

1. 데이터베이스

예제 1)

```
import java.sql.*;
class JDBC_Connect01{
    public static void main(String[] args) {
        String driver = "sun.jdbc.odbc.JdbcOdbcDriver";

        //Oracle
        String driver="oracle.jdbc.driver.OracleDriver";

        //Mysql
        String driver="com.mysql.jdbc.Driver";

        try{
            Class.forName(driver);
            System.out.println("데이터베이스 연결 성공!");
        }
        catch(Exception e){
            System.out.println("데이터베이스 연결 실패!");
        }
    }
}
```

예제2)

```
import java.sql.*;
class JDBC_Connect02{
    public static void main(String[] args) {
        String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
        String url = "jdbc:odbc:customerDSN";

        //Oracle
        // String driver="oracle.jdbc.driver.OracleDriver";
        // String url="jdbc:oracle:thin:@localhost:1521:orcl";
        // String user="db사용자";
        // String psword="db패스워드";

        //Mysql
        // String driver="com.mysql.jdbc.Driver";
        // String url="jdbc:mysql://localhost:3306/test";
        // String user="db사용자";
        // String psword="db패스워드";

        Connection con = null;
        try{
            Class.forName(driver);
            con = DriverManager.getConnection(url, "scott", "tiger" );
            System.out.println("데이터베이스 연결 성공!");
        }
        catch(Exception e){
            System.out.println("데이터베이스 연결 실패!");
        }
        finally{
            try{
                if( con != null )            con.close();
            }
            catch(Exception e){
                System.out.println( e.getMessage( ));
            }
        }
    }
}
```

예제3)

```
import java.sql.*;
class JDBC_Connect03{
    public static void main(String[] args) {
        String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
        String url = "jdbc:odbc:customerDSN";
        Connection con = null;
        Statement stmt = null ;
        try{
            Class.forName(driver);
            con = DriverManager.getConnection(url, "scott", "tiger" );
            stmt= con.createStatement();
            System.out.println("데이터베이스 연결 성공!");
        }
        catch(Exception e){
            System.out.println("데이터베이스 연결 실패!");
        }
        finally{
            try{
                if( stmt != null )        stmt.close();
                if( con != null )        con.close();
            }
            catch(Exception e){
                System.out.println( e.getMessage( ));
            }
        }
    }
}
```

예제4)

```
import java.sql.*;
class JDBC_Select{
    public static void main(String[] args) {
        String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
        String url = "jdbc:odbc:customerDSN";
        Connection con = null;
        Statement stmt = null ;
        //---JDBC_Select 추가된 내용 -----
        ResultSet rs = null;
        int no = 0;
        String name, email, tel; //데이터베이스에서 얻어온 필드값 저장할 변수 선언
        String sql;             //SQL문을 저장할 변수 선언
        try{
            Class.forName(driver);
            con = DriverManager.getConnection(url, "scott", "tiger" );
            stmt= con.createStatement();
            //---JDBC_Select 추가된 내용 -----
            sql = "SELECT * FROM customer";
            System.out.printf("번호 Wt 이름 WtWt 이메일 WtWt 전화번호 Wn");
            System.out.printf("-----Wn");
            rs = stmt.executeQuery(sql); //얻어진 레코드를 가져옴
            while( rs.next() ){
                no = rs.getInt("no");
                name = rs.getString("name");
                email = rs.getString("email");
                tel = rs.getString("tel");
                System.out.printf(" %d Wt %s Wt %s Wt %sWn", no, name,email,tel);
            }
        }
        catch(Exception e){
            System.out.println("데이터베이스 연결 실패!");
        }
        finally{
            try{//rs, stmt, con 객체를 close() 메서드를 호출해 해제
                if( rs != null )      rs.close();
                if( stmt != null )    stmt.close();
                if( con != null )     con.close();
            }
            catch(Exception e){
                System.out.println( e.getMessage( ));
            }
        }
    }
}
```

예제5)

```
import java.sql.*;
import java.io.*; // 도스 콘솔 창에서 사용자 입력을 받아들이기 위해 BufferedReader

class JDBC_Insert{
public static void main(String[] args) {
    String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
    String url = "jdbc:odbc:customerDSN";
    Connection con = null;
    Statement stmt = null ;

    ResultSet rs = null;
    String sql;

    String name, email, tel, no ;

    try{
        Class.forName(driver);
        con = DriverManager.getConnection(url, "scott", "tiger" );
        stmt= con.createStatement();
        //---JDBC_Insert 추가된 내용-----
        // 테이블에 추가할 내용을 도스 콘솔 창에서 사용자의 입력을 받도록 한다.
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        System.out.println(" customer 테이블에 값 입력하기 .....");
        System.out.print(" 번호 입력: ");
        no = br.readLine();
        System.out.print(" 이름 입력: ");
        name = br.readLine();           //테이블에 추가할 name 필드 값을 입력 받음
        System.out.print(" 이메일 입력: ");
        email = br.readLine();          //테이블에 추가할 email 필드 값을 입력 받음
        System.out.print(" 전화번호 입력: ");
        tel = br.readLine();            //테이블에 추가할 tel 필드 값을 입력 받음

        // INSERT 쿼리문을 작성
        sql = "INSERT into customer(no, name, email, tel) values " ;
        sql += "(" + no + "," + name + "," + email + "," + tel + ")" ;

        //Statement 객체의 executeUpdate(sql) 메서드를 이용해
        stmt.executeUpdate(sql) ; //데이터베이스 파일에 새로운 값을 추가시킴
    }
    catch(Exception e){
        System.out.println("데이터베이스 연결 실패!");
    }
    finally{
        try{
```

```

        if( rs != null )    rs.close();
        if( stmt != null ) stmt.close();
        if( con != null )  con.close();
    }
    catch(Exception e){
        System.out.println( e.getMessage());
    }
}
}
}
}

```

예제6)

```

import java.sql.*;
import java.io.*; // 도스 콘솔 창에서 사용자 입력을 받아들이기 위해 BufferedReader

class JDBC_Update{
public static void main(String[] args) {
    String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
    String url = "jdbc:odbc:customerDSN";
    Connection con = null;
    Statement stmt = null ;

    String sql;
    String name, email, tel ;

    try{
        Class.forName(driver);
        con = DriverManager.getConnection(url, "scott", "tiger" );
        stmt= con.createStatement();
        //---JDBC_Insert 추가된 내용-----
        // 테이블에 추가할 내용을 도스 콘솔 창에서 사용자의 입력을 받도록 한다.
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        System.out.println(" customer 테이블에 값 갱신하기 .....");
        System.out.print("어떤 분의 내용을 갱신할지 그분의 이름을 입력하세요: ");
        name = br.readLine();           //테이블에 추가할 name 필드 값을 입력 받음
        System.out.print("변경할 이메일 입력: ");
        email = br.readLine();          //테이블에 추가할 email 필드 값을 입력 받음
        System.out.print("변경할 전화번호 입력: ");
        tel = br.readLine();            //테이블에 추가할 tel 필드 값을 입력 받음
    }
    catch(Exception e){
        System.out.println(e.getMessage());
    }
}
}

```

```

// INSERT 쿼리문을 작성
sql = "UPDATE customer SET email=" + email;
    sql += " , tel=" + tel + " WHERE name =";
sql += name + """;

//Statement 객체의 executeUpdate(sql) 메서드를 이용해
stmt.executeUpdate(sql) ; //데이터베이스 파일의 내용을 변경시킴
}
catch(Exception e){
    System.out.println("데이터베이스 연결 실패!");
}
finally{
    try{
        if( stmt != null ) stmt.close();
        if( con != null ) con.close();
    }
    catch(Exception e){
        System.out.println( e.getMessage());
    }
}
}
}

```

예제7)

```
import java.sql.*;
import java.io.*; // 도스 콘솔 창에서 사용자 입력을 받아들이기 위해 BufferedReader

class JDBC_Delete{
public static void main(String[] args) {
    String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
    String url = "jdbc:odbc:customerDSN"; Connection con = null;
    Statement stmt = null ;

    String sql;
    String name, email, tel ;

    try{
        Class.forName(driver);
        con = DriverManager.getConnection(url, "scott", "tiger" );
        stmt= con.createStatement();
        ///---JDBC_Delete 변경된 내용-----
        // 테이블에 추가할 내용을 도스 콘솔 창에서 사용자의 입력을 받도록 한다.
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        System.out.println(" customer 테이블에서 레코드 삭제하기 .....");
        System.out.print("어떤 분의 내용을 삭제할지 그분의 이름을 입력하세요: ");
        name = br.readLine();    //테이블에서 삭제할 name 필드 값을 입력 받음

        // DELETE 쿼리문을 작성
        sql = "DELETE FROM customer WHERE name ='";
        sql += name + "'";

        //Statement 객체의 executeUpdate(sql) 메서드를 이용해
        stmt.executeUpdate(sql) ; //데이터베이스 파일에서 레코드 삭제시킴
    }
    catch(Exception e){
        System.out.println("데이터베이스 연결 실패!");
    }
    finally{
        try{
            if( stmt != null ) stmt.close();
            if( con != null ) con.close();
        }
        catch(Exception e){
            System.out.println( e.getMessage());
        }
    }
}
}
```


예제8)

```
import java.io.*;
import java.sql.*;
class JDBC_Select2{
    public static void main(String[] args)    {
        String driver="oracle.jdbc.driver.OracleDriver";
        String url="jdbc:oracle:thin:@localhost:1521:ORCL";
        String user="scott";
        String pwd="tiger";

        Connection con=null;
        PreparedStatement pstmt= null;

        String s_name, s_email, s_tel;
        String sql;

        ResultSet rs    = null;

        try{
            Class.forName(driver);
            con = DriverManager.getConnection(url, user, pwd);

            BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

            System.out.println(
                "customer 테이블에서 찾고자하는 사람이름이나 이메일이나 전화번호를 입력하기");
            System.out.print(" 이름 입력: ");
            s_name = br.readLine();           //테이블에서 찾고자하는 사람의 이름을 입력 받음
            System.out.print(" 이메일 입력: ");
            s_email = br.readLine();          //테이블에서 찾고자하는 사람의 이메일을 입력 받음
            System.out.print(" 전화번호 입력: ");
            s_tel = br.readLine();            //테이블에서 찾고자하는 사람의 전화번호를 입력 받음

            // SELECT 쿼리문을 작성
            sql = "SELECT * FROM customer WHERE name LIKE ? AND email LIKE ? AND tel
            LIKE ? ";
            pstmt= con.prepareStatement(sql);

            if( s_name.equals("") )
                s_name = "%";
            if( s_email.equals("") )
                s_email = "%";
            if( s_tel.equals("") )
                s_tel = "%";
```

```

pstmt.setString(1, s_name);
pstmt.setString(2, s_email);
pstmt.setString(3, s_tel);

//PreparedStatement 객체의 executeQuery( ) 메소드로 데이터 검색
rs = pstmt.executeQuery();
System.out.println("WnWn ----- 검색 결과 출력 ----- " );
while( rs.next() ){
    s_name = rs.getString("name");
    s_email = rs.getString("email");
    s_tel = rs.getString("tel");
    //컬럼 값을 화면에 출력
    System.out.println(s_name + " Wt " + s_email + " Wt " + s_tel );
} //while

rs.close();
pstmt.close();
con.close();
} //try
catch(IOException e){
    e.printStackTrace( );
}
}
}

```

예제9)

```
import java.io.*;
import java.sql.*;
class JDBC_Insert2{
    public static void main(String[] args)  {
        Connection con=null;
        String url="jdbc:oracle:thin:@localhost:1521:ORCL";

        PreparedStatement pstmt= null; //PreparedStatement 객체 변수 pstmt를 선언
        int no;
        String name, email, tel, temp; //데이터베이스에서 얻어온 필드 값 저장할 변수 선언
        String sql;                //SQL문을 저장할 변수 선언

        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            con = DriverManager.getConnection(url , "scott", "tiger");

            //---JDBC_Insert2 추가된 내용-----
            // 테이블에 추가할 내용을 도스 콘솔 창에서 사용자의 입력을 받도록 한다.
            BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
            System.out.println(" customer 테이블에 값 입력하기 .....");
            System.out.print(" 번호 입력: ");
            temp = br.readLine();
            System.out.print(" 이름 입력: ");
            name = br.readLine();           //테이블에 추가할 name 필드 값을 입력 받음
            System.out.print(" 이메일 입력: ");
            email = br.readLine();          //테이블에 추가할 email 필드 값을 입력 받음
            System.out.print(" 전화번호 입력: ");
            tel = br.readLine();            //테이블에 추가할 tel 필드 값을 입력 받음

            // ? 문자가 포함된 INSERT 문을 작성
            sql = "INSERT INTO customer(no, name, email, tel) values(?,?,?,?)" ;

            //PreparedStatement를 객체를 생성
            pstmt = con.prepareStatement(sql);
            no = Integer.parseInt(temp);
            pstmt.setInt(1,no); //첫 번째 ? 에 사용자로부터 입력받은 번호를 대입
            pstmt.setString(2,name); //두 번째 ? 에 사용자로부터 입력받은 이름을 대입
            pstmt.setString(3,email); //세 번째 ? 에 사용자로부터 입력받은 이메일을 대입
            pstmt.setString(4,tel); //네 번째 ? 에 사용자로부터 입력받은 전화번호를 대입

            //PreparedStatement 객체의 executeUpdate( ) 메서드를 이용해
            pstmt.executeUpdate() ; //데이터베이스 파일에 새로운 값을 추가시킴
```

```

    }catch(ClassNotFoundException e){
        System.out.println( e );
    }catch(SQLException e){
        System.out.println( e );
    }catch(IOException e){
        System.out.println( e );
    }finally{
        try{
            if( pstmt != null ) pstmt.close();    //PreparedStatement 객체를 메모리에서 해제
            if( con != null ) con.close();
        }
        catch(Exception e){
            System.out.println( e.getMessage());
        }
    }//finally 끝
}
}

```

예제10)

```

import java.io.*;
import java.sql.*;
class CallableStatementTest{
    public static void main(String[] args)  {
        Connection con=null;
        String url="jdbc:oracle:thin:@localhost:1521:ORCL";
        String sql=null;
        CallableStatement cs = null;

        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            con = DriverManager.getConnection(url , "scott", "tiger");

            sql = "{call del_all()}";

            //CallableStatement를 객체를 생성
            cs = con.prepareCall(sql);
            cs.execute();
            cs.close();
            con.close();
        }catch(Exception e){
            System.out.println(e);
        }
    }
}

```