

```

1 package ch_34_db;
2
3 import java.sql.Connection;
4 import java.sql.PreparedStatement;
5 import java.sql.ResultSet;
6 import java.sql.DriverManager;
7 import java.sql.SQLException;
8
9 /*
10 DBMS와 연결하기.
11 java.sql 패키지가 제공하는 Connection 객체를 사용.
12 Connection 객체는 역시 java.sql 패키지에 포함되어 있는 DriverManager 클래스의 getConnection() 메서드를 이용.
13 getConnection(String url, String user, String password) 메서드는 static 메서드로서
14 url은 서버와 데이터베이스 위치를 나타내고, user는 사용자 접속 아이디, password는 접속 비밀번호를 지정.
15 */
16 class DB {
17     Connection conn = null;
18     PreparedStatement preparedStatement = null;
19     ResultSet resultSet = null; // 쿼리시에 결과를 저장하는 용도로 사용. select 에 주로 사용
20
21     public void connectDB() {
22         final String driver = "org.mariadb.jdbc.Driver";
23         final String DB_HOST = "127.0.0.1";
24         final String DB_PORT = "3306";
25         final String DB_NAME = "sample"; // 데이터 베이스 이름
26         final String DB_URL = "jdbc:mariadb://" + DB_HOST + ":" + DB_PORT + "/" + DB_NAME;
27         final String DB_USER = "root";
28         final String DB_PASS = "3432";
29
30         // 디비 접속
31         try {
32             Class.forName(driver);
33             conn = DriverManager.getConnection(DB_URL, DB_USER, DB_PASS);
34             if (conn != null) {
35                 System.out.println("DB 접속 성공");
36             }
37         } catch (ClassNotFoundException e) {
38             System.out.println("드라이버 로드 실패");
39             e.printStackTrace();
40         } catch (SQLException e) {
41             System.out.println("DB 접속 실패");
42             e.printStackTrace();
43         }
44     }
45
46     public void closeDB() {
47         try {
48             if (resultSet != null) { // 열린 경우 닫음
49                 resultSet.close();
50             }
51             if (preparedStatement != null) { // 열린 경우 닫음
52                 preparedStatement.close();
53             }
54
55             if (conn != null && !conn.isClosed()) {
56                 conn.close();
57                 System.out.println("DB 접속 해제");
58             }
59         } catch (SQLException e) {
60             e.printStackTrace();
61         }
62     }
63 }
64
65 public class My_02_ConnectDB {
66     // Connection 객체를 생성해 DataBase 연결하기
67
68     public static void main(String[] args) {
69         // TODO Auto-generated method stub
70         DB myDB = new DB();
71         myDB.connectDB();
72         myDB.closeDB();
73     }
74
75 }
76

```