



PROJECT

# BANKRUPT

JAVA-TEAM-PROJECT-2ND-2017

DATE

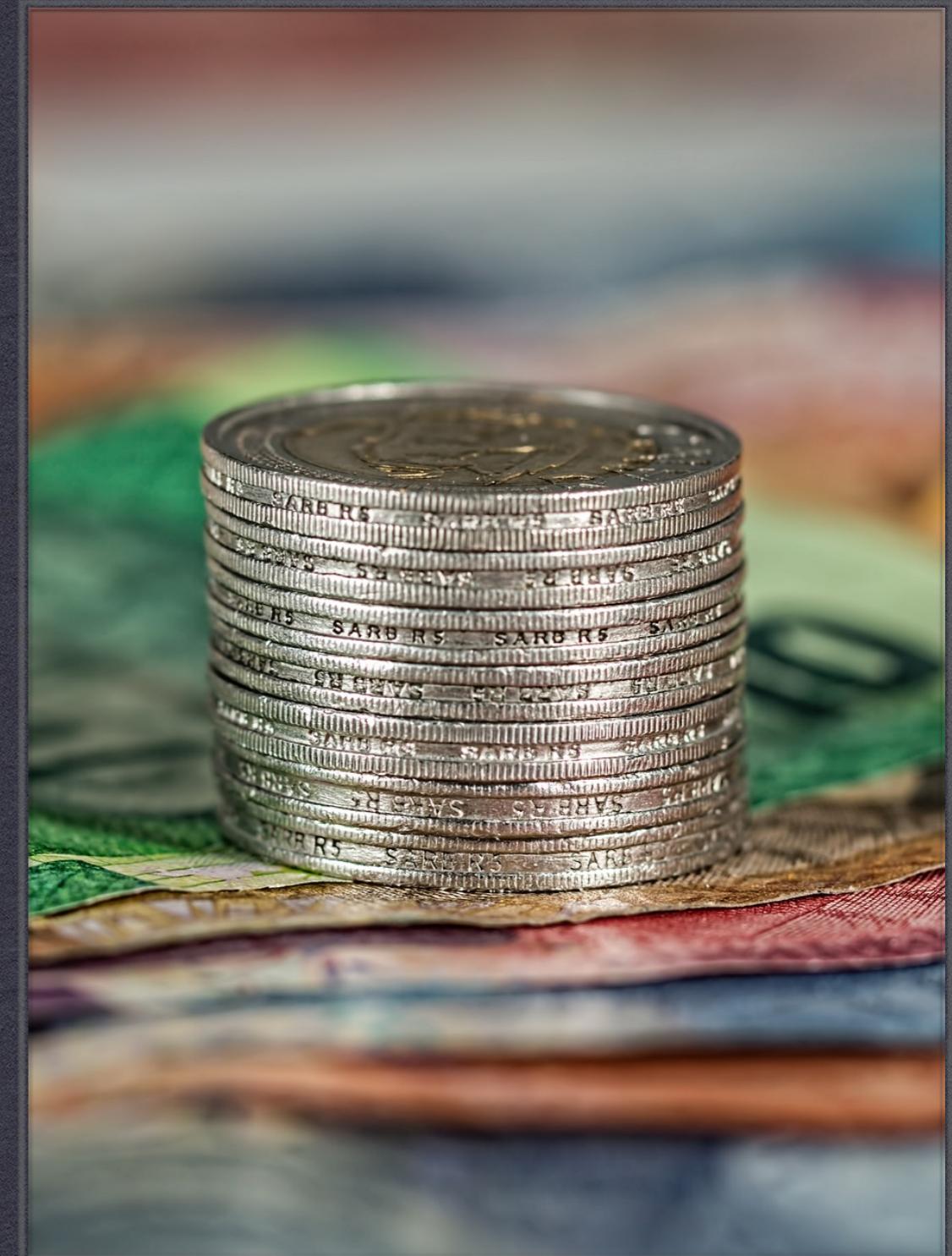
2017/12

NAME

201635922 김동주 201635924 김민석 201635954 김효재 201635988 성현식

# INDEX

- 1. ABOUT BANKRUPT**
- 2. CLASS RELATIONSHIP**
- 3. DATABASE SCHEMA**
- 4. CODE DESCRIPTION**
- 5. DEMO**



# ABOUT BANKRUPT JAVA



# About Bankrupt

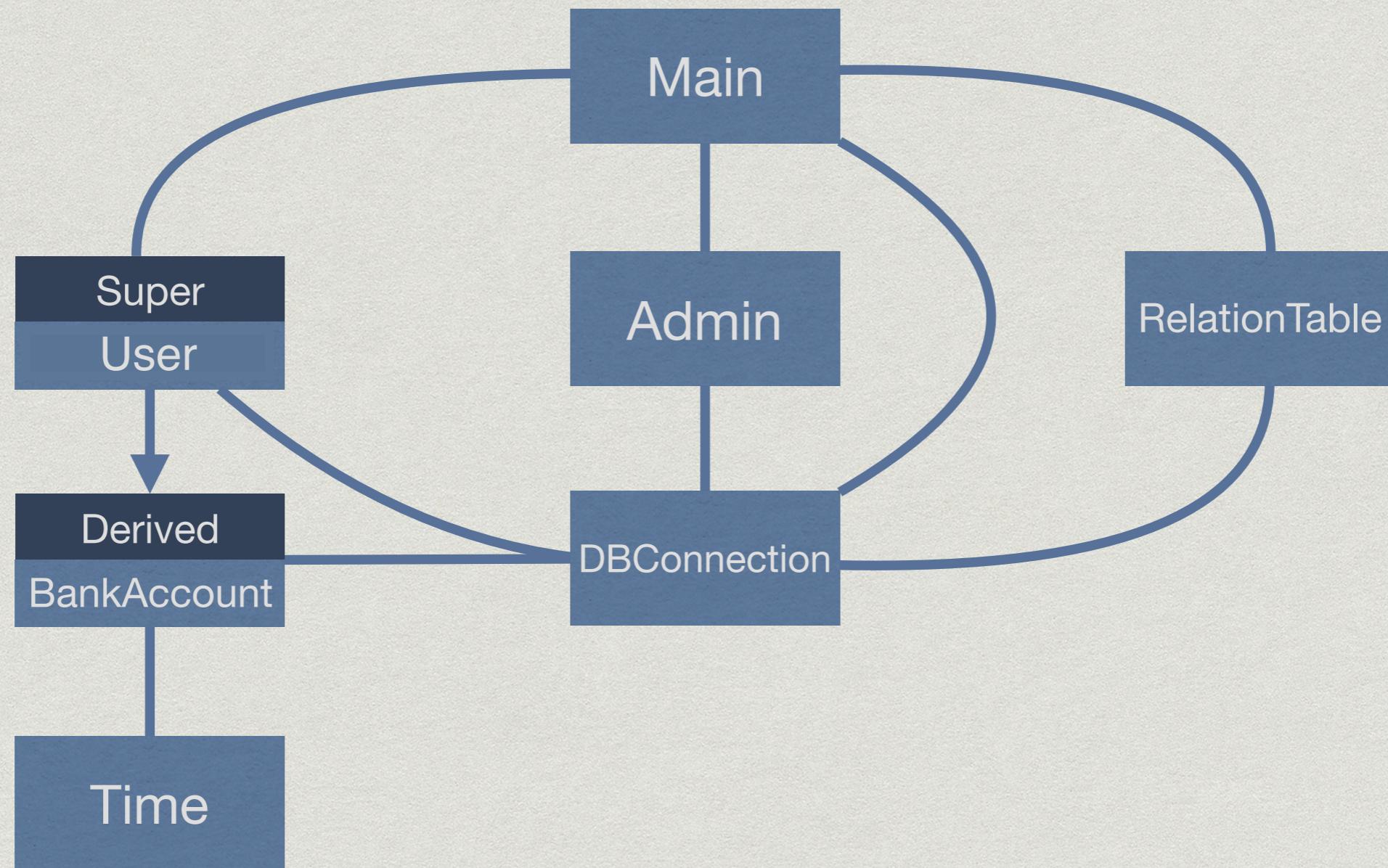
- \* Basic banking program
  - User CRUD / Administer account mode
  - Withdraw/Deposit between users
  - Exchange with nation code(KOR, USA, CHN)
- \* Connects to Database
  - MariaDB
  - Oracle

# CLASS RELATIONSHIP

## BANKRUPT



# Class relationship



# DATABASE SCHEMA BANKRUPT



# RelationTable.java

```
...
public void MakeTable() throws SQLException{
    DbConnect();

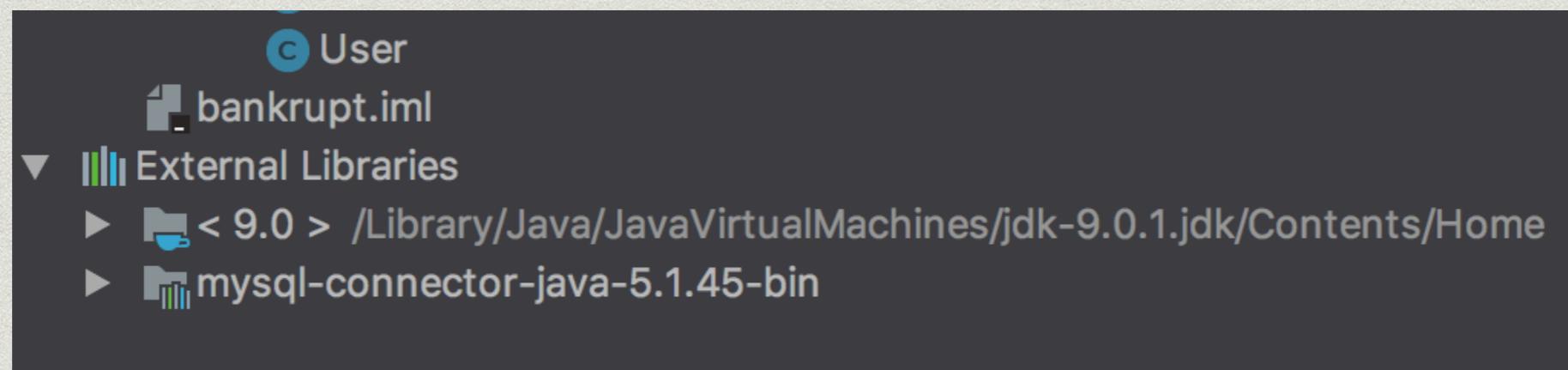
    String table1= "CREATE TABLE BankUser ("+
        "Id          VARCHAR(20)      NOT NULL, "+  
        +"AdminId    VARCHAR(20), "+  
        +"Passwd     VARCHAR(20), "+  
        +"Name       VARCHAR(20), "+  
        +"Country    VARCHAR(20), "+  
        +"PRIMARY KEY(Id), "+  
        +"FOREIGN KEY(Country) REFERENCES F_Money(Country), "+  
        +"FOREIGN KEY(AdminId) REFERENCES Admin(Id))";  
  
String table2= "CREATE TABLE BankAccount ("+
        "UserId      VARCHAR(20)      NOT NULL, "+  
        +"CreateAccountTime VARCHAR(40)  NOT NULL, "+  
        +"AccountNumber VARCHAR(40)    NOT NULL, "+  
        +"Money       INT, "+  
        +"PRIMARY KEY(UserId, CreateAccountTime, AccountNumber), "+  
        +"FOREIGN KEY (UserId) REFERENCES BankUser(Id))";  
  
String table3= "CREATE TABLE Admin ("+
        "Id          VARCHAR(20)      NOT NULL, "+  
        +"Passwd     VARCHAR(20), "+  
        +"Name       VARCHAR(20), "+  
        +"PRIMARY KEY(Id))";  
  
String table4= "CREATE TABLE F_Money ("+
        "Country    VARCHAR(20) NOT NULL, "+  
        +"Rate       DOUBLE, "+  
        +"PRIMARY KEY(Country))";  
  
...
```

# CODE DESCRIPTION

## BANKRUPT

```
TagName(s)[0];
turn;
id = id;
.net/en_US/sdk.js#xfbml=1&version=v2.6&appId=200000000000000
js, fjs);
ok-jssdk'));</script>
n-reader-text" href="#content"><?php esc_html_el '$
ass="site-header" role="banner">
nding">
tn pull-left">
home() && $xpanel['homepage-style'] == 1) { ?>
id="openMenu"><i class="fa fa-bars fa-3x"></i></a>
{ ?>
id="openMenu2"><i class="fa fa-bars fa-3x"></i></a>
jo pull-left">
?php echo esc_url( home_url() ) ?>">
rc=<?php echo $xpanel['logo']['url'] ?>">
search-box hidden-xs hidden-sm pull-left ml-10">
_search_form(); ?>
ubmit-btn hidden-xs hidden-sm pull-left ml-10">
<?php echo get_page_link($xpanel['submit-link']) ?>
user-info pull-right mr-10">
user_logged_in() ) {
- current user;
```

# Code description



mysql/oracle connector library is necessary

# DBConnection.java

```
package bankrupt;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class DBConnection {
    public static Connection dbConn;
    public static Connection conn = null; // DB연결된 상태(세션)을
    담은 객체
    public static Statement stmt=null; //statement 객체
    public static PreparedStatement pstm = null; // SQL 문을 나타
    내는 객체
    public static ResultSet rs = null; // 쿼리문을 날린것에 대한 반환값
    을 담을 객체
    public static Connection getConnection() {
        /*
         * public UserDao() {
         * try {
         *     // connect to database
         *     Class.forName("com.mysql.jdbc.Driver");
         *     // replace url with own ip
         *     con = DriverManager.getConnection("jdbc:mysql://"
         * localhost:3306/mvc_board", "root", "joo6969");
         */
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    Connection conn = null;
    try {
        String user = "root";
        String pw = "joo6969";
        String url = "jdbc:mysql://localhost:3306/bankrupt";

        Class.forName("com.mysql.jdbc.Driver");
        conn = DriverManager.getConnection(url, user, pw);
    } catch (ClassNotFoundException ce) {
        System.out.println("DB 드라이버 로딩 실패 :" +ce.toString());
    } catch (SQLException se) {
        System.out.println("DB 접속실패 :" +se.toString());
    } catch (Exception e) {
        System.out.println("Unknown error");
        e.printStackTrace();
    }
    return conn;
}
```

# RelationTable.java

```
...
public void MakeTable() throws SQLException{
    DbConnect();

    String table1= "CREATE TABLE BankUser (" +
        "Id          VARCHAR(20)      NOT NULL," +
        "AdminId     VARCHAR(20)," +
        "Passwd      VARCHAR(20)," +
        "Name        VARCHAR(20)," +
        "Country     VARCHAR(20)," +
        "PRIMARY KEY(Id)," +
        "FOREIGN KEY(Country) REFERENCES F_Money(Country)," +
        "FOREIGN KEY(AdminId) REFERENCES Admin(Id))";

    String table2= "CREATE TABLE BankAccount (" +
        "UserId       VARCHAR(20)      NOT NULL," +
        "CreateAccountTime VARCHAR(40)    NOT NULL," +
        "AccountNumber VARCHAR(40)      NOT NULL," +
        "Money        INT," +
        "PRIMARY KEY(UserId, CreateAccountTime, AccountNumber)," +
        "FOREIGN KEY (UserId) REFERENCES BankUser(Id))";

    String table3= "CREATE TABLE Admin (" +
        "Id          VARCHAR(20)      NOT NULL," +
        "Passwd      VARCHAR(20)," +
        "Name        VARCHAR(20)," +
        "PRIMARY KEY(Id))";

    String table4= "CREATE TABLE F_Money (" +
        "Country     VARCHAR(20) NOT NULL," +
        "Rate         DOUBLE," +
        "PRIMARY KEY(Country))";
    ...
}
```

**NUMBER(Oracle)**  
**DOUBLE(Mysql/MariaDB)**

# User.java

```
...
public class User {
    static Connection conn = null; // DB연결된 상태(세션)을 담은 객체
    static Statement stmt=null;    //statement 객체
    static PreparedStatement pstm = null; // SQL 문을 나타내는 객체
    static ResultSet rs = null; // 쿼리문을 날린것에 대한 반환값을 담을 객체
    static Scanner key = new Scanner(System.in);
    protected String id;
    protected String passwd;
    protected String name;
    protected String country;
    //생성자들
    public User(){
        id=null;
        passwd=null;
        name=null;
        country=null;
    }
    public User(String inpId, String inpPasswd, String inpName, String inpCountry){
        id=inpId;
        passwd=inpPasswd;
        name=inpName;
        country=inpCountry;
    }
}
...
```

# BankAccount.java

```
...
public class BankAccount extends User {
    static Connection conn = null; // DB연결된 상태(세션)을 담은 객체
    static Statement stmt=null;      //statement 객체
    static PreparedStatement pstm = null; // SQL 문을 나타내는 객체
    static ResultSet rs = null; // 쿼리문을 날린것에 대한 반환값을 담을 객체
    static Scanner key = new Scanner(System.in);
    private String createAccountTime;
    private String accountNumber;
    private int money;
    Time t = new Time();
    public BankAccount(){
        super();
        createAccountTime=null;
        accountNumber=null;
        money=0;
    }
    public BankAccount(String inpld, String inpPasswd, String inpName, String inpCountry, String inpCreateAccountTime, String
inpAccountNumber, int inpMoney){
        super(inpld, inpPasswd, inpName, inpCountry);
        createAccountTime=inpCreateAccountTime;
        accountNumber=inpAccountNumber;
        money=inpMoney;
    }
...
}
```

# Admin.java \_USER CRUD

```
...
//User정보 열람/찾기
public void FindUserInfo() throws SQLException{
    DbConnect();
    String query="select * from Bankuser";
    pstmt=conn.prepareStatement(query);
    rs=pstmt.executeQuery();

    System.out.println("Id"+ "\t"+ "AdminId"+ "\t\t"+ "Passwd"+ "\t\t"+ "Name"
        +"\t"+ "Country");

    System.out.println("=====");
    =====;
    while(rs.next()){
        String id=rs.getString(1);
        String adminId=rs.getString(2);
        String passwd=rs.getString(3);
        String name=rs.getString(4);
        String country=rs.getString(5);
        String
result=id+ "\t"+adminId+ "\t\t"+passwd+ "\t\t"+name+ "\t"+country;
        System.out.println(result);
    }
    DBDisConnect();
}
//User 1명의 정보 삭제
public void OneUserDelete(String inpName) throws SQLException{
    DbConnect();
```

```
    String query="select name from Bankuser where name =
    "+inpName+"";
    pstmt=conn.prepareStatement(query);
    rs=pstmt.executeQuery();
    while(rs.next()){
        if(inpName.equals(rs.getString("Name"))){
            String deleteSql="delete from bankUser where Name
= "+inpName+"";
            stmt.executeUpdate(deleteSql);
            stmt.executeUpdate("commit");
            System.out.println("사용자 삭제 완료!");
            DBDisConnect();
            return;
        }
    }
    System.out.println("존재하지 않는 사용자입니다!");
    DBDisConnect();
}
//User 전체 정보 삭제
public void AllUserDelete() throws SQLException{
    DbConnect();
    String deleteSql="delete from Bankuser";
    stmt.executeUpdate(deleteSql);
    stmt.executeUpdate("commit");
    System.out.println("사용자 전체 삭제 완료!");
    DBDisConnect();
}
...
```

# Main.java \_USER CRUD

```
//회원가입
public static void MakeUser() throws SQLException {
    String name,passwd,country,id;
    System.out.println("=====");
    System.out.println("  [회원가입]");
    System.out.println("-----");
    System.out.print("이름: ");
    name=key.next();
    System.out.print("아이디: ");
    id=key.next();
    System.out.print("비밀번호: ");
    passwd=key.next();
    System.out.print("국적: ");
    country=key.next();
    System.out.println("=====");
    User u = new User(id,passwd,name,country);
    u.MakeUser();
}
//회원탈퇴
public static void DeleteUser() throws SQLException{
    String id,passwd;
    System.out.println("=====");
    System.out.println("  [회원탈퇴]");
    System.out.println("-----");
    System.out.print("아이디: ");
    id=key.next();
    System.out.print("비밀번호: ");
    passwd=key.next();
    System.out.println("=====");
    User u = new User(id,passwd);
    u.DeleteUser();
}
```

= User Update not implemented

**DEMO**  
**BANKRUPT**



## **Code below is specific to users;**

```
/Library/Java/JavaVirtualMachines/jdk-9.0.1.jdk/Contents/Home/bin/java "-javaagent:/Applications/IntelliJ IDEA.app/  
Contents/lib/idea_rt.jar=62993:/Applications/IntelliJ IDEA.app/Contents/bin" -Dfile.encoding=UTF-8 -classpath /Users/  
hjdev/Documents/Projects/Java/bankrupt/out/production/bankrupt:/Users/hjdev/Downloads/mysql-connector-java-5.1.45/  
mysql-connector-java-5.1.45-bin.jar bankrupt.Main
```