

[A. HLL-Report-1910896-통계학과- 천옥희]

HLL 사용 언어 : Java

데이터 베이스 : Mysql

데이터베이스 생성과정은 web report에 적었습니다.

주제 : 눈송이 도서관 도서의 대출, 반납, 예약, 회원 관리.

구현 내용 상세 설명

First.java

```
import java.sql.*;
import java.util.Scanner;

public class First {

    public First() {
        Scanner scan = new Scanner(System.in);
        int id = 0;

        while(true) {
            System.out.println("");
            System.out.printf("=" .repeat(55));
            System.out.println("");
            System.out.println("눈송이 도서관에 오신걸 환영합니다.\n 이용할 서비스를 선택해 주세요.");
            System.out.println("1.회원가입, 2.도서추가, 3.도서 검색 & 대출, 4.나의 도서 현황, 5.종료" );
            id = scan.nextInt();
            System.out.printf("=" .repeat(50));
            System.out.println("");
            if(id == 1) {
                Memberinsert member = new Memberinsert();
            }else if(id == 2) {
                Bookinsert book = new Bookinsert();
            }else if(id == 3) {
                Booksearch bsearch = new Booksearch();
            }else if(id == 4) {
                History history = new History();
            }else {
                System.out.println("프로그램을 종료합니다.");
                System.exit(0);
            }
        }
    }
}
```

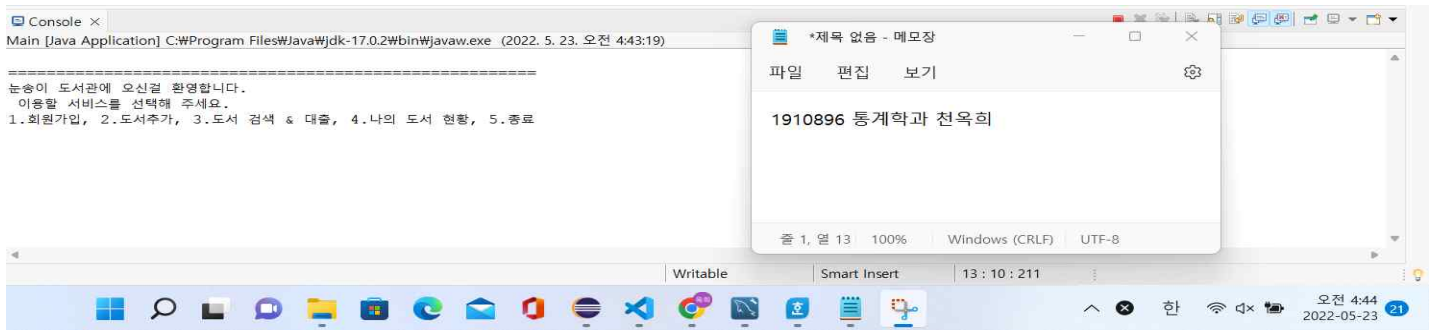
간단 요약

1번 : 회원가입 librarymember 테이블의 회원정보를 insert하는 함수 구현.

2번 : 도서 book 테이블의 도서 정보를 insert하는 함수 구현.

3번 : 도서를 검색하는 select함수 구현 및 rent 테이블에 대여한 도서 정보를 insert하는 함수 구현. 도서 정보의 rentstate 부분 업데이트.

4번 : 대출한 내역 및 모든 테이블 조인하여 예약한 내역 select함수로 나타내줌. 책 반납 및 예약 내역 delete하는 함수 구현.



1번으로 이동!

Memberinsert.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Scanner;
```

```
public class Memberinsert {
```

```
Scanner scan = new Scanner(System.in);
private Connection conn;
private static final String USERNAME = "root";
private static final String PASSWORD = "try1234";
private static final String URL = "jdbc:mysql://localhost:3306/library";
public Memberinsert(){
```

1. `String sql = "INSERT INTO libraryMember(mnum, pword, mname, major, email, birth)VALUES(?,?,?,?,?,?)";`

2. `String sql2 = "select * from libraryMember where mnum=?";`

```
try {
    System.out.println("회원가입");
```

```
Class.forName("com.mysql.jdbc.Driver");
conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);
```

```
PreparedStatement prestmt = conn.prepareStatement(sql);
```

```
PreparedStatement prestmt2 = conn.prepareStatement(sql2);
```

```
System.out.print("학과 : ");
String major = scan.next();
System.out.print("이름 : ");
String mname = scan.next();
```

```
int mnum = 0;
while(true) {
    System.out.print("학번 : ");
```

```
mnum = scan.nextInt();
```

```
        prestmt2.setInt(1, mnum);
        ResultSet result = prestmt2.executeQuery();

        int title = 0;
        while(result.next()) {
            title = result.getInt("mnum");
        }

        if(title == 0) {
            System.out.println("사용가능한 학번입니다.");
            break;
        }else {
            System.out.println("이미 회원가입한 학번입니다.");
        }
        break;
    }
}
```

```
System.out.print("비밀번호 : ");
String pword = scan.next();
System.out.print("이메일 : ");
String email = scan.next();
System.out.print("생년월일 : ");
String birth = scan.next();
```

```
    prestmt.setInt(1, mnum);
    prestmt.setString(2, pword);
    prestmt.setString(3, mname);
    prestmt.setString(4, major);
    prestmt.setString(5, email);
    prestmt.setString(6, birth);

    prestmt.executeUpdate();

    System.out.println("회원가입이 되었습니다.");
```

```
} catch (Exception e) {
    System.out.println("드라이버 로딩 실패 ");
    try {
        conn.close();
    } catch (SQLException e1) { }
}
}
```

```
}
```

```
1910896 cjsdhrm1 천옥희 통계학과 cjsdhrm1000@google.com 2000-08-26
```

```
회원가입  
Loading class `com.mysql.jdbc.  
학과 : 통계학과  
이름 : 천옥희  
학번 : 1910896  
이미 회원가입한 학번입니다.  
학번 :
```

위의 파란 박스를 보면, 입력된 학번을 조건으로 회원테이블(librarymember)의 정보를 가져와 정보가 있으면, 이미 가입한 회원으로 다른 학번을 사용하도록 해주었습니다.

오른쪽은 가입된 저의 학번을 넣어 봤고, 이미 가입한 학번이라 뜨고, 다시 학번을 입력하도록 하는 것을 볼 수 있습니다.

```
회원가입  
Loading class `com.mysql.jdbc.Driver'. This is deprecated. J  
학과 : it공학과  
이름 : 김눈송  
학번 : 2012345  
사용가능한 학번입니다.  
비밀번호 : rlasnsthd  
이메일 : rlasnsthd@gamil.com  
생년월일 : 2001-05-09  
회원가입이 되었습니다.
```

위의 빨간 박스 부분을 보면 insert함수로 회원 정보를 회원테이블(librarymember)에 입력하였으며, 데이터베이스에도 잘 올라갔음을 확인할 수 있습니다.

```
2012345 rlasnsthd 김눈송 it공학과 rlasnsthd@gamil.com 2001-05-09  
NULL NULL NULL NULL NULL NULL
```

```
=====  
눈송이 도서관에 오신걸 환영합니다.  
이용할 서비스를 선택해 주세요.  
1. 회원가입, 2. 도서추가, 3. 도서 검색 & 대출, 4. 나의 도서 현황, 5. 종료
```

2번 도서 추가하기로 넘어가 보도록 하겠습니다.

Bookinsert.java

```
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.PreparedStatement;  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.sql.Statement;  
import java.util.Scanner;
```

```
public class Bookinsert {
```

```
Scanner scan = new Scanner(System.in);  
private Connection conn;  
private static final String USERNAME = "root";  
private static final String PASSWORD = "try1234";  
private static final String URL = "jdbc:mysql://localhost:3306/library";  
public Bookinsert(){
```

```
String sql = "INSERT INTO Book(bnum, bname, author, publisher, bookyear, rentState,  
reserveState)VALUES(?,?,?,?,?,0,0)";
```

```
String sql2 = "select * from Book";
```

```
try {
```

```
System.out.println("이용하실 서비스를 골라주세요!");  
System.out.println("1.모든 도서 목록보기, 2.도서 추가");  
int id = scan.nextInt();
```

```

Class.forName("com.mysql.jdbc.Driver");
conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);
if(id == 1) {
    System.out.println("도서 목록");
    Statement stmt = conn.createStatement();
    ResultSet result = stmt.executeQuery(sql2);
    System.out.printf("%-5s %-15s %-8s %-8s %-15s %-5s\n","도서번호","도서이름","작가","출판사","
출판년도","대출여부");
    System.out.printf("-".repeat(70));
    System.out.println("");
    while(result.next()) {
        int bnum = result.getInt("bnum");
        String bname = result.getString("bname");
        String author = result.getString("author");
        String publisher = result.getString("publisher");
        String bookyear = result.getString("bookyear");
        int rentState = result.getInt("rentState");
        int reserveState = result.getInt("reserveState");
        String state = null;
        if(reserveState == 0 && rentState == 0 ) {
            state = "대출가능";
        }else if(rentState == 1) {
            state = "대출중";
        }else if (reserveState == 1&& rentState == 0) {
            state = "예약중";
        }
        System.out.format(" %-5d %-15s %-8s %-8s %-15s %-5s",bnum, bname, author,
publisher, bookyear, state);
        System.out.print("\n");
    }
}

```

```

}else if(id == 2) {
    System.out.println("도서 추가");

    PreparedStatement prestmt = conn.prepareStatement(sql);

    System.out.print("도서 번호 : ");
    int bnum = scan.nextInt();
    scan.nextLine();
    System.out.print("도서 이름 : ");
    String bname = scan.nextLine();
    System.out.print("작가 : ");
    String author = scan.next();
    System.out.print("출판사 : ");
    String publisher = scan.next();
    System.out.print("출판년도 : ");
    String bookyear = scan.next();
}

```

```

        prestmt.setInt(1, bnum);
        prestmt.setString(2, bname);
        prestmt.setString(3, author);
        prestmt.setString(4, publisher);
        prestmt.setString(5, bookyear);

        int cnt = prestmt.executeUpdate();

        System.out.println("도서가 추가되었습니다!");
    }
}

```

```

} catch (Exception e) {
    System.out.println("드라이버 로딩 실패 ");
    try {
        conn.close();
    } catch (SQLException e1) { }
}
}

```

도서 추가 안에서 모든 도서의 목록보기, 도서추가 2가지로 나눕니다. 먼저 1번은 위의 빨간색 박스를 통해 볼 수 있듯이 모든 책의 정보를 select함수를 통해 가져와 테이블 형식으로 만들어 주었습니다.

눈송이 도서관에 오신걸 환영합니다.
이용할 서비스를 선택해 주세요.
1. 회원가입, 2. 도서추가, 3. 도서 검색 & 대출, 4. 나의 도서 현황, 5. 종료

2

=====

이용하실 서비스를 골라주세요!
1. 모든 도서 목록보기, 2. 도서 추가

1

도서 목록

도서번호	도서이름	작가	출판사	출판년도	대출여부
1	Java의 정석	남궁성	도우출판	2019-12-30	대출가능
2	c++	황기태	생능출판	2018-10-16	대출가능
3	자바의 정석	남궁성	도우출판	2019-12-30	대출중
4	사회조사분석사	소정현	시대고사	2021-08-13	대출가능
5	작별인사	김영하	북북서가	2022-05-02	대출가능
6	기분을 관리하면 인생이 관리된다	김다슬	클라우드미	2022-04-12	대출가능
7	불편한 편의점	김호연	나무옆의자	2021-04-20	대출가능
8	나로서 충분히 괜찮은 사람	김재식	북로망스	2022-05-10	대출가능
9	책들의 부엌	김지혜	팩토리나인	2022-05-12	대출가능
10	금쪽이들의 진짜 마음속	오은영	오온라이프사이언스	2022-05-25	대출가능

=====

다음은 도서를 추가해보도록 하겠습니다.

이 부분은 위의 파란 박스를 통해 볼 수 있습니다. 위의 회원가입과 비슷하게 책의 정보를 입력된 값들을 받아 book 테이블에 추가할 수 있도록 했습니다.

이용할 서비스를 선택해 주세요.
1.회원가입, 2.도서추가, 3.도서 검색 & 대출, 4.나의 도서 현황, 5.종료

2

이용하실 서비스를 골라주세요!

1.모든 도서 목록보기, 2.도서 추가

2

Loading class 'com.mysql.jdbc.Driver'. This is deprecated. The new driver class is 'com.mysql.cj.jdbc.Driver'. The driver is automatically regi:

도서 추가

도서 번호 : 11

도서 이름 : 나에게 고맙다

작가 : 전승환

출판사 : 북로망스

출판년도 : 2022-02-18

도서가 추가되었습니다!

눈송이 도서관에 오신걸 환영합니다.

이용할 서비스를 선택해 주세요.

1.회원가입, 2.도서추가, 3.도서 검색 & 대출, 4.나의 도서 현황, 5.종료



이렇게 도서를 추가하면 도서가 추가했다는 문구가 뜹니다.

이를 한번 웹에서 Cross check 해보도록 하겠습니다.

Cross check- Insert

도서 내역					
도서번호	도서이름	작가	출판사	출판년도	대출여부
1	Java의 정석	남궁성	도우출판	2019-12-30	대출가능
2	c++	황기태	생능출판	2018-10-16	대출가능
3	자바의 정석	남궁성	도우출판	2019-12-30	대출중
4	사회조사분석사	소정현	시대고사	2021-08-13	대출가능
5	작별인사	김영하	북복서가	2022-05-02	대출가능
6	기분을 관리하면 인생이 관리된다	김다슬	클라우드아	2022-04-12	대출가능
7	불편한 편의점	김호연	나무옆의자	2021-04-20	대출가능
8	나로서 충분히 괜찮은 사람	김재식	북로망스	2022-05-10	대출가능
9	책들의 부엌	김지혜	팩토리나인	2022-05-12	대출가능
10	금쪽이들의 진짜 마음속	오은영	오은라이프사 이언스	2022-05-25	대출가능
11	나에게 고맙다	전승환	북로망스	2022-02-18	대출가능

제대로 추가되었음을 확인할 수 있습니다.

3번 도서 검색으로 넘어가도록 하겠습니다.

Booksearch.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Scanner;

public class Booksearch {
    Scanner scan = new Scanner(System.in);
    private Connection conn;
    private static final String USERNAME = "root";
    private static final String PASSWORD = "try1234";
    private static final String URL = "jdbc:mysql://localhost:3306/library";

    public Booksearch() {
        String sql = "select * from book where bname=?";

        try {
            System.out.println("도서 검색");

            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);

            PreparedStatement pstmt = conn.prepareStatement(sql);

            System.out.print("책 제목 : ");
            String bname = scan.nextLine();

            pstmt.setString(1, bname);
            ResultSet result = pstmt.executeQuery();
            System.out.printf("%-5s %-15s %-8s %-8s %-15s %-5s\n", "도서번호", "도서이름", "작가", "출판사", "출판년도", "대출여부");
            System.out.printf("-".repeat(70));
            System.out.println("");

            while(result.next()) {
                int bnum = result.getInt("bnum");
                String author = result.getString("author");
                String publisher = result.getString("publisher");
                String bookyear = result.getString("bookyear");
```

```

        int rentState = result.getInt("rentState");
        int reserveState = result.getInt("reserveState");
        String state = null;
        if(reserveState == 0 && rentState ==0 ) {
            state = "대출가능";
        }else if(rentState ==1) {
            state = "대출중";
        }else if (reserveState == 1&& rentState == 0) {
            state = "예약중";
        }
        System.out.format(" %-5d %-15s %-8s %-8s %-15s %-5s",bnum, bname,
author, publisher, bookyear, state);
        System.out.print("\n");
    }
}

```

```

System.out.println("대출하시겠습니까?\n1.예, 2.아니오");
int id = scan.nextInt();
if(id == 1) {
    Bookrent rent = new Bookrent();
}
else {
    First first = new First();
}
}

```

```

} catch (Exception e) {
    System.out.println("드라이버 로딩 실패 ");
    try {
        conn.close();
    } catch (SQLException e1) { }
}
}
}

```

```

}

```

빨간박스를 보면 도서제목을 검색하게 되면, 제목에 대한 정보가 나오게 되며, rentState(대여 여부)와 reserveState(예약 여부)로 도서가 대출 중인지, 예약 중인지, 대출 가능인지 확인할 수 있도록 하였습니다.

도서 검색					
책 제목 : 나에게 고맙다					
도서번호	도서이름	작가	출판사	출판년도	대출여부
11	나에게 고맙다	전승환	북로망스	2022-02-18	대출가능
대출하시겠습니까?					
1.예, 2.아니오					

검색을 하게 되면, 대출할 것인지 물어보도록 하였습니다. 이는 파란 박스를 보면 알 수 있듯이 1을 선택하게 되면, Bookrent클래스로 넘어가게 된다. bookrent 클래스를 한번 살펴보겠습니다.

Bookrent.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.Scanner;

public class Bookrent {
    Scanner scan = new Scanner(System.in);
    private Connection conn;

    private static final String USERNAME = "root";
    private static final String PASSWORD = "try1234";
    private static final String URL = "jdbc:mysql://localhost:3306/library";

    public Bookrent() {
        String sql = "select * from book where bnum=?";
        String sql2 = "select * from librarymember where mnum=?";
        String sql3 = "INSERT INTO rent(bnum, mnum, rentDate, returnDate, returnState)VALUES(?,?,Now(),?,0)";
        String sql4 = "UPDATE book SET rentState = 1 WHERE bnum in (SELECT bnum FROM rent WHERE book.bnum = rent.bnum and rent.bnum = ?)";

        Date now = new Date();
        Calendar cal = Calendar.getInstance();
        cal.setTime(now);
        cal.add(Calendar.DATE, 10);
        DateFormat df = new SimpleDateFormat("yyyy-MM-dd");
        String returndate = df.format(cal.getTime());

        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);

            PreparedStatement prestmt = conn.prepareStatement(sql);
            PreparedStatement prestmt2 = conn.prepareStatement(sql2);
            PreparedStatement prestmt3 = conn.prepareStatement(sql3);
            PreparedStatement prestmt4 = conn.prepareStatement(sql4);

            System.out.print("도서번호 : ");
            int bnum = scan.nextInt();
```

```

prestmt.setInt(1, bnum);
ResultSet result = prestmt.executeQuery();
while(result.next()) {
    int reserveState = result.getInt("reserveState");
    int rentState = result.getInt("rentState");
    if(reserveState == 0 && rentState == 0 ) {
        while(true) {

            System.out.print("학번 : ");
            int mnum = scan.nextInt();
            System.out.print("비밀번호 : ");
            String pword = scan.next();

            prestmt2.setInt(1, mnum);
            ResultSet result2 = prestmt2.executeQuery();

            int checkmnum = 0;
            String checkpword = null;

            while(result2.next()) {
                checkmnum = result2.getInt("mnum");
                checkpword = result2.getString("pword");
            }

            if(checkmnum == 0) {
                System.out.println("\n학번이 올바르지 않습니다.");
            }else {
                if(checkpword.equals(pword)) {
                    prestmt3.setInt(1, bnum);
                    prestmt3.setInt(2, mnum);
                    prestmt3.setString(3, returndate);

                    prestmt3.executeUpdate();

                    prestmt4.setInt(1, bnum);
                    prestmt4.executeUpdate();

                    System.out.println("대출되었습니다.");
                    break;
                }else {
                    System.out.println("비밀번호가 올바르지 않습니다.");
                }
            }
        }
    }
}

```

```

    }
}
else if(rentState ==1) {
    System.out.println("대출 중인 책입니다. 예약하시겠습니까?");
    System.out.print("1.예, 2.아니오 ");
    int id = scan.nextInt();
    if(id == 1) {
        Bookreserve reserve = new Bookreserve();
    }else {
        First first = new First();
    }
}

}else if (reserveState == 1 && rentState == 0) {
    System.out.println("대출 중인 책입니다. 예약하시겠습니까?");
    System.out.print("1.예, 2.아니오 ");
    int id = scan.nextInt();
    if(id == 1) {
        Bookreserve reserve = new Bookreserve();
    }else {
        First first = new First();
    }
}
}
} catch (Exception e) {
    System.out.println("드라이버 로딩 실패 ");
    try {
        conn.close();
    } catch (SQLException e1) { }
}
}
}

```

빨간 박스를 보면 도서 번호를 입력받아 book테이블을 도서번호를 조건으로 불러왔으며, reserveState(예약여부)와 rentState(대여여부)를 통해 대출과 예약을 나눴습니다. rentState와 reserveState 둘 중 하나라도 1이면 예약으로 넘어가고, 둘 다 0이면 대출로 넘어가도록 했습니다.

대출 먼저 살펴보도록 하겠습니다.

파란 박스의 sql2에서 입력받은 학번으로 회원 정보를 불러오고, 여기서 회원정보가 없으면 학번이 없다고 뜨며, 입력받은 비밀번호와 회원 정보테이블의 비밀번호가 다르다면 비밀번호가 다르다고 뜨면서 다시 입력할 수 있도록 했습니다. 둘다 일치하게 되면, 위 sql3쿼리문을 보면 알 수 있듯이 rent 테이블(대여테이블)에 대여 정보가 입력되는 것을 확인할 수 있습니다. 여기서 책이 대여 되면, book테이블의 rentState도 1로 변경되어야 합니다. 따라서 위 sql4를 보면 rent 테이블의 bnum(책번호)와 book테이블의 bnum(책번호)로 조인시켜 book테이블의 rentState를 업데이트 해줬습니다.

책 번호가 11번인 나에게 고맙다를 대출했습니다.

```

11  나에게 고맙다      전승환      북로망스      2022-02-18      대출가능
대출하시겠습니까?
1. 예, 2. 아니오
1
도서번호 : 11
학번 : 1910896
비밀번호 : cjsdhrqml
대출되었습니다.
=====

```

update에 대한 cross check를 해보도록 하겠습니다.

Cross check- update

insert crosscheck 했던 부분에서는 대출 여부가 대출 가능이었으나, 이번엔 rentState가 1로 변함으로서 대출중으로 바뀌었습니다.

도서 내역

도서번호	도서이름	작가	출판사	출판년도	대출여부
1	Java의 정석	남궁성	도우출판	2019-12-30	대출가능
2	c++	황기태	생능출판	2018-10-16	대출가능
3	자바의 정석	남궁성	도우출판	2019-12-30	대출중
4	사회조사분석사	소정현	시대고사	2021-08-13	대출가능
5	작별인사	김영하	북복서가	2022-05-02	대출가능
6	기분을 관리하면 인생이 관리된다	김다슬	클라우드아	2022-04-12	대출가능
7	불편한 편의점	김호연	나무옆의자	2021-04-20	대출가능
8	나로서 충분히 괜찮은 사람	김재식	북로망스	2022-05-10	대출가능
9	책들의 부엌	김지혜	팩토리나인	2022-05-12	대출가능
10	금쪽이들의 진짜 마음속	오은영	오은라이프사이언스	2022-05-25	대출가능
11	나에게 고맙다	전승환	북로망스	2022-02-18	대출중

다음은 대출이 아닌 예약하기로 들어가 보도록 하겠습니다.

Bookreserve.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Scanner;
```

```
public class Bookreserve {
    Scanner scan = new Scanner(System.in);
    private Connection conn;
    private static final String USERNAME = "root";
    private static final String PASSWORD = "try1234";
    private static final String URL = "jdbc:mysql://localhost:3306/library";
```

```
    public Bookreserve() {
        String sql = "select * from librarymember where mnum=?";
        String sql2 = "INSERT INTO reserve (bnum, mnum, reserveDate) VALUES (?, ?, NOW())";
        String sql3 = "UPDATE book SET reserveState=1 WHERE bnum in (SELECT bnum FROM reserve
WHERE book.bnum = reserve.bnum and book.bnum = ?)";
```

```
    try {
        Class.forName("com.mysql.jdbc.Driver");
        conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);

        PreparedStatement prestmt = conn.prepareStatement(sql);
        PreparedStatement prestmt2 = conn.prepareStatement(sql2);
        PreparedStatement prestmt3 = conn.prepareStatement(sql3);

        System.out.print("도서번호 : ");
        int bnum = scan.nextInt();
        System.out.print("학번 : ");
        int mnum = scan.nextInt();
        System.out.print("비밀번호 : ");
        String pword = scan.next();

        prestmt.setInt(1, mnum);
        ResultSet result = prestmt.executeQuery();

        int checkmnum = 0;
```

```

String checkpword = null;

while(result.next()) {
    checkmnum = result.getInt("mnum");
    checkpword = result.getString("pword");
}

if(checkmnum == 0) {
    System.out.println("\n학번이 올바르지 않습니다.");
}else {
    if(checkpword.equals(pword)) {
        prestmt2.setInt(1, bnum);
        prestmt2.setInt(2, mnum);

        prestmt2.executeUpdate();

        prestmt3.setInt(1, bnum);
        prestmt3.executeUpdate();
        System.out.println("예약되었습니다.");
    }else {
        System.out.println("비밀번호가 올바르지 않습니다.");
    }
}

} catch (Exception e) {
    System.out.println("드라이버 로딩 실패 ");
    try {
        conn.close();
    } catch (SQLException e1) { }
}
}
}

```

아까 대출하기와 비슷하기 때문에 간단하게 설명하도록 하겠습니다. 학번을 입력받아 회원 정보를 불러오고, 학번과 비밀번호가 일치해야지만, 예약할 수 있도록 했습니다. 또한, reserve 테이블에 예약 정보들을 입력할 수 있도록 했으며, book 테이블의 reserveState 또한 1로 업데이트 되도록 하였습니다.

```

도서 검색
책 제목 : 나에게 고맙다
도서번호  도서이름      작가      출판사      출판년도      대출여부
-----
11  나에게 고맙다      전승환      북로망스      2022-02-18      대출중
대출하시겠습니까?
1.예, 2.아니오
1
도서번호 : 11
대출 중인 책입니다. 예약하시겠습니까?
1.예, 2.아니오 1
도서번호 : 11
학번 : 190704
비밀번호 : thdduswj
예약되었습니다.

```


마지막으로 쉘 복잡한 4번 나의 도서 현황 보기로 가보도록하겠습니다.

History.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Scanner;

public class History {
    Scanner scan = new Scanner(System.in);
    private Connection conn;
    private static final String USERNAME = "root";
    private static final String PASSWORD = "try1234";
    private static final String URL = "jdbc:mysql://localhost:3306/library";

    public History() {
        String sql = "SELECT rnum, book.bnum, bname, author, publisher, rentDate, returnDate,
returnState "
                + "FROM book, rent, librarymember "
                + "WHERE librarymember.mnum = ? "
                + "AND librarymember.pword = ? "
                + "AND rent.mnum = librarymember.mnum "
                + "AND book.bnum = rent.bnum";

        String sql2 = "SELECT reserve.Renum, book.bnum, bname, author, publisher, reserveDate,
rentDate"
                + " FROM reserve"
                + " left join rent"
                + "     on reserve.mnum = rent.mnum"
                + "     and reserve.rnum = rent.rnum"
                + " left join book"
                + "     on reserve.bnum = book.bnum "
                + " INNER JOIN librarymember"
                + "     on reserve.mnum = librarymember.mnum"
                + "     and librarymember.mnum = ?"
                + "     and librarymember.pword = ?";

        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);

            PreparedStatement prestmt = conn.prepareStatement(sql);
            PreparedStatement prestmt2 = conn.prepareStatement(sql2);

            System.out.println("이용할 서비스를 선택해 주세요.");
            System.out.print("1.대출내역보기, 2.예약내역보기 ");
            int id = scan.nextInt();

            System.out.print("학번 : ");
            int mnum = scan.nextInt();
            System.out.print("비밀번호 : ");
            String pword = scan.next();

            if(id == 1) {
                prestmt.setInt(1, mnum);
                prestmt.setString(2, pword);
                ResultSet result = prestmt.executeQuery();

                System.out.printf("%-5s %-5s %-15s %-8s %-8s %-15s %-15s\n", "대출번호", "도서번호", "도서이름", "작가", "출판사", "예약일", "대출일", "반납");
                System.out.print("-".repeat(100));
                System.out.println("");
                while(result.next()) {
                    int rnum = result.getInt("rnum");
                    int bnum = result.getInt("bnum");
                }
            }
        }
    }
}
```

모든 테이블 조인!!

```

        String bname = result.getString("bname");
        String author = result.getString("author");
        String publisher = result.getString("publisher");
        String rentDate = result.getString("rentDate");
        String returnDate = result.getString("returnDate");
        int returnState = result.getInt("returnState");
        String state = null;
        if(returnState == 0) {
            state = "반납X";
        }else {
            state = "반납O";
        }
        System.out.format("%-5s %-5d %-15s %-8s %-8s %-15s %-15s
%-5s",rnum, bnum, bname, author, publisher, rentDate,returnDate, state);
        System.out.print("\n");
    }

```

```

        System.out.print("1.반납하기, 2.처음으로 돌아가기 ");
        int id2 = scan.nextInt();
        if(id2 == 1) {
            Bookreturn bookreturn = new Bookreturn();
        }else {
            First first = new First();
        }
    }else {
        prestmt2.setInt(1, mnum);
        prestmt2.setString(2, pword);
        ResultSet result2 = prestmt2.executeQuery();

        System.out.printf("%-5s %-5s %-15s %-8s %-8s %-15s %-15s
%-5s\n","예약번호","도서번호","도서이름","작가","출판사","예약일","대출일","반납");
        System.out.printf("-".repeat(100));
        System.out.println("");
        while(result2.next()) {
            int Renum = result2.getInt("Renum");
            int bnum = result2.getInt("bnum");
            String bname = result2.getString("bname");
            String author = result2.getString("author");
            String publisher = result2.getString("publisher");
            String reserveDate = result2.getString("reserveDate");
            String rentDate = result2.getString("rentDate");
            String state = null;
            if(rentDate == null) {
                state = "취소O";
            }else {
                state = "취소X";
            }
            System.out.format("%-5d %-5d %-15s %-8s %-8s %-15s %-15s %-5s",Renum, bnum,
bname, author, publisher, reserveDate, rentDate, state);
            System.out.print("\n");
        }

        System.out.print("1.예약 취소 하기, 2.예약 순위 보기 3. 처음으로 돌아가기 ");
        int id3 = scan.nextInt();
        if(id3 == 1) {
            Reservedelete reservedelete = new Reservedelete();
        }else if(id3 == 2){
            Reserverank rank = new Reserverank();
        }else {
            First first = new First();
        }
    }
}

```

```

}
}catch (Exception e) {
    System.out.println("드라이버 로딩 실패 ");
    try {
        conn.close();
    } catch (SQLException e1) {
    }
}
}
}

```

여기서는 대출 내역 보기와 예약 내역 보기로 나뉜다. 대출 내역보기에서는 반납하기로 넘어가고, 예약 내역보기는 예약 취소하고, 예약 순서 보기로 나눠집니다.

먼저 대출 내역 보기부터 보도록 하겠습니다.

파란색 부분이 대출 내역보기 입니다. 여기서는 rent, librarymember, book 3개의 테이블을 조인해 대출 내역을 볼 수 있게 하였습니다.

1. 회원가입, 2. 도서추가, 3. 도서 검색 & 대출, 4. 나의 도서 현황, 5. 종료

4

=====

이용할 서비스를 선택해 주세요.
1. 대출내역보기, 2. 예약내역보기 1

학번 : 1910896
비밀번호 : cjsdhrgm1

대출번호	도서번호	도서이름	작가	출판사	예약일	대출일	반납
27	1	Java의 정석	남궁성	도우출판	2022-05-22 01:05:19	2022-05-31	반납O
30	2	c++	황기태	생능출판	2022-05-22 01:52:03	2022-05-31	반납O
33	2	c++	황기태	생능출판	2022-05-22 02:08:42	2022-05-31	반납O
35	2	c++	황기태	생능출판	2022-05-22 03:00:43	2022-05-31	반납O
37	2	c++	황기태	생능출판	2022-05-22 03:18:13	2022-05-31	반납O
39	3	자바의 정석	남궁성	도우출판	2022-05-22 03:20:09	2022-05-31	반납O
40	1	Java의 정석	남궁성	도우출판	2022-05-22 03:25:15	2022-05-31	반납O
42	3	자바의 정석	남궁성	도우출판	2022-05-22 15:11:25	2022-06-01	반납O
43	3	자바의 정석	남궁성	도우출판	2022-05-22 15:18:55	2022-06-01	반납O
44	2	c++	황기태	생능출판	2022-05-23 02:39:22	2022-06-02	반납O
45	3	자바의 정석	남궁성	도우출판	2022-05-23 03:07:40	2022-06-02	반납O
47	11	나에게 고맙다	전승환	북로망스	2022-05-23 06:09:43	2022-06-02	반납X

1. 반납하기, 2. 처음으로 돌아가기

이어서 바로 반납할 수 있도록 해렸습니다.

Bookreturn.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Scanner;

public class Bookreturn {
    Scanner scan = new Scanner(System.in);
    private Connection conn;
    private static final String USERNAME = "root";
    private static final String PASSWORD = "try1234";
    private static final String URL = "jdbc:mysql://localhost:3306/library";

    public Bookreturn() {
        String sql = "UPDATE rent SET returnState = 1 WHERE rnum = ?";
        String sql2 = "UPDATE book, rent SET rentState = 0 WHERE rnum = ? AND book.bnum = rent.bnum";

        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);

            PreparedStatement prestmt = conn.prepareStatement(sql);
            PreparedStatement prestmt2 = conn.prepareStatement(sql2);

            System.out.println("도서 반납");
            System.out.print("대출 번호 : ");
            int rnum = scan.nextInt();

            prestmt.setInt(1, rnum);
            prestmt.executeUpdate();

            prestmt2.setInt(1, rnum);
            prestmt2.executeUpdate();

            System.out.println("반납되었습니다!");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```

    } catch (Exception e) {
        System.out.println("드라이버 로딩 실패 ");
        try {
            conn.close();
        } catch (SQLException e1) { }
    }
}
}
}

```

입력받은 대출번호로 rent테이블의 returnState(반납여부)부분을 1로 업데이트 시켜주었으며, 대출에서 book 테이블의 rentState를 1로 시켜줬던 것을 반납해줬으니 0으로 업데이트해주었습니다.

다음은 **History.java**의 예약내역 부분입니다.
 이는 빨간 박스로 표시 해놓았습니다. 여기서 1. 예약 취소, 2. 예약순위 보기입니다.
 먼저 예약 취소를 보도록 하겠습니다.

Reservedelete.java

```

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Scanner;

```

```

public class Reservedelete {
    Scanner scan = new Scanner(System.in);
    private Connection conn; //DB 커넥션 연결 객체
    private static final String USERNAME = "root";//DBMS접속 시 아이디
    private static final String PASSWORD = "try1234";//DBMS접속 시 비밀번호
    private static final String URL = "jdbc:mysql://localhost:3306/library";//DBMS접속 url

```

```

    public Reservedelete() {
        String sql = "DELETE FROM reserve where Renum = ?";
        String sql2 = "UPDATE book, reserve SET reserveState = 0 where book.bnum = ? and not exists (select Renum from reserve where reserve.bnum = ? and reserve.bnum = book.bnum and reserve.rnum is null)";
    }

```

```

        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);

            PreparedStatement prestmt = conn.prepareStatement(sql);
            PreparedStatement prestmt2 = conn.prepareStatement(sql2);

```

```

            System.out.println("예약 도서 취소");
            System.out.print("예약 번호 : ");
            int Renum = scan.nextInt();
            System.out.print("도서 번호 : ");
            int bnum = scan.nextInt();

```

```

            prestmt.setInt(1, Renum);
            prestmt.executeUpdate();

```

```

            prestmt2.setInt(1, bnum);
            prestmt2.setInt(2, bnum);
            prestmt2.executeUpdate();

```

```

            System.out.println("예약이 취소되었습니다.!");

```

```

        } catch (Exception e) {
            System.out.println("드라이버 로딩 실패 ");
            try {
                conn.close();
            } catch (SQLException e1) { }
        }
    }
}

```

빨간 박스의 sql을 보면 입력받은 예약번호의 예약 정보를 삭제하도록 했으며, 예약 정보가 삭제됐으니 book의 reserveState도 0이 되도록 해야합니다. 그런데 무조건 내 예약 정보만이 사라졌다고 그 책을 예약한 사람이 모두 사라지는 것이 아니기 때문에 reserve테이블에서 rnum이 null인 것들만 예약중인 것이다. 따라서 is null이 존재하지 않을 경우에 0으로 바뀌도록 하였다.

눈송이 도서관에 오신걸 환영합니다.

이용할 서비스를 선택해 주세요.

1.회원가입, 2.도서추가, 3.도서 검색 & 대출, 4.나의 도서 현황, 5.종료

4

이용할 서비스를 선택해 주세요.

1.대출내역보기, 2.예약내역보기 2

학번 : 190704

비밀번호 : thdduswjdd

예약번호	도서번호	도서이름	작가	출판사	예약일	대출일	반납
11	1	Java의 정석	남궁성	도우출판	2022-05-22 01:05:31	2022-05-22 01:06:42	취소X
13	2	c++	황기태	생능출판	2022-05-22 01:52:22	2022-05-22 01:53:06	취소X
17	2	c++	황기태	생능출판	2022-05-22 03:01:04	2022-05-22 03:01:25	취소X
19	2	c++	황기태	생능출판	2022-05-22 03:18:27	2022-05-22 03:18:52	취소X
21	1	Java의 정석	남궁성	도우출판	2022-05-22 03:28:09	2022-05-22 03:28:33	취소X
28	3	자바의 정석	남궁성	도우출판	2022-05-23 03:33:04	null	취소O
29	11	나에게 고맙다	전숙화	북로망스	2022-05-23 06:17:08	null	취소O

예약내역을 봤으니 ‘나에게 고맙다’ 책의 예약을 취소하도록 하겠습니다.

1.예약 취소 하기, 2.예약 순위 보기 3. 처음으로 돌아가기 1
예약 도서 취소
예약 번호 : 29
도서 번호 : 11
예약이 취소되었습니다.!

취소한(삭제) 내역을 웹에서 Cross Check하도록 하겠습니다.

Cross Check - delete

<예약 내역>

도서번호	도서이름	작가	출판사	예약일	대출일	예약취소
1	Java의 정석	남궁성	도우출판	2022-05-22 01:05:31	2022-05-22 01:06:42	대출완료
2	c++	황기태	생능출판	2022-05-22 01:52:22	2022-05-22 01:53:06	대출완료
2	c++	황기태	생능출판	2022-05-22 03:01:04	2022-05-22 03:01:25	대출완료
2	c++	황기태	생능출판	2022-05-22 03:18:27	2022-05-22 03:18:52	대출완료
1	Java의 정석	남궁성	도우출판	2022-05-22 03:28:09	2022-05-22 03:28:33	대출완료
3	자바의 정석	남궁성	도우출판	2022-05-23 03:33:04		예약취소

나에게 고맙다 부분이 잘 삭제되었음을 확인 할 수 있습니다.

Reserverank.java

마지막으로 예약 순위 보기입니다.

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.Scanner;

public class Reserverank {
    Scanner scan = new Scanner(System.in);
    private Connection conn; //DB 커넥션 연결 객체
    private static final String USERNAME = "root";//DBMS접속 시 아이디
    private static final String PASSWORD = "try1234";//DBMS접속 시 비밀번호
    private static final String URL = "jdbc:mysql://localhost:3306/library";//DBMS접속 url

    public Reserverank() {
        String sql = "select mnum, book.bnum, rentState,rank() over(order by Renum) As ranking
from reserve, book where reserve.bnum = ? and reserve.bnum = book.bnum and rnum is null";
        String sql2 = "SELECT * FROM libraryMember where mnum = ?";
        String sql3 = "INSERT INTO rent(bnum, mnum, rentDate, returnDate,
returnState)VALUES(?,?,Now(),?,0)";
        String sql4 = "UPDATE book SET rentState = 1 WHERE bnum = ?";
        String sql5 = "UPDATE reserve, rent SET reserve.rnum= rent.rnum WHERE rent.mnum =
? AND rent.bnum = ? AND rent.returnState = 0 AND reserve.bnum = rent.bnum AND reserve.mnum =
rent.mnum AND reserve.rnum is null";
        String sql6 = "UPDATE book, reserve SET reserveState = 0 where book.bnum = ? and not
exists (select Renum from reserve where reserve.bnum = ? and reserve.bnum = book.bnum and
reserve.rnum is null)";
        Date now = new Date();
        Calendar cal = Calendar.getInstance();
        cal.setTime(now);
        cal.add(Calendar.DATE, 10);
        DateFormat df = new SimpleDateFormat("yyyy-MM-dd");
        String returndate = df.format(cal.getTime());
        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);

            PreparedStatement prestmt = conn.prepareStatement(sql);
            PreparedStatement prestmt2 = conn.prepareStatement(sql2);
            PreparedStatement prestmt3 = conn.prepareStatement(sql3);
            PreparedStatement prestmt4 = conn.prepareStatement(sql4);
            PreparedStatement prestmt5 = conn.prepareStatement(sql5);
            PreparedStatement prestmt6 = conn.prepareStatement(sql6);

            System.out.println("예약 순위");
            System.out.print("도서 번호 : ");
            int bnum = scan.nextInt();

            prestmt.setInt(1, bnum);
            ResultSet result = prestmt.executeQuery();
```

```

System.out.format("%-10s %-3s %-6s\n","학번","순위","대출여부");
System.out.printf("-".repeat(30));
System.out.println("");

int mnumcheck = 0;
int mnum1 = 0;
while(result.next()) {
    int mnum = result.getInt("mnum");
    int rentState = result.getInt("rentState");
    int ranking = result.getInt("ranking");
    String rent = null;
    if(rentState == 0 && ranking == 1) {
        rent = "대출가능";
    }else {
        rent = "대출대기";
    }
    System.out.format("%-10d %-5d %-6s", mnum, ranking, rent);
    System.out.println("");
    if (rent.equals("대출가능")) {
        mnumcheck=1;
        mnum1 = result.getInt("mnum");
    }
}
System.out.println(mnum1+"님 입니까? 1. 네, 2. 아니오");
int id = scan.nextInt();
if(id == 1) {
    System.out.println("대출하시겠습니까? 1. 네, 2. 아니오");
    int id2 = scan.nextInt();
    if(id2 == 1) {
        if(mnumcheck == 1) {
            while(true) {
                System.out.print("비밀번호 : ");
                String pword = scan.next();

                pstmt2.setInt(1, mnum1);
                ResultSet result2 = pstmt2.executeQuery();

                while(result2.next()) {
                    String pwordcheck = result2.getString("pword");
                    if(pword.equals(pwordcheck)) {
                        pstmt3.setInt(1, bnum);
                        pstmt3.setInt(2, mnum1);
                        pstmt3.setString(3, returndate);

                        pstmt3.executeUpdate();

                        pstmt4.setInt(1, bnum);

                        pstmt4.executeUpdate();

                        pstmt5.setInt(1, mnum1);
                        pstmt5.setInt(2, bnum);

                        pstmt5.executeUpdate();

                        pstmt6.setInt(1, bnum);
                        pstmt6.setInt(2, bnum);

                        pstmt6.executeUpdate();
                    }
                }
            }
        }
    }
}

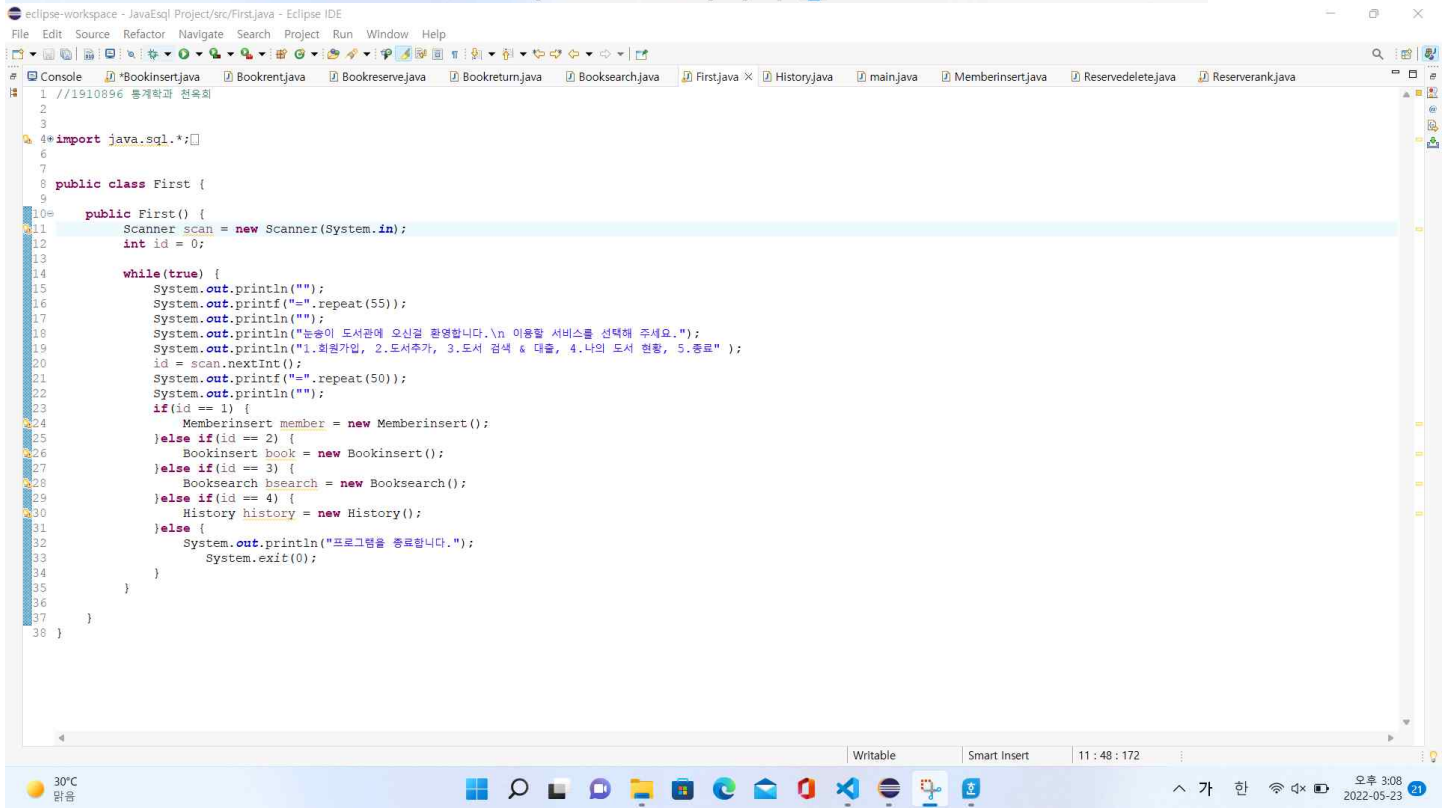
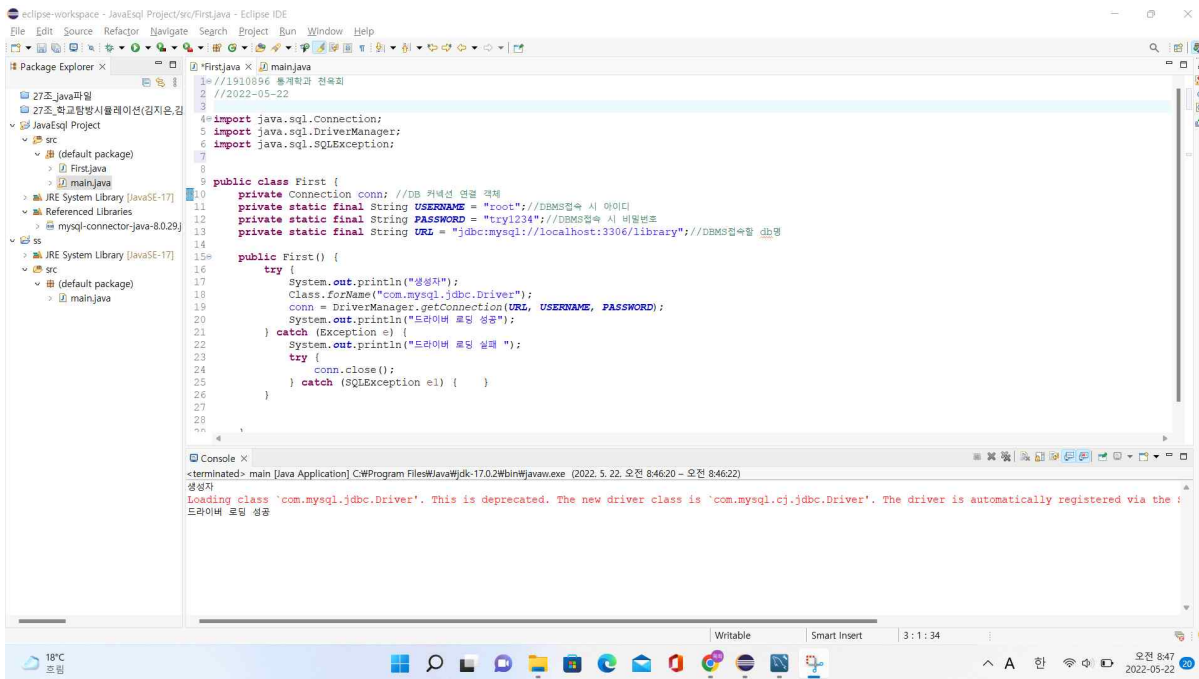
```


빨간 박스 부분을 보면 입력한 책 번호의 예약순위가 보이고, 만약 대출했던 사람이 반납을 하면, 순위가 1등인 사람이 바로 대출할 수 있도록 하였습니다. 이 부분은 파란 박스 부분인데 아까 대출하기랑 비슷하기 때문에 간단하게 설명하도록 하겠습니다. 1번 순위인 학번으로 회원정보를 불러와 입력한 비밀번호와 회원정보가 맞으면 rent테이블에 대여 정보가 입력되고, book 테이블의 rentState는 1로 변경됩니다. 또한, 이는 예약한 후 대출한 것이기 때문에 book테이블의 reserveState도 변경 해줘야합니다. 삭제 했었던것과 같이 reserveState의 rnum이 null인 부분은 아직 예약이 된 것이기 때문에 0으로 바꾸면 안됩니다. 그래서 파란 박스의 sql6처럼 해주었습니다.

웹이 일찍 끝날 줄 알고, 웹 먼저 시작했다가 일주일이라는 시간이 걸려 hll부분은 2일 정도 밤새면서 구현했습니다.

7시부터 일어나 <https://sso-feeling.tistory.com/105> 를 보며 이클립스와 mysql를 연동시키는데 성공했습니다. 처음엔 객체지향시간 때 배운 내용으로 panel로 예쁘게 꾸며보고자 시도했지만, 1시간 동안 다시 panel을 공부하는데 시간을 썼습니다. web언어에서도 꾸미는데 시간이 오래 걸리기도 했고, 더 이상 지체할 시간도 없었기에 밥먹는 시간, 알바가 있는 시간 빼고는 컴퓨터 앞에 앉아있었습니다. web언어를 하면서 조인 등 어떻게 연결해야 할 지에 대해 이미 많이 시간을 쏟고 고민했기 때문에 고민하는 시간은 많이 들지 않았습니다. 하지만, 중간중간 오류가 뜨기도 했고, 알바를 끝나고 너무 피곤해 포기하고 싶었지만, 꼭 완성해서 제출하고자하는 의지로 22~23일 밤새가며 완료했습니다.

각 파일을 캡처하는 것을 잊어버려 뒤늦게라도 캡처 화면 올립니다!



eclipse-workspace - JavaEsq Project/src/Bookinsrt.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

JavaEsq Project

src

(default package)

Bookinsrt.java

Bookkrent.java

Bookreserve.java

Bookkretum.java

Booksearch.java

First.java

History.java

main.java

Memberinsrt.java

Reservedelete.java

Reserverank.java

JRE System Library

Referenced Libraries

Console

Bookinsrt.java

Bookkrent.java

Bookreserve.java

Bookkretum.java

Booksearch.java

First.java

History.java

main.java

Memberinsrt.java

Reservedelete.java

Reserverank.java

1 //1910896 통계학과 천우희

2

3 import java.sql.Connection;

4 import java.sql.DriverManager;

5 import java.sql.PreparedStatement;

6 import java.sql.ResultSet;

7 import java.sql.SQLException;

8 import java.sql.Statement;

9 import java.util.Scanner;

10

11 public class Bookinsrt {

12

13 Scanner scan = new Scanner(System.in);

14 private Connection conn;

15 private static final String USERNAME = "root";

16 private static final String PASSWORD = "try1234";

17 private static final String URL = "jdbc:mysql://localhost:3306/library";

18 public Bookinsrt() {

19 String sql = "INSERT INTO Book(bnum, bname, author, publisher, bookyear, rentState, reserveState) VALUES (?, ?, ?, ?, ?, 0, 0)";

20 String sql2 = "select * from Book";

21 try {

22 System.out.println("이용하실 서비스를 알려주세요!");

23 System.out.println("1. 모든 도서 목록보기, 2. 도서 추가");

24 int id = scan.nextInt();

25 Class.forName("com.mysql.jdbc.Driver");

26 conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);

27 if(id == 1) {

28 System.out.println("도서 목록");

29 Statement stmt = conn.createStatement();

30 ResultSet result = stmt.executeQuery(sql2);

31 System.out.printf("%-5s %-15s %-8s %-15s %-5s\n", "도서번호", "도서이름", "작가", "출판사", "출판년도", "대출여부");

32 System.out.printf("-".repeat(70));

33 System.out.println("");

34 while(result.next()) {

35 int bnum = result.getInt("bnum");

36 String bname = result.getString("bname");

37 String author = result.getString("author");

38 String publisher = result.getString("publisher");

39 String bookyear = result.getString("bookyear");

40 int rentState = result.getInt("rentState");

41 int reserveState = result.getInt("reserveState");

42 String state = null;

43 if(reserveState == 0 && rentState == 0) {

Bookinsrt.java - JavaEsq Project/src

30°C 맑음

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

JavaEsq Project

src

(default package)

Bookinsrt.java

Bookkrent.java

Bookreserve.java

Bookkretum.java

Booksearch.java

First.java

History.java

main.java

Memberinsrt.java

Reservedelete.java

Reserverank.java

JRE System Library

Referenced Libraries

Console

Bookinsrt.java

Bookkrent.java

Bookreserve.java

Bookkretum.java

Booksearch.java

First.java

History.java

main.java

Memberinsrt.java

Reservedelete.java

Reserverank.java

49 System.out.format("%-5d %-15s %-8s %-15s %-5s", bnum, bname, author, publisher, bookyear, state);

50 System.out.print("\n");

51 }

52 }

53 }else if(id == 2) {

54 System.out.println("도서 추가");

55

56 PreparedStatement pstmt = conn.prepareStatement(sql);

57

58 System.out.print("도서 번호 : ");

59 int bnum = scan.nextInt();

60 scan.nextLine();

61 System.out.print("도서 이름 : ");

62 String bname = scan.nextLine();

63 System.out.print("작가 : ");

64 String author = scan.next();

65

66 System.out.print("출판사 : ");

67 String publisher = scan.next();

68 System.out.print("출판년도 : ");

69 String bookyear = scan.next();

70

71 pstmt.setInt(1, bnum);

72 pstmt.setString(2, bname);

73 pstmt.setString(3, author);

74 pstmt.setString(4, publisher);

75 pstmt.setString(5, bookyear);

76

77 int cnt = pstmt.executeUpdate();

78

79 System.out.println("도서가 추가되었습니다!");

80 }

81 }

82 } catch (Exception e) {

83 System.out.println("드라이버 로딩 실패 ");

84 try {

85 conn.close();

86 } catch (SQLException e1) { }

87 }

88 }

89 }

90 }

91 }

Bookinsrt.java - JavaEsq Project/src

30°C 맑음

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

JavaEsq Project

src

(default package)

Bookinsrt.java

Bookkrent.java

Bookreserve.java

Bookkretum.java

Booksearch.java

First.java

History.java

main.java

Memberinsrt.java

Reservedelete.java

Reserverank.java

JRE System Library

Referenced Libraries

Console

Bookinsrt.java

Bookkrent.java

Bookreserve.java

Bookkretum.java

Booksearch.java

First.java

History.java

main.java

Memberinsrt.java

Reservedelete.java

Reserverank.java

오후 3:12 2022-05-23

```
1 //1910896 출제학과 전유희
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5 import java.sql.PreparedStatement;
6 import java.sql.ResultSet;
7 import java.sql.SQLException;
8 import java.text.DateFormat;
9 import java.text.SimpleDateFormat;
10 import java.util.Calendar;
11 import java.util.Date;
12 import java.util.Scanner;
13
14 public class Bookrent {
15     Scanner scan = new Scanner(System.in);
16     private Connection conn;
17     private static final String USERNAME = "root";
18     private static final String PASSWORD = "try1234";
19     private static final String URL = "jdbc:mysql://localhost:3306/library";
20
21     public Bookrent() {
22         String sql1 = "select * from book where bnum=?";
23         String sql2 = "select * from librarymember where lnum=?";
24         String sql3 = "INSERT INTO rent(bnum, lnum, rentDate, returnDate, returnState) VALUES(?, ?, Now(), ?, 0)";
25         String sql4 = "UPDATE book SET rentState = 1 WHERE bnum in (SELECT bnum FROM rent WHERE book.bnum = rent.bnum and rent.bnum = ?)";
26         Date now = new Date();
27         Calendar cal = Calendar.getInstance();
28         cal.setTime(now);
29         cal.add(Calendar.DATE, 10);
30         DateFormat df = new SimpleDateFormat("yyyy-MM-dd");
31         String returndate = df.format(cal.getTime());
32
33         try {
34             Class.forName("com.mysql.jdbc.Driver");
35             conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);
36
37             PreparedStatement pstmt1 = conn.prepareStatement(sql1);
38             PreparedStatement pstmt2 = conn.prepareStatement(sql2);
39             PreparedStatement pstmt3 = conn.prepareStatement(sql3);
40             PreparedStatement pstmt4 = conn.prepareStatement(sql4);
41
42             System.out.println("도서번호 : ");
43             int bnum = scan.nextInt();
```

```
53         System.out.print("도판 : ");
54         int lnum = scan.nextInt();
55         System.out.print("비밀번호 : ");
56         String pword = scan.next();
57
58         pstmt2.setInt(1, lnum);
59         ResultSet result2 = pstmt2.executeQuery();
60
61         int checknum = 0;
62         String checkpword = null;
63
64         while(result2.next()) {
65             checknum = result2.getInt("lnum");
66             checkpword = result2.getString("pword");
67         }
68
69         if(checknum == 0) {
70             System.out.println("\n책번호를 올바르게 입력하셨습니다.");
71         } else {
72             if(checkpword.equals(pword)) {
73                 pstmt3.setInt(1, bnum);
74                 pstmt3.setInt(2, lnum);
75                 pstmt3.setString(3, returndate);
76                 pstmt3.executeUpdate();
77
78                 pstmt4.setInt(1, bnum);
79                 pstmt4.executeUpdate();
80
81                 System.out.println("대출되었습니다.");
82                 break;
83             } else {
84                 System.out.println("비밀번호가 올바르지 않습니다.");
85             }
86         }
87     }
88
89     }
90
91     else if(rentState == 1) {
92         System.out.println("대출 중인 책입니다. 예약하시겠습니까?");
93         System.out.print("1.예, 2.아니오 ");
94         int id = scan.nextInt();
95         if(id == 1) {
96             Bookreserve reserve = new Bookreserve();
97             First first = new First();
98
99         }
100     }
101
102     else if (reserveState == 1 && rentState == 0) {
103         System.out.println("대출 중인 책입니다. 예약하시겠습니까?");
104         System.out.print("1.예, 2.아니오 ");
105         int id = scan.nextInt();
106         if(id == 1) {
107             Bookreserve reserve = new Bookreserve();
108             First first = new First();
109
110         }
111     }
112 } catch (Exception e) {
113     System.out.println("드라이버 로드 실패 ");
114 } try {
115     conn.close();
116 } catch (SQLException e1) {
117 }
118 }
119
120 }
```

```
78         pstmt3.executeUpdate();
79
80         pstmt4.setInt(1, bnum);
81         pstmt4.executeUpdate();
82
83         System.out.println("대출되었습니다.");
84         break;
85     } else {
86         System.out.println("비밀번호가 올바르지 않습니다.");
87     }
88 }
89
90 }
91
92 else if(rentState == 1) {
93     System.out.println("대출 중인 책입니다. 예약하시겠습니까?");
94     System.out.print("1.예, 2.아니오 ");
95     int id = scan.nextInt();
96     if(id == 1) {
97         Bookreserve reserve = new Bookreserve();
98         First first = new First();
99
100     }
101
102     else if (reserveState == 1 && rentState == 0) {
103         System.out.println("대출 중인 책입니다. 예약하시겠습니까?");
104         System.out.print("1.예, 2.아니오 ");
105         int id = scan.nextInt();
106         if(id == 1) {
107             Bookreserve reserve = new Bookreserve();
108             First first = new First();
109
110         }
111     }
112 } catch (Exception e) {
113     System.out.println("드라이버 로드 실패 ");
114 } try {
115     conn.close();
116 } catch (SQLException e1) {
117 }
118 }
119
120 }
```

eclipse-workspace - JavaEsql Project/src/Bookreserve.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Console Bookinsert... Bookrentjava Bookreserve... Bookreturnjava Booksearchjava Firstjava Historyjava mainjava Memberinser... Reserverank...

```
1 import java.sql.Connection;
2
3 public class Bookreserve {
4     Scanner scan = new Scanner(System.in);
5     private Connection conn;
6     private static final String USERNAME = "root";
7     private static final String PASSWORD = "try1234";
8     private static final String URL = "jdbc:mysql://localhost:3306/library";
9
10    public Bookreserve() {
11        String sql = "select * from librarymember where mnum=?";
12        String sql2 = "INSERT INTO reserve (bnum, mnum, reserveDate) VALUES (?, ?, NOW())";
13        String sql3 = "UPDATE book SET reserveState=1 WHERE bnum in (SELECT bnum FROM reserve WHERE book.bnum = reserve.bnum and book.bnum = ?)";
14
15        try {
16            Class.forName("com.mysql.jdbc.Driver");
17            conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);
18
19            PreparedStatement pstmt = conn.prepareStatement(sql);
20            PreparedStatement pstmt2 = conn.prepareStatement(sql2);
21            PreparedStatement pstmt3 = conn.prepareStatement(sql3);
22
23            System.out.print("도서번호 : ");
24            int bnum = scan.nextInt();
25            System.out.print("학번 : ");
26            int mnum = scan.nextInt();
27            System.out.print("비밀번호 : ");
28            String pword = scan.next();
29
30            pstmt.setInt(1, mnum);
31            ResultSet result = pstmt.executeQuery();
32
33            int checkmnum = 0;
34            String checkpword = null;
35
36            while(result.next()) {
37                checkmnum = result.getInt("mnum");
38                checkpword = result.getString("pword");
39            }
40
41            if(checkmnum == 0) {
42                System.out.println("\n학번이 올바르지 않습니다.");
43            }
44        } catch (Exception e) {
45            System.out.println("드라이버 로딩 실패");
46        }
47    }
48 }
```

30°C 맑음

Writable Smart Insert 13:77:465

오후 3:15 2022-05-23

eclipse-workspace - JavaEsql Project/src/Bookreserve.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

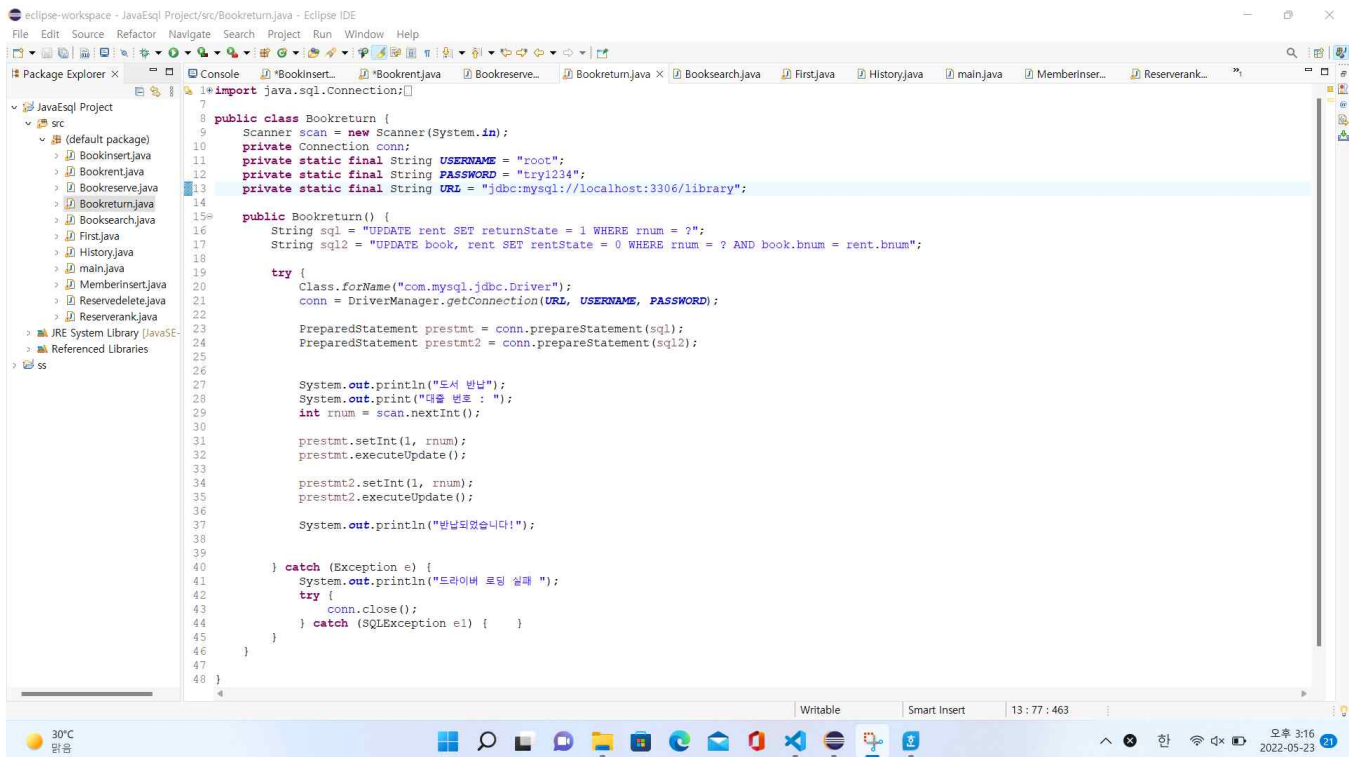
Package Explorer Console Bookinsert... Bookrentjava Bookreserve... Bookreturnjava Booksearchjava Firstjava Historyjava mainjava Memberinser... Reserverank...

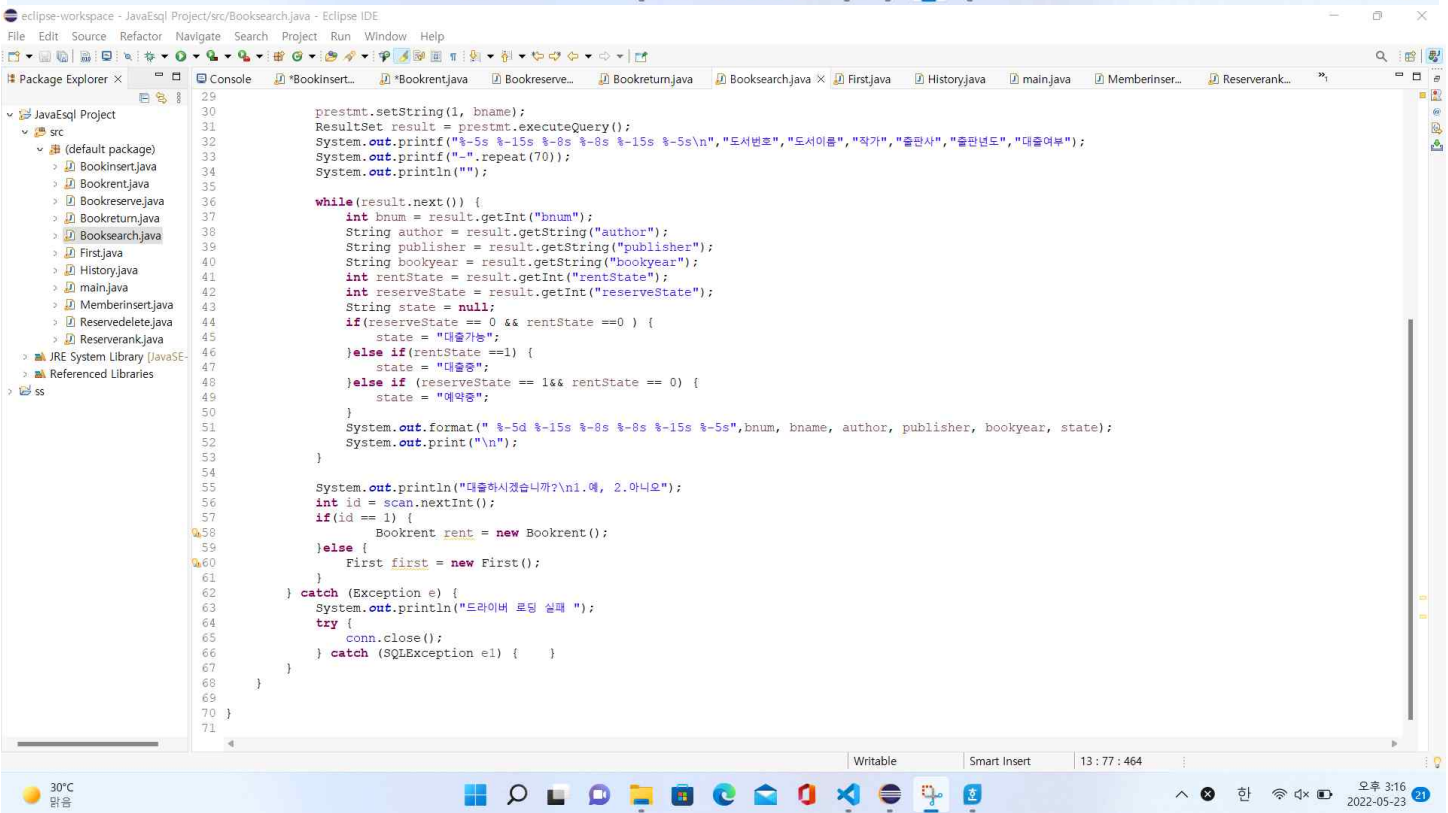
```
34    System.out.print("비밀번호 : ");
35    String pword = scan.next();
36
37    pstmt.setInt(1, mnum);
38    ResultSet result = pstmt.executeQuery();
39
40    int checkmnum = 0;
41    String checkpword = null;
42
43    while(result.next()) {
44        checkmnum = result.getInt("mnum");
45        checkpword = result.getString("pword");
46    }
47
48    if(checkmnum == 0) {
49        System.out.println("\n학번이 올바르지 않습니다.");
50    } else {
51        if(checkpword.equals(pword)) {
52            pstmt2.setInt(1, bnum);
53            pstmt2.setInt(2, mnum);
54
55            pstmt2.executeUpdate();
56
57            pstmt3.setInt(1, bnum);
58            pstmt3.executeUpdate();
59            System.out.println("예약되었습니다.");
60        } else {
61            System.out.println("비밀번호가 올바르지 않습니다.");
62        }
63    }
64
65    catch (Exception e) {
66        System.out.println("드라이버 로딩 실패");
67        try {
68            conn.close();
69        } catch (SQLException e1) {}
70    }
71 }
72
73
74
75 }
```

30°C 맑음

Writable Smart Insert 13:77:465

오후 3:16 2022-05-23





```
eclipse-workspace - JavaEsq Project/src/History.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Console Bookinsert... Bookrentjava Bookreserve... Bookreturnjava Booksearchjava Firstjava Historyjava mainjava Memberinser... Reserverank...

JavaEsq Project
  src
    (default package)
      Bookinsert.java
      Bookrent.java
      Bookreserve.java
      Bookreturn.java
      Booksearch.java
      First.java
      History.java
      main.java
      Memberinsert.java
      Reserverank.java
      Reserverank.java
      JRE System Library [JavaSE-
      Referenced Libraries
      ss

1=import java.sql.Connection;
2 import java.sql.DriverManager;
3 import java.sql.PreparedStatement;
4 import java.sql.ResultSet;
5 import java.sql.SQLException;
6 import java.util.Scanner;
7
8 public class History {
9     Scanner scan = new Scanner(System.in);
10    private Connection conn;
11    private static final String USERNAME = "root";
12    private static final String PASSWORD = "try1234";
13    private static final String URL = "jdbc:mysql://localhost:3306/library";
14
15    public History() {
16        String sql = "SELECT rnum, book.bnum, bname, author, publisher, rentDate, returnDate, returnState "
17            + "FROM book, rent, librarymember "
18            + "WHERE librarymember.mnum = ? "
19            + "AND librarymember.pword = ? "
20            + "AND rent.mnum = librarymember.mnum "
21            + "AND book.bnum = rent.bnum";
22
23        String sql2 = "SELECT reserve.Renum, book.bnum, bname, author, publisher, reserveDate, rentDate"
24            + "FROM reserve "
25            + "left join rent "
26            + "on reserve.mnum = rent.mnum "
27            + "and reserve.rnum = rent.rnum "
28            + "left join book "
29            + "on reserve.bnum = book.bnum "
30            + "INNER JOIN librarymember "
31            + "on reserve.mnum = librarymember.mnum "
32            + "and librarymember.mnum = ? "
33            + "and librarymember.pword = ?";
34
35
36
37    try {
38        Class.forName("com.mysql.jdbc.Driver");
39        conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);
40
41        PreparedStatement pstmt = conn.prepareStatement(sql);
42        PreparedStatement pstmt2 = conn.prepareStatement(sql2);
43
30°C 맑음
Writeable Smart Insert 1:1:0
오후 3:20 2022-05-23
```

```
eclipse-workspace - JavaEsq Project/src/History.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Console Bookinsert... Bookrentjava Bookreserve... Bookreturnjava Booksearchjava Firstjava Historyjava mainjava Memberinser... Reserverank...

JavaEsq Project
  src
    (default package)
      Bookinsert.java
      Bookrent.java
      Bookreserve.java
      Bookreturn.java
      Booksearch.java
      First.java
      History.java
      main.java
      Memberinsert.java
      Reserverank.java
      Reserverank.java
      JRE System Library [JavaSE-
      Referenced Libraries
      ss

43
44    System.out.println("이용할 서비스를 선택해 주세요.");
45    System.out.print("1.대출내역보기, 2.예약내역보기 ");
46    int id = scan.nextInt();
47
48    System.out.print("학번 : ");
49    int mnum = scan.nextInt();
50    System.out.print("비밀번호 : ");
51    String pword = scan.next();
52
53    if(id == 1) {
54        pstmt.setInt(1, mnum);
55        pstmt.setString(2, pword);
56        ResultSet result = pstmt.executeQuery();
57
58        System.out.printf("%-5s %-5s %-15s %-8s %-8s %-15s %-15s %-5s\n", "대출번호", "도서번호", "도서이름", "작가", "출판사", "예약일", "대출일", "반납");
59        System.out.println("-".repeat(100));
60        System.out.println("");
61        while(result.next()) {
62            int rnum = result.getInt("rnum");
63            int bnum = result.getInt("bnum");
64            String bname = result.getString("bname");
65            String author = result.getString("author");
66            String publisher = result.getString("publisher");
67            String rentDate = result.getString("rentDate");
68            String returnDate = result.getString("returnDate");
69            int returnState = result.getInt("returnState");
70            String state = null;
71            if(returnState == 0) {
72                state = "반납X";
73            }else {
74                state = "반납O";
75            }
76            System.out.format("%-5s %-5s %-15s %-8s %-8s %-15s %-15s %-5s", rnum, bnum, bname, author, publisher, rentDate, returnDate, state);
77            System.out.println("");
78
79
80    System.out.print("1.반납하기, 2.처음으로 돌아가기 ");
81    int id2 = scan.nextInt();
82    if(id2 == 1) {
83        Bookreturn bookreturn = new Bookreturn();
84    }else {
85        First first = new First();
30°C 맑음
Writeable Smart Insert 1:1:0
오후 3:20 2022-05-23
```

```
eclipse-workspace - JavaEsq Project/src/History.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Console *Bookinsert... *Bookrent.java *Bookreserve... Bookreturn.java Booksearch.java First.java History.java main.java Memberinser... Reserverank...

JavaEsq Project
src
  (default package)
    Bookinsert.java
    Bookrent.java
    Bookreserve.java
    Bookreturn.java
    Booksearch.java
    First.java
    History.java
    main.java
    Memberinser.java
    Reserverank.java
    Reserverank.java
  JRE System Library [JavaSE-
  Referenced Libraries
  ss

102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134

if(id2 == 1) {
    Bookreturn bookreturn = new Bookreturn();
} else {
    First first = new First();
}
} else {
    prestmt2.setInt(1, mnum);
    prestmt2.setString(2, pword);
    ResultSet result2 = prestmt2.executeQuery();

    System.out.printf("%-5s %-5s %-15s %-8s %-8s %-15s %-5s\n", "예약번호", "도서번호", "도서이름", "작가", "출판사", "예약일", "대출일", "반납");
    System.out.println("");
    while(result2.next()) {
        int Renum = result2.getInt("Renum");
        int bnum = result2.getInt("bnum");
        String bname = result2.getString("bname");
        String author = result2.getString("author");
        String publisher = result2.getString("publisher");
        String reserveDate = result2.getString("reserveDate");
        String rentDate = result2.getString("rentDate");
        String state = null;
        if(rentDate == null) {
            state = "취소O";
        } else {
            state = "취소X";
        }
        System.out.format("%-5d %-5d %-15s %-8s %-8s %-15s %-5s", Renum, bnum, bname, author, publisher, reserveDate, rentDate, state);
        System.out.print("\n");
    }

    System.out.print("1.예약 취소 하기, 2.예약 순위 보기 3. 처음으로 돌아가기 ");
    int id3 = scan.nextInt();
    if(id3 == 1) {
        Reserverank reserverank = new Reserverank();
    } else if(id3 == 2) {
        Reserverank rank = new Reserverank();
    } else {
        First first = new First();
    }
}

} catch (Exception e) {
    System.out.println("드라이버 로딩 실패 ");
    try {
        conn.close();
    } catch (SQLException e1) {
    }
}
}
```

```
eclipse-workspace - JavaEsq Project/src/History.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Console *Bookinsert... *Bookrent.java *Bookreserve... Bookreturn.java Booksearch.java First.java History.java main.java Memberinser... Reserverank...

JavaEsq Project
src
  (default package)
    Bookinsert.java
    Bookrent.java
    Bookreserve.java
    Bookreturn.java
    Booksearch.java
    First.java
    History.java
    main.java
    Memberinser.java
    Reserverank.java
    Reserverank.java
  JRE System Library [JavaSE-
  Referenced Libraries
  ss

92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134

System.out.printf("%-5s %-5s %-15s %-8s %-8s %-15s %-5s\n", "예약번호", "도서번호", "도서이름", "작가", "출판사", "예약일", "대출일", "반납");
System.out.println("");
while(result2.next()) {
    int Renum = result2.getInt("Renum");
    int bnum = result2.getInt("bnum");
    String bname = result2.getString("bname");
    String author = result2.getString("author");
    String publisher = result2.getString("publisher");
    String reserveDate = result2.getString("reserveDate");
    String rentDate = result2.getString("rentDate");
    String state = null;
    if(rentDate == null) {
        state = "취소O";
    } else {
        state = "취소X";
    }
    System.out.format("%-5d %-5d %-15s %-8s %-8s %-15s %-5s", Renum, bnum, bname, author, publisher, reserveDate, rentDate, state);
    System.out.print("\n");
}

System.out.print("1.예약 취소 하기, 2.예약 순위 보기 3. 처음으로 돌아가기 ");
int id3 = scan.nextInt();
if(id3 == 1) {
    Reserverank reserverank = new Reserverank();
} else if(id3 == 2) {
    Reserverank rank = new Reserverank();
} else {
    First first = new First();
}
}

} catch (Exception e) {
    System.out.println("드라이버 로딩 실패 ");
    try {
        conn.close();
    } catch (SQLException e1) {
    }
}
}
```



```
eclipse-workspace - JavaEsql Project/src/MemberInsert.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Console *Bookinsert... *Bookrentjava *Bookreserve... Bookreturnjava Booksearchjava Firstjava Historyjava mainjava Memberinser... Reserverank...
JavaEsql Project
  src
    (default package)
      Bookinsert.java
      Bookrent.java
      Bookreserve.java
      Bookreturn.java
      Booksearch.java
      First.java
      History.java
      main.java
      Memberinsert.java
      Reserveddelete.java
      Reserverank.java
      JRE System Library [JavaSE-11]
      Referenced Libraries
      ss

1 import java.sql.Connection;
2 import java.sql.DriverManager;
3 import java.sql.PreparedStatement;
4 import java.sql.ResultSet;
5 import java.sql.SQLException;
6 import java.sql.Statement;
7 import java.util.Scanner;
8
9 public class Memberinsert {
10
11     Scanner scan = new Scanner(System.in);
12     private Connection conn;
13     private static final String USERNAME = "root";
14     private static final String PASSWORD = "try1234";
15     private static final String URL = "jdbc:mysql://localhost:3306/library";
16     public Memberinsert() {
17         String sql = "INSERT INTO libraryMember(mnum, pword, mname, major, email, birth) VALUES(?, ?, ?, ?, ?, ?)";
18         String sql2 = "select * from libraryMember where mnum=?";
19         try {
20             System.out.println("회원가입");
21
22             Class.forName("com.mysql.jdbc.Driver");
23             conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);
24
25             PreparedStatement pstmt = conn.prepareStatement(sql);
26             PreparedStatement pstmt2 = conn.prepareStatement(sql2);
27
28             System.out.print("학과 : ");
29             String major = scan.next();
30             System.out.print("이름 : ");
31             String mname = scan.next();
32
33             int mnum = 0;
34             while(true) {
35                 System.out.print("학번 : ");
36                 mnum = scan.nextInt();
37
38                 pstmt2.setInt(1, mnum);
39                 ResultSet result = pstmt2.executeQuery();
40
41                 int title = 0;
42                 while(result.next()) {
43                     title = result.getInt("mnum");
44                 }
45             }
46         } catch (Exception e) {
47             System.out.println("드라이버 로딩 실패");
48         }
49     }
50
51     System.out.print("비밀번호 : ");
52     String pword = scan.next();
53     System.out.print("이메일 : ");
54     String email = scan.next();
55     System.out.print("생년월일 : ");
56     String birth = scan.next();
57
58     pstmt.setInt(1, mnum);
59     pstmt.setString(2, pword);
60     pstmt.setString(3, mname);
61     pstmt.setString(4, major);
62     pstmt.setString(5, email);
63     pstmt.setString(6, birth);
64
65     pstmt.executeUpdate();
66
67     System.out.println("회원가입이 되었습니다.");
68
69     catch (Exception e) {
70         System.out.println("드라이버 로딩 실패");
71     }
72     try {
73         conn.close();
74     } catch (SQLException e1) { }
75 }
76
77 }
```

```
eclipse-workspace - JavaEsql Project/src/MemberInsert.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Console *Bookinsert... *Bookrentjava *Bookreserve... Bookreturnjava Booksearchjava Firstjava Historyjava mainjava Memberinser... Reserverank...
JavaEsql Project
  src
    (default package)
      Bookinsert.java
      Bookrent.java
      Bookreserve.java
      Bookreturn.java
      Booksearch.java
      First.java
      History.java
      main.java
      Memberinsert.java
      Reserveddelete.java
      Reserverank.java
      JRE System Library [JavaSE-11]
      Referenced Libraries
      ss

40
41     int title = 0;
42     while(result.next()) {
43         title = result.getInt("mnum");
44     }
45
46     if(title == 0) {
47         System.out.println("사용가능한 학번입니다.");
48         break;
49     } else {
50         System.out.println("이미 회원가입한 학번입니다.");
51     }
52 }
53
54 System.out.print("비밀번호 : ");
55 String pword = scan.next();
56 System.out.print("이메일 : ");
57 String email = scan.next();
58 System.out.print("생년월일 : ");
59 String birth = scan.next();
60
61 pstmt.setInt(1, mnum);
62 pstmt.setString(2, pword);
63 pstmt.setString(3, mname);
64 pstmt.setString(4, major);
65 pstmt.setString(5, email);
66 pstmt.setString(6, birth);
67
68 pstmt.executeUpdate();
69
70 System.out.println("회원가입이 되었습니다.");
71
72 catch (Exception e) {
73     System.out.println("드라이버 로딩 실패");
74 }
75 try {
76     conn.close();
77 } catch (SQLException e1) { }
78
79 }
80
81 }
82 }
```

eclipse-workspace - JavaEsq Project/src/ReserveDelete.java - Eclipse IDE

```
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
JavaEsq Project
  src
    (default package)
      Bookinsert.java
      Bookrent.java
      Bookreserve.java
      Bookreturn.java
      Booksearch.java
      First.java
      History.java
      main.java
      Memberinsert.java
      ReserveDelete.java
      Reserverank.java
      JRE System Library [JavaSE-
      Referenced Libraries
      ss

1 import java.sql.Connection;
2
3 public class ReserveDelete {
4     Scanner scan = new Scanner(System.in);
5     private Connection conn;
6     private static final String USERNAME = "root";
7     private static final String PASSWORD = "try1234";
8     private static final String URL = "jdbc:mysql://localhost:3306/library";
9
10    public ReserveDelete() {
11        String sql = "DELETE FROM reserve where Renum = ?";
12        String sql2 = "UPDATE book, reserve SET reserveState = 0 where book.bnum = ? and not exists (select Renum from reserve where reserve.bnum = ? and re:
13
14    try {
15        Class.forName("com.mysql.jdbc.Driver");
16        conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);
17
18        PreparedStatement pstmt = conn.prepareStatement(sql);
19        PreparedStatement pstmt2 = conn.prepareStatement(sql2);
20
21        System.out.println("예약 도서 취소");
22        System.out.print("예약 번호 : ");
23        int Renum = scan.nextInt();
24        System.out.print("도서 번호 : ");
25        int bnum = scan.nextInt();
26
27        pstmt.setInt(1, Renum);
28        pstmt.executeUpdate();
29
30        pstmt2.setInt(1, bnum);
31        pstmt2.setInt(2, bnum);
32        pstmt2.executeUpdate();
33
34        System.out.println("예약이 취소되었습니다.!");
35
36    } catch (Exception e) {
37        System.out.println("드라이버 로딩 실패 ");
38        try {
39            conn.close();
40        } catch (SQLException e1) { }
41    }
42 }
```

30°C 맑음 12:77:439 오후 3:22 2022-05-23

eclipse-workspace - JavaEsq Project/src/main.java - Eclipse IDE

```
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
JavaEsq Project
  src
    (default package)
      Bookinsert.java
      Bookrent.java
      Bookreserve.java
      Bookreturn.java
      Booksearch.java
      First.java
      History.java
      main.java
      Memberinsert.java
      ReserveDelete.java
      Reserverank.java
      JRE System Library [JavaSE-
      Referenced Libraries
      ss

1 public class main {
2     public static void main(String[] args) {
3         First first = new First();
4     }
5 }
6
7
8
```

30°C 맑음 1:1:0 오후 3:23 2022-05-23

```
eclipse-workspace - JavaEsq Project/src/Reserverank.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer X
JavaEsq Project
src
  (default package)
    Bookinsert.java
    Bookrent.java
    Bookreserve.java
    Bookreturn.java
    Booksearch.java
    First.java
    History.java
    main.java
    Memberinsert.java
    Reservedele.java
    Reserverank.java
  JRE System Library [JavaSE-
  Referenced Libraries
  ss

11 import java.sql.Connection;
12
13 public class Reserverank {
14     Scanner scan = new Scanner(System.in);
15     private Connection conn;
16     private static final String USERNAME = "root";
17     private static final String PASSWORD = "try1234";
18     private static final String URL = "jdbc:mysql://localhost:3306/library";
19
20     public Reserverank() {
21         String sql1 = "select mnum, book.bnum, rentState,rank() over(order by Renum) As ranking from reserve, book where reserve.bnum = ? and reserve.bnum = ?";
22         String sql2 = "SELECT * FROM libraryMember where mnum = ?";
23         String sql3 = "INSERT INTO rent(bnum, mnum, rentDate, returnDate, returnState)VALUES(?,?,Now(),?,0)";
24         String sql4 = "UPDATE book SET rentState = 1 WHERE bnum = ?";
25         String sql5 = "UPDATE reserve, rent SET reserve.rnum= rent.rnum WHERE rent.mnum = ? AND rent.bnum = ? AND rent.returnState = 0 AND reserve.bnum = ?";
26         String sql6 = "UPDATE book, reserve SET reserveState = 0 where book.bnum = ? and not exists (select Renum from reserve where reserve.bnum = ? and reserveState = 0)";
27         Date now = new Date();
28         Calendar cal = Calendar.getInstance();
29         cal.setTime(now);
30         cal.add(Calendar.DATE, 10);
31         DateFormat df = new SimpleDateFormat("yyyy-MM-dd");
32         String returndate = df.format(cal.getTime());
33         try {
34             Class.forName("com.mysql.jdbc.Driver");
35             conn = DriverManager.getConnection(URL, USERNAME, PASSWORD);
36
37             PreparedStatement prestmt = conn.prepareStatement(sql1);
38             PreparedStatement prestmt2 = conn.prepareStatement(sql2);
39             PreparedStatement prestmt3 = conn.prepareStatement(sql3);
40             PreparedStatement prestmt4 = conn.prepareStatement(sql4);
41             PreparedStatement prestmt5 = conn.prepareStatement(sql5);
42             PreparedStatement prestmt6 = conn.prepareStatement(sql6);
43
44             System.out.println("예약 순위");
45             System.out.print("도서 번호 : ");
46             int bnum = scan.nextInt();
47
48             prestmt.setInt(1, bnum);
49             ResultSet result = prestmt.executeQuery();
50
51             System.out.format("%-10s %-3s %-6s\n", "학번", "순위", "대출여부");
52             System.out.printf("-".repeat(30));
```

```
eclipse-workspace - JavaEsq Project/src/Reserverank.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer X
JavaEsq Project
src
  (default package)
    Bookinsert.java
    Bookrent.java
    Bookreserve.java
    Bookreturn.java
    Booksearch.java
    First.java
    History.java
    main.java
    Memberinsert.java
    Reservedele.java
    Reserverank.java
  JRE System Library [JavaSE-
  Referenced Libraries
  ss

52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94

    int mnumcheck = 0;
    int mnum1 = 0;
    while(result.next()) {
        int mnum = result.getInt("mnum");
        int rentState = result.getInt("rentState");
        int ranking = result.getInt("ranking");
        String rent = null;
        if(rentState == 0 && ranking == 1) {
            rent = "대출가능";
        }else {
            rent = "대출대기";
        }
        System.out.format("%-10d %-5d %-6s", mnum, ranking, rent);
        System.out.println("");
        if (rent.equals("대출가능")) {
            mnumcheck=1;
            mnum1 = result.getInt("mnum");
        }
    }
    System.out.println(mnum1+"님 읽니까? 1. 네, 2. 아니요");
    int id = scan.nextInt();
    if(id == 1) {
        System.out.println("대출하시겠습니까? 1. 네, 2. 아니요");
        int id2 = scan.nextInt();
        if(id2 == 1) {
            if(mnumcheck == 1) {
                while(true) {
                    System.out.print("비밀번호 : ");
                    String pword = scan.next();
                    prestmt2.setInt(1, mnum1);
                    ResultSet result2 = prestmt2.executeQuery();
                    while(result2.next()) {
                        String pwordcheck = result2.getString("pword");
                        if(pword.equals(pwordcheck)) {
                            prestmt3.setInt(1, bnum);
                            prestmt3.setInt(2, mnum1);
                            prestmt3.setString(3, returndate);
                            prestmt3.executeUpdate();
                        }
                    }
                }
            }
        }
    }
}
```

```

88         while(result2.next()) {
89             String pwordcheck = result2.getString("pword");
90             if(pword.equals(pwordcheck)) {
91                 pstmt3.setInt(1, bnum);
92                 pstmt3.setInt(2, mnum);
93                 pstmt3.setString(3, returndate);
94                 pstmt3.executeUpdate();
95
96                 pstmt4.setInt(1, bnum);
97                 pstmt4.executeUpdate();
98
99                 pstmt5.setInt(1, mnum);
100                pstmt5.setInt(2, bnum);
101                pstmt5.executeUpdate();
102
103                pstmt6.setInt(1, bnum);
104                pstmt6.setInt(2, bnum);
105                pstmt6.executeUpdate();
106
107                System.out.println("대출되었습니다.");
108                break;
109            }
110            else {
111                System.out.println("비밀번호가 올바르지 않습니다.");
112            }
113        }
114    }
115
116    }else {
117        First first= new First();
118    }
119    }else {
120        First first= new First();
121    }
122
123    } catch (Exception e) {
124        System.out.println("드라이버 로딩 실패 ");
125        try {
126            conn.close();
127        } catch (SQLException e1) { }
128    }
129
130    }

```

평가내용	요구점수	이유
최종 보고서 자체 평가	2.5	해당 날짜에 대한 구현 내용 캡처를 못했으나 보고서에 구현 완료 후에라도 캡처해서 넣었기 때문에 0.5점을 깎았습니다.
고급/웹 언어 성숙도 평가	5	대출, 반납, 예약에 필요한 기능은 다 추가했다고 생각했기 때문에 5점을 요구합니다.
테이블 조인 기능 존재 여부	1	17쪽에 해당 내용이 있으므로 1점을 요구합니다.
Cross Check	1	8, 14, 21쪽에 해당 내용이 있으므로 1점을 요구합니다.