQ+1
202 ";
203             sql+="where BOARD_RE_REF=? and BOARD_RE_SEQ>?";
204
205             pstmt = con.prepareStatement(sql);
206             pstmt.setInt(1,re_ref);
207             pstmt.setInt(2,re_seq);
208             result=pstmt.executeUpdate();
209
210             re_seq = re_seq + 1;
211             re_lev = re_lev+1;
212
213             sql="insert into memberboard
214 (BOARD_NUM,BOARD_ID,BOARD_SUBJECT,";
215
216             sql+="BOARD_CONTENT,BOARD_FILE,BOARD_RE_REF,BOARD_RE_LE
217 V,";
218             sql+="BOARD_RE_SEQ,BOARD_READCOUNT,BOARD_DATE) ";
219             sql+="values(?,?,?,?,?,?,?,sysdate)";
220
221             pstmt = con.prepareStatement(sql);
222             pstmt.setInt(1, num);
223             pstmt.setString(2, board.getBOARD_ID());
224             pstmt.setString(3, board.getBOARD_SUBJECT());
225             pstmt.setString(4, board.getBOARD_CONTENT());
226             pstmt.setString(5, ""); //답장에는 파일을 업로드하지 않음.
227             pstmt.setInt(6, re_ref);
228             pstmt.setInt(7, re_lev);
229             pstmt.setInt(8, re_seq);
230             pstmt.setInt(9, 0);
231             pstmt.executeUpdate();
232             return num;
233         }catch(SQLException ex){
234             System.out.println("boardReply 에러 : "+ex);
235         }finally{
236             if(rs!=null) try{rs.close();}catch(SQLException ex){}
237             if(pstmt!=null) try{pstmt.close();}catch(SQLException
238 ex){}
```

```
234         if(con!=null)try{con.close();}catch(SQLException ex){}
235     }
236     return 0;
237 }
238
239 //글 수정.
240 public boolean boardModify(BoardBean modifyboard) throws
Exception{
241     String sql="update memberboard set BOARD_SUBJECT=?,";
242     sql+="BOARD_CONTENT=? where BOARD_NUM=?";
243
244     try{
245         con = ds.getConnection();
246         pstmt = con.prepareStatement(sql);
247         pstmt.setString(1, modifyboard.getBOARD_SUBJECT());
248         pstmt.setString(2, modifyboard.getBOARD_CONTENT());
249         pstmt.setInt(3, modifyboard.getBOARD_NUM());
250         pstmt.executeUpdate();
251         return true;
252     }catch(Exception ex){
253         System.out.println("boardModify 에러 : " + ex);
254     }finally{
255         if(rs!=null)try{rs.close();}catch(SQLException ex){}
256         if(pstmt!=null)try{pstmt.close();}catch(SQLException
ex){}
257         if(con!=null)try{con.close();}catch(SQLException ex){}
258     }
259     return false;
260 }
261
262 //글 삭제.
263 public boolean boardDelete(int num){
264     String board_delete_sql=
265         "delete from memberboard where BOARD_num=?";
266
267     int result=0;
268
269     try{
270         con = ds.getConnection();
271         pstmt=con.prepareStatement(board_delete_sql);
272         pstmt.setInt(1, num);
273         result=pstmt.executeUpdate();
274         if(result==0)return false;
275
276         return true;
277     }catch(Exception ex){
278         System.out.println("boardDelete 에러 : "+ex);
279     }finally{
280         try{
281             if(pstmt!=null)pstmt.close();
282             if(con!=null)con.close();
283         }catch(Exception ex){}
```

```
284     }
285
286     return false;
287 }
288
289 //조회수 업데이트.
290 public void setReadCountUpdate(int num) throws Exception{
291     String sql="update memberboard set BOARD_READCOUNT = "+
292         "BOARD_READCOUNT+1 where BOARD_NUM = "+num;
293
294     try{
295         con=ds.getConnection();
296         pstmt=con.prepareStatement(sql);
297         pstmt.executeUpdate();
298     }catch(SQLException ex){
299         System.out.println("setReadCountUpdate 에러 : "+ex);
300     }
301     finally{
302         try{
303             if(pstmt!=null)pstmt.close();
304             if(con!=null)con.close();
305         }catch(Exception ex){}
306     }
307 }
308
309 //글쓴이인지 확인.
310 public boolean isBoardWriter(int num,String id){
311     System.out.println("id="+id);
312     String board_sql=
313         "select * from memberboard where BOARD_NUM=?";
314
315     try{
316         con=ds.getConnection();
317         pstmt=con.prepareStatement(board_sql);
318         pstmt.setInt(1, num);
319         rs=pstmt.executeQuery();
320         rs.next();
321
322         if(id.equals(rs.getString("BOARD_ID"))){
323             return true;
324         }
325     }catch(SQLException ex){
326         System.out.println("isBoardWriter 에러 : "+ex);
327     }
328     finally{
329         try{
330             if(pstmt!=null)pstmt.close();
331             if(con!=null)con.close();
332         }catch(Exception ex){}
333     }
334     return false;
335 }
```

```
336 }  
337
```