

프로그래밍 언어 응용

2022/09/15

임민지

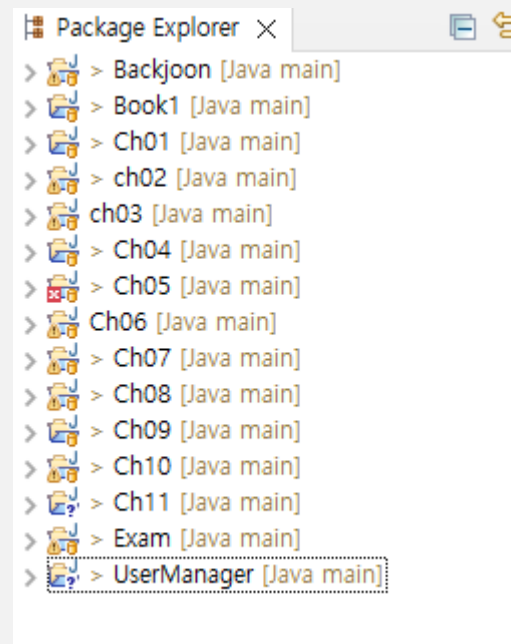
목

차

- 1. 프로젝트 생성
- 2. 라이브러리 설정
- 3. 프로젝트 구성
- 4. 프로젝트 작성

문제 1

- 프로젝트 생성
- 프로젝트 명 : UserManager
- 자바 버전 : JAVA8 이상
- DBMS : MYSQL 8
- 개발도구 : Eclipse,
Workbench



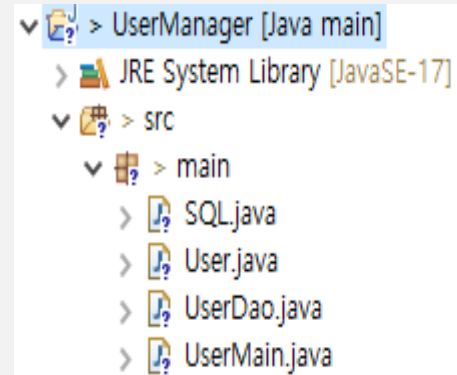
문제 2

- 라이브러리 구성
- _프로젝트_새 폴더-lib 생성 –
mysql-connector-java-8.x.x.jar
이동
- _프로젝트_-Build Path- Add
External Archives... 선택 –
mysql-connector-java-8.x.x.jar
선택



문제 3

- 프로젝트 구성
- -프로젝트-new-Class-패키지명:
main/클래스명:SQL 생성
- -프로젝트-new-Class-패키지명:
main/클래스명:User 생성
- -프로젝트-new-Class-패키지명:
main/클래스명:UserDao 생성
- -프로젝트-new-Class-패키지명:
main/클래스명:UserMain 생성



문제 4-1

```
public class SQL {  
  
    public static final String INSERT_USER = "INSERT INTO `User3` VALUES (?, ?, ?, ?);";  
    public static final String SELECT_USERS = "SELECT * FROM `User3`;";  
    public static final String SELECT_USER = "SELECT * FROM `User3` WHERE `name`=?;";  
    public static final String DELETE_USER = "DELETE FROM `User3` WHERE `uid`=?;";  
}
```

문제 4-2

```
public class User {
    private String uid;
    private String name;
    private String hp;
    private int age;

    public String getUid() {
        return uid;
    }

    public void setUid(String uid) {
        this.uid = uid;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getHp() {
        return hp;
    }

    public void setHp(String hp) {
        this.hp = hp;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }

    @Override
    public String toString() {
        return uid+","+name+","+hp+","+age;
    }
}
```

문제 4-3

```
public class UserDao {  
    // 싱글톤  
    private static UserDao instance = new UserDao();  
    public static UserDao getInstance() {  
        return instance;  
    }  
    private UserDao() {}  
  
    // DB정보  
    String host = "jdbc:mysql://127.0.0.1:3306/java1db";  
    private String user = "root";  
    private String pass = "1234";  
  
    private Connection getConnection() throws ClassNotFoundException, SQLException {  
        Class.forName("com.mysql.cj.jdbc.Driver");  
        Connection conn = DriverManager.getConnection(host, user, pass);  
        return conn;  
    }  
  
    private Connection conn;  
    private PreparedStatement psmt;  
    private ResultSet rs;  
  
    public void insertUser(User user) {  
        try {  
            conn = getConnection();  
            psmt = conn.prepareStatement(SQL.INSERT_USER);  
            psmt.setString(1, user.getId());  
            psmt.setString(2, user.getName());  
            psmt.setString(3, user.getHp());  
            psmt.setInt(4, user.getAge());  
            psmt.executeUpdate();  
            close();  
        } catch (SQLException e) {  
            System.out.println("아이디 또는 휴대폰 중복입니다.");  
        } catch (ClassNotFoundException e) {  
            e.printStackTrace();  
        } catch (SQLException e) {  
            e.printStackTrace();  
        }  
    }  
  
    public List<User> selectUsers() {  
        List<User> users = new ArrayList<>();  
  
        try {  
            conn = getConnection();  
            psmt = conn.prepareStatement(SQL.SELECT_USERS);  
            rs = psmt.executeQuery();  
  
            while(rs.next()) {  
                User user = new User();  
                user.setId(rs.getString(1));  
                user.setName(rs.getString(2));  
                user.setHp(rs.getString(3));  
                user.setAge(rs.getInt(4));  
                users.add(user);  
            }  
            close();  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```
        return users;  
    }  
  
    public User selectUser(String name) {  
        User user = new User();  
  
        try {  
            conn = getConnection();  
            psmt = conn.prepareStatement(SQL.SELECT_USER);  
            psmt.setString(1, name);  
            rs = psmt.executeQuery();  
  
            if(rs.next()) {  
                user.setId(rs.getString(1));  
                user.setName(rs.getString(2));  
                user.setHp(rs.getString(3));  
                user.setAge(rs.getInt(4));  
            }  
        }  
        close();  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
    return user;  
}  
  
    public int deleteUser(String name) {  
        int result = 0;  
  
        try {  
            conn = getConnection();  
            psmt = conn.prepareStatement(SQL.DELETE_USER);  
            psmt.setString(1, name);  
            result = psmt.executeUpdate();  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
        return result;  
}  
  
    private void close() throws SQLException {  
        if(rs != null) {  
            rs.close();  
        }  
  
        if(psmt != null) {  
            psmt.close();  
        }  
  
        if(conn != null) {  
            conn.close();  
        }  
    }  
}
```


문제 4-4

```
package com.scribble;

public static void main(String[] args) {
    System.out.println("-----");
    System.out.println("회원관리메니저 v1.0");
    System.out.println("-----");

    Scanner sc = new Scanner(System.in);

    while(true) {
        System.out.println("-----");
        System.out.println("종료:0, 등록:1, 조회:2, 검색:3, 삭제:4");
        System.out.print("선택 : ");
        int answer = sc.nextInt();

        if(answer == 0) {
            break;
        } else if(answer == 1) {
            // 데이터 입력
            User user = new User();

            System.out.print("아이디 입력 : ");
            user.setid(sc.next());
            System.out.print("이름 입력 : ");
            user.setName(sc.next());

            System.out.print("휴대폰 번호 : ");
            user.sethp(sc.next());

            System.out.print("나이 입력 : ");
            user.setage(sc.nextInt());

            UserDao.getInstance().insertUser(user);
            System.out.println("입력완료...");

        } else if(answer == 2) {
            // 데이터 조회
            List<User> users = UserDao.getInstance().selectUsers();

            System.out.println("-----현재조회-----");
            for(User user : users) {
                System.out.println(user);
            }

        } else if(answer == 3) {
            // 데이터 검색
            System.out.print("이름검색 : ");
            String name = sc.next();

            User user = UserDao.getInstance().selectUser(name);

            System.out.println("-----검색결과-----");
            System.out.println(user);

        } else if(answer == 4) {
            // 데이터 삭제
            System.out.print("삭제아이디 : ");
            String uid = sc.next();

            int result = UserDao.getInstance().deleteUser(uid);

            if(result >= 1) {
                System.out.println("아이디 " + uid + " 삭제완료");
            } else {
                System.out.println("아이디 " + uid + " 사용자가 존재하지 않습니다.");
            }
        }
    }

    sc.close();
    System.out.println("프로그램 종료...");
}
```

결과

```
*****
회원관리매니저 v1.0
*****

-----
종료:0, 입력:1, 조회:2, 검색:3, 삭제:4
선택 : 2
-----전체조회-----
a101,김유신,010-1234-1001,25
a102,김춘추,010-1234-1002,23
-----
종료:0, 입력:1, 조회:2, 검색:3, 삭제:4
선택 : 1
아이디 입력 : a103
이름 입력 : 장보고
휴대폰 입력 : 010-1234-1003
나이 입력 : 31
입력완료...
-----
종료:0, 입력:1, 조회:2, 검색:3, 삭제:4
선택 : 0
프로그램 종료...
```

```
*****
회원관리매니저 v1.0
*****

-----
종료:0, 입력:1, 조회:2, 검색:3, 삭제:4
선택 : 3
이름검색 : 김유신
-----검색결과-----
a101,김유신,010-1234-1001,25
-----
종료:0, 입력:1, 조회:2, 검색:3, 삭제:4
선택 : 4
삭제아이디 : a101
아이디 a101 삭제완료
-----
종료:0, 입력:1, 조회:2, 검색:3, 삭제:4
선택 : 2
-----전체조회-----
a102,김춘추,010-1234-1002,23
a103,장보고,010-1234-1003,31
-----
종료:0, 입력:1, 조회:2, 검색:3, 삭제:4
선택 : 0
프로그램 종료...
```

