

Library System

2016311821 한승하

Overview

본 Project는 이번학기 학습한 ER Diagram과 이를 통해 만들어지는 Relation Data Model을 이용한 Database 설계, 또한 SQL이라는 언어를 통해 이를 DBMS에 적용하는 과정을 학습할 수 있는 Project이었습니다. 또한 학기 중 수업시간에 다루지 않은 php와 Html언어에 대해서도 학습할 수 있는 기회였습니다. 저는 여러 Topics 중 Library System을 선택하였고, 아래와 같은 5가지 기능들을 구현하였습니다.

1) 회원가입, 로그인, 로그아웃

아이디 및 비밀번호 등을 모두 입력하지 않았을 경우, 이미 존재하는 아이디인 경우 등의 대한 에러처리

2) [Must] 키워드를 통한 도서검색 및 정보 표출

도서 제목 및 저자 이름으로 도서 검색 및, LIKE를 이용해 유사한 제목 및 저자이름을 가진 도서 검색

3) 도서 대여 및 이미 대여된 도서일 경우 예약

이미 대여된 도서에 대한 탐색과 예약 순위 부여

4) [Plus] 스터디룸 조회 및 가능한시간 예약

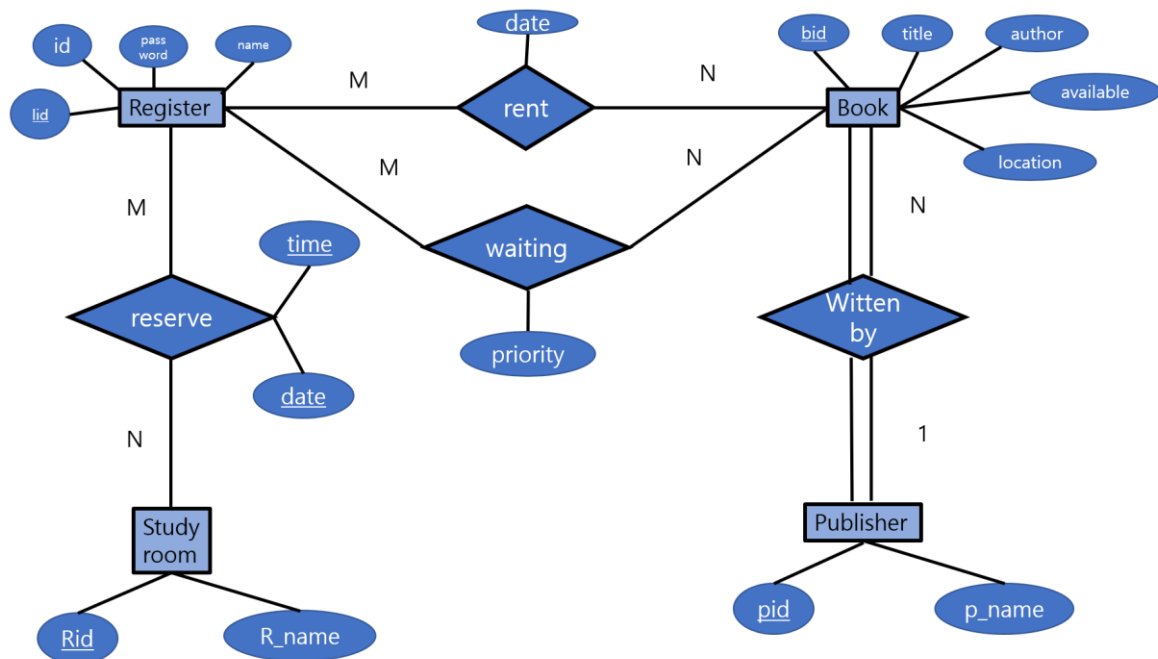
이미 예약된 시간에 대한 처리

5) My page를 통한 기본정보, 대여 정보, 예약 정보, 등의 확인

예약 및 대여 정보에 대한 검색 및 표출

화면전환 및 Variable들의 전달, 등은 php와 Html 언어로 구현하였고, Database 관리와 Insertion, Deletion 등은 Mysql을 통해 구현하였습니다.

ER diagram



위는 제가 이번 Project DBMS Modeling을 위해 디자인한 ER Diagram입니다.

총 4개의 Entity와 4개의 Relation으로 구성되어 있으며 Entity와 Relation에 대한 설명은 아래와 같습니다

<Entity>

1) Register

Register는 회원들의 정보를 저장하고 있습니다. 사용된 Data와 Type는 아래와 같습니다.

-Lid: Library id의 약자로 회원 가입한 순서대로 하나의 고유번호를 받습니다. Int 형 변수로 Register Entity의 Key입니다.

-Id: 회원가입시 입력한 아이디로 varchar형을 사용했습니다.

-Password: 회원가입시 입력한 password로 varchar형을 사용했습니다.

-Name: 회원가입시 입력한 name으로 varchar형을 사용했습니다.

2) Book

Book은 Library에 저장되어 있는 책의 정보를 기록하고 있습니다. 본 과제에서 Book insert 기능은 구현하지 않았으므로 사전 책 정보들을 insert해 줘야 합니다. Data와 Type은 아래와 같습니다

-Bid: BOOK ID의 약자로 lid와 마찬가지로 책 하나마다 고유의 번호를 부여 받았습니다. Book entry의 key입니다.

-title: 도서의 제목을 기록하고 있으며 varchar형을 사용했습니다.

-author: 도서의 저자를 기록하고 있으며 varchar형을 사용했습니다.

-available: 도서가 대여 가능 상태인지 (이미 대여 중인지) 를 기록하는 변수로 tinyint 형 변수를 사용하고 있고 0은 available하지 않음을 1은 대여 가능함을 나타냅니다.

-location: 도서를 보관중인 장소를 기록합니다.

2) Publisher

도서를 기록한 출판사에 대한 정보를 기록하고 있으며 BOOK과 마찬가지로 사전 기록해 주어야 합니다. 사용한 Data와 Type은 아래와 같습니다.

-pid: Publisher ID의 약자로 lid, bid와 같이 Publisher마다 하나의 고유번호를 부여 받으며, Publisher의 key입니다.

-p_name: Publisher의 이름을 기록합니다.

3) Study room

스터디룸의 정보를 기록하고 있으며 스터디룸 또한 BOOK과 마찬가지로 사전 기록해 주어야 합니다. 사용한 Data와 Type은 아래와 같습니다.

-Rid: Room ID의 약자로 Room별로 고유번호를 부여 받으며, Study room의 Key입니다.

-R_name: Room의 이름을 기록하고 있습니다.

<Relations>

위의 ER Diagram에서 사용된 Relations들은 다음과 같습니다.

1) Reserve

Reserve는 회원과 회원이 예약한 Studyroom 간의 관계입니다. 여러 명의 회원이 여러 스터디룸을 예약할 수 있으므로, M:N 관계로 표시해 주었으며, 하나의 회원이 하나의 스터디룸을 여러 시간대에 예약할 수 있으므로, time과 date또한 key로 표시해 주었습니다.

2) Rent

Rent는 회원과 회원이 대여한 도서에 대한 관계입니다. 하나의 회원이 다양한 책을 대여할 수 있으므로 1:M 관계로 표시해 주었으며, 예약한 날짜인 Date를 명시해 주었습니다.

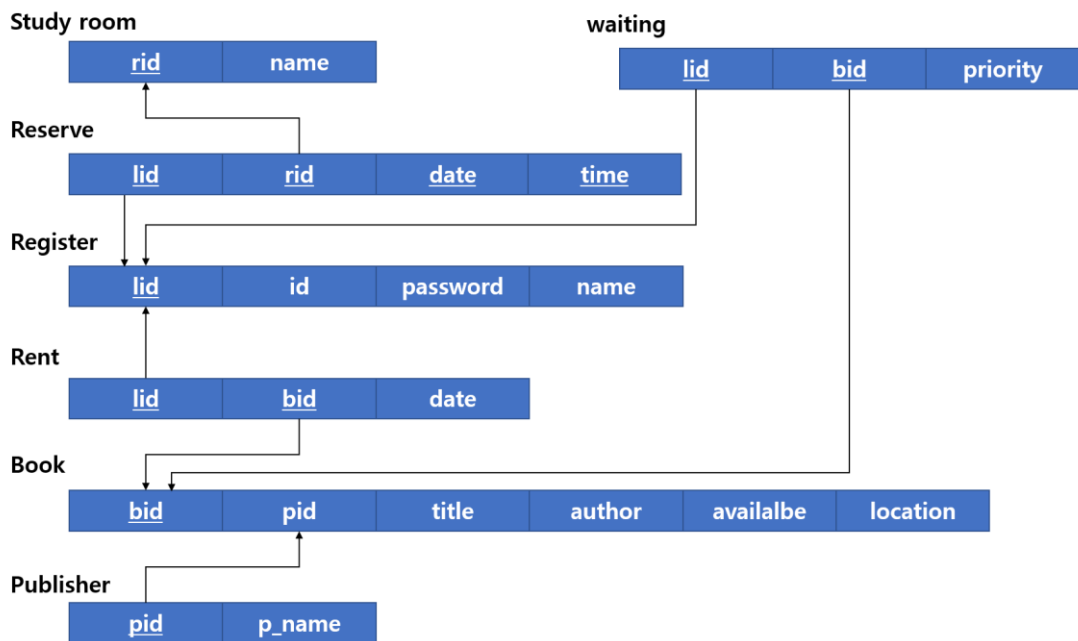
3) Waiting

Waiting은 회원과 회원이 예약한 도서의 관계로 여러 회원이 여러 도서를 예약할 수 있으므로 M:N관계로 표시해 주었으며, 예약 순서인 Priority 값을 표시해 주었습니다.

4) Written by

도서와 출판사의 관계로, 모든 책들은 반드시 어떤 출판사에 소속되어 있어야 하고, 여러 책이 하나의 출판사에 속해 있어야 하므로 M:1 관계를 표시해 주었습니다.

Relational Data Model



위는 ER diagram으로 만든 Relation data model입니다.

ER diagram -> Relation table 교안을 참고하여 만들었으며, 각 table은 위에서 ER diagram에서 설명 드렸던 항목들로 이루어져 있습니다.

Reserve와 waiting은 M:N Relation으로 새로운 Table을 만들어 관리해 주었으며, Rent는 M:1 Relation이나, Date변수와 더 효율적인 관리를 위해 새로운 Table로 관리해 주었습니다. Book은 Written by 관계에 전부 포함되어 있으므로 pid를 Book table에 삽입해 주었습니다.

모든 Data 들은 NOT NULL로 정의되어 있으며, lid, rid, pid, bid는 전부 AUTO_INCREMENT를 사용하여 자동으로 독립적인 변수가 설정될 수 있게 해 주었습니다.

Realtion을 Table로 만든 Table들은 (rent, reserve, waiting) FK들을 PK로 가지고 있으며, Reserve

table에선 하나의 회원이 같은 스터디룸을 여러 시간, 여러 날짜에 예약할 수 있으므로 date와 time도 PK로 선언해 주었습니다.

Implementation and Results

(같이 첨부한 DB_Project PPT를 같이 보시면 더 이해가 잘 될 수 있습니다.)

저는 아래와 같은 5개의 기능을 구현하였습니다.

1) 회원가입, 로그인, 로그아웃

아이디 및 비밀번호 등을 모두 입력하지 않았을 경우, 이미 존재하는 아이디인 경우 등의 대한 에러처리

2) [Must] 키워드를 통한 도서검색 및 정보 표출

도서 제목 및 저자 이름으로 도서 검색 및, LIKE를 이용해 유사한 제목 및 저자이름을 가진 도서 검색

3) 도서 대여 및 이미 대여된 도서일 경우 예약

이미 대여된 도서에 대한 탐색과 예약 순위 부여

4) [Plus] 스터디룸 조회 및 가능한시간 예약

이미 예약된 시간에 대한 처리

5) My page를 통한 기본정보, 대여 정보, 예약 정보, 등의 확인

예약 및 대여 정보에 대한 검색 및 표출

기능에 대해 하나씩 설명 드리겠습니다.

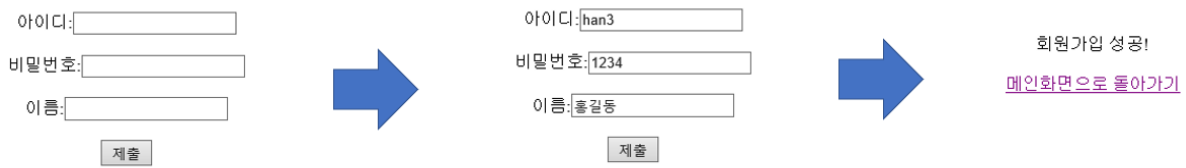
<회원가입, 로그인, 로그아웃>

SKKU Library (Made by 한승하(2016311821))

[로그인](#)

[회원가입](#)

Local host web을 열면 초기에 위와 같은 page를 만나실 수 있습니다. 회원가입은 아래 그림과 같은 순서로 진행됩니다.



회원은 Register table에서 관리하고 있으며, id, pw, name 변수는 POST방식으로 받아 Register_mysql.php 페이지에서 insert 및 결과를 프린트해 줍니다.

사용한 쿼리는 다음과 같습니다. if(\$id==NULL || \$pw==NULL || \$name==NULL) 을 이용하여, 한 칸 이라도 빈칸인 상태로 제출하였을 경우 에러처리를 해주었고, \$sql="SELECT * from Register WHERE id = '\$id'"; 을 이용하여, 이미 같은 아이디로 가입한 사람이 있는지 체크와, 에러처리를 해주었습니다.

만약 정상적인 input값들만 들어온 경우 \$sql="INSERT INTO register (id,password,name) VALUES ('\$id','\$pw','\$name')"; 를 이용해 DB에 회원 정보를 입력해 주었습니다.

에러 화면은 다음과 같습니다.

ERROR CASES

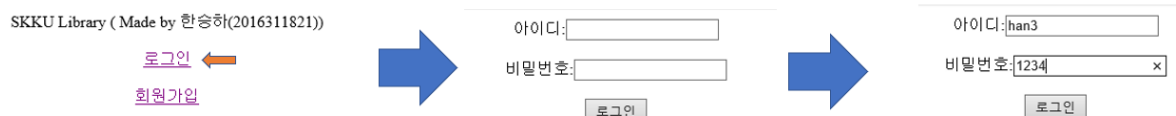
모든 칸을 입력 해 주십시오

[입력확인으로 돌아가기](#)

이미 존재하는 아이디 입니다.

[돌아가기](#)

다음은 로그인 입니다.



Id, password를 DB에서 체크하여 동일한 값이 있을 경우 로그인 처리하게 해 주었습니다.

if(\$id==NULL || \$pw==NULL) 를 이용하여 빈칸이 있을 경우 에러처리를 해 주었으며,

\$sql="SELECT * from Register WHERE id = '\$id' AND password = '\$pw'"; 를 이용하여 동일 id, password로 가입된 회원이 있는지 탐색합니다. 있다면 로그인 페이지로 이동하며, Session을 사용하여 id, password, name, lid를 저장해 줍니다. 없다면 에러 페이지로 이동합니다. 에러 페이지는 아래와 같습니다.

ERROR CASES

모든 칸을 입력 해 주십시오

[로그인 창 으로 돌아가기](#)

아이디 혹은 패스워드가 잘못되었습니다.

[돌아가기](#)

마지막으로 로그아웃 입니다.

SKKU Library (Made by 한승하(2016311821))

안녕하세요 홍길동 님!

[<도서검색>](#)

[<스터디룸 예약>](#)

[<마이페이지>](#)

[<로그아웃>](#)



SKKU Library (Made by 한승하(2016311821))

[로그인](#)

[회원가입](#)

로그인을 하게 되면 다음과 같은 메인 창이 뜨게 되는데, 로그아웃을 누르게 되면

session_destroy(); 를 이용하여 session을 해지 후, 초기창으로 이동하게 됩니다.

[Must] 키워드를 통한 도서검색 및 정보 표출

다음은 도서 검색 기능입니다. 로그인 후 메인 페이지에서 도서검색 페이지로 이동할 수 있는데 도서검색 페이지는 다음과같이 구성되어 있습니다.

SKKU Library (Made by 한승하(2016311821))

안녕하세요 홍길동 님!

[<도서검색>](#)

[<스터디룸 예약>](#)

[<마이페이지>](#)

[<로그아웃>](#)



제목 혹은 저자

검색어 :

[메인 페이지로 돌아가기](#)



제목 혹은 저자

검색어 :

[메인 페이지로 돌아가기](#)

검색은 제목, 저자 모두 검색 가능 하게 해 주었으며,

\$sql = "SELECT bid, title, author, available, location FROM book WHERE title LIKE '%\$value%' OR author LIKE '%\$value%'"; 쿼리를 이용해서 동일한 값이 아닌 유사한 값이 들어와도 검색될 수 있게 해 주었습니다.

책이 검색되면 다음과 같은 창이 뜨게 됩니다. (더 자세한 사진은 ppt에 첨부되어 있습니다)

등록번호: 1
제목: LOL WORLDS
저자: FAKER
보관 위치: LCK_SECTION

대출가능

대출하기

등록번호: 6
제목: LEDGEND NEVER DIE
저자: FAKER
보관 위치: LCK_SECTION

대출중

예약하기

available을 확인하여 대출가능여부를 확인했습니다.

if(\$row[3] == 0) (\$row =
mysqli_fetch_array(\$result) 를 이용한 row[3]에는
available 값이 저장되어 있습니다)

대출 가능한 경우 <대출하기> 버튼이, 대출되어
대출 불가능한 경우 <예약하기> 버튼이 표시되게
됩니다.

<대출하기> 를 클릭한 경우 대출을

<예약하기> 를 클릭한 경우 예약을 진행할 수
있습니다.

책에 대한 정보는 등록번호 (bid) , 제목, 저자, 보
관위치와 대출가능 여부가 표시되게 하였습니다.'

만일 책이 없을 경우

IF No Books

검색 결과가 존재하지 않습니다.

검색창으로 돌아가기

와 같은 에러창이 표시되게 됩니다.

<도서 대여 및 이미 대여된 도서일 경우 예약>

대출과 예약은 아래와 같이 진행되게 됩니다.

대출은 rent table을, 예약은 waiting table을 이용하므로 각각의 table에 insert해주는 쿼리들을 사

용하였습니다.

```
$sql = "INSERT INTO rent(lid,bid,date) values ('".$lid."','".$bid."',NOW());";
```

```
$sql = "UPDATE book SET available = 0 where bid = '".$bid.'";
```

 를 이용하여 rent table 에 insert 와 available 값을 update 해 주었습니다.

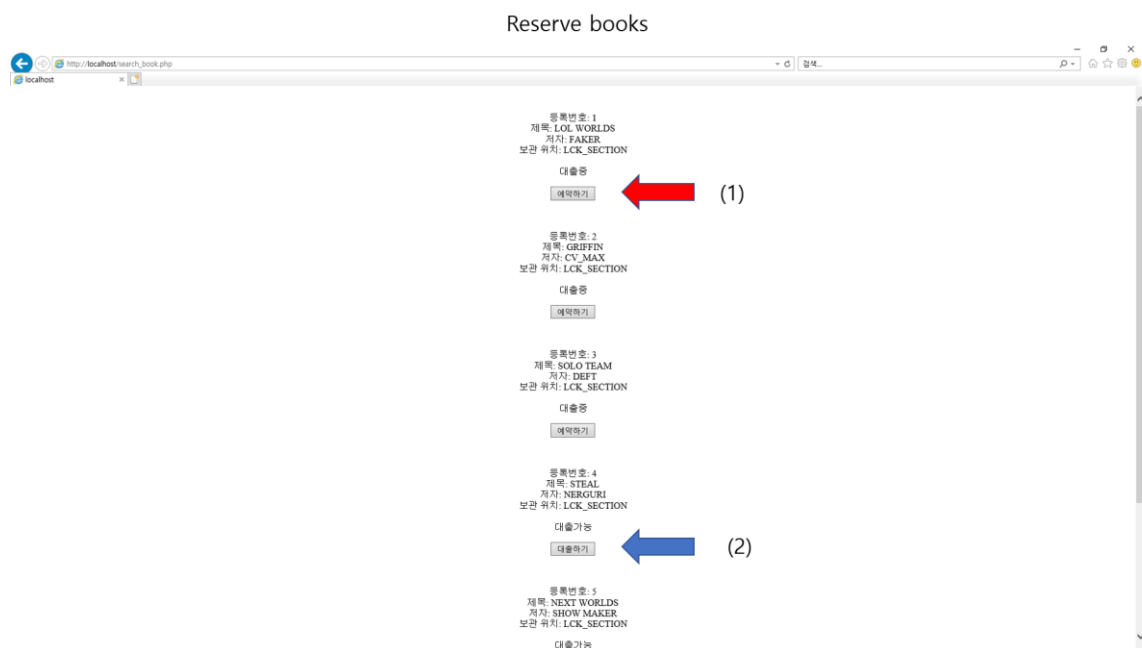
예약은

```
$sql = "SELECT bid from waiting where bid = '".$bid.'";
```

 를 이용하여 이미 같은 도서를 예약한 사람 수를 체크하여 예약 순번을 부여하게 됩니다. 이후

```
$sql2 = "INSERT INTO waiting(lid,bid,priority) values ('".$lid."','".$bid."','".$count.'");
```

 를 이용하여 lid, bid 그리고 순번을 입력해 주었습니다. 결과창은 아래와 같습니다.



Reserve books

(1)

도서 예약 완료

마이 페이지에서 확인할 수 있습니다

마이페이지

다른 도서 검색

(2)

도서 대여 완료

마이 페이지에서 확인할 수 있습니다

마이페이지

다른 도서 검색

< [Plus] 스터디룸 조회 및 가능한시간 예약 >

SKKU Library (Made by 한승하(2016311821))

안녕하세요 홍길동 님!

<도서검색>

<스터디룸 예약>



<마이페이지>

<로그아웃>

다음 페이지를 클릭하면 스터디룸을 예약할 수 있습니다.

원하시는 시간을 선택해 주세요 (1회 예약시 사용시간은 1시간 입니다)

<스터디룸 예약>
약을 원하시는 날짜를 선택해 주세요

11-28



11-29

11-30

12-01

12-02

메인 페이지로 돌아가기



< 400111호 >

11:00

12:00시는 이미 예약되었습니다

13:00시는 이미 예약되었습니다

14:00

15:00

16:00



17:00

18:00

19:00

20:00

< 400117호 >

11:00

스터디룸은 위와 같이 진행되게 되며 \$sql = "SELECT rid,name FROM studyroom;"; 를 이용하여 존재하는 스터디룸 각각에 대해 예약 가능한 시간을 print합니다.

\$sql2 = "SELECT time FROM reserve WHERE rid = ".\$row[0]."" AND date = ".\$value.""; 을 이용하여 예약된 시간이 있는지 체크하고 있다면 이미 예약되었다는 메시지를 print합니다.

빈 시간을 클릭하면 page가 넘어가며

\$sql = "INSERT Into reserve (lid,rid,date,time) values

(".\$_SESSION["ses_lid"].",".\$rid.",".\$date.",".\$time.");"; 쿼리를 이용하여 예약 시간 및 장소, 예약자 lid를 저장해 줍니다. 저장완료시 다음과 같은 창이 표시됩니다.

11-28 일 16:00 시
스터디룸 예약 완료되었습니다.

마이 페이지에서 확인할 수 있습니다

마이페이지

<My page를 통한 기본정보, 대여 정보, 예약 정보, 등의 확인>

SKKU Library (Made by 한승하(2016311821))

안녕하세요 홍길동 님!

[<도서검색>](#)

[<스터디룸 예약>](#)

[<마이페이지>](#)

[<로그아웃>](#)



홍길동 님의 정보입니다.

아이디 : han3
비밀번호 : 1234

나의 도서 대여 정보

나의 도서 예약 정보

나의 스터디룸 예약 정보

메인 페이지로 돌아가기

로그인 메인 창에서 마이페이지를 클릭하면 다음과 같은 마이페이지 창이 뜨게 되는데 마이페이지의 메인 창에서는 가입자의 이름과 아이디, 비밀번호를 확인할 수 있습니다. 이 정보들은 session에 저장된 정보들로 프린트해 주었습니다.

이후 3가지 영역에 대한 예약정보를 확인할 수 있습니다.

먼저 대여한 도서에 대한 정보입니다. Rent table을 다음과 같은 쿼리로 서칭하여

```
$sql = "SELECT B.bid,title,author,date FROM book B, rent R WHERE B.bid = R.bid AND R.lid =  
        ''.$lid.'''";
```

대여한 도서를 서칭 하여 프린트합니다. 이때 현재 날짜를 기준으로 연체 여부를 계산하여 연체 여부와 연체료를 표시하였습니다. 대여정보 창은 아래와 같습니다.

홍길동 님의 정보입니다.

아이디 : han3
비밀번호 : 1234

나의 도서 대여 정보

나의 도서 예약 정보

나의 스터디룸 예약 정보

메인 페이지로 돌아가기

오늘 날짜 : 19-11-28

<대여 도서 목록>

등록번호: 4
제목: STEAL
저자: NERGURI
대출일: 2019-11-29
반납일: 2019-12-14

마이페이지로 돌아가기

혹은

오늘 날짜 : 19-11-28

<대여 도서 목록>

등록번호: 1
제목: LOL WORLDS
저자: FAKER
대출일: 2019-10-29
반납일: 2019-11-13

<< 15 일 연체중 >>

현재까지 연체료는 1500원 입니다
(연체료는 하루당 100원 입니다)

등록번호: 3
제목: SOLO TEAM
저자: DEFT
대출일: 2019-11-28
반납일: 2019-12-13

마이페이지로 돌아가기

혹은

대여한 도서가 존재하지 않습니다.

[마이페이지로 돌아가기](#)

와 같이 표시되게 됩니다.

예약도서는 대여한 도서와 같은 방법으로 waiting table을 서칭하여 표시합니다.

\$sql = "SELECT B.bid,title,author,priority FROM book B, waiting W WHERE B.bid = W.bid AND W.lid = ' ".\$lid." "; 쿼리를 이용하여 서칭하며 결과는 아래와 같습니다.

오늘 날짜 : 19-11-28

<예약 도서 목록>

등록번호: 1
제목: LOL WORLDS
저자: FAKER
예약순위: 3

등록번호: 2
제목: GRIFFIN
저자: CV_MAX
예약순위: 1

[마이페이지로 돌아가기](#)

홍길동 님의 정보입니다.

아이디 : han3
비밀번호 : 1234

[나의 도서 대여 정보](#)

[나의 도서 예약 정보](#)

[나의 스터디룸 예약 정보](#)

[메인 페이지로 돌아가기](#)

혹은

예약한 도서가 존재하지 않습니다.

[마이페이지로 돌아가기](#)

와 같은 창으로 나타나게 됩니다.

마지막으로 스터디룸 예약은 reserve table을 서칭하여 출력하게 되며

\$sql = "SELECT name,date,time from studyroom S, reserve R where R.lid = ' ".\$lid." AND S.rid = R.rid"; 쿼리를 사용하여 서칭 해 주었습니다. 결과창은 아래와 같습니다

홍길동 님의 정보입니다.

아이디 : han3
비밀번호 : 1234

나의 도서 대여 정보

나의 도서 예약 정보

나의 스터디룸 예약 정보

메인 페이지로 돌아가기



400111호
11-28 일 : 16:00 시에 예약되어있습니다.:

마이페이지로 돌아가기



혹은

스터디룸 예약 내역이 존재하지 않습니다.

마이페이지로 돌아가기

와 같은 창이 나타나게 됩니다.

이상 데이터베이스개론 project에 대한 final 보고서였습니다.

감사합니다.