Database 팀 프로젝트 11조

1-1. 사용자 요구 명세서

도서 관리 시스템은 사원이 책과 회원들을 관리하는 시스템이다.

<로그인>

시스템에 접근 가능한 사원은 이미 등록되어 있으며, 로그인 시도 시 등록 되어 있는 사원은 시스템에 접근이 가능하나, 그렇지 않은 사원은 시스템을 이용할 수 없다.

<책 추가>

사원은 책의 장르, 책 이름, 저자 정보를 바탕으로 새 책을 등록할 수 있다.

<회원 추가>

또한, 사원은 회원 이름, 전화 번호, 주소를 바탕으로 새 회원을 등록할 수 있다.

<예약>

책의 이름으로 사원이 검색을 하여 예약이 가능한지 아닌지 확인할 수 있다. 이미 어떤 회원이 예약을 했다면, 예약한 회원의 아이디와 예약한 날짜를 보여준다. 예약이 가능하면, 예약을 요청하는 회원의 아이디로 예약해준다.

<선호도 표시>

도서 관리 시스템을 통해 다음과 같이 3가지의 Best 5 또는 Best 3를 볼 수 있다.

첫번째로, 도서관에 있는 전체 책들 중에서 Best 5를 볼 수 있다.

두번째로, 도서관에 있는 전체 책들 중에 인기가 많은 장르 Best 3를 볼 수 있다.

세번째로, 검색한 장르의 책 Best 3를 볼 수 있다.

<대여>

사원은 책의 대여 시, 책 목록에서 원하는 책을 클릭, 회원 목록에서 원하는 회원을 클릭 후 ‘Borrow’ 버튼을 누른다. 대여가 가능 한 책은 책의 대여자 정보에 대여자의 회원 id가 입력된다. 대여가 불가능한 책은 대여가 불가능함을 알리는 메세지를 출력한다.(재욱오빠 코딩 완료 후 수정~)

<반납>

사원은 책의 반납 시, 반납하고자 하는 책을 책 목록에서 선택 후 ‘Return’버튼을 누른다. 반납이 된 책은 책의 대여자 정보가 빈칸으로 변경된다.

<종료>

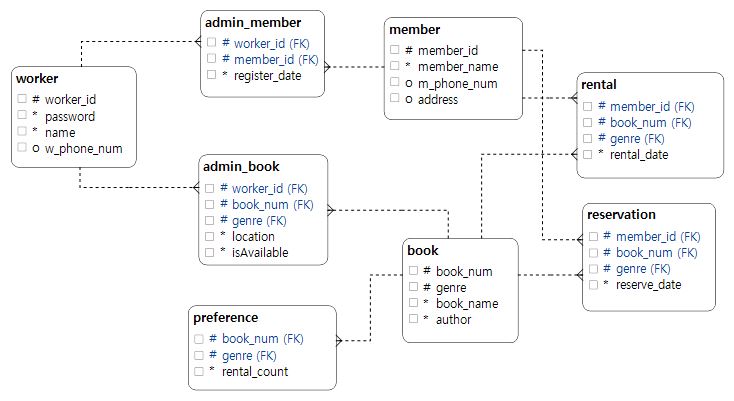
사원은 프로그램 종료 시 ‘Exit’ 버튼을 클릭한다.

1-2. 기능 요구 명세서

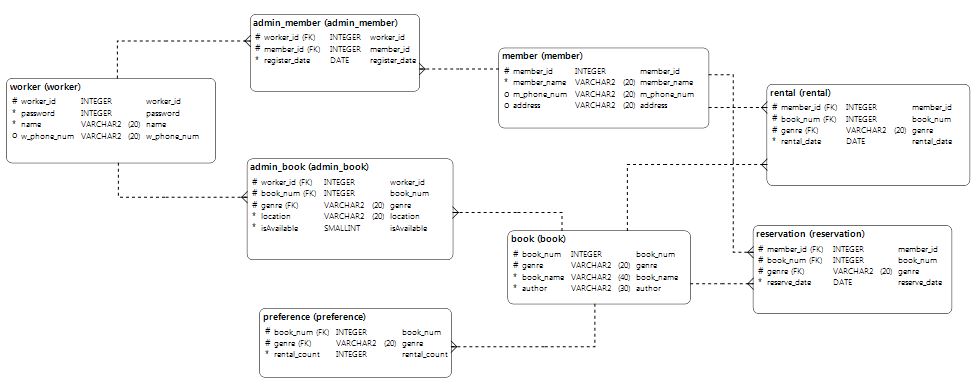
|  |  |  |  |
| --- | --- | --- | --- |
| No | 기능명세 | 입력 | 출력 |
| 1 | 사원이 로그인한다. |  |  |
| 2 | 사원이 회원을 등록한다. |  |  |
| 3 | 사원이 책을 등록한다. |  |  |
| 4 | 사원이 회원정보를 바탕으로 책을 대여 / 예약 시켜준다. |  |  |
| 5 | 빌려간 책의 정보를 확인한다. | book\_num | member\_id  rental\_date |
| 6 | 사원이 회원정보와 책 정보를 바탕으로 반납처리한다. |  |  |
| 7 | 책의 대여 횟수를 통해 Best 5 책을 선출하여 출력한다. |  | book\_name  author |
| 8 | 책의 예약정보를 확인한다. | book\_num | member\_id  book\_name  author  genre  reserve\_date |
| 9 | 책이 꽂혀있는 위치를 찾는다. | book\_num | location  isAvailable |
| 10 | 회원의 정보를 확인한다. | member\_id | m\_phone\_number  address |
| 11 | 책의 관리자를 찾는다. | book\_num | worker\_id  w\_phone\_num |
| 12 | 장르별 Best 3 책을 선출하여 출력한다. | genre | book\_name  author |
| 13 | Best 3 장르를 해당 장르 책들의 총 대여횟수와 함께 출력한다. |  | genre, sum(rental\_count) |

2-1. E-R 다이어그램

1. 논리적인 설계



1. 물리적인 설계



2-2. DB Schema 구축 스크립터

별도 파일 첨부(DDL.sql)

2-3. 테이블 명세서

별도 파일 첨부(테이블정의서.hwp)

3-1. 데이터 입력 스크립터

별도 파일 첨부(InputData.sql)

3-2. 기능구현 SQL

1. select worker\_id, password

from worker

where worker\_id = ‘INPUT’;

2. INSERT INTO MEMBER

VALUES (MEMBER\_ID,MEMBER\_NAME,M\_PHONE\_NUM,ADDRESS)

INSERT INTO ADMIN\_MEMBER

VALUES (WORKER\_ID,MEMBER\_ID,REGISTER\_DATE)

3. INSERT INTO BOOK

VALUES (BOOK\_NUM, GENRE, BOOK\_NAME, AUTHOR)

INSERT INTO ADMIN\_BOOK

VALUES (WORKER\_ID, BOOK\_NUM, GENRE, LOCATION, ISAVAILABLE)

4. UPDATE admin\_book SET isAvailable=0 WHERE book\_num = ‘INPUT’

INSERT INTO rental VALUES (member\_id,book\_num,genre,rental\_date)

UPDATE preference SET rental\_count=’prefer’ WHERE book\_num = ‘INPUT’

prefer변수는 rental\_count를 받아와서 ++한 값

5. select member\_id, rental\_date

from rental

where book\_num = ‘INPUT’

6. UPDATE admin\_book SET isAvailable = 1 WHERE book\_name = ‘INPUT’

7. SELECT BOOK\_NAME, AUTHOR

FROM BOOK NATURAL JOIN PREFERENCE

ORDER BY RENTAL\_COUNT DESC;

8. select member\_id, book\_num, reserve\_date

from reservation

where reserve\_num = ‘INPUT’

9. select location, isAvailable

from admin\_book

where book\_num = ‘INPUT’

10. select m\_phone\_number, address

from member

where member\_id = ‘INPUT’

11. select worker\_id, w\_phone\_num

from admin\_book natural join worker

where book\_num = 'INPUT'

12. SELECT BOOK\_NAME, AUTHOR

FROM BOOK NATURAL JOIN PREFERENCE

WHERE GENRE = 'INPUT'

ORDER BY RENTAL\_COUNT DESC;

13. SELECT GENRE, SUM(RENTAL\_COUNT)

FROM PREFERENCE

GROUP BY GENRE

ORDER BY SUM(RENTAL\_COUNT) DESC;

4. 화면설명서

별도 파일 첨부(화면설명서.docx)

5. 데모 파워포인트 파일

별도 파일 첨부(도서관리시스템.pptx)