



Chapter 14 게시판 제작

목차

01 게시판의 기능과 구성

02 글쓰기

03 글 목록 보기

04 글 내용 보기

05 첨부 파일 다운로드

06 글 수정 및 삭제

학습목표

- 게시판의 기능과 구성을 이해할 수 있다.
- 게시판 DB 테이블을 생성할 수 있다.
- 게시판의 글쓰기 폼 양식을 작성할 수 있다.
- 게시판의 글을 DB에 저장할 수 있다.
- 게시글의 목록과 내용을 DB에서 가져올 수 있다.
- DB에서 게시글을 수정 및 삭제할 수 있다.

- 이 장에서 제작하는 게시판은 글쓰기, 목록 보기, 내용 보기, 수정하기, 삭제하기, 파일 첨부하기 등의 기능을 가지고 있음.
- 게시판의 글쓰기 페이지를 불러오기 위해 웹 브라우저 주소 창에 다음 주소 입력.

`http://localhost/part3/board_form.php`

[TIP] 실습 사이트의 게시판은 로그인한 사용자만 글쓰기가 가능하도록 설계되어 있으므로 주소를 입력하기 전에 로그인을 해야 함.

01 게시판의 기능과 구성

PHP 프로그래밍 입문

루바토(rubato)님[Level:9, Point:0] | 로그아웃(12장) | 정보 수정(12장)

HOME

쪽지 만들기(13장)

게시판 만들기(14장)

사이트 완성하기(15장)

PHP Coding
Server Side Web Programming



게시판 > 글쓰기

이름 :

루바토

제목 :

내용 :

첨부 파일

찾아보기...

완료

목록

그림 14-1 게시판의 글쓰기 페이지(board_form.php)

- 실습 사이트 게시판의 기능
 - ✓ 로그인한 회원이 게시판에 글쓰기
 - ✓ 회원과 비회원의 글 목록 보기, 글 내용 보기
 - ✓ 글을 쓴 회원이 글을 수정 및 삭제하기
 - ✓ 글쓰기 시 파일 첨부하기
 - ✓ 글을 쓴 회원에게 포인트 부여하기
 - ✓ 첨부 파일 다운로드하기

표 14-1 게시판 기능을 구현하는 데 필요한 파일

구분	파일	기능
CSS	css/common.css	사이트에 공통으로 사용되는 CSS 파일
	css/board.css	게시판 페이지의 CSS 파일
페이지	board_form.php	글쓰기 폼 양식
	board_list.php	글 목록 보기
	board_view.php	글 내용 보기
	board_modify_form.php	글 수정 폼 양식
기능 페이지	board_insert.php	DB에 글 저장
	board_modify.php	DB에 저장된 글 수정
	board_delete.php	DB에 저장된 글 삭제
	board_download.php	첨부 파일 다운로드

- 이 게시판은 글 목록 보기, 글쓰기 폼 양식, 글 내용 보기, 글 수정 폼 양식 웹 페이지와 DB에서 게시글을 처리하는 기능을 구현하는 4개의 파일로 구성되어 있음.

1. 게시판 DB 테이블 설계

- 게시판을 이루는 DB 테이블의 구조는 [표 14-2]와 같음.
- 테이블 필드는 회원 정보 관련 필드와 게시판 글 관련 필드로 구분. 그 외 조회수와 파일 첨부 관련 필드가 있음.

표 14-2 게시판 DB 테이블(테이블명: board)

필드명	데이터형	null	추가 사항	설명
num	int	not null	primary key, auto_increment	일련번호
id	char(15)	not null		글쓴이 아이디
name	char(10)	not null		글쓴이 이름
subject	char(200)	not null		글 제목
content	text	not null		글 내용
regist_day	char(20)	not null		작성 일시
hit	int	not null		조회 수
file_name	char(40)			파일명
file_type	char(40)			파일 형식
file_copied	char(40)			저장 파일명

- num, id, name, subject, content, regist_day, hit 필드는 not null로 설정되는 필수 입력 항목. 레코드 일련번호를 의미하는 num은 주 키와 auto_increment로 설정.
- num 필드는 게시글의 고유한 일련번호. 이 필드를 이용하여 해당 레코드를 수정하거나 삭제.
- id, name, subject, content 필드는 각각 글쓴이의 아이디, 이름, 게시글의 제목과 내용을 의미. regist_day 필드는 게시글을 작성한 일시를 의미.
- hit 필드는 글 조회 수를 세는 데 사용, file_name, file_type, file_copied 필드는 게시글에 파일을 첨부하는 데 사용.

2. 게시판 DB 테이블 생성

- 게시판 DB 테이블을 생성하는 데 필요한 SQL 명령을 담은 board.sql 파일

게시판 DB 테이블(board)생성 파일 `part3/db/board.sql`

```
create table board (  
    num int not null auto_increment,  
    id char(15) not null,  
    name char(10) not null,  
    subject char(200) not null,  
    content text not null,  
    regist_day char(20) not null,  
    hit int not null,  
    file_name char(40),  
    file_type char(40),  
    file_copied char(40),  
    primary key(num)  
);
```

- phpMyAdmin에서 board.sql 파일을 이용하여 게시판 DB 테이블 생성.
- user1 계정(비밀번호: 12345)으로 phpMyAdmin에 접속.
- sample 데이터베이스 선택, [가져오기] 메뉴에 이어 <찾아보기> 클릭하여 board.sql 파일 실행.
- board 테이블 생성.

01 게시판의 기능과 구성

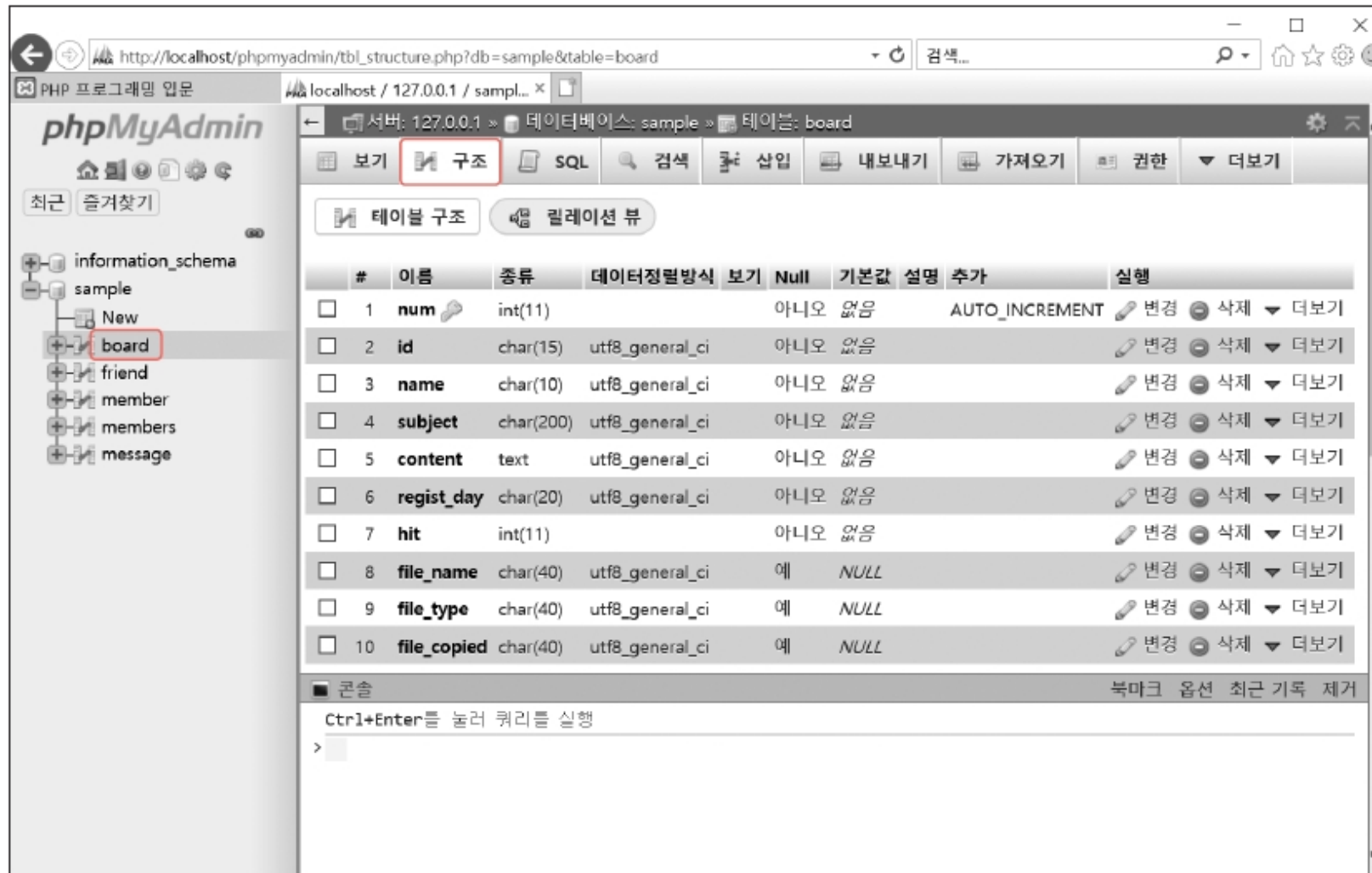


그림 14-2 board 테이블의 구조

1. 글쓰기 폼 양식

게시판 > 글쓰기

이름 :

루바토

제목 :

내용 :

첨부 파일

찾아보기...

완료

목록

그림 14-3 글쓰기 폼 양식

[예제 14-1] 글쓰기 폼 양식 페이지 `part3/board_form.php`

```
<!-- 생략 -->
```

```
<script>
```

```
function check_input() {
```

```
    if (!document.board_form.subject.value)
```

```
    {
```

```
        alert("제목을 입력하세요!");
```

```
        document.board_form.subject.focus();
```

```
        return;
```

```
    }
```

```
    if (!document.board_form.content.value)
```

```
    {
```

```
        alert("내용을 입력하세요!");
```

```
        document.board_form.content.focus();
```

```
        return;
```

```
    }
```

①

[예제 14-1] 글쓰기 폼 양식 페이지 `part3/board_form.php`

```
        document.board_form.submit();
    }
</script>
<!-- 생략 -->
<section>
<div id="main_img_bar">
    
</div>
<div id="board_box">
<h3 id="board_title">
    게시판 > 글쓰기
</h3>
    <form name="board_form" method="post" action="board_insert.php" ②
        enctype="multipart/form-data">
        <ul id="board_form">
```

[예제 14-1] 글쓰기 폼 양식 페이지 `part3/board_form.php`

```
<li>
    <span class="col1">이름 : </span>
    <span class="col2"><?=$username?> </span>
</li>
<li>
    <span class="col1">제목 : </span>
    <span class="col2"><input name="subject" type="text"> </span> ③
</li>
<li id="text_area">
    <span class="col1">내용 : </span>
    <span class="col2">
        <textarea name="content"> </textarea> ④
    </span>
</li>
<li>
```


[예제 14-1] 글쓰기 폼 양식 페이지 `part3/board_form.php`

```
<span class="col1"> 첨부 파일</span>
<span class="col2"> <input type="file" name="upfile"> </span> ⑤
</li>
</ul>
<ul class="buttons">
<li> <button type="button" onclick="check_input()">완료</button> </li>
<li> <button type="button" onclick="location.href='board_list.php'">
    목록</button> </li>
</ul>
</form>
</div> <!-- board_box -->
</section>
<!-- 생략 -->
```

① check_input() 함수

- 글쓰기 폼 양식 하단의 <완료> 버튼([그림 14-3]) 클릭하면 호출되는 check_input() 함수 정의.
- 사용자가 입력 창에 내용을 입력했는지 검사.
- 이상이 없으면 ②의 action에 설정한 board_insert.php로 이동.

② <form> 문

- <form> 태그의 name 속성 값인 board_form은 ①의 check_input() 함수에서 사용.
- 파일 첨부 기능을 사용할 때는 enctype 속성을 multipart/form-data로 설정.

③ 글 제목 설정

- 글쓰기 폼 양식에서 제목 입력 창의 name 속성을 subject로 설정.
- 사용자가 입력한 제목이 POST 방식인 \$_POST["subject"]로 board_insert.php에 전달됨.

④ 글 내용 설정

사용자가 입력한 글 내용이 \$_POST["content"]로 board_insert.php에 전달.

⑤ 첨부 파일 선택

- <input> 태그의 type 속성을 file로 설정하면 업로드 파일을 선택할 수 있는 <찾아보기> 버튼 생성.
- name 속성 값인 upfile은 첨부 파일을 서버에 업로드할 때 사용됨.

2. DB에 폼 양식 데이터 저장

- 글쓰기 폼 양식에서 사용자가 입력한 데이터는 [예제 14-1]의 ②에서 action 속성으로 설정한 board_insert.php에 의해 DB에 저장.

[예제 14-2] DB에 폼 양식 데이터 저장하기

part3/board_insert.php

```
<meta charset="utf-8">
<?php
    session_start();
    if (isset($_SESSION["userid"])) $userid = $_SESSION["userid"];
    else $userid = "";
    if (isset($_SESSION["username"])) $username = $_SESSION["username"];
    else $username = "";
    if (!$userid)
    {
        echo("
        <script>
            alert('게시판 글쓰기는 로그인 후 이용해 주세요!');
            history.go(-1)
        </script>
        ");
        exit;
    }
```

①

[예제 14-2] DB에 폼 양식 데이터 저장하기

part3/board_insert.php

```
$subject = $_POST["subject"];  
$content = $_POST["content"];
```

②

```
$subject = htmlspecialchars($subject, ENT_QUOTES);  
$content = htmlspecialchars($content, ENT_QUOTES);
```

```
$regist_day = date("Y-m-d (H:i)"); // 현재의 '연-월-일-시-분' 저장
```

```
$upload_dir = './data/';
```

③

```
$upfile_name = $_FILES["upfile"]["name"];  
$upfile_tmp_name = $_FILES["upfile"]["tmp_name"];  
$upfile_type = $_FILES["upfile"]["type"];  
$upfile_size = $_FILES["upfile"]["size"];  
$upfile_error = $_FILES["upfile"]["error"];
```

④

[예제 14-2] DB에 폼 양식 데이터 저장하기

part3/board_insert.php

```
if ($upfile_name && !$upfile_error)
{
    $file = explode(".", $upfile_name);
    $file_name = $file[0];
    $file_ext = $file[1];
    $new_file_name = date("Y_m_d_H_i_s");
    $new_file_name = $new_file_name;
    $copied_file_name = $new_file_name.".".$file_ext;
    $uploaded_file = $upload_dir.$copied_file_name;
    if ($upfile_size > 1000000) {
        echo("
        <script>
            alert('업로드 파일 크기가 지정된 용량(1MB)을
            초과합니다!<br>파일 크기를 체크해주세요! ');
            history.go(-1)
        </script>
        ");
    }
}
```

⑤

⑥

⑦

[예제 14-2] DB에 폼 양식 데이터 저장하기 part3/board_insert.php

```
        exit;
    }

    if (!move_uploaded_file($upfile_tmp_name, $uploaded_file) )
    {
        echo("
        <script>
            alert('파일을 지정한 디렉터리에 복사하는 데 실패했습니다.');
```

⑧

```
            history.go(-1)
        </script>
        ");
        exit;
    }
}
else
{
```


[예제 14-2] DB에 폼 양식 데이터 저장하기

part3/board_insert.php

```
$upfile_name = "";
$upfile_type = "";
$copied_file_name = "";
}
$con = mysqli_connect("localhost", "user1", "12345", "sample");
$sql = "insert into board (id, name, subject, content, regist_day,
    hit, file_name, file_type, file_copied) ";
$sql = "values('$userid', '$username', '$subject', '$content',
    '$regist_day', 0, ";
$sql .= "'$upfile_name', '$upfile_type', '$copied_file_name')";
mysqli_query($con, $sql); // $sql에 저장된 명령 실행

// 포인트 부여하기
$point_up = 100;
$sql = "select point from members where id='$userid'";
$result = mysqli_query($con, $sql);
$row = mysqli_fetch_array($result);
```

⑨

⑩

[예제 14-2] DB에 폼 양식 데이터 저장하기 [part3/board_insert.php](#)

```
$new_point = $row["point"] + $point_up;
```

```
$sql = "update members set point=$new_point where id='$userid'";  
mysqli_query($con, $sql);
```

⑪

```
mysqli_close($con); // DB 연결 끊기
```

```
echo "
```

```
<script>
```

```
    location.href = 'board_list.php';
```

⑫

```
</script>
```

```
";
```

```
?>
```

① 로그인 아이디와 이름 가져오기

- 실습 사이트의 게시판에 글을 쓰려면 로그인한 상태여야 함.
- 사용자의 아이디와 이름을 의미하는 세션 변수인 `$_SESSION["userid"]`, `$_SESSION["username"]` 값을 각각 `$userid`, `$username`에 저장.

② 글 제목과 내용 전달받기

- 사용자가 글쓰기 폼 양식에서 입력한 제목과 내용을 `$_POST["subject"]`, `$_POST["content"]`로 전달받아 `$subject`, `$content`에 저장.

③ 업로드 폴더 설정

업로드 파일을 저장할 폴더인 `$upload_dir` 값을 `./data`로 설정.

④ 업로드 파일 정보 가져오기

- [예제 14-1]의 ⑤에서 업로드 파일의 name 속성을 upfile로 설정하여 업로드 파일의 정보가 배열 변수로 전달됨.

표 14-3 name 속성이 upfile일 때 전달되는 배열 변수

배열	설명
<code>\$_FILES["upfile"]["name"]</code>	업로드 파일명
<code>\$_FILES["upfile"]["tmp_name"]</code>	실제 서버에 저장되는 임시 파일명
<code>\$_FILES["upfile"]["type"]</code>	업로드 파일의 형식
<code>\$_FILES["upfile"]["size"]</code>	업로드 파일의 크기
<code>\$_FILES["upfile"]["error"]</code>	오류 발생

- Test.zip 파일을 업로드한다면 `$_FILES["upfile"]["name"]`에는 파일명인 test.zip 저장.
- `$_FILES["upfile"]["tmp_name"]`에는 파일명(test.zip) 대신 실제 서버에 저장되는 임시 파일명 저장.
- `$_FILES["upfile"]["type"]`에는 test.zip 파일의 형식 저장.
- `$_FILES["upfile"]["size"]`에는 test.zip 파일의 크기 저장.
- `$_FILES["upfile"]["error"]`에는 업로드시 발생하는 오류 정보 저장.

⑤ 파일명과 확장자 분리

- explode() 함수를 이용하여 업로드 파일인 \$upfile_name에서 파일명과 확장자를 분리.
- 파일명은 \$file_name에, 확장자는 \$file_ext에 저장.

⑥ 실제 업로드 파일명 구하기

- date() 함수로 실제 업로드할 파일명을 구한 다음 확장자와 붙여서 \$copied_file_name에 저장.
- \$uploaded_file에는 저장할 폴더인 ./data를 포함.

⑦ 업로드 파일의 크기 제한

- 업로드 파일의 크기가 1,000,000바이트(약 1메가바이트)를 초과하면 경고 메시지를 출력.
- 이전의 폼 양식 페이지로 돌아감.

⑧ 업로드 파일을 ./data 폴더에 저장

- `move_uploaded_file()` 함수로 서버에 임시 저장된 `$upfile_tmp_name`을 `$uploaded_file`의 값인 경로/파일명 형태로 저장. 업로드 파일명이 중복되는 것을 피할 수 있음.
- if문 조건식으로 `move_uploaded_file()` 함수를 수행할 때 오류가 발생하면 '파일을 지정한 디렉터리에 복사하는 데 실패했습니다.'라는 메시지 경고 창을 띄우고 이전 폼 양식 페이지로 돌아감.

move_uploaded_file() 함수	
형식	<code>move_uploaded_file(file, newloc)</code>
기능	업로드된 파일을 새로운 위치의 파일(경로 포함)로 이동한다.
인자	file: 업로드된 파일의 임시 파일 newloc: 경로를 포함한 파일명.확장자
반환 값	성공: true, 실패: false

⑨ DB에 게시글 저장

- board 테이블의 각 필드에 해당하는 데이터를 저장.
- \$userid와 \$username은 로그인한 사용자의 아이디와 비밀번호, \$subject와 \$content는 게시글의 제목과 내용, \$regist_day는 작성 일시, 0은 조회 수의 초깃값을 나타냄.
- \$upfile_name, \$upfile_type, \$copied_file_name은 각각 업로드 파일명, 업로드 파일 형식, 실제 업로드 파일명을 나타냄.

⑩ 새로운 포인트 구하기

- 게시판에 글을 쓰면 회원에게 포인트 100점 부여. 부여되는 포인트인 \$point_up을 100으로 설정.
- members 테이블에서 로그인 아이디인 \$userid의 현재 포인트를 \$row["point"]로 가져와 \$point_up과 합산한 값을 새로운 포인트인 \$new_point에 저장.

⑪ DB에 포인트 업데이트

SQL의 update 명령으로 members 테이블의 point 필드를 \$new_point 값으로 업데이트.

⑫ 글 목록 보기 페이지로 이동

location 객체의 속성 location.href에 board_list.php를 설정하여 글 목록 보기 페이지인 board_list.php로 이동.

- 글 목록 보기를 알아보기 위해 글쓰기 폼 양식에 먼저 내용을 입력.

게시판 > 글쓰기

이름 : 루비토

제목 : 안녕하세요.

내용 : 반갑습니다.
오랜만이군요~~~

첨부 파일

그림 14-4 글쓰기 폼 양식에 제목과 내용 입력

03 글 목록 보기

- 글쓰기 폼 양식에 내용을 입력한 다음 하단의 <완료> 클릭, 글 정보가 DB에 저장되고 글 목록 보기 페이지가 나타남.

PHP 프로그래밍 입문

루바토(rubato)님 [Level:9, Point:0] | 로그인(12장) | 정보 수정(12장)

HOME

꼭지 만들기(13장)

게시판 만들기(14장)

사이트 완성하기(15장)

PHP Coding
Server Side Web Programming



게시판 > 목록 보기

번호	제목	글쓴이	첨부	등록일	조회
1	만남하세요,	루바토		2019-04-11 (03:05)	0

1

목록글쓰기

PHP 프로그래밍 입문 | 한빛아카데미

예제 소스 다운로드

- 한빛아카데미(<http://hanbit.co.kr>)

- 코딩스쿨(<http://codingschool.info>)

저자 문의 메일

- 메일 주소 : goldmont@naver.com

그림 14-5 글 목록 보기 페이지(board_list.php)

03 글 목록 보기

- 글 목록 보기 페이지에서는 DB에 저장된 게시글을 한 행씩 가져옴. 입력한 게시글이 DB에 저장되어 있는지 phpMyAdmin에서 확인.



그림 14-6 board 테이블에 저장된 데이터 확인

[예제 14-3] 글 목록 보기 페이지 `part3/board_list.php`

```
<!-- 생략 -->
<section>
<div id="main_img_bar">
    
</div>
<div id="board_box">
    <h3>
        게시판 > 목록 보기
    </h3>
    <ul id="board_list">
    <li>
        <span class="col1">번호</span>
        <span class="col2">제목</span>
        <span class="col3">글쓴이</span>
        <span class="col4">첨부</span>
        <span class="col5">등록일</span>
        <span class="col6">조회</span>
```

[예제 14-3] 글 목록 보기 페이지 `part3/board_list.php`

```
</li>
<?php
    if (isset($_GET["page"]))
        $page = $_GET["page"];
    else
        $page = 1;

    $con = mysqli_connect("localhost", "user1", "12345", "sample");
    $sql = "select * from board order by num desc";
    $result = mysqli_query($con, $sql);
    $total_record = mysqli_num_rows($result); // 전체 글 수

    $scale = 10;
    // 전체 페이지 수($total_page) 계산
    if ($total_record % $scale == 0)
        $total_page = floor($total_record/$scale);
    else
        $total_page = floor($total_record/$scale) + 1;
```

①

②

③

④

[예제 14-3] 글 목록 보기 페이지 `part3/board_list.php`

```
// 표시할 페이지($page)에 따라 $start 계산
```

```
$start = ($page - 1) * $scale;
```

```
$number = $total_record - $start;
```

```
for ($i=$start; $i<$start+$scale && $i < $total_record; $i++)
```

⑤

```
{
```

```
    mysqli_data_seek($result, $i);
```

```
    // 가져올 레코드로 위치(포인터) 이동
```

```
    $row = mysqli_fetch_array($result);
```

⑥

```
    // 하나의 레코드 가져오기
```

```
    $num = $row["num"];
```

```
    $id = $row["id"];
```

```
    $name = $row["name"];
```

```
    $subject = $row["subject"];
```

```
    $regist_day = $row["regist_day"];
```

```
    $hit = $row["hit"];
```

```
    if ($row["file_name"])
```

⑦

[예제 14-3] 글 목록 보기 페이지 `part3/board_list.php`

```
$file_image = "<img src='./img/file.gif'>";
else
    $file_image = " ";
?>

<li>
<span class="col1"> <?=$number?> </span>
<span class="col2">
    <a href="board_view.php?num= <?=$num?> &page= <?=$page?>">
        <?=$subject?> </a> </span>
<span class="col3"> <?=$name?> </span>
<span class="col4"> <?=$file_image?> </span>
<span class="col5"> <?=$regist_day?> </span>
<span class="col6"> <?=$hit?> </span>
</li>
<?php
    $number--;
}
```

⑧

[예제 14-3] 글 목록 보기 페이지 `part3/board_list.php`

```
mysqli_close($con);
?>
</ul>
<ul id="page_num">
<?php
if ($total_page >= 2 && $page >= 2)
{
    $new_page = $page-1;
    echo "<li><a href='message_box.php?mode=$mode&page=$new_page'>
        ◀ 이전</a> </li>";
}
else
    echo "<li>&nbsp;</li>";
// 게시판 목록 하단에 페이지 링크 번호 출력
for ($i = 1; $i <= $total_page; $i++)
{
    if ($page == $i) // 현재 페이지 번호를 링크하지 않음
```


[예제 14-3] 글 목록 보기 페이지 `part3/board_list.php`

```
{
    echo "<li><b> $i </b></li>";
}
else
{
    echo "<li> <a href='board_list.php?mode=$mode&page=$i'> $i </a>
    </li>";
}
}
if ($total_page >= 2 && $page != $total_page)
{
    $new_page = $page+1;
    echo "<li> <a href='board_list.php?mode=$mode&page=$new_page'>
        다음 ▶ </a> </li>";
}
else
    echo "<li>&nbsp;</li>";
```

[예제 14-3] 글 목록 보기 페이지 `part3/board_list.php`

```
?>
    </ul> <!-- page -->
    <ul class="buttons">
        <li> <button onclick="location.href='board_list.php'">목록</button> </li>
        <li>
<?php
    if ($userid) {
?>
        <button onclick="location.href='board_form.php'">글쓰기</button>
<?php
    } else {
?>
        <a href="javascript:alert('로그인 후 이용해 주세요!')">
            <button>글쓰기</button> </a>
```

⑩

[예제 14-3] 글 목록 보기 페이지 `part3/board_list.php`

```
<?php
    }
?>
    </li>
    </ul>
</div> <!-- board_box -->
</section>
<!-- 생략 -->
```

① GET 방식으로 페이지 번호 전달받기

- \$_GET["page"]에 값이 설정되어 있는지 isset() 함수를 이용하여 검사.
- 값이 설정되어 있으면 페이지 번호인 \$page에 \$_GET["page"] 저장, 설정되어 있지 않으면 1을 초깃값으로 저장.

② DB에서 전체 게시글 가져오기

- board 테이블에서 레코드 일련번호인 num을 기준으로 내림차순한 전체 레코드를 가져와 \$result에 저장.
- mysqli_num_rows() 함수를 이용하여 전체 레코드 수(게시글 수)를 \$total_record에 저장.

③ 한 페이지에 표시되는 행의 개수

- \$scale은 글 목록 화면에 표시되는 글의 수 즉, 행의 수를 의미.
- 현재 \$scale 값이 10이므로 한 페이지에 글 10개가 출력.

④ 전체 페이지 수, 시작 레코드 번호, 일련번호 구하기

\$total_page, \$start, \$number는 전체 페이지 수, 시작 레코드 번호, 화면에 표시되는 글의 일련번호를 의미.

⑤ 반복 루프 설명

DB에서 글 목록을 가져오기 위해 반복 루프 설정.

⑥ 글 목록의 항목 가져오기

- mysqli_fetch_array() 함수로 글 목록의 한 행, 즉, 레코드 하나를 가져와 \$row에 저장.
- \$row["num"], \$row["id"], \$row["name"], \$row["subject"], \$row["regist_day"], \$row["hit"]는 레코드 일련번호, 글쓴이 아이디, 글쓴이 이름, 제목, 작성 일시, 조회 수 의미.

⑦ 첨부 파일 존재 시 아이콘 표시

- 첨부 파일 있는 게시글에는 첨부 파일 아이콘 표시.
- \$row["file_name"]은 업로드 파일명을 나타냄.
- 업로드 파일이 존재하면 \$file_image에 HTML 코드인 를 저장하고 그렇지 않으면 공백인 " " 저장.

게시판 > 목록 보기					
번호	제목	글쓴이	첨부	등록일	조회
2	파일 첨부합니다.	루바토		2019-04-11 (03:41)	0
1	안녕하세요.	루바토		2019-04-11 (03:05)	0

1

목록

글쓰기

그림 14-7 글 목록 보기 페이지에 표시된 첨부 파일 아이콘(file.gif)

⑧ 글 목록의 항목 출력

- 글 목록의 각 항목을 화면에 출력.
- \$number는 게시글 번호, \$subject는 제목, \$name은 글쓴이, \$file_image는 첨부 파일 아이콘을 삽입하는 문자열, \$regist_day는 작성 일시, \$hit는 조회 수.

⑨ 페이지 번호 표시

- 글 목록 보기 페이지의 하단에 페이지 번호 출력. 현재 페이지 번호는 굵은 글씨체로 표시, 나머지 페이지 번호에는 해당 페이지로 이동하는 링크 설정.
- 2페이지 이상일 때는 '◀ 이전', '다음 ▶' 표시. 클릭 시 이전 페이지, 다음 페이지를 보여줌.

⑩ <목록>, <글쓰기> 버튼 삽입

- <button> 태그로 <목록> 버튼 삽입. 버튼 클릭 시 글 목록 보기 페이지(board_list.php)로 이동하도록 링크 설정.
- <글쓰기> 버튼은 if문의 조건식에 있는 \$userid가 값을 가질 때, 즉 로그인 상태에서만 글쓰기 페이지(board_form.php)로 이동하도록 설정. \$userid에 값이 없으면 로그인 후 버튼을 클릭하라는 경고 창 띄움.

- [예제 14-3]의 ⑧에서 제목을 나타내는 \$subject에 다음 URL을 링크하여 글 목록 보기 페이지에서 제목을 클릭하면 글 내용을 볼 수 있음.

```
board_view.php?num=<?=$num?>&page=<?=$page?>
```

- 글 내용 보기 페이지인 board_view.php로 이동할 때 레코드 일련번호인 \$num과 페이지 번호인 \$page가 함께 GET 방식으로 전달.

게시판 > 내용 보기

제목 : 파일 첨부합니다. 루바토 | 2019-04-11 (03:48)

▷ 첨부파일 : 사진파일.zip (951 Byte) [저장]

다운로드 후 압축을 풀어 사용해 주세요. ^^

목록 수정 삭제 글쓰기

그림 14-8 글 내용 보기 페이지(board_view.php)

[예제 14-4] 글 내용 보기 페이지 `part3/board_view.php`

```
<!-- 생략 -->
<section>
<div id="main_img_bar">
    
</div>
<div id="board_box">
    <h3 class="title">
        게시판 > 내용 보기
    </h3>
<?php
    $num = $_GET["num"];
    $page = $_GET["page"];

    $con = mysqli_connect("localhost", "user1", "12345", "sample");
    $sql = "select * from board where num=$num";
    $result = mysqli_query($con, $sql);
```

①

②

[예제 14-4] 글 내용 보기 페이지 `part3/board_view.php`

```
$row = mysqli_fetch_array($result);  
$id = $row["id"];  
$name = $row["name"];  
$regist_day = $row["regist_day"];  
$subject = $row["subject"];  
$content = $row["content"];  
$file_name = $row["file_name"];  
$file_type = $row["file_type"];  
$file_copied = $row["file_copied"];  
$hit = $row["hit"];
```

③

```
$content = str_replace(" ", "&nbsp;", $content);  
$content = str_replace("\n", "<br>", $content);
```

④

```
$new_hit = $hit + 1;  
$sql = "update board set hit=$new_hit where num=$num";  
mysqli_query($con, $sql);
```

⑤

```
?>
```

part3/board_view.php

7

[예제 14-4] 글 내용 보기 페이지 `part3/board_view.php`

```
    }  
?>  
    <?=$content?>  
    </li>  
    </ul>  
    <ul class="buttons">  
    <li> <button onclick="location.href='board_list.php?  
        page=<?=$page?>'>목록 </button> </li>  
    <li> <button onclick="location.href='board_modify_form.php?  
        num=<?=$num?>&page=<?=$page?>'>수정 </button> </li>  
    <li> <button onclick="location.href='board_delete.php?  
        num=<?=$num?>&page=<?=$page?>'>삭제 </button> </li>  
    <li> <button onclick="location.href='board_form.php'">  
        글쓰기 </button> </li>  
    </ul>  
</div> <!-- board_box -->  
</section> <!-- 생략 -->
```

⑧

⑨

① 레코드 일련번호와 페이지 번호 전달받기

글 목록 보기 페이지(board_list.php)로부터 레코드 일련번호와 페이지 번호를 각각 `$_GET["num"]`, `$_GET["page"]`로 전달받음.

② DB에서 해당 글 검색

레코드 번호 `$num`을 가진 레코드를 검색하여 `$result`에 저장.

③ `$result`에서 데이터 가져오기

`$row["id"]` → 글쓴이의 아이디

`$row["regist_day"]` → 작성 일시

`$row["content"]` → 내용

`$row["file_type"]` → 첨부 파일 형식

`$row["hit"]` → 조회 수

`$row["name"]` → 이름

`$row["subject"]` → 제목

`$row["file_name"]` → 첨부 파일명

`$row["file_copied"]` → 서버에 저장된 첨부 파일명

④ 공백과 줄 바꿈 코드 변경

공백(" ")을 웹 페이지에 표시하기 위해 `str_replace()` 함수로 HTML 특수 기호인 ` `로 변경.
같은 방식으로 줄 바꿈 코드인 `\n`은 HTML의 줄 바꿈 태그인 `
`로 변경.

⑤ 조회 수 값 증가와 DB 업데이트

조회 수를 나타내는 `$hit` 값을 1 증가시키고 MySQL의 `update` 명령을 이용하여 `board` 테이블의 `hit` 필드를 업데이트.

⑥ 제목, 글쓴이 이름, 작성 일시 출력

게시글 제목(`$subject`), 글쓴이 이름(`$name`), 작성 일시(`$regist_day`) 출력.

⑦ 첨부 파일 정보 출력

\$file_name에 값이 설정되어 있으면 첨부 파일 존재. 게시글 제목 아래에 파일명(\$file_name)과 파일 크기(\$file_size) 표시, 파일을 다운로드할 수 있게 <저장> 버튼에 board_download.php 링크.

⑧ 글 내용 출력

첨부 파일 정보 다음 줄에 글 내용인 \$content 출력.

⑨ 버튼 삽입

화면 하단에 <목록>, <수정>, <삭제>, <글쓰기> 버튼 삽입하고 해당 파일로 이동할 수 있는 링크 설정.

- 글 내용 보기 페이지(board_view.php)에서 첨부 파일명 옆에 있는 <저장>을 클릭하면 파일 다운로드 가능. 이 버튼은 [예제 14-4]의 ⑦에서 다음 URL로 링크되어 있음.

`board_download.php?num=$num&real_name=$real_name&file_name=$file_name&file_type=$file_type`

- 게시판 사용자가 첨부 파일을 다운로드할 수 있게 하는 board_download.php에서는 레코드 일련번호(\$num), 서버 저장 파일명(\$real_name), 파일명(\$file_name), 파일 형식(\$file_type)을 글 내용 보기 페이지로부터 GET 방식으로 전달받음.

[예제 14-5] 첨부 파일 다운로드 페이지

part3/board_download.php

```
<?php
    $real_name = $_GET["real_name"];
    $file_name = $_GET["file_name"];
    $file_type = $_GET["file_type"];
    $file_path = "./data/".$real_name;

    $ie = preg_match('~MSIE|Internet Explorer~i', $_SERVER['HTTP_USER_
        AGENT']) || (strpos($_SERVER['HTTP_USER_AGENT'], 'Trident/7.0')
        !== false && strpos($_SERVER['HTTP_USER_AGENT'], 'rv:11.0') !==
        false);

    // IE인 경우 한글 파일명이 깨지는 것을 방지하기 위한 코드
    if ($ie){
        $file_name = iconv('utf-8', 'euc-kr', $file_name);
    }
```


① 첨부 파일 정보 전달받기

글 내용 보기 페이지(예제 14-4)에서 전달받은 `$_GET["real_name"]`, `$_GET["file_name"]`, `$_GET["file_type"]`은 각각 서버에 저장된 파일명, 파일명, 파일 형식을 의미.

② 브라우저가 인터넷 익스플로러인지 판단

- 인터넷 익스플로러에서 첨부 파일을 다운로드 할 때 파일명이 한글이면 한글이 깨지는 현상이 발생할 수 있음. 이를 대비해 사용자 브라우저가 인터넷 익스플로러인지 검사 필요.
- `$_SERVER ['HTTP_USER_AGENT']`는 사용자 브라우저 이름.
- `preg_match()` 함수로 브라우저 이름에 MSIE나 Internet Explorer가 포함되어 있는지 검사.
- `strpos()` 함수로 브라우저 이름에 문자열 Trident/7.0 또는 rv:11.0이 포함되어 있는지 검사. 인터넷 익스플로러는 브라우저 이름에 이러한 문자열이 포함되어 있기 때문.

preg_match() 함수	
형식	<code>preg_match(pattern, string)</code>
기능	문자열에서 해당 패턴이 존재하는지 검사한다.
인자	pattern: 찾는 패턴 string: 검사 대상 문자열
반환 값	해당 패턴이 문자열에 존재하는 경우: true 해당 패턴이 문자열에 존재하지 않는 경우: false

③ 인터넷 익스플로러인 경우 파일명의 문자셋 변경

- 인터넷 익스플로러에서 한글이 깨지는 현상을 방지하기 위해 파일명의 문자셋을 utf-8에서 euc-kr로 변경.
- `iconv('utf-8', 'euc-kr', $file_name)`은 `$file_name`의 문자셋을 utf-8에서 euc-kr로 변경하라는 의미.

iconv() 함수	
형식	<code>iconv(in_charset, out_charset, string)</code>
기능	문자열의 문자셋을 변경한다.
인자	in_charset: 기존 문자셋 out_charset: 변경할 문자셋 string: 대상 문자열
반환 값	성공: 문자셋이 변경된 문자열 반환 실패: false

④ 다운로드 파일의 존재 확인

`file_exists($file_path)`는 다운로드하려는 파일인 `$file_path`가 존재하는지 검사.

file_exists() 함수	
형식	<code>file_exists(filename)</code>
기능	해당 파일이 존재하는지 검사한다.
인자	filename: 경로를 포함한 파일명
반환 값	해당 파일 존재: true, 해당 파일 부재: false

⑤ 파일 열기

\$file_path 파일 열기. rb는 파일을 읽고 쓸 수 있는 모드로 열라는 의미.

fopen() 함수	
형식	fopen(filename, mode)
기능	파일을 열어서 그 파일이 지시하는 파일 포인터를 얻는다. mode에 따라 파일 모드가 결정된다.
인자	filename: 경로를 포함한 파일명 mode: r(읽기 모드), w(쓰기 모드), b(이진 모드)
반환 값	성공: 열린 파일의 파일 포인터 반환 실패: false

⑥ 파일 정보 알려주기

다운로드할 파일 정보를 Header() 함수로 클라이언트 브라우저에 알려줌.

Header() 함수	
형식	Header(string)
기능	HTTP 헤더를 클라이언트의 브라우저에 보낸다. 예를 들어 Header('Content-type: application/pdf')는 전송하려는 파일의 형식이 pdf라는 것을 알려준다. Header('Content-Disposition: attachment; filename="downloaded.pdf"')는 전송하려는 파일명이 downloaded.pdf라는 것을 클라이언트의 브라우저에 알려준다.
인자	string: HTTP의 헤더 정보 문자열
반환 값	없음

⑦ 파일 전송 및 파일 포인터 닫기

파일 포인터인 `$fp`에 저장된 파일 데이터를 `fpassthru()` 함수로 출력 버퍼에 저장하면 파일이 다운로드 됨. 다운로드가 완료되면 `fclose()` 함수로 파일을 닫음.

fpassthru() 함수	
형식	fpassthru(fp)
기능	현재 파일 포인터가 지시하는 위치부터 끝까지 파일을 읽어 출력 버퍼에 저장한다. 이렇게 함으로써 사용자의 컴퓨터로 파일이 전송된다.
인자	fp: 파일 포인터
반환 값	성공: 전송된 문자의 개수 반환 실패: false

fclose() 함수	
형식	fclose(fp)
기능	파일 포인터가 지시하는 파일을 닫는다.
인자	fp: 파일 포인터
반환 값	성공: true, 실패: false

- 글 내용 보기 페이지(board_view.php) 하단의 <수정>을 클릭하면 [예제 14-4]의 ⑨에 의해 글 수정 페이지(board_modify_form.php)로 이동.

```
board_modify_form.php?num=<?=$num?>&page=<?=$page?>
```

- 글 수정 페이지로 이동할 때 해당 레코드의 일련번호인 \$num과 페이지 번호인 \$page를 함께 전달.

1. 글 수정 폼 양식

- 브라우저 화면에 나타나는 글 수정 페이지의 폼 양식.

게시판 > 글쓰기

이름 :

루바토

제목 :

파일 첨부합니다.

내용 :

다운로드 후 압축을 풀어 사용해 주세요. ^^

첨부 파일 :

사진파일.zip

수정하기

목록

그림 14-9 글 수정 폼 양식

[예제 14-6] 글 수정 폼 양식 `part3/board_modify_form.php`

```
<!-- 생략 -->
<script>
function check_input() {
    if (!document.board_form.subject.value)
    {
        alert("제목을 입력하세요!");
        document.board_form.subject.focus();
        return;
    }
    if (!document.board_form.content.value)
    {
        alert("내용을 입력하세요!");
        document.board_form.content.focus();
        return;
    }
}
```

[예제 14-6] 글 수정 폼 양식 `part3/board_modify_form.php`

```
        document.board_form.submit();
    }
</script>
<!-- 생략 -->
<section>
<div id="main_img_bar">
    
</div>
<div id="board_box">
    <h3 id="board_title">
        게시판 > 글쓰기
    </h3>
<?php
    $num = $_GET["num"];
    $page = $_GET["page"];
```

[예제 14-6] 글 수정 폼 양식 `part3/board_modify_form.php`

```
$con = mysqli_connect("localhost", "user1", "12345", "sample"); ②
$sql = "select * from board where num=$num";
$result = mysqli_query($con, $sql);
$row = mysqli_fetch_array($result);
$name = $row["name"];
$subject = $row["subject"];
$content = $row["content"];
$file_name = $row["file_name"];
?>

<form name="board_form" method="post" ③
    action="board_modify.php?num=<?=$num?>&page=<?=$page?>"
    enctype="multipart/form-data">
    <ul id="board_form">
    <li>
        <span class="col1">이름 : </span> ④
```

[예제 14-6] 글 수정 폼 양식 `part3/board_modify_form.php`

```
<span class="col2"> <?=$name?> </span>
</li>
<li>
  <span class="col1">제목 : </span>
  <span class="col2"> <input name="subject" type="text"
    value="<?=$subject?>" </span>
</li>
<li id="text_area">
  <span class="col1">내용 : </span>
  <span class="col2">
    <textarea name="content"> <?=$content?> </textarea>
  </span>
</li>
<li>
  <span class="col1">첨부 파일 : </span>
```

⑤

⑥

⑦

[예제 14-6] 글 수정 폼 양식 `part3/board_modify_form.php`

```
<span class="col2"> <?=$file_name?> </span>
</li>
</ul>
<ul class="buttons">
<li> <button type="button" onclick="check_input()">수정하기 ⑧
    </button> </li>
<li> <button type="button" onclick="location.href='board_list.php'">목록
    </button> </li>
</ul>
</form>
</div> <!-- board_box -->
</section>
<!-- 생략 -->
```


① 레코드 번호와 페이지 번호 전달받기

글 내용 보기 페이지(board_view.php)로부터 레코드 번호와 페이지 번호를 \$_GET["num"], \$_GET["page"]로 전달받음.

② DB에서 글 정보 가져오기

board 테이블에서 글쓴이, 제목, 내용, 파일명 정보를 가져와 \$name, \$subject, \$content, \$file_name에 저장.

③ <form> 태그 설정

<수정하기> 버튼을 클릭했을 때 DB에 저장된 글을 수정할 수 있도록 action 속성을 board_modify.php?num=<?=\$num?>&page=<?=\$page?>로 지정. board_modify.php 파일에서 글 수정 폼 양식의 데이터를 전달받아 DB를 업데이트한다는 의미.

④ 이름 출력

'이름 : ' 옆에 글쓴이 이름인 \$name 출력.

⑤ 입력 창에 글 제목 삽입

'제목 : ' 옆의 입력 창에 수정 전 제목인 \$subject 삽입.

⑥ 다중 입력 창에 글 내용 삽입

'내용 : ' 옆의 다중 입력 창에 수정 전 내용인 \$content 삽입.

⑦ 첨부 파일명 출력

'첨부 파일 : ' 옆에 첨부 파일명인 \$file_name 출력.

⑧ <수정하기> 버튼 삽입

- <수정하기> 버튼을 클릭하면 check_input() 호출.
- 폼 양식에 데이터가 입력되어 있는지 검사. 이상이 없으면 ③의 action 속성에 설정된 board_modify.php로 이동하여 폼 양식에 입력된 데이터로 DB 업데이트.

[예제 14-7] 수정 내용으로 DB를 업데이트하는 페이지 [part3/board_modify.php](#)

```
<?php
    $num = $_GET["num"];
    $page = $_GET["page"];
    $subject = $_POST["subject"];
    $content = $_POST["content"];
    $con = mysqli_connect("localhost", "user1", "12345", "sample");
    $sql = "update board set subject='$subject', content='$content' ";
    $sql .= " where num=$num";
    mysqli_query($con, $sql);
    mysqli_close($con);

    echo "
    <script>
        location.href = 'board_list.php?page=$page';
    </script>
    ";
?>
```

①

②

③

① 레코드 번호, 페이지 번호, 제목, 내용 전달받기

글 수정 페이지(board_modify_form.php)로부터 레코드 번호, 페이지 번호, 수정된 글 제목과 내용을 \$_GET["num"], \$_GET["page"], \$_POST["subject"], \$_POST["content"]로 전달받음.

② 수정된 글 제목과 내용으로 DB 업데이트

수정된 글 제목(\$subject)과 내용(\$content)으로 board 테이블의 해당 필드 업데이트.

③ 글 목록 보기 페이지로 이동

글 목록 보기 페이지인 board_list.php 파일로 이동. 페이지 번호를 GET 방식으로 전달하여 글 목록 보기 페이지에서 수정된 글 제목을 볼 수 있음.

3. 글 삭제

- 글 내용 보기 페이지(board_view.php) 하단의 <삭제>를 클릭하면 [예제 14-4]의 ⑨에 의해 글 삭제하기 페이지(board_delete.php)로 이동.

```
board_delete.php?num=<?=$num?>&page=<?=$page?>
```

- 글 삭제하기 페이지에서는 레코드 일련번호인 \$num과 페이지 번호인 \$page를 글 내용 보기 페이지로부터 전달받음.
- 레코드 일련번호로 해당 글을 삭제할 수 있으며 글 삭제 후 글 목록 보기 페이지로 돌아갈 때는 페이지 번호를 전달하여 해당 페이지의 글 목록을 볼 수 있음.

[예제 14-8] 글 삭제하기 페이지 `part3/board_delete.php`

```
<?php
    $num = $_GET["num"];
    $page = $_GET["page"];

    $con = mysqli_connect("localhost", "user1", "12345", "sample");
    $sql = "select * from board where num = $num";
    $result = mysqli_query($con, $sql);
    $row = mysqli_fetch_array($result);
    $copied_name = $row["file_copied"];

    if ($copied_name)
    {
        $file_path = "./data/".$copied_name;
        unlink($file_path);
    }
```

[예제 14-8] 글 삭제하기 페이지 `part3/board_delete.php`

```
$sql = "delete from board where num = $num";  
mysqli_query($con, $sql);  
mysqli_close($con);
```

④

```
echo "  
<script>  
    location.href = 'board_list.php?page=$page';  
</script>  
";
```

⑤

```
?>
```

① 레코드 번호와 페이지 번호 전달받기

글 내용 보기 페이지(board_view.php)로부터 GET 방식으로 전달받은 레코드 번호(\$_GET["num"])와 페이지 번호(\$_GET["page"])를 \$num, \$page에 저장.

② 삭제할 첨부 파일명 가져오기

- 첨부 파일을 삭제하기 위해 DB에 저장된 첨부 파일명을 가져옴.
- \$num을 이용하여 해당 레코드 정보를 가져와 \$result에 저장.
- mysqli_fetch_array() 함수로 첨부 파일명인 \$row["file_copied"]를 구하여 \$copied_name에 저장.

③ 첨부 파일 삭제

\$copied_name에 값이 설정되어 있으면 첨부 파일이 존재한다는 의미. unlink() 함수로 서버에 저장된 첨부 파일 삭제.

④ DB에서 해당 레코드 삭제

레코드 번호 \$num을 이용하여 board 테이블의 해당 레코드를 삭제.

⑤ 페이지 이동

글 목록 보기 페이지(board_list.php)로 이동. 페이지 번호인 \$page를 전달하므로 해당 페이지의 글 목록을 화면에 출력할 수 있음.