

게시판 강좌 따라하기 13강

게시판 글 삭제하기 기능 구현하기

#BbsDAO

강좌 카테고리를 정주행 하셨다면 코드를 이미 입력하셨겠지만 삭제하기 포스팅으로 바로 접속하신 분들을 위해서 'BbsDAO' 코드를 다시 한 번 올리겠습니다.

```
155 //게시글 삭제 메소드
156 public int delete(int bbsID) {
157     //실제 데이터를 삭제하는 것이 아니라 게시글 유효숫자를 '0'으로 수정한다
158     String sql = "update bbs set bbsAvailable = 0 where bbsID = ?";
159     try {
160         PreparedStatement pstmt = conn.prepareStatement(sql);
161         pstmt.setInt(1, bbsID);
162         return pstmt.executeUpdate();
163     } catch (Exception e) {
164         e.printStackTrace();
165     }
166     return -1; //데이터베이스 오류
167 }
168
```

```
package bbs;
```

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.ArrayList;
```

```
public class BbsDAO {

    private Connection conn;
    private ResultSet rs;

    //기본 생성자
```

```

public BbsDAO() {
    try {
        String dbURL = "jdbc:mariadb://localhost:3306/bbs";
        String dbID = "root";
        String dbPassword = "001025";
        Class.forName("org.mariadb.jdbc.Driver");
        conn = DriverManager.getConnection(dbURL, dbID, dbPassword);
    } catch (Exception e) {
        e.printStackTrace();
    }
}

```

//작성일자 메소드

```

public String getDate() {
    String sql = "select now()";
    try {
        PreparedStatement pstmt = conn.prepareStatement(sql);
        rs = pstmt.executeQuery();
        if(rs.next()) {
            return rs.getString(1);
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
    return ""; //데이터베이스 오류
}

```

//게시글 번호 부여 메소드

```

public int getNext() {
    //현재 게시글을 내림차순으로 조회하여 가장 마지막 글의 번호를 구한다
    String sql = "select bbsID from bbs order by bbsID desc";
    try {
        PreparedStatement pstmt = conn.prepareStatement(sql);
        rs = pstmt.executeQuery();
        if(rs.next()) {
            return rs.getInt(1) + 1;
        }
        return 1; //첫 번째 게시글인 경우
    }
}

```

```

    }catch (Exception e) {
        e.printStackTrace();
    }
    return -1; //데이터베이스 오류
}

```

//글쓰기 메소드

```

public int write(String bbsTitle, String userID, String bbsContent) {
    String sql = "insert into bbs values(?, ?, ?, ?, ?, ?)";
    try {
        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setInt(1, getNext());
        pstmt.setString(2, bbsTitle);
        pstmt.setString(3, userID);
        pstmt.setString(4, getDate());
        pstmt.setString(5, bbsContent);
        pstmt.setInt(6, 1); //글의 유효번호
        return pstmt.executeUpdate();
    }catch (Exception e) {
        e.printStackTrace();
    }
    return -1; //데이터베이스 오류
}

```

//게시글 리스트 메소드

```

public ArrayList<Bbs> getList(int pageNumber){
    String sql = "select * from bbs where bbsID < ? and bbsAvailable = 1
order by bbsID desc limit 10";
    ArrayList<Bbs> list = new ArrayList<Bbs>();
    try {
        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setInt(1, getNext() - (pageNumber - 1) * 10);
        rs = pstmt.executeQuery();
        while(rs.next()) {
            Bbs bbs = new Bbs();
            bbs.setBbsID(rs.getInt(1));
            bbs.setBbsTitle(rs.getString(2));
            bbs.setUserID(rs.getString(3));

```

```

        bbs.setBbsDate(rs.getString(4));
        bbs.setBbsContent(rs.getString(5));
        bbs.setBbsAvailable(rs.getInt(6));
        list.add(bbs);
    }
} catch (Exception e) {
    e.printStackTrace();
}
return list;
}

```

//페이징 처리 메소드

```

public boolean nextPage(int pageNumber) {
    String sql = "select * from bbs where bbsID < ? and bbsAvailable = 1";
    try {
        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setInt(1, getNext() - (pageNumber - 1) * 10);
        rs = pstmt.executeQuery();
        if(rs.next()) {
            return true;
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
    return false;
}

```

//하나의 게시글을 보는 메소드

```

public Bbs getBbs(int bbsID) {
    String sql = "select * from bbs where bbsID = ?";
    try {
        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setInt(1, bbsID);
        rs = pstmt.executeQuery();
        if(rs.next()) {
            Bbs bbs = new Bbs();
            bbs.setBbsID(rs.getInt(1));
            bbs.setBbsTitle(rs.getString(2));

```

```

        bbs.setUserID(rs.getString(3));
        bbs.setBbsDate(rs.getString(4));
        bbs.setBbsContent(rs.getString(5));
        bbs.setBbsAvailable(rs.getInt(6));
        return bbs;
    }
} catch (Exception e) {
    e.printStackTrace();
}
return null;
}

```

//게시글 수정 메소드

```

public int update(int bbsID, String bbsTitle, String bbsContent) {
    String sql = "update bbs set bbsTitle = ?, bbsContent = ? where bbsID
= ?";

    try {
        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setString(1, bbsTitle);
        pstmt.setString(2, bbsContent);
        pstmt.setInt(3, bbsID);
        return pstmt.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
    return -1; //데이터베이스 오류
}

```

//게시글 삭제 메소드

```

public int delete(int bbsID) {
    //실제 데이터를 삭제하는 것이 아니라 게시글 유효숫자를 '0'으로 수정한다
    String sql = "update bbs set bbsAvailable = 0 where bbsID = ?";
    try {
        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setInt(1, bbsID);
        return pstmt.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

```

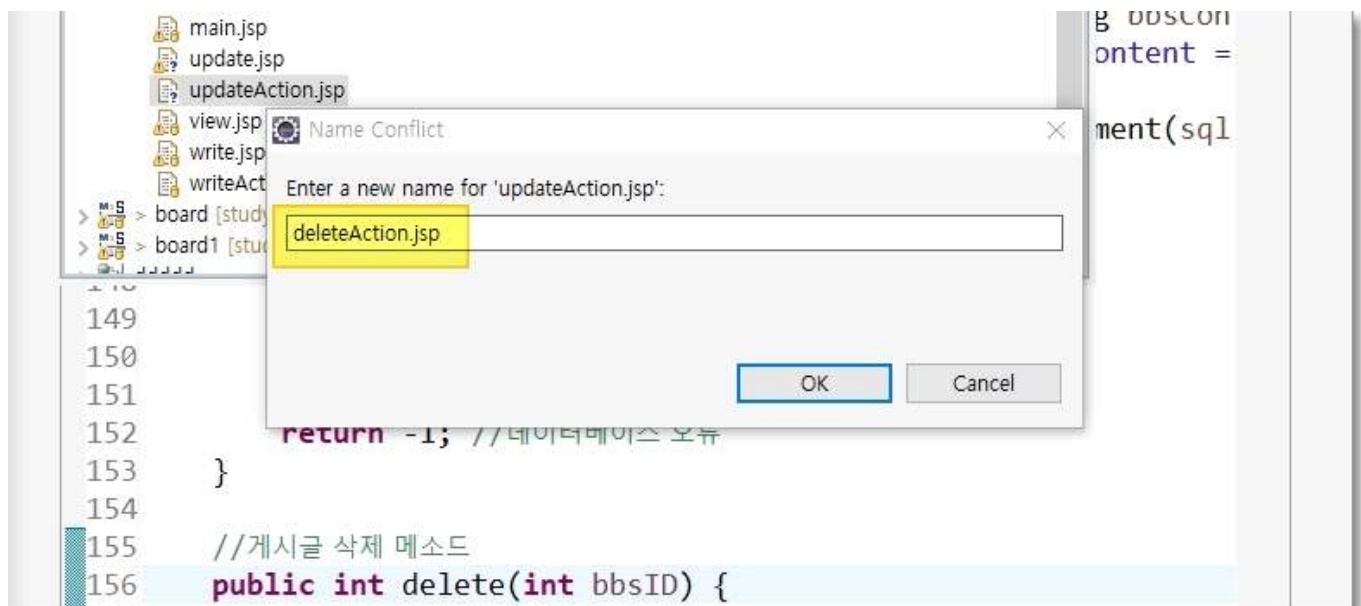
```

    }
    return -1; //데이터베이스 오류
}

}

```

deleteAction.jsp



'updateAction.jsp' 페이지를 복사하여 'deleteAction.jsp' 페이지를 생성합니다



필요 없는 코드들을 지웁니다

```
46      script.println("location.href='bbs.jsp'");
47      script.println("</script>");
48  } else{
49      // 글 삭제 로직을 수행한다
50      BbsDAO bbsDAO = new BbsDAO();
51      int result = bbsDAO.delete(bbsID);
52      // 데이터베이스 오류인 경우
53      if(result == -1){
54          PrintWriter script = response.getWriter();
55          script.println("<script>");
56          script.println("alert('글 수정하기에 실패했습니다')");
57          script.println("history.back()");
58          script.println("</script>");
59      // 글 삭제가 정상적으로 실행되면 알림창을 띄우고 게시판 메인으로 이동한다
60      }else {
61          PrintWriter script = response.getWriter();
62          script.println("<script>");
63          script.println("alert('글 삭제하기 성공')");
64          script.println("location.href='bbs.jsp'");
65          script.println("</script>");
66      }
67  }
68
69  %>
70 </body>
71 </html>
```

html/body/jsp:scriptlet/#text

글 삭제 기능을 처리해줄 코드를 추가합니다

```
<%@page import="java.io.PrintWriter"%>
<%@page import="bbs.BbsDAO"%>
<%@page import="bbs.Bbs"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<% request.setCharacterEncoding("utf-8"); %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>JSP 게시판 웹 사이트</title>
```

</head>

<body>

<%

// 현재 세션 상태를 체크한다

String userID = null;

if(session.getAttribute("userID") != null){

 userID = (String)session.getAttribute("userID");

}

// 로그인을 한 사람만 글을 쓸 수 있도록 코드를 수정한다

if(userID == null){

 PrintWriter script = response.getWriter();

 script.println("<script>");

 script.println("alert('로그인을 하세요')");

 script.println("location.href='login.jsp'");

 script.println("</script>");

}

int bbsID = 0;

if(request.getParameter("bbsID") != null){

 bbsID = Integer.parseInt(request.getParameter("bbsID"));

}

if(bbsID == 0){

 PrintWriter script = response.getWriter();

 script.println("<script>");

 script.println("alert('유효하지 않은 글입니다')");

 script.println("location.href='bbs.jsp'");

 script.println("</script>");

}

//해당 'bbsID'에 대한 게시글을 가져온 다음 세션을 통하여 작성자 본인이 맞는지 체크한다

Bbs bbs = new BbsDAO().getBbs(bbsID);

if(!userID.equals(bbs.getUserID())){

 PrintWriter script = response.getWriter();

 script.println("<script>");

 script.println("alert('권한이 없습니다')");

 script.println("location.href='bbs.jsp'");

 script.println("</script>");

} else{


```

// 글 삭제 로직을 수행한다
BbsDAO bbsDAO = new BbsDAO();
int result = bbsDAO.delete(bbsID);
// 데이터베이스 오류인 경우
if(result == -1){
    PrintWriter script = response.getWriter();
    script.println("<script>");
    script.println("alert('글 수정하기에 실패했습니다')");
    script.println("history.back()");
    script.println("</script>");
// 글 삭제가 정상적으로 실행되면 알림창을 띄우고 게시판 메인으로
이동한다

}else {
    PrintWriter script = response.getWriter();
    script.println("<script>");
    script.println("alert('글 삭제하기 성공')");
    script.println("location.href='bbs.jsp'");
    script.println("</script>");
}
}

%>
</body>
</html>

```

글 삭제하기 테스트

게시판 글 보기

글 제목	수정한 제목
작성자	admin
작성일자	2020-07-18 18시55분
내용	수정한 내용

목록

수정

삭제

로그인을 하고 직접 쓴 게시글을 눌러서 '삭제' 버튼을 눌러봅니다

번호	제목	작성자	작성일
13	<script>악성스크립트 방지 </script>	admin	2020-07-17 14시10분
12	특수문자 테스트	admin	2020-07-17 14시04분
11	테스트11	admin	2020-07-15 18시06분
10	테스트10	admin	2020-07-15 18시06분
9	테스트9	admin	2020-07-15 18시06분
8	테스트8	admin	2020-07-15 18시06분
7	테스트7	admin	2020-07-15 18시05분
6	테스트6	admin	2020-07-15 18시05분
5	테스트5	admin	2020-07-15 18시05분
4	테스트44	admin	2020-07-15 18시05분

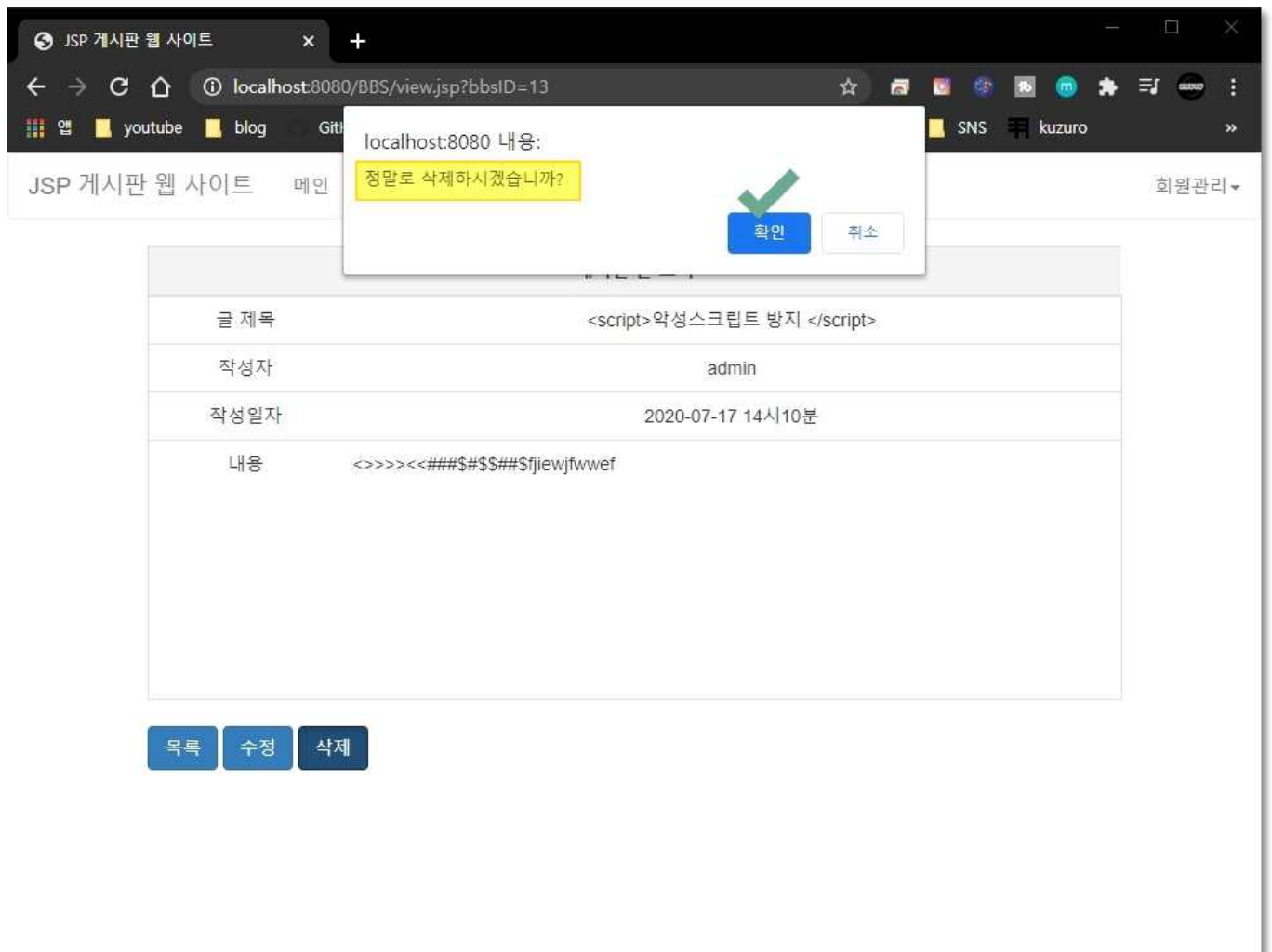
다음

글쓰기

삭제 기능이 정상적으로 실행이 되면 게시판 메인 페이지('bbs.jsp')로 이동하며 내가 삭제했던 게시글은 사라지고 바로 밑에 있던 게시글이 상단으로 올라온 것을 확인할 수 있습니다

```
136      <!-- 해당 글의 작성자가 본인이라면 수정과 삭제가 가능하도록 코드 추가 -->
137      <%
138          if(userID != null && userID.equals(bbs.getUserID())){
139              %>
140                  <a href="update.jsp?bbsID=<%= bbsID %>" class="btn btn-primary">수정</a>
141                  <a onclick="return confirm('정말로 삭제하시겠습니까?')" href=
142                      "deleteAction.jsp?bbsID=<%= bbsID %>" class="btn btn-primary">삭제</a>
143              <%
144          }
145      %>
146  </div>
147  </div>
```

'view.jsp' 를 열어서 글을 삭제할 때 실수를 방지하기 위한 더블체크를 해주는 코드를 추가합니다



'삭제' 버튼을 누르면 한 번 더 물어보는 알림창이 뜨는 것을 확인할 수 있습니다

번호	제목	작성자	작성일
12	특수문자 테스트	admin	2020-07-17 14시 04분
11	테스트11	admin	2020-07-15 18시 06분
10	테스트10	admin	2020-07-15 18시 06분
9	테스트9	admin	2020-07-15 18시 06분
8	테스트8	admin	2020-07-15 18시 06분
7	테스트7	admin	2020-07-15 18시 05분
6	테스트6	admin	2020-07-15 18시 05분
5	테스트5	admin	2020-07-15 18시 05분
4	테스트44	admin	2020-07-15 18시 05분
3	테스트3333	admin	2020-07-15 18시 00분

다음

글쓰기

정상적으로 게시글 삭제가 처리된 것을 확인할 수 있습니다



이제 준비한 JSP 게시판의 모든 기능이 구현 되었습니다. 아직 메인 페이지를 꾸미는 과정이 하나 더 남아 있지만 공부가 목적이시라면 1강부터 13강까지에 해당하는 내용만 잘 이해하시더라도 JSP를 공부함에 있어서 매우 큰 도움이 될 거라고 생각합니다. 이 강좌를 유튜브로 먼저 접했는데 이해가 안 되서 돌아보다가 이 곳에 들르셔서 공부하시는 분이나, 바로 제 블로그로 오셔서 공부하시는 분들 모두 제가 경험했던 것처럼 JSP 게시판 무작정 따라하기를 해보셔서 조금이라도 더 도움이 되었으면 하는 바람입니다. 혹시 1강부터 지금까지 포스팅을 보시고 따라하셨다면 축하드리고 봐주셔서 감사하다는 말 전해드리고 싶습니다.