

AI3

도서관리 페이지

2021 / TEAM PORTFOLIO 박재형,신봉규,손병규,전규환,채예진



AN OVERVIEW

저희 팀은 이번 프로젝트로
도서 관리 페이지를 제작하였습니다.

Frontend:

박재형 채예진

Backend:

신봉규 손병규

DBA:

전규환

01

프로젝트 설계

03

프로젝트 일정

05

사이트 시연

02

프로젝트 기술

04

코드 리뷰

06

QnA

1.

프로젝트 설계

순서도

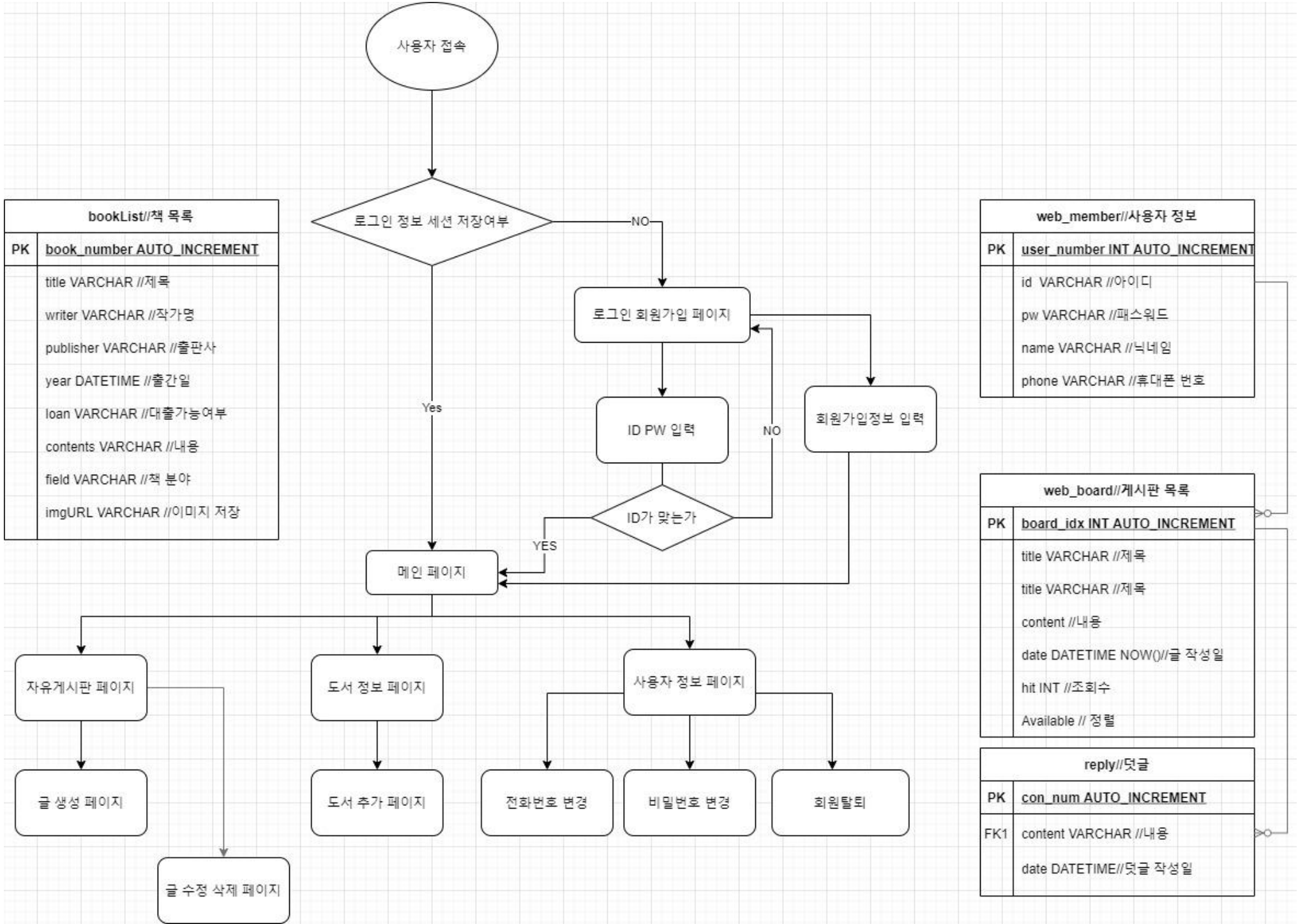
FlowChart & ERD

페이지의 간략한 순서도 입니다.

인증후에만 메인에 접속할수있는구조를 가지고 있으며

메인 페이지에서는 도서정보를 볼수 있으며

자유 게시판에서는 소통기능을 구현하였습니다.



도서 목록

로그인후 접속할수 있는
메인페이지의 도서목록 입니다.

추천하는 책의 제목, 설명 등등
여러가지 정보가 표시되는 페이지 입니다.

게시판

저음으로 도서관리 게시판 로그아웃

번호	제목	작성자	작성일	조회수
15	테스트	c	2021-11-14 08:21:42	1
14	아이	c	2021-11-14 08:13:36	2
13	asdf	testuser	2021-11-12 16:01:06	1
12	asdf	testuser	2021-11-12 16:01:06	1
11	asdf	testuser	2021-11-12 16:01:06	1
10	a	testuser	2021-11-12 16:01:06	1
9	as	testuser	2021-11-12 16:01:06	1
8	asd	testuser	2021-11-12 16:01:06	3
7	asdf	testuser	2021-11-12 16:01:06	3
6	test	admin	2021-11-11 16:01:06	3

글쓰기

12

책

홈 도서관리 게시판 로그아웃 내정보

원하는 책을 마음껏 검색해보세요!

2021-11-14 출간

피터 래빗 시리즈 전집

베아트릭스 포터 지음 | 윤후남 옮김

2021-11-14 출간

피터래빗 시리즈

피터 래빗 시리즈는 단순히 그림만 예쁜 그림 동화가 아니다. 각 이야기들은 단순한 듯하면서도 그 안에 해학이 있고, 의인화된 동물들은 사람들이 사는 현실 세계의 모습을 잘 반영하고 있어 어린이에게는 물론이거니와, ‘어른을 위한 동화’로도 더없는 즐거움을 선사한다.

Read more →

★단골손님을 찾습니다★

달라구트

꿈백화점²

A D O N T E C L E P

C F J U S T Y J U S T

E U D O U V D O L T K

G U S J C I T I Z E W

J O D M P B X H K S B

Y O U R E A H I U G L

C D O M C E E V E R Y

admin님

책 리뷰 회원정보 로그아웃

도서 검색

Enter search term... 검색

분야별 찾기

소설 건강
인문 사전
음악 해외도서

인생책, 100리스트

인생에 한번쯤은 꼭 읽어야 할 도서 시리즈

자유롭게 글을 쓸수있는 게시판 입니다
로그인 시에만 글쓰기가 가능하며

글쓰기,편집,삭제
기능이 구현되어 있습니다

박재형 신봉규 손병규 전규환 채예진

2.

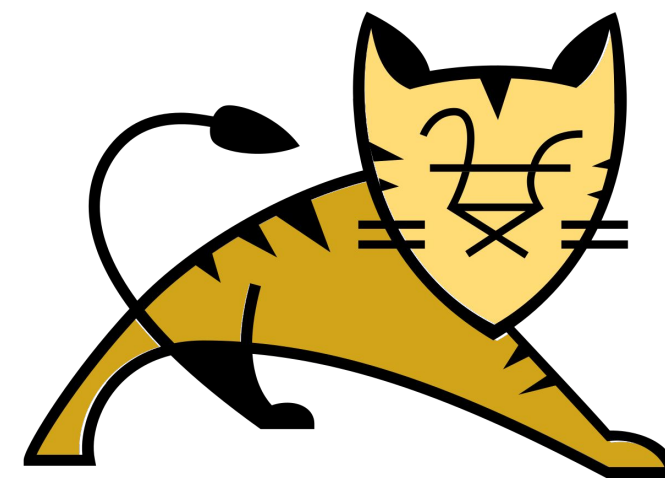
프로젝트
기술

2-1

OS WEB WAS

저희 팀은 이클립스와 깃허브로 협업이 끝난 후
서버에 웹 페이지를 구축하였습니다

부트스트랩을 이용하여 반응형 페이지를 제작하였습니다.



Ubuntu Nginx Setting

ubuntu

유동 IP를 도메인과 연결하기 위해
ddclient를 이용하여 DDNS 설정을 하였습니다

https 설정을 위한 SSL인증서는
certbot 으로 Let's Encrypt 인증서를 발급받았습니다.

nginx

인증서로 https 접속을 설정하였으며
도메인이 아닌 ip로 접속시 444에러를 출력하게 하였습니다.

프록시패스를 설정하여 톰캣서버와 연동하였습니다.

ddclient 설정

```
# Configuration file for ddclient generated by debconf
#
# /etc/ddclient.conf
daemon=300
ssl=yes
use=web
protocol=googledomains

# Log file (if you want to use it)
#logfile="/var/log/ddclient.log"

# Login and password for the web interface
#login="admin"
#password="password"

login=gyuhwan
password=1234567890
zone=gyuhwan.com
gyuhwan.com,www.gyuhwan.com
```

nginx 설정

```
server {
    #gyuhwan.com
    #listen 80 default_server;
    #listen [::]:80 default_server;

    listen 443 ssl;
    #listen [::]: ssl;

    ssl_certificate /etc/letsencrypt/live/gyuhwan.com/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/gyuhwan.com/privkey.pem;

    root /var/www/html;

    index index.html index.htm index.nginx-debian.html;

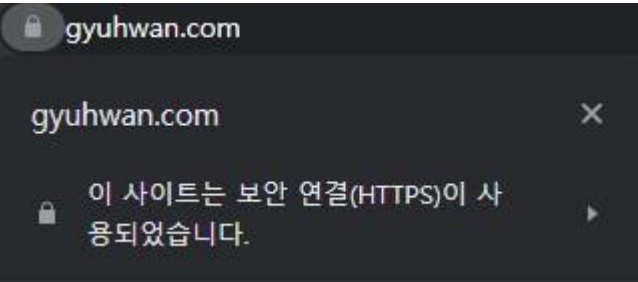
    server_name www.gyuhwan.com gyuhwan.com;

    location / {
        proxy_pass http://localhost:8081;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $http_host;
        #try_files $uri $uri/ =404;
    }
}

server {
    #default_server
    listen 80;
    #listen 443 ssl;

    root /var/www/html;
    index index.html;
    server_name ~^(.+)$;
    return 444;
    error_page 403 404 405 406 411 497 500 501 502 503 504 505 /index.html;
}
```

https 적용



2-2

Tomcat setting

web.xml을 설정하여 주소창에 .jsp가 보이지 않게 하였습니다.
server.xml을 수정하여 에러페이지에
정보가 나오지 않게 하였습니다

server.xml

```
<Service name="Catalina">
  <Connector port="8081" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="4443" />
  <Connector port="4009" protocol="AJP/1.3" redirectPort="4443" URIEncoding="UTF-8"/>

  <Engine name="Catalina" defaultHost="localhost">
    <Realm className="org.apache.catalina.realm.LockOutRealm">
      <Realm className="org.apache.catalina.realm.UserDatabaseRealm"
        resourceName="UserDatabase"/>
    </Realm>

    <Host name="localhost" appBase="/tom/"
      unpackWARs="true" autoDeploy="true">
      <Context path="" docBase="/tom/" reloadable="true"/>
      <Valve className="org.apache.catalina.valves.ErrorReportValve" showReport="false" showServerInfo="false"/>
    </Host>
  </Engine>
</Service>
```

웹 에러 발생시

HTTP Status 404 - /

type Status report

message!

description The requested resource (/) is not available.

Apache Tomcat/7.0.28

HTTP 상태 404 – 찾을 수 없음

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>  
<web-app metadata-complete="true">  
  <display-name>T1</display-name>  
  <welcome-file-list>  
    <welcome-file>index.html</welcome-file>  
    <welcome-file>index.jsp</welcome-file>  
  </welcome-file-list>  
  
  <servlet>  
    <servlet-name>main</servlet-name>  
    <jsp-file>/main.jsp</jsp-file>  
  </servlet>  
  
  <servlet>  
    <servlet-name>addBook</servlet-name>  
    <jsp-file>/addBook.jsp</jsp-file>  
  </servlet>  
  
  <servlet>  
    <servlet-name>addBookAction</servlet-name>  
    <jsp-file>/addBookAction.jsp</jsp-file>  
  </servlet>  
  
  <servlet>  
    <servlet-name>board</servlet-name>  
    <jsp-file>/board.jsp</jsp-file>  
  </servlet>  
  
  <servlet>  
    <servlet-name>delete</servlet-name>  
    <jsp-file>/delete.jsp</jsp-file>  
  </servlet>  
  
  <servlet>  
    <servlet-name>deleteAction</servlet-name>  
    <jsp-file>/deleteAction.jsp</jsp-file>  
  </servlet>  
  
  <servlet>  
    <servlet-name>index</servlet-name>  
    <jsp-file>/index.jsp</jsp-file>  
  </servlet>  
  
  <servlet>  
    <servlet-name>join</servlet-name>  
    <jsp-file>/join.jsp</jsp-file>  
  </servlet>  
  
  <servlet>
```

"web.xml" [dos] 184L, 4996B

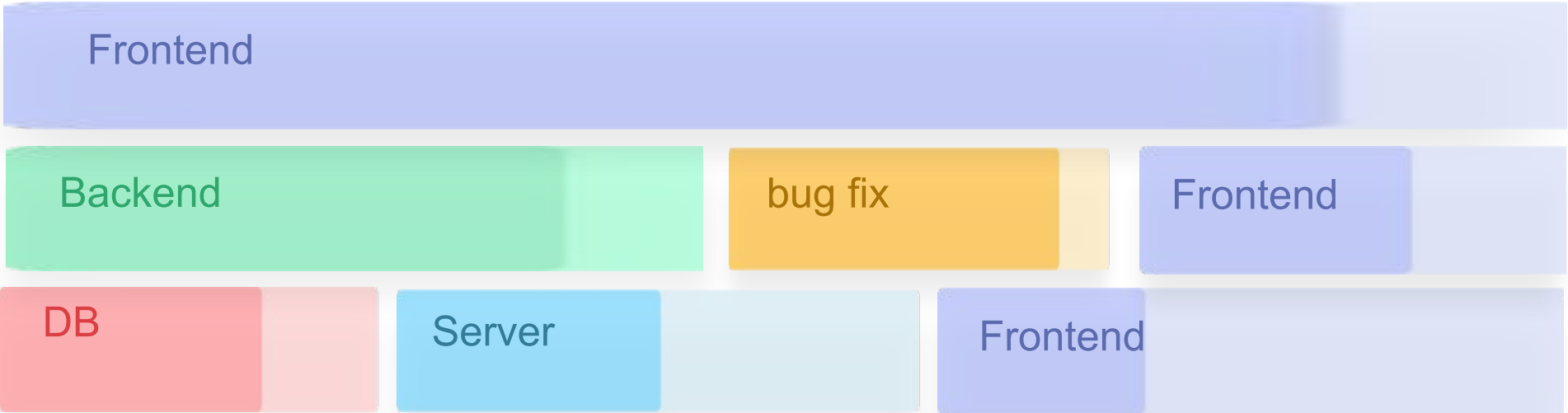
2-3 그 외 사용 프로그램



3.

프로젝트 일정

일정



처음 분배된 역할을
먼저 끝낸 팀원이

작업량이 많은
다른 팀원의 작업을
도와주는 방식으로 진행 하였습니다

4.

코드
리뷰

4-1

로그인

/login

```

<form method="post" action="LoginAction">
  <h3 style="text-align: center;">로그인 화면</h3>
  <div class="form-group">
    <input type="text" class="form-control" placeholder="아이디" name="Id" maxlength="20">
  </div>
  <div class="form-group">
    <input type="password" class="form-control" placeholder="비밀번호" name="Pw" maxlength="20">
  </div>
  <input type="submit" class="btn btn-primary form-control" value="로그인"><br><br>
</form>
<form method="post" action="join">
  <input type="submit" class="btn btn-primary form-control" value="회원가입">
</form>

```

memberDAO

```

public int login(String id, String pw) {
    String SQL = "SELECT pw FROM web_member WHERE id=?";
    try {
        pstmt = con.prepareStatement(SQL);
        pstmt.setString(1, id);
        rs = pstmt.executeQuery();
        if (rs.next()) {
            if(rs.getString(1).equals(pw)) {
                return 1; //로그인 성공
            }
            else {
                return 0; //비밀번호 X
            }
        }
    }
    return -1; //ID X
}

```

/LoginAction

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>A팀 프로젝트-로그인</title>
  </head>
  <body>
    <%
      String userID=request.getParameter("Id");
      String userPassword=request.getParameter("Pw");

      memberDAO dao=new memberDAO();
      int result=dao.login(userID,userPassword);
      if(result == 1){
        session.setAttribute("userID",userID);
        PrintWriter script = response.getWriter();
        script.println("<script>");
        script.println("alert('로그인 성공')");
        script.println("location.href='index'");
        script.println("</script>");
      }
    %>
  </body>
</html>

```


회원가입

회원가입

```

/join
<form method="post" action="joinAction">
  <h3 style="text-align: center;">회원가입 화면</h3>
  <div class="form-group">
    <input type="text" class="form-control" placeholder="아이디" name="Id" maxlength="20" required>
  </div>
  <div class="form-group">
    <input type="password" class="form-control" placeholder="비밀번호" name="Pw" maxlength="20" required>
  </div>
  <div class="form-group">
    <input type="text" class="form-control" placeholder="이름" name="Name" maxlength="20" required>
  </div>
  <div class="form-group">
    <input type="number" class="form-control" placeholder="전화번호" name="Phone" maxlength="20" required>
  </div>
  <input type="submit" class="btn btn-primary form-control" value="회원가입">
</form>

```

memberDAO

```
public void signUp(String id, String pw, String name, String phone) {  
    String SQL="insert into web_member(id,pw,name,phone) "  
        + "values(?,?,?,?)";  
    try {  
        pstmt = con.prepareStatement(SQL);  
        pstmt.setString(1, id);  
        pstmt.setString(2, pw);  
        pstmt.setString(3, name);  
        pstmt.setString(4, phone);  
        pstmt.executeUpdate();  
    } catch (Exception e) {  
        System.out.println("#####\n\n"+e.getMessage());  
    }  
}
```

```

/joinAction
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>A팀 프로젝트-로그인</title>
  </head>
  <body>
    <%PrintWriter pw = response.getWriter(); %>
    <%
String userID=request.getParameter("Id");
String userPassword=request.getParameter("Pw");
String userName=request.getParameter("Name");
String userPhone=request.getParameter("Phone");

if(userID.contains(" ")||userPassword.contains(" ")){
    pw.println("<script>");
    pw.println("alert('ID혹은 비밀번호에 띄어쓰기가 존재합니다')");
    pw.println("history.back()");
    pw.println("</script>");
}else {
memberDAO dao=new memberDAO();
dao.signUp(userID,userPassword,userName,userPhone);
System.out.println("회원가입을 시도함");
response.sendRedirect("./index");
}
%>
  </body>
</html>

```


4-3

게시판 글 쓰기

boardDAO

```
public int writeBoard(String id, String title, String content) {
    String SQL = "INSERT INTO web_board values (?, ?, ?, ?, ?, ?, ?, ?)";
    try {
        PreparedStatement pstmt = con.prepareStatement(SQL);
        pstmt.setInt(1, getNext());
        pstmt.setString(2, id);
        pstmt.setString(3, title);
        pstmt.setString(4, content);
        pstmt.setString(5, getDate());
        pstmt.setInt(6, 1);
        pstmt.setInt(7, 1); // 유효번호
        return pstmt.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

write

```
<tr>
    <td><input type="text" class="form-control"
        placeholder="글 제목" name="title" maxlength="100"></td>
</tr>
<tr>
    <td><textarea class="form-control" placeholder="글 내용"
        name="content" maxlength="1024" style="height: 350px;"></textarea></td>
</tr>
```

writeAction

```
<%
String userID = null;
if (session.getAttribute("userID") != null) {
    userID = (String) session.getAttribute("userID");
}
%>
<%

String title=request.getParameter("title");
String content=request.getParameter("content");

        boardDAO boardDAO = new boardDAO();
        int result = boardDAO.writeBoard(userID,title ,content);
        // 데이터베이스 오류인 경우
        if (result == -1) {
            PrintWriter script = response.getWriter();
            script.println("<script>");
            script.println("alert('글쓰기에 실패했습니다')");
            script.println("history.back()");
            script.println("</script>");
            // 글쓰기가 정상적으로 실행되면 알림창을 띄우고 게시판 메인으로 이동한다
        } else {
            PrintWriter script = response.getWriter();
            script.println("<script>");
            script.println("alert('글쓰기 성공')");
            script.println("location.href='board'");
            script.println("</script>");
        }
    }
}
```


4-4

게시판 수정

boardDAO

```
// 게시글수정
public int update(int board_idx, String title, String content) {
    String SQL = "update web_board set title=?, content=? where board_idx=?";
    try {
        PreparedStatement pstmt = con.prepareStatement(SQL);
        pstmt.setString(1, title);
        pstmt.setString(2, content);
        pstmt.setInt(3, board_idx);
        return pstmt.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
    return -1; // 데이터베이스 오류
}
```

update

```
<%=board_idx%>
action="updateAction?board_idx=<%=board_idx%>"
<table class="table table-striped"
    style="text-align: center; border: 1px solid #dddddd; width: 600px">
    <thead>
        <tr>
            <th colspan="2"
                style="background-color: #eeeeee; text-align: center;">작성 글
                수정</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <td><input type="text" class="form-control"
                placeholder="글 제목" name="title" maxlength="100" value="<%=board.getTitle() %>"></td>
        </tr>
        <tr>
            <td><textarea class="form-control" placeholder="글 내용"
                name="content" maxlength="1024" style="height: 350px;"><%=board.getContent() %>
            </td>
        </tr>
    </tbody>
</table>
<!-- 글쓰기 버튼 생성 -->
<input type="submit" class="btn btn-primary pull-right"
    value="수정하기 ">
```

updateAction

```
<%
String userID = null;
if (session.getAttribute("userID") != null) {
    userID = (String) session.getAttribute("userID");
}
%>
<%
int board_idx = Integer.parseInt(request.getParameter("board_idx"));
String title = request.getParameter("title");
String content = request.getParameter("content");

boardDAO boardDAO = new boardDAO();
int result = boardDAO.update(board_idx, title, content);
// 데이터베이스 오류인 경우
if (result == -1) {
    PrintWriter script = response.getWriter();
    script.println("<script>");
    script.println("alert('글수정에 실패했습니다')");
    script.println("history.back()");
    script.println("</script>");
    // 글쓰기가 정상적으로 실행되면 알림창을 띄우고 게시판 메인으로 이동한다
}
```


4-5

게시판 삭제

boardDAO

```
//게시글삭제
public int delete(int board_idx) {
    String SQL = "update web_board set Available=0 where board_idx=?";
    try {
        PreparedStatement pstmt = con.prepareStatement(SQL);
        pstmt.setInt(1, board_idx);
        return pstmt.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
    return -1; // 데이터베이스 오류
}
```

view

```
<!-- 해당 글의 작성자가 본인이라면 수정과 삭제가 가능하도록 코드 추가 -->
<%
if (userID != null && userID.equals(board.getId())) {
%>
<a href="update?board_idx=<%=board_idx%>" class="btn btn-primary">수정</a>
<a onclick="return confirm('삭제하시겠습니까?')"
href="board_deleteAction?board_idx=|
<%=board_idx%>
"
class="btn btn-primary">삭제</a>
<%
}
```

board_deleteAction

```
String userID = null;
if (session.getAttribute("userID") != null) {
    userID = (String) session.getAttribute("userID");
}
%>
<%
int board_idx = Integer.parseInt(request.getParameter("board_idx"));

boardVO bbs = new boardDAO().getBoard(board_idx);
if (!userID.equals(bbs.getId())) {
    PrintWriter script = response.getWriter();
    script.println("<script>");
    script.println("alert('권한이 없습니다')");
    script.println("location.href='bbs'");
    script.println("</script>");
} else {
    boardDAO BoardDAO = new boardDAO();
    int result = BoardDAO.delete(board_idx);
    // 데이터베이스 오류인 경우
    if (result == -1) {
        PrintWriter script = response.getWriter();
        script.println("<script>");
        script.println("alert('글수정에 실패했습니다')");
        script.println("history.back()");
        script.println("</script>");
        // 글쓰기가 정상적으로 실행되면 알림창을 띄우고 게시판 메인으로 이동한다
    } else {
        PrintWriter script = response.getWriter();
        script.println("<script>");
        script.println("alert('삭제 성공')");
        script.println("location.href='board'");
        script.println("</script>");
    }
}
```

5.

사이트
시연

원하는 책을 마음껏 검색해보세요!



2021-11-14 출간

피터래빗 시리즈

피터 래빗 시리즈는 단순히 그림만 예쁜 그림 동화가 아니다. 각 이야기들은 단순한 듯하면서도 그 안에 해학이 있고, 의인화된 동물들은 사람들이 사는 현실 세계의 모습을 잘 반영하고 있어 어린이에게는 물론이거니와, '어른을 위한 동화'로도 더없는 즐거움을 선사한다.

[Read more →](#)

admin님

[책 리뷰](#)

[회원정보](#)

[로그아웃](#)

도서 검색

Enter search term...

[검색](#)

분야별 찾기

[소설](#)

[인문](#)

[유아](#)

[건강](#)

[사전](#)

[해외도서](#)

인생책, 100리스트

인생에 한번쯤은 꼭 읽어야 할 도서 시리즈



A	D	O	N	T	E	C	L	E	P	
C	F	J	U	S	T	Y	J	U	S	T
E	U	D	O	U	V	D	O	L	T	K
G	U	S	J	C	I	T	!	Z	E	W
J	O	D	M	P	B	X	H	K	S	B
Y	O	U	R	E	A	H	I	U	G	L
C	D	O	M	C	E	E	V	E	R	Y

6.

QnA

THANK
YOU