

2021학년도

데이터베이스 실습 최종보고서

제목: 소비생각

개발자: 19 윤지연

19 한지은

20 이영민

목차

1. 프로그램 설명 -
2. 시스템 설계 -
3. ER Diagram -
4. 팀원 역할 -
5. 개발일정 -
6. 개발환경 -
7. 개발결과 -
8. 자체평가 -

1. 프로그램 설명

사용자가 자신의 지출을 날짜, 시간, 분류, 금액, 내용, 결제 방법 등의 상세정보를 토대로 입력하고 일별, 분류별(식비, 생활용품, 교통비, 의류, 저축 등)로 사용 내역을 확인할 수 있도록 하여 더 체계적으로 지출 할 수 있도록 돕는 웹앱이다.

2. 시스템 설계

어떤 식으로 프로그램이 작동하는지 최대한 자세히 설명

(1) 초기화면

- 로그인과 회원가입 버튼을 누를 수 있다.
- 로그인 버튼을 누르면 (2) 로그인 화면으로 간다.
- 회원가입 버튼을 누르면 (3) 회원가입 화면으로 간다.

(2) 로그인 화면

- ID/PW입력: 사용자로부터 ID와 PW를 입력받는다.
- DB 확인: 사용자가 입력한 ID와 PW가 DB에 저장된 값과 일치하는지 확인한다.
- 만약 일치하지 않는다면 “로그인에 실패하였습니다.”라는 문구가 뜨고, 사용자가 입력한 ID와 비밀번호를 지우고 다시 입력받는다.
- 만약 일치한다면 (4) 메인화면으로 간다.

(3) 회원가입 화면

- ID입력 : 사용자로부터 ID를 입력받는다
- ID 중복확인 : DB에 저장되어 있는 기존 ID와 중복되는지 확인한다.
- 중복된다면 “아이디가 중복되었습니다다”를 메세지 박스로 출력한다.
- 중복되지 않다면 “회원가입이 완료되었습니다”를 메세지 박스로 출력한다.
- PW 입력 : 사용자로부터 PW를 입력받는다.
- DB 저장 : 입력받은 ID와 PW를 DB에 있는 user테이블에 저장한다.

(4) 메인화면

(4-1) 네비게이션바

- 화면 상단에 고정되어 있다.
- “일별 지출 내역 확인”을 누르면 **(4-2)** 화면으로 넘어 간다.
- “분류별 지출 내역 확인”을 누르면 **(4-3)** 화면으로 넘어 간다.
- “지출 내역 작성”을 누르면 **(5)** 화면으로 넘어 간다.

(4-2) 일별 지출내역 화면

- 일별 지출내역을 DB에서 받아 리스트에 입력하여 화면에 출력한다.
- 해당 리스트 클릭 시 수정 및 삭제가 가능하다. **(6)** 수정 및 삭제 화면으로 간다.

(4-3) 분류별(카테고리별) 총 지출 화면

- 분류별로 해당되는 지출을 DB에서 받아 총합을 계산하여 화면에 출력한다.
- 해당 리스트 버튼 클릭시 해당 분류에 따라 다른 내역이 출력되는 **(4-4)** 화면으로 넘어간다

(4-4) 분류별(카테고리별) 지출내역 화면

- 선택된 카테고리의 지출내역을 DB에서 받아 화면에 최근시간순으로 출력한다.

(5) 지출 내용 작성화면

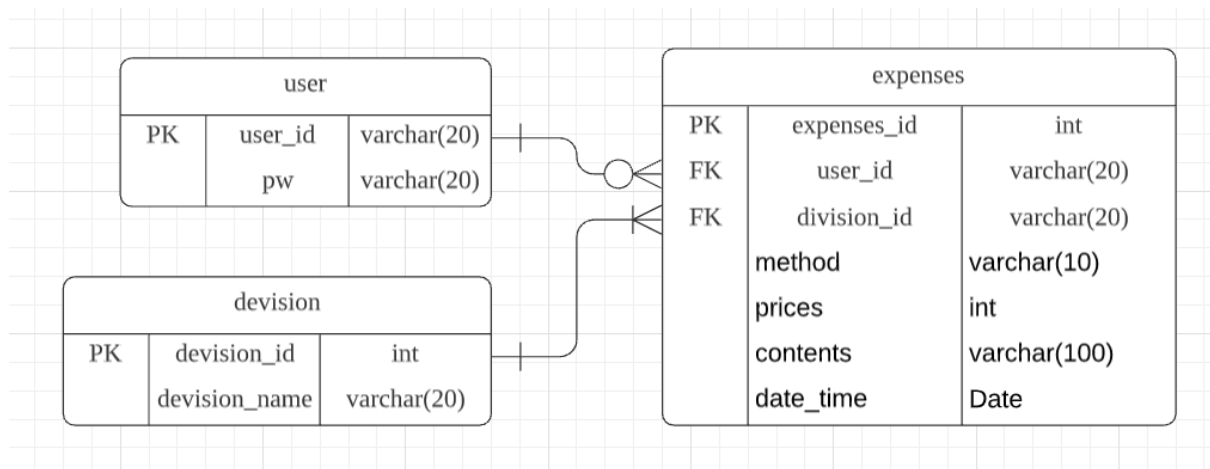
- 사용자로 부터 날짜 및 시간, 금액, 내용 입력을 받는다.
- 사용자로 부터 자산, 분류의 아이템 중 선택한 내용을 받는다.
- 위 내용을 DB에 있는 **exepenses** 테이블에 저장한다.

(6) 수정 및 삭제 화면

- 수정

- 수정버튼을 누르면 다음의 내용을 입력받는다.
- 사용자로 부터 날짜 및 시간, 금액, 내용을 입력받고, 결제 방법, 분류는 드롭박스, 라디오 버튼을 통해 선택한 내용을 받는다.
- 위 내용을 DB에 있는 **expenses** 테이블에서 수정한다.
- 삭제
- 해당 지출내역에 대한 내용을 DB에 있는 **expenses** 테이블에서 삭제한다.

3. ER Diagram



4. 팀원역할

팀원 기여도 평가입니다. 해당 팀원이 구체적으로 어떤 역할을 했는지 기록.

윤지연 : 지출내용 작성(수정, 삭제) 화면 구현, DB 연동

한지은 : 지출목록 확인 화면 구현(일별, 분류별, 분류별 총지출), DB구축 및 연동

이영민 : 로그인 및 회원가입 화면 구현, 서버구축, DB 구축 및 연동

5. 개발일정

5월 13일

- 제안서 작성 및 설계

5월 29일

- 각 화면 프론트 개발 및 DB연동

6월 1일

- 개발 마무리 및 오류 수정을 위한 통합 테스트
- 보고서 작성 및 발표 준비

6. 개발환경

IDE: visual studio code

서버활용유무: Apache 웹서버 활용

사용언어: html, php, javascript, css

Apache웹 서버, mySQL, php 연동을 위해 WAMP server사용

7. 개발결과

- 코드 중 일부는 생략되어 있으므로 전체 코드는 부록 링크 참고
- DB 접근 방법은 모두 아래의 방법 사용

DB 접근 방법

mysqli_connect()함수를 사용해 접근 후

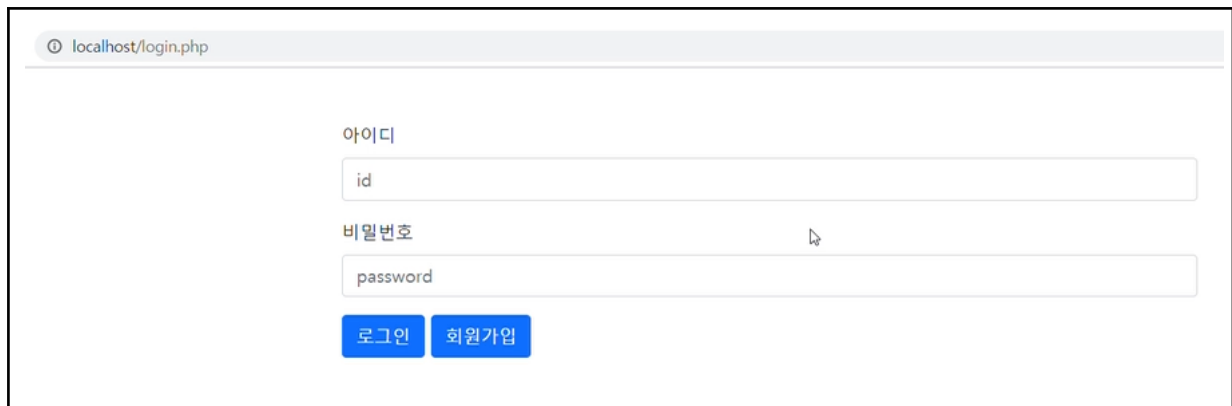
```
$conn = mysqli_connect("localhost", "root", "091405", "think", 3306);
```

동일한 php 파일에서 DB에 한글도 저장해주기 위해 mysqli_set_charset()함수를 사용

```
mysqli_set_charset($conn, 'utf8');
```

(1) 초기화면: 이영민

결과 캡처 화면



설명(login.php 화면)
 로그인 버튼을 누르면 **post**를 통해 아이디와 비밀번호를 전달 **loginprocess.php**로 전달하면서 **loginprocess.php**를 실행함.
 회원가입을 누르면 **form** 태그 와 **input(button)**태그 및 **summit**을 통해 **signup.php**가 실행된다.

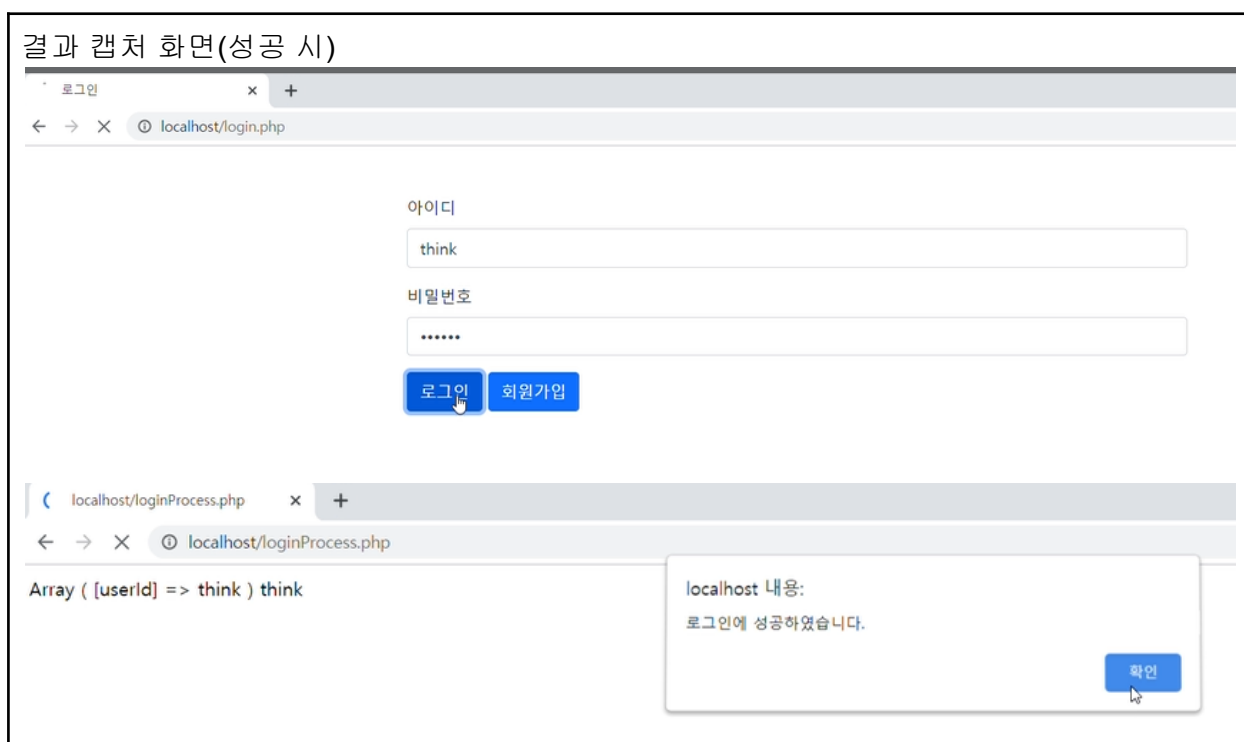
스타일

```
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/5.0.0-alpha2/css/boo
tstrap.min.css"
integrity="sha384-DhY6onE6f3zzKbjUPRc2hOzGAdEf4/Dz+WJwBvEYL/lkkIsI3ihuf
q9hk9K4lVoK" crossorigin="anonymous">
```

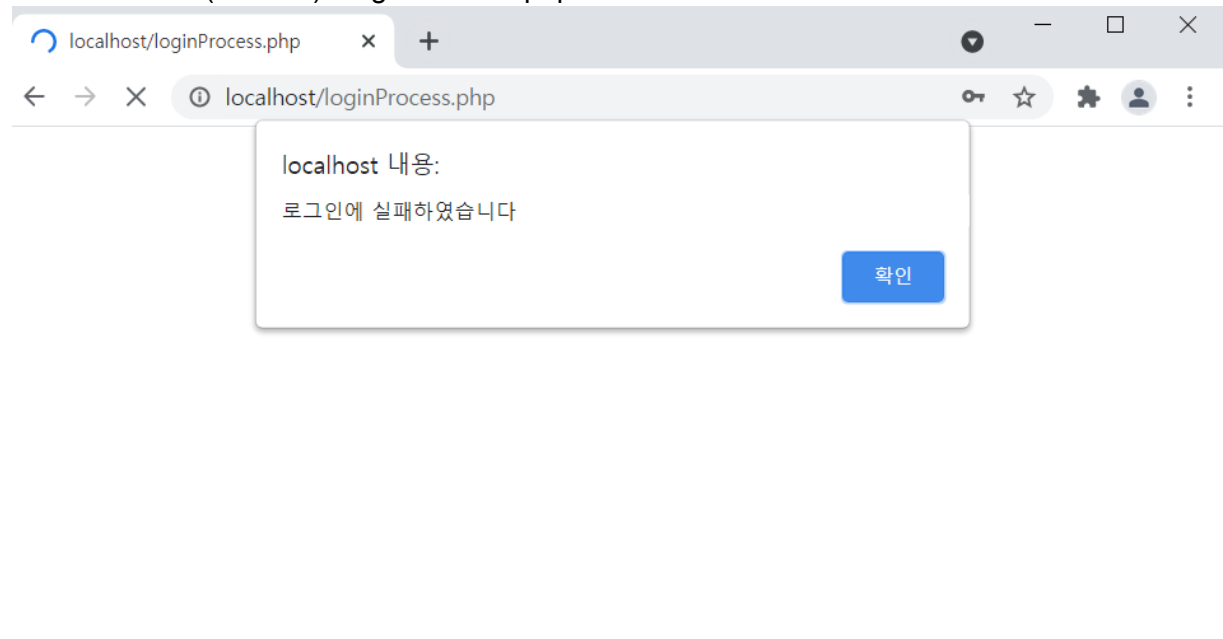
위 링크를 통해 버튼 텍스트 스타일을 받아온다. 이 부분은 회원가입 화면도 동일하다.

(2) 로그인 화면: 이영민

결과 캡처 화면(성공 시)



결과 캡처 화면(실패 시) - loginProcess.php



설명(loginprocess.php)

login.php에서 POST를 통해 아이디와 비밀번호를 받아서 DB에 해당 아이디와 비밀번호가 일치하는지 검색(Select)후 일치한다면 로그인 성공 메세지 박스와 함께 메인화면으로 넘어가고 이때 user 정보 중 userid를 session으로 메인화면으로 보낸다. 일치하지 않는다면 다시 로그인화면(login.php)로 돌아간다.

DB 활용한 내용

✓ 0 - 3행 표시 중 (합계 4, 질의 실행시간 0.0008 초.)

`SELECT * FROM `user``

☐ 프로파일링 [\[인라인 편집\]](#) [\[수정\]](#) [\[SQL 해석\]](#) [\[PHP 코드 보기\]](#)

☐ 모두 보기 | 행 갯수: 25 ▼ 행 필터링: 현재 테이블 검색 Sort by key: 없음 ▼

+ 옵션

← T →

	user_id	pw
<input type="checkbox"/>	lm	\$2y\$10\$XopToJrchOJGUEfsmXfN6ebMRBskrKdfVtYcftMms3J...
<input type="checkbox"/>	ym	\$2y\$10\$LDPs8L7B6Cs0Cp0jqgPiXuNH.IDWffUffBo4bUf0pxO...
<input type="checkbox"/>	dudals	\$2y\$10\$NxfMSsbF.ZuuR3bScI2mZuSPdTgXsAl9bcwFolZUsN4...
<input type="checkbox"/>	think	\$2y\$10\$JUCN4ab78t4c0qMBUZcB.GIQLVWfzt0gei76z9JsRy...

위와 같이 user테이블의 user_id와 pw를 DB에서 select를 통해 비교하는데 비밀번호는 회원가입시 보안을위해 hash를 사용해 저장되어 있다. pw는 password_verify함수를 통해 hash된 user테이블의 pw와 사용자가 입력한 pw를 비교한다. 일치한다면 세션을 실행해 user id를 세션에 저장하고 로그인 완료 메세지 박스와 함께 메인화면 해당 php를 실행한다. 아닐경우 로그인실패 메세지 박스와 함께 로그인화면으로 돌아감

코드(일부)

```
$sql = "SELECT * FROM user WHERE `user_id` = '$userid'";  
$result = mysqli_query($conn, $sql) or die(mysqli_error($conn));;
```



```

$row = mysqli_fetch_array($result) or die(mysqli_error($conn));
$hashedPassword = $row['pw'];
$row['user_id'];

// echo $row['id'];
// DB 정보를 가져왔으니
// 비밀번호 검증 로직을 실행하면 된다.
$passwordResult = password_verify($pw, $hashedPassword);

if ($passwordResult === true) {
    // 로그인 성공
    // 세션에 id 저장
    session_start();
    $_SESSION['userId'] = $row['user_id'];
    print_r($_SESSION);
    echo $_SESSION['userId'];
?>

<script>

    alert("로그인에 성공하였습니다.")
    location.href = "daily.php";
</script>
<?php
} else {
    // 로그인 실패
?>

<script>
    alert("로그인에 실패하였습니다");
    location.href = "login.php";
</script>
<?php
}
?>

```

(3) 회원가입 화면: 이영민

결과 캡처 화면 (성공 시)

회원가입

아이디

think

비밀번호

비밀번호 체크

회원가입

DB에 추가된 내용 캡처 화면(user 테이블)

phpMyAdmin

현재 서버: MySQL

최근 즐겨찾기

새로운

- heart
- information_schema
- mysql
- performance_schema
- sys
- think
 - New
 - division
 - expenses
 - user

서버: MySQL:3306 > 데이터베이스: think > 테이블: user

보기 구조 SQL 검색 삽입 내보내기 가져오기 권한 더보기

0 - 3행 표시 중 (합계 4, 질의 실행시간 0.0005 초.)

SELECT * FROM `user`

☐ 프로파일링 [인라인 편집] [수정] [SQL 해석] [PHP 코드 보기] [다시 보기]

☐ 모두 보기 | 행 갯수: 25 | 행 필터링: 현재 테이블 검색 | Sort by key: 없음

+ 옵션

	user_id	pw
<input type="checkbox"/> 수정 <input type="checkbox"/> 복사 <input type="checkbox"/> 삭제	lm	\$2y\$10\$XopToJrchOJGUefsmXfN6ebMRBskrKdfVtYcftMms3J...
<input type="checkbox"/> 수정 <input type="checkbox"/> 복사 <input type="checkbox"/> 삭제	ym	\$2y\$10\$LDPs8L7B6Cs0Cp0jqgPiXuNHjIDWfUFfBo4bUf0pxO...
<input type="checkbox"/> 수정 <input type="checkbox"/> 복사 <input type="checkbox"/> 삭제	dudals	\$2y\$10\$NxfMsSbfZuuR3bScl2mZuSPdTgXsAI9bcwFolZUsN4...

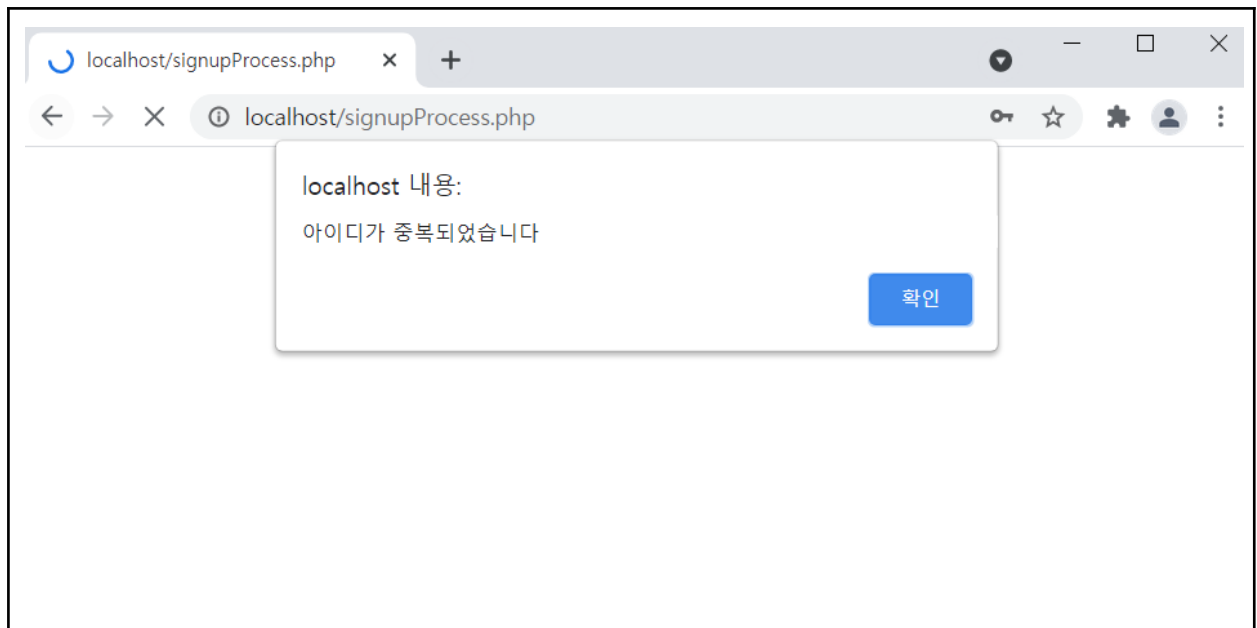
☐ 모두 체크 | 선택한 것을: ☐ 수정 ☐ 복사 ☐ 삭제 ☐ 내보내기

☐ 모두 보기 | 행 갯수: 25 | 행 필터링: 현재 테이블 검색 | Sort by key: 없음

쿼리 결과 처리방법

☐ 인쇄 ☐ 클립보드에 복사하기 ☐ 내보내기 ☐ 차트 표시 ☐ 뷰 생성

결과 캡처 화면(실패 시)- signupProcess.php



설명(signup.php)

회원가입 버튼을 누르면 비밀번호와 비밀번호 확인이 똑같은 문자열인지 확인후 맞으면 해당 id 와 비밀번호를 POST 형식으로 **signupProcess.php**를 실행한다.

설명(signupProcess.php)

POST를 통해 id, pw를 받아 insert를 실행한다. 이과정에서 에러가 발생하면 id가 pk임으로 id가 중복되었다고 판단하고 ID가 중복되었다는 메세지와 함께 **signup.php**로 돌아간다, 만약 제대로 되었다고 회원가입 완료 메세지와 함께 **login.php**를 실행한다.

DB 활용한 내용

비밀번호의 암호화를 위해 POST를 받은 비밀번호를 password_hash를 통해 암호화하여 그 값과 해당 id를 DB user 테이블에 insert 한다.

코드

```
<?php
$conn = mysqli_connect("localhost", "root", "091405", "think", 3306);
$hashedPassword = password_hash($_POST['password'], PASSWORD_DEFAULT);
//echo $hashedPassword;
$sql = "
    INSERT INTO user
    (user_id, pw)
    VALUES('".$_POST['email']."', '{$hashedPassword}'
    )";
$result = mysqli_query($conn, $sql);
//echo $sql;
if ($result === false) {
    ?>
    <script>
        alert("아이디가 중복되었습니다");
        location.href = "signup.php";
    </script>
}
```

```

</script>
<?php
} else {
?>

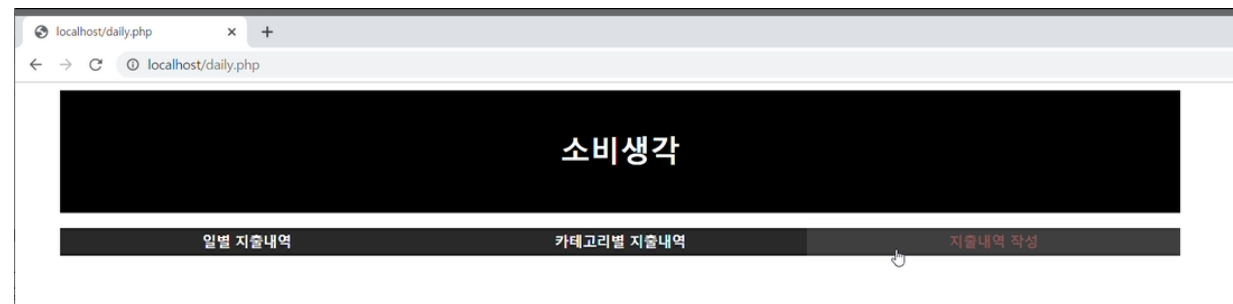
<script>
    alert("회원가입이 완료되었습니다");
    location.href = "login.php";
</script>
<?php
}
?>

```

(4) 메인 화면

(4-1) 네비게이션바: 한지은

결과 캡처 화면



설명

(아래의 코드는 daily.php, category.php, category_detail.php 세 파일의 <body> 안에 존재)

```

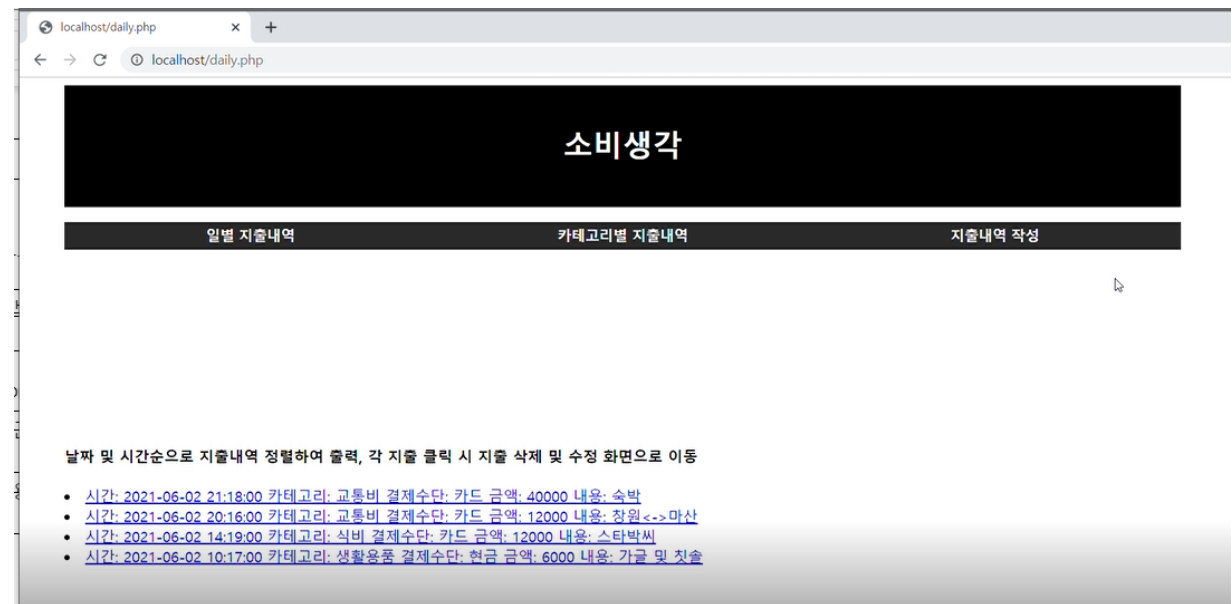
<nav id="topMenu">
<ul>
    <li><a class="menuLink" href="daily.php?userid=<?php echo $userid;?>">일별 지출내역</a></li>
    <li><a class="menuLink" href="category.php?userid=<?php echo $userid;?>">카테고리별 지출내역</a></li>
    <li><a class="menuLink" href="expenditureInput.php?userid=<?php echo $userid;?>">지출내역 작성</a></li>
</ul>
</nav>

```

각 메뉴를 클릭 시 링크를 통해 일별 지출내역 화면, 분류별 총 지출 화면, 분류별 지출내역 화면 으로 이동한다. 스타일은 style.css에서 설정한다.

(4-2) 일별 지출내역 화면: 한지은

결과 캡처 화면



설명

daily.php 파일에서 php문법을 통해 로그인에 성공한 user_id값을 php문법을 사용해 받아오고,

```
$userid = $_SESSION['userId'];
```

DB에서 받아온 지출내용 및 목록을 전체 출력한다.

```
if (mysqli_num_rows($result) > 0) {
    while($row = mysqli_fetch_assoc($result)) {
        //링크 이동과 함께 선택한 지출의 expenses_id를 넘겨준다.
        echo "<li> <a href = \"expenditurechange.php?id={$row['expenses_id']}
            &userid={$userid}\">";
        //지출내용 출력
        echo "시간: " . $row["date_time"] . " 카테고리: " .
            $row["division_name"] . " 결제수단: " . $row["method"] .
            " 금액: " . $row["prices"] . " 내용: " . $row["contents"];
    }
}
```

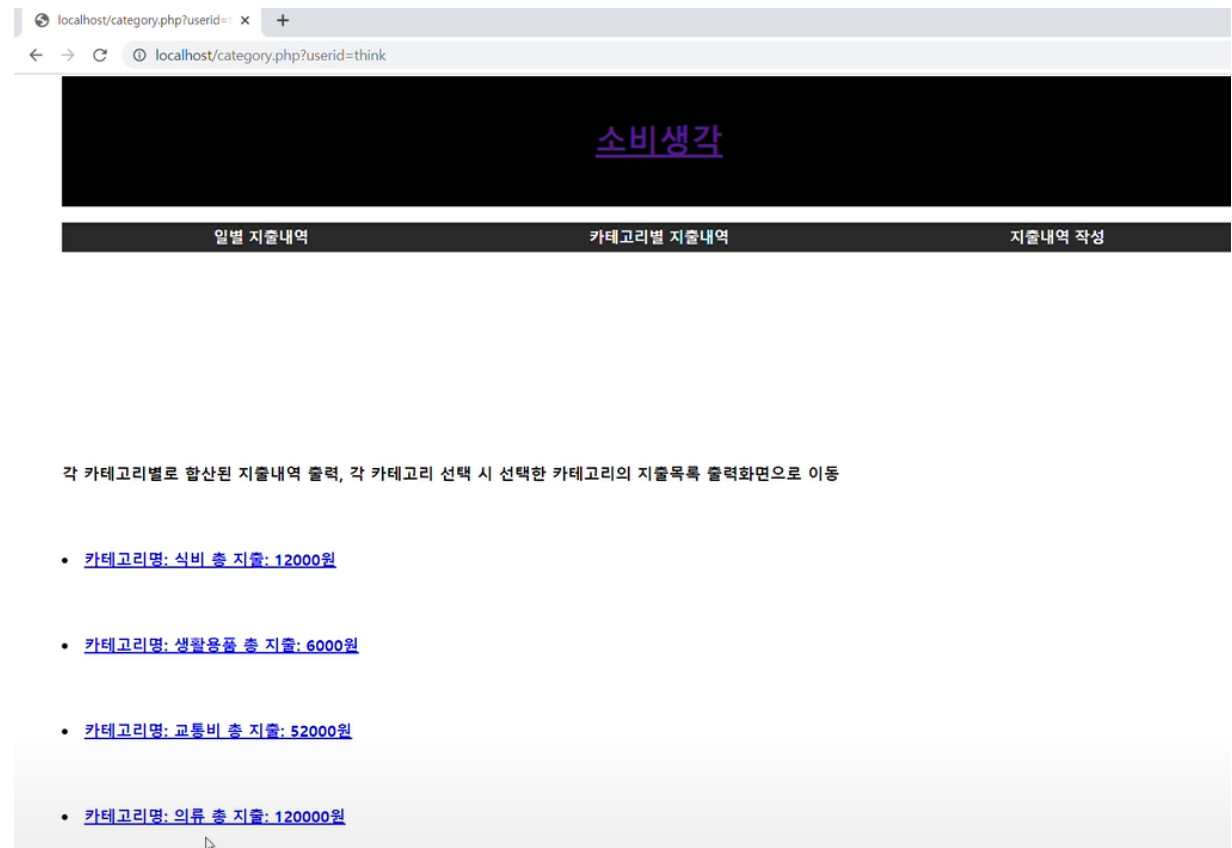
DB 활용한 내용

```
$sql = "SELECT * From expenses LEFT JOIN division ON
expenses.division_id = division.division_id WHERE user_id='{$userid}'
ORDER BY expenses.date_time DESC";
$result = mysqli_query($conn, $sql) or die(mysqli_error($conn));
expenses테이블에 division테이블을 division_id을 기준으로 LEFT JOIN 하여 만들어진
```

테이블에서 이전 세션에서 전달받은 `userid`와 테이블의 `user_id`가 일치하는 행만 내림차순(시간순)으로 정렬한다. 그리고 완성된 테이블과 `mysql_query()` 함수를 사용하여 로그인한 계정에서 작성한 지출 내역을 최신순으로 정렬하여 가져온다. 이때 확인을 위해 `mysqli_error()` 함수를 사용한다

(4-3) 분류별(카테고리별) 총 지출 화면: 한지은

결과 캡처 화면



설명

`category.php` 파일에서 php문법을 통해 로그인에 성공한 `user_id`값을 php문법을 사용해 받아오고,

```
$userid = $_SESSION['userId'];
```

DB에서 가져온 카테고리명과 각 카테고리별 총 지출을 선택 가능한 리스트로 출력한다. 리스트 선택 시 `category_detail.php`파일로 `user_id`와 `division_id`를 전달하며 이동한다.

```
$sum_prices = 0;
```

```
$divisionId = 0;
```

```
while($row = mysqli_fetch_array($result)) {
```

```
    $sum_prices = $row['sum(prices)'];
```

```
    //링크 이동과 함께 선택한 카테고리의 division_id, user_id, expenses_id를  
    넘겨준다
```

```

echo "<br> <h4> <li> <a href = \"category_detail.php?userid={\$userid}
    &divisionid={\$row['division_id']}\>";
//카테고리명: division_name    총 지출: $sum_prices
echo "카테고리명: ". $row["division_name"]. "총 지출: ".
    $sum_prices. "원";
echo "</a> </li> </h4>";
}

```

DB 활용한 내용

```

$sql = "SELECT * From expenses LEFT JOIN division ON
expenses.division_id = division.division_id WHERE user_id='$userid' AND
expenses.division_id='$division_id' ORDER BY expenses.date_time DESC";
$result = mysqli_query($conn, $sql) or
die(mysqli_error($conn));

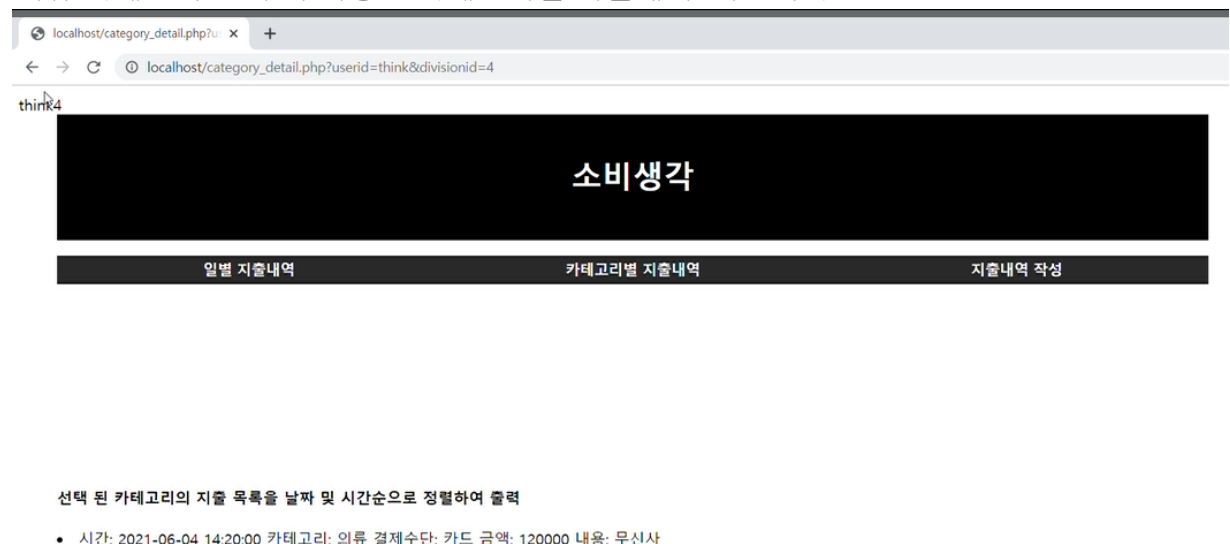
```

expenses테이블에 division테이블을 division_id를 기준으로 LEFT JOIN 하여 이전 category.php에서 전달받은 user_id와 division_id가 모두 일치하는 행만 내림차순(시간순)으로 정렬한다. 그리고 완성된 테이블과 mysqli_query() 함수를 사용하여 로그인한 계정에서 작성한 지출 내역을 최신순으로 정렬하여 가져온다. 이때 확인을 위해 mysqli_error() 함수를 사용한다

(4-4) 분류별(카테고리별) 지출내역 확인: 한지은

결과 캡처 화면

:의류 카테고리 선택 시 이동한 카테고리별 지출내역 화면이다.



설명

category_detail.php 파일에서 php문법을 통해 로그인에 성공한 user_id값과 division_id, expenses id를 php문법을 사용해 받아오고,

```
$userid = $_GET['userid']??'';  
$expense_id = $_GET['id'];  
$division_id = $_GET['divisionid'];
```

DB에서 받아와 선택한 카테고리에 속한 지출내용 및 목록을 화면에 출력한다.

```
if (mysqli_num_rows($result) > 0) {  
    while($row = mysqli_fetch_assoc($result)) {  
        echo "<li>";  
        echo "시간: " . $row["date_time"]. " 카테고리: " .  
            $row["division_name"]. " 결제수단: " . $row["method"]. " 금액: " .  
            $row["prices"]. " 내용: ". $row["contents"]; //지출내용 출력  
        echo "</li>";  
    }  
}
```

DB 활용한 내용

(5) 지출 내용 작성화면: 윤지연

결과 캡처 화면

지출내역 입력

localhost/expenditureInput.php?userid=think

지출내역 입력

- [일별 지출내역](#)
- [카테고리별 지출내역](#)
- [지출내역 작성](#)

날짜 2021-06-02

시간 오후 02:19

분류 식비

금액 12000

내용 내용을 입력해주세요.

현금과 카드 중 어떤 것을 사용하셨나요?

☐ 현금

☐ 카드

저장

localhost/write.php

localhost/write.php

localhost 내용:
지출내역이 등록되었습니다.

확인

DB에 추가된 내용 캡처 화면(expenses 테이블)

✓ 0 - 15행 표시 중 (합계 16, 질의 실행시간 0.0002 초.)

SELECT * FROM `expenses`

☐ 프로파일링 [\[인라인 편집\]](#) [\[수정\]](#) [\[SQL 해석\]](#) [\[PHP 코드 보기\]](#) [\[다시 보기\]](#)

☐ 모두 보기

행 갯수: 25 ▼

행 필터링: 현재 테이블 검색

Sort by key: 없음 ▼

+ 옵션

←T→	expenses_id	user_id	division_id	method	prices	contents	date_time
<input type="checkbox"/> 수정 복사 삭제	12	lm	1	현금	364354	사용	2021-06-18 21:45:00
<input type="checkbox"/> 수정 복사 삭제	6	lm	1	현금	364354	사용	2021-06-18 19:15:00
<input type="checkbox"/> 수정 복사 삭제	11	lm	1	현금	34634	444	2021-06-25 21:44:00
<input type="checkbox"/> 수정 복사 삭제	13	lm	1	현금	364354	사용	2021-06-17 21:56:00
<input type="checkbox"/> 수정 복사 삭제	14	lm	1	현금	364354	사용	2021-06-24 21:02:00
<input type="checkbox"/> 수정 복사 삭제	15	lm	1	카드	60000	김밥천국	2021-06-26 22:55:00
<input type="checkbox"/> 수정 복사 삭제	17	lm	2	현금	500000	스스	2021-06-17 00:30:42
<input type="checkbox"/> 수정 복사 삭제	18	lm	2	현금	50000	스스	2021-06-10 10:30:42
<input type="checkbox"/> 수정 복사 삭제	20	think	3	카드	12000	창원<->마산	2021-06-02 20:16:00
<input type="checkbox"/> 수정 복사 삭제	21	think	2	현금	6000	가글 및 칫솔	2021-06-02 10:17:00
<input type="checkbox"/> 수정 복사 삭제	22	think	3	카드	40000	숙박	2021-06-02 21:18:00
<input type="checkbox"/> 수정 복사 삭제	23	think	4	카드	120000	무신사	2021-06-04 14:20:00

설명

expenditureInput.php파일에서 로그인에 성공한 user_id값을 php문법을 사용해 받아오고, html 문법 중 <form> 태그와 <input>태그를 사용해 사용자로부터 날짜, 시간, 분류, 금액, 내용, 결제 방법에 대한 정보를 입력받아 “저장” 버튼을 누르면 write.php파일에 입력받은 내용을 전달

```
<?php
    error_reporting (E_ALL & ~E_DEPRECATED & ~E_STRICT & ~E_NOTICE);
    session_start();
    $userid = $_GET['userid'];
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>지출내역 입력</title>
</head>
<body>
    <h1> 지출내역 입력 </h1>
    <form method = "post" action = "write.php">
        날짜 <input type = "date" name = "date" value = "dateValue"/><br>
```

```

시간 <input type = "time" name = "time" value = "timeValue"/><br>

분류 <select name = "category" size = "1">
    <option value = "1" selected> 식비 </option>
    <option value = "2"> 생활용품 </option>
    <option value = "3"> 교통비 </option>
    <option value = "4"> 의류 </option>
    <option value = "5"> 저축 </option>
</select><br>

금액 <input type = "text" name = "price" placeholder = "금액을
입력해주세요."/><br>
내용 <input type = "text" name = "content" placeholder = "내용을
입력해주세요."/><br>
현금과 카드 중 어떤 것을 사용하셨나요?<br>
<input type = "radio" name = "method" value = "현금"/> 현금 <br>
<input type = "radio" name = "method" value = "카드"/> 카드 <br>
<input type = hidden name = "userid" value = "<?php echo $userid;?>"
/>

<input type = "submit" value = "저장"/>
</form>

</body>
</html>

```

write.php파일에서 DB에 접근해 Insert쿼리문을 사용해 DB에 사용자로부터 입력받은 내용 저장 후 daily.php파일 호출

```

<?php
// mysql 접속 계정 정보 설정
$mysql_host = 'localhost';
$mysql_user = 'root';
$mysql_pw = '091405';
$mysql_dbname = 'think';

header("Content-Type: text/html; charset=utf-8");
$userid = $_POST['userid']??'';
$date = $_POST['date']??'';
$time = $_POST['time']??'';
$date_time = (string)$date . " " . (string)$time;
$category_value = $_POST['category']??'';
//echo $category_value;
switch($category_value){
    case 1: {

```

```

        $division_id = 1;
        $division_name = '식비';
        global $division_id, $division_name;
        break;
    }

    case 2: {
        $division_id = 2;
        $division_name = '생활용품';
        global $division_id, $division_name;
        break;}

    case 3: {
        $division_id = 3;
        $division_name = '교통비';
        global $division_id, $division_name;
        break;}

    case 4: {
        $division_id = 4;
        $division_name = '의류';
        global $division_id, $division_name;
        break;}

    case 5: {
        $division_id = 5;
        $division_name = '저축';
        global $division_id, $division_name;
        break;}
}

$price = $_POST['price']??'';
$content = $_POST['content']??'';
$method_value = $_POST['method']??'';

//Create connection
$conn = mysqli_connect("localhost", "root", "091405", "think", 3306);

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

mysqli_set_charset($conn, 'utf8');

// 쿼리문
$insertQuery = "INSERT INTO

```

```

expenses(user_id,division_id,method,prices,contents,date_time) VALUES
('$userid','$division_id','$method_value', '$price', '$content',
'$date_time');"
$expense_result = mysqli_query($conn, $insertQuery) or
die(mysqli_error($conn));

if($expense_result)
{
    session_start();
    $_SESSION['userId'] = $userid;
    print_r($_SESSION);
    echo $_SESSION['userId'];
?>
<script>
    alert("지출내역이 등록되었습니다.");
    location.href = "daily.php";
</script>
<?php
}
?>

```

DB 활용한 내용

write.php 파일에서 INSERT 쿼리문과 mysqli_query()함수를 사용해 expenses테이블에
사용자로부터 입력받은 내용을 저장, 이때 에러 확인을 위해 mysqli_error()함수 사용

```

$insertQuery = "INSERT INTO
expenses(user_id,division_id,method,prices,contents,date_time) VALUES
('$userid','$division_id','$method_value', '$price', '$content',
'$date_time');"
echo $insertQuery;
$expense_result = mysqli_query($conn, $insertQuery) or
die(mysqli_error($conn));

```

(6) 지출 내용 수정 및 삭제 화면: 윤지연

결과 캡처 화면

think19

지출내역 수정 삭제

- [일별 지출내역](#)
- [카테고리별 지출내역](#)
- [지출내역 작성](#)

날짜 연도-월-일

시간 --:--

분류 식비

금액 금액을 입력해주세요.

내용 내용을 입력해주세요.

현금과 카드 중 어떤 것을 사용하셨나요?

☐ 현금

☐ 카드

수정 삭제

localhost 내용:
업데이트 되었습니다.

확인

localhost 내용:
삭제가 되었습니다.

확인

DB에 수정 전 내용 캡처 화면(expenses 테이블)

✓ 0 - 12행 표시 중 (합계 13, 질의 실행시간 0.0002 초.)

`SELECT * FROM `expenses``

☐ 프로파일링 [\[인라인 편집\]](#) [\[수정\]](#) [\[SQL 해석\]](#) [\[PHP 코드 보기\]](#) [\[다시 보기\]](#)

☐ 모두 보기 | 행 갯수: 25 ▼ 행 필터링: 현재 테이블 검색 Sort by key: 없음 ▼

+ 옵션

		expenses_id	user_id	division_id	method	prices	contents	date_time
<input type="checkbox"/>	 수정  복사  삭제	24	think	5	현금	3000000	노후자금	2021-06-25 20:29:00
<input type="checkbox"/>	 수정  복사  삭제	12	lm	1	현금	364354	사용	2021-06-18 21:45:00
<input type="checkbox"/>	 수정  복사  삭제	6	lm	1	현금	364354	사용	2021-06-18 19:15:00
<input type="checkbox"/>	 수정  복사  삭제	11	lm	1	현금	34634	444	2021-06-25 21:44:00
<input type="checkbox"/>	 수정  복사  삭제	13	lm	1	현금	364354	사용	2021-06-17 21:56:00
<input type="checkbox"/>	 수정  복사  삭제	14	lm	1	현금	364354	사용	2021-06-24 21:02:00
<input type="checkbox"/>	 수정  복사  삭제	15	lm	1	카드	60000	김밥천국	2021-06-26 22:55:00
<input type="checkbox"/>	 수정  복사  삭제	17	lm	2	현금	500000	ㅅㅅ	2021-06-17 00:30:42
<input type="checkbox"/>	 수정  복사  삭제	18	lm	2	현금	50000	ㅅㅅ	2021-06-10 10:30:42
<input type="checkbox"/>	 수정  복사  삭제	20	think	3	카드	12000	창원<->마산	2021-06-02 20:16:00
<input type="checkbox"/>	 수정  복사  삭제	21	think	2	현금	6000	가글 및 칫솔	2021-06-02 10:17:00
<input type="checkbox"/>	 수정  복사  삭제	22	think	3	카드	40000	숙박	2021-06-02 21:18:00
<input type="checkbox"/>	 수정  복사  삭제	23	think	4	카드	120000	무신사	2021-06-04 14:20:00

DB에 수정 후 내용 캡처 화면(expenses 테이블, expenses_id = 24 수정)









✓ 0 - 12행 표시 중 (합계 13, 질의 실행시간 0.0002 초.)

`SELECT * FROM `expenses``

☐ 프로파일링 [\[인라인 편집\]](#) [\[수정\]](#) [\[SQL 해석\]](#) [\[PHP 코드 보기\]](#) [\[다시 보기\]](#)

☐ 모두 보기 | 행 갯수: 25 ▼ 행 필터링: 현재 테이블 검색 Sort by key: 없음 ▼

+ 옵션

		expenses_id	user_id	division_id	method	prices	contents	date_time
<input type="checkbox"/>	 수정  복사  삭제	24	think	5	현금	5000000	노후자금	2021-06-26 20:28:00
<input type="checkbox"/>	 수정  복사  삭제	12	lm	1	현금	364354	사용	2021-06-18 21:45:00
<input type="checkbox"/>	 수정  복사  삭제	6	lm	1	현금	364354	사용	2021-06-18 19:15:00
<input type="checkbox"/>	 수정  복사  삭제	11	lm	1	현금	34634	444	2021-06-25 21:44:00
<input type="checkbox"/>	 수정  복사  삭제	13	lm	1	현금	364354	사용	2021-06-17 21:56:00
<input type="checkbox"/>	 수정  복사  삭제	14	lm	1	현금	364354	사용	2021-06-24 21:02:00
<input type="checkbox"/>	 수정  복사  삭제	15	lm	1	카드	60000	김밥천국	2021-06-26 22:55:00
<input type="checkbox"/>	 수정  복사  삭제	17	lm	2	현금	500000	ㅅㅅ	2021-06-17 00:30:42
<input type="checkbox"/>	 수정  복사  삭제	18	lm	2	현금	50000	ㅅㅅ	2021-06-10 10:30:42
<input type="checkbox"/>	 수정  복사  삭제	20	think	3	카드	12000	창원<->마산	2021-06-02 20:16:00
<input type="checkbox"/>	 수정  복사  삭제	21	think	2	현금	6000	가글 및 칫솔	2021-06-02 10:17:00
<input type="checkbox"/>	 수정  복사  삭제	22	think	3	카드	40000	숙박	2021-06-02 21:18:00
<input type="checkbox"/>	 수정  복사  삭제	23	think	4	카드	120000	무신사	2021-06-04 14:20:00

DB에서 삭제 전 내용 캡처 화면(expenses 테이블)

✓ 0 - 12행 표시 중 (합계 13, 질의 실행시간 0.0002 초.)

SELECT * FROM `expenses`

☐ 프로파일링 [인라인 편집] [수정] [SQL 해석] [PHP 코드 보기] [다시 보기]

☐ 모두 보기 | 행 갯수: 25 ▼ 행 필터링: 현재 테이블 검색 Sort by key: 없음 ▼

+ 옵션

수정 복사 삭제	expenses_id	user_id	division_id	method	prices	contents	date_time
<input type="checkbox"/> 수정 복사 삭제	24	think	5	현금	5000000	노후자금	2021-06-26 20:28:00
<input type="checkbox"/> 수정 복사 삭제	12	lm	1	현금	364354	사용	2021-06-18 21:45:00
<input type="checkbox"/> 수정 복사 삭제	6	lm	1	현금	364354	사용	2021-06-18 19:15:00
<input type="checkbox"/> 수정 복사 삭제	11	lm	1	현금	34634	444	2021-06-25 21:44:00
<input type="checkbox"/> 수정 복사 삭제	13	lm	1	현금	364354	사용	2021-06-17 21:56:00
<input type="checkbox"/> 수정 복사 삭제	14	lm	1	현금	364354	사용	2021-06-24 21:02:00
<input type="checkbox"/> 수정 복사 삭제	15	lm	1	카드	60000	김밥천국	2021-06-26 22:55:00
<input type="checkbox"/> 수정 복사 삭제	17	lm	2	현금	500000	ㅅㅅ	2021-06-17 00:30:42
<input type="checkbox"/> 수정 복사 삭제	18	lm	2	현금	50000	ㅅㅅ	2021-06-10 10:30:42
<input type="checkbox"/> 수정 복사 삭제	20	think	3	카드	12000	창원<->마산	2021-06-02 20:16:00
<input type="checkbox"/> 수정 복사 삭제	21	think	2	현금	6000	가글 및 칫솔	2021-06-02 10:17:00
<input type="checkbox"/> 수정 복사 삭제	22	think	3	카드	40000	숙박	2021-06-02 21:18:00
<input type="checkbox"/> 수정 복사 삭제	23	think	4	카드	120000	무신사	2021-06-04 14:20:00

DB에서 삭제 후 내용 캡처 화면(expenses 테이블, expenses_id = 24 삭제)

✓ 0 - 11행 표시 중 (합계 12, 질의 실행시간 0.0002 초.)

SELECT * FROM `expenses`

☐ 프로파일링 [인라인 편집] [수정] [SQL 해석] [PHP 코드 보기] [다시 보기]

☐ 모두 보기 | 행 갯수: 25 ▼ 행 필터링: 현재 테이블 검색 Sort by key: 없음 ▼

+ 옵션

수정 복사 삭제	expenses_id	user_id	division_id	method	prices	contents	date_time
<input type="checkbox"/> 수정 복사 삭제	12	lm	1	현금	364354	사용	2021-06-18 21:45:00
<input type="checkbox"/> 수정 복사 삭제	6	lm	1	현금	364354	사용	2021-06-18 19:15:00
<input type="checkbox"/> 수정 복사 삭제	11	lm	1	현금	34634	444	2021-06-25 21:44:00
<input type="checkbox"/> 수정 복사 삭제	13	lm	1	현금	364354	사용	2021-06-17 21:56:00
<input type="checkbox"/> 수정 복사 삭제	14	lm	1	현금	364354	사용	2021-06-24 21:02:00
<input type="checkbox"/> 수정 복사 삭제	15	lm	1	카드	60000	김밥천국	2021-06-26 22:55:00
<input type="checkbox"/> 수정 복사 삭제	17	lm	2	현금	500000	ㅅㅅ	2021-06-17 00:30:42
<input type="checkbox"/> 수정 복사 삭제	18	lm	2	현금	50000	ㅅㅅ	2021-06-10 10:30:42
<input type="checkbox"/> 수정 복사 삭제	20	think	3	카드	12000	창원<->마산	2021-06-02 20:16:00
<input type="checkbox"/> 수정 복사 삭제	21	think	2	현금	6000	가글 및 칫솔	2021-06-02 10:17:00
<input type="checkbox"/> 수정 복사 삭제	22	think	3	카드	40000	숙박	2021-06-02 21:18:00
<input type="checkbox"/> 수정 복사 삭제	23	think	4	카드	120000	무신사	2021-06-04 14:20:00

설명

수정 및 삭제

exchange.php파일에서 수정 및 삭제 화면을 **html**문법을 사용해 구현 (**html** 내용은 (5) 지출내역작성화면의 코드와 유사하기 때문에 생략)

로그인한 **user_id**와 사용자가 수정 혹은 삭제하고자 하는 지출내역의 **expense_id**를 받아와 **php**변수에 저장하여, “수정” 버튼을 누르면 **update.php**파일에서 **input** 태그에서 입력받은 내용을 **Update**쿼리문을 사용해 **DB**에 있는 내용 수정. “삭제” 버튼을 누르면 **delete.php**파일에서 **Delete**쿼리문을 사용해 **DB**에 있는 내용 삭제.

```
<?php
error_reporting (E_ALL & ~E_DEPRECATED & ~E_STRICT & ~E_NOTICE);
    session_start();
    $userid = $_GET['userid']??'';
    $expense_id = $_GET['id'];
    echo $userid;
    echo $expense_id;
?>
// 생략
<script language=javascript>
    function btn_click(str){
        if(str=="update"){
            frm.action = "update.php?userid=<?php echo
$userid;?>&id=<?php echo $expense_id;?>";
        } else if(str=="delete"){
            frm.action ="delete.php?userid=<?php echo
$userid;?>&id=<?php echo $expense_id;?>";
        }
        frm.submit();
    }
</script>
```

DB 활용한 내용

수정

php 파일에서 **UPDATE** 쿼리문과 **mysqli_query()**함수를 사용해 **DB**에 있는 **expenses** 테이블의 내용 중 **\$expeses_id**에 해당하는 내용 수정

```
$updateQuery = "UPDATE expenses SET `division_id`='<?php echo $new_division_id;?>',
`method`='<?php echo $method_value;?>', `prices`='<?php echo $price;?>', `contents` = '<?php echo $content;?>',
`date` = '<?php echo $date_time;?>'
WHERE expenses_id = '<?php echo $expense_id;?>'";
$expense_result = mysqli_query($conn, $updateQuery);
```

삭제

php 파일에서 **DELETE** 쿼리문과 **mysqli_query()**함수를 사용해 **DB**에 있는 **expenses** 테이블의 내용 중 **\$expeses_id**에 해당하는 내용 삭제

```
$deleteQuery = "DELETE FROM expenses WHERE
expenses_id='<?php echo $expensis_id;?>'";
$expense_result = mysqli_query($conn, $deleeQuery);
```

8. 자체평가

다음 평가 기준을 3단계(우수, 보통, 미흡)로 나눠 평가.

8.1. 프로그램의 SQL CRUD 구현 (우수)

C: Insert문

지출내역 작성화면 및 회원가입 화면 구현에서 사용

R: Select문

지출 내역 확인 화면 및 로그인 화면 구현에서 사용

U: Update문

지출 내역 수정 및 삭제 화면 중 “수정” 버튼 구현에서 사용

D: Delete문

지출 내역 수정 및 삭제 화면 중 “삭제” 버튼 구현에서 사용

8.2. ERD 완성도 (우수)

설계 시 작성한 요구분석에 따라 ER 다이어그램을 정확히 맞춰 생성하였고, 실제 개발 시 ERD를 기반으로 구현한 DB테이블을 유용하게 사용하였기 때문에 우수라고 판단

8.3. JOIN의 활용 (우수)

분류별 지출내역 확인을 위해 LEFT JOIN사용 (daily.php, category.php, category_detail.php)

[daily.php]

```
//mysql 명령문 : expenses테이블에 division테이블을 division_id을 기준으로  
LEFT JOIN 하여 전달받은 user_id가 모두 일치하는 행만 내림차순(시간순)으로 정렬  
  
$sql = "SELECT * From expenses LEFT JOIN division ON  
expenses.division_id = division.division_id WHERE user_id='$userid'  
ORDER BY expenses.date_time DESC";
```

[category.php]

```
/*mysql 명령문 : expenses 테이블에 division테이블을 division_id을 기준으로  
LEFT JOIN 하여 expenses.division_id, division_name, sum(prices)을 선택한  
후 이전 category.php에서 전달받은 uer_id와 division_id가 모두 일치하는 행만  
오름차순(시간)으로 정렬*/  
  
$sql = "SELECT expenses.division_id, division_name, sum(prices) From  
expenses LEFT JOIN division ON expenses.division_id =  
division.division_id WHERE user_id='$userid' GROUP BY division_id ORDER  
BY expenses.division_id ASC";
```

[category_detail.php]

```
/*mysql 명령문 : expenses테이블에 division테이블을 division_id을 기준으로  
LEFT JOIN 하여 이전 category.php에서 전달받은 uer_id와 division_id가 모두  
일치하는 행만 내림차순(시간순)으로 정렬  
  
$sql = "SELECT * From expenses LEFT JOIN division ON  
expenses.division_id = division.division_id WHERE user_id='$userid' AND  
expenses.division_id='$division_id' ORDER BY expenses.date_time DESC";
```

8.4. 테이블 3개 이상 (우수)

user, expenses, division 테이블로 총 3개의 사용

8.5. Primary key 및 auto_increment 적용 (우수)

user 테이블의 user_id (Primary key 적용)

expenses 테이블의 expenses_id (Primary key 적용, auto_increment적용)

division 테이블의 division_id (Primary key 적용)

위와 같은 근거로 각 평가 기준별 모두 우수로 평가.

부록

소비생각 전체 코드 참고용 깃허브 주소: <https://github.com/jungbosong/ConsumThinkWeb>