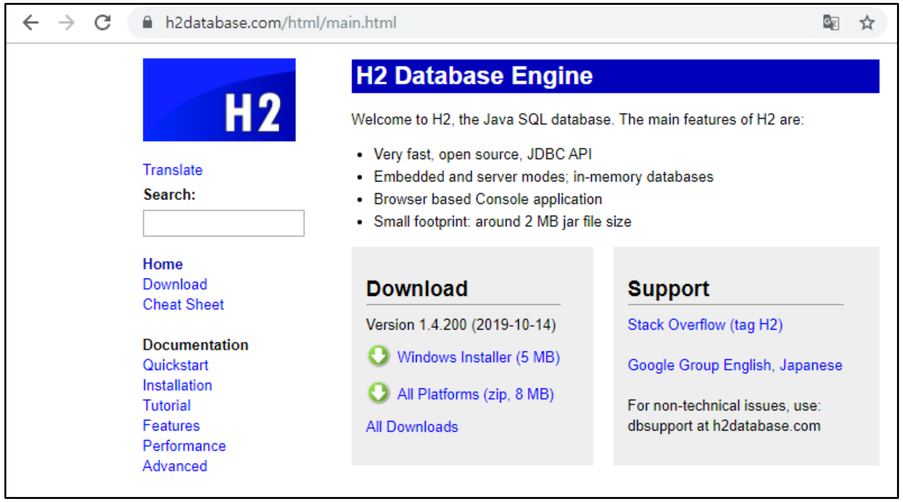
Database

H2



h2database.com/html/main.html

Translate

Search:

Home
Download
Cheat Sheet

Documentation
Quickstart
Installation
Tutorial
Features
Performance
Advanced

H2 Database Engine

Welcome to H2, the Java SQL database. The main features of H2 are:

- Very fast, open source, JDBC API
- Embedded and server modes; in-memory databases
- Browser based Console application
- Small footprint: around 2 MB jar file size

Download

Version 1.4.200 (2019-10-14)

Windows Installer (5 MB)

All Platforms (zip, 8 MB)

All Downloads

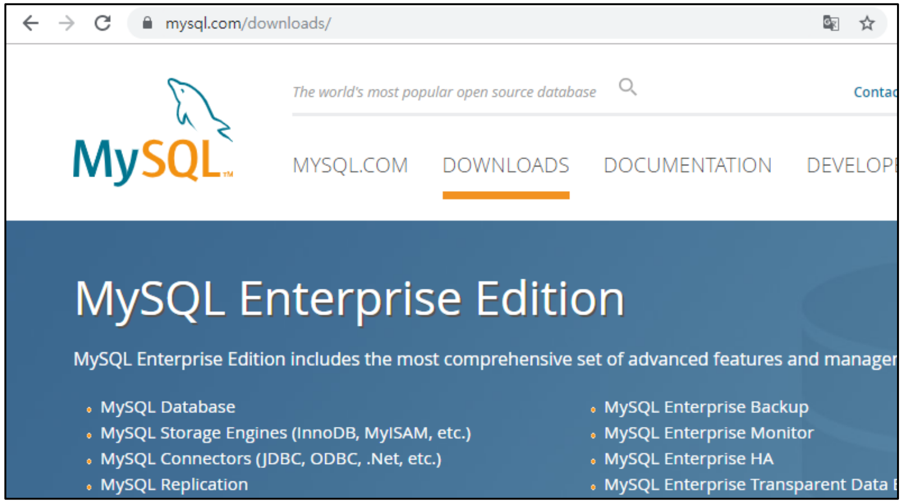
Support

Stack Overflow (tag H2)

Google Group English, Japanese

For non-technical issues, use:
dbsupport at h2database.com

MySQL



mysql.com/downloads/

The world's most popular open source database

Contac

MYSQL.COM DOWNLOADS DOCUMENTATION DEVELOP

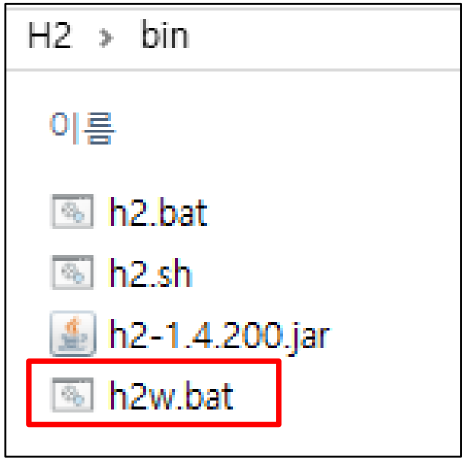
MySQL Enterprise Edition

MySQL Enterprise Edition includes the most comprehensive set of advanced features and manager

- MySQL Database
- MySQL Storage Engines (InnoDB, MyISAM, etc.)
- MySQL Connectors (JDBC, ODBC, .Net, etc.)
- MySQL Replication
- MySQL Enterprise Backup
- MySQL Enterprise Monitor
- MySQL Enterprise HA
- MySQL Enterprise Transparent Data E

■ H2

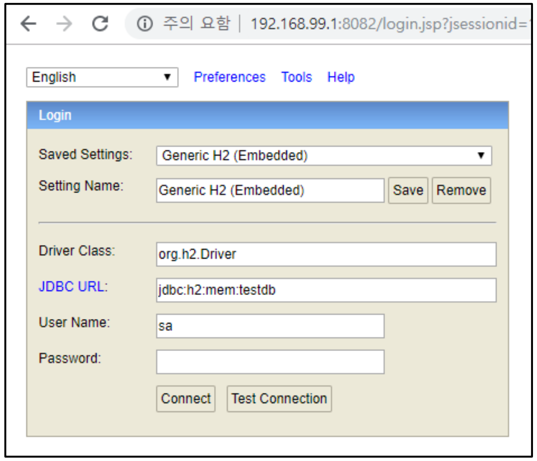
● Server Start 및 Console 접속



H2 › bin

이름

- h2.bat
- h2.sh
- h2-1.4.200.jar
- h2w.bat



ⓘ 주의 요함 | 192.168.99.1:8082/login.jsp?jsessionid=

Login

Saved Settings: Generic H2 (Embedded)

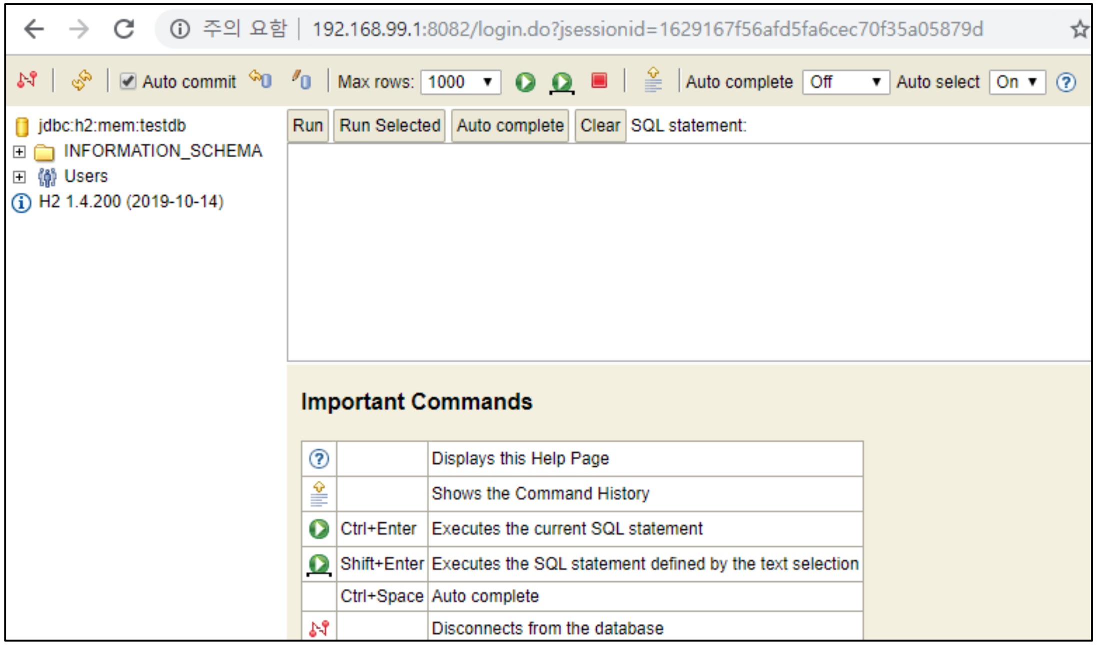
Setting Name: Generic H2 (Embedded)

Driver Class: org.h2.Driver

JDBC URL: jdbc:h2:mem:testdb

User Name: sa

Password:



ⓘ 주의 요함 | 192.168.99.1:8082/login.do?jsessionid=1629167f56afd5fa6cec70f35a05879d

☑ Auto commit

Max rows: 1000

Auto complete: Off

Auto select: On

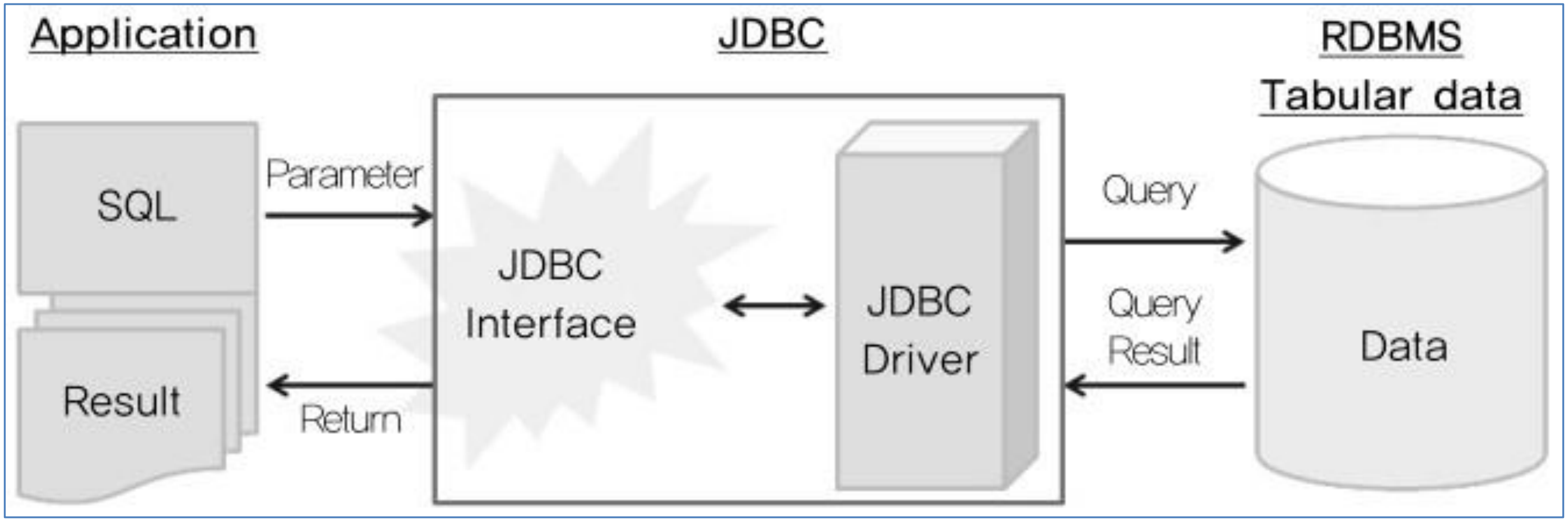
Important Commands

?		Displays this Help Page
		Shows the Command History
▶	Ctrl+Enter	Executes the current SQL statement
▶	Shift+Enter	Executes the SQL statement defined by the text selection
	Ctrl+Space	Auto complete
		Disconnects from the database

■ JDBC (Java Database Connectivity)

- ● 자바에서 데이터베이스에 접속할 수 있도록 하는 자바 API

- ● CRUD
  - Create
  - Read (Retrieve)
  - Update
  - Delete (Destroy)



■ JDBC (Java Database Connectivity)

● 데이터베이스 작업 흐름

1. 드라이버 로딩

```
Class.forName("oracle.jdbc.driver.OracleDriver");
Class.forName("com.mysql.jdbc.Driver");
Class.forName("org.h2.Driver");
```

2. 데이터베이스 연결

```
Connection con = DriverManager.getConnection(url, id, pw);
String url = "jdbc:oracle:thin:@localhost:1521:xe";
String url = "jdbc:mysql://localhost:3306/java";
String url = "jdbc:h2:tcp://localhost/~/testdb";
```

■ JDBC (Java Database Connectivity)

● 데이터베이스 작업 흐름

3. SQL (Query) 실행 준비

```
String sql = "select * from emp";

String sql = "insert into (id, pw) values (?, ?);

PreparedStatement stmt = conn.prepareStatement(sql);
stmt.setString(1, "userid");
stmt.setString(2, "password");
```

4. SQL (Query) 실행

- insert, update, delete : stmt.executeUpdate();
- select : stmt.executeQuery();

5. (조회시) 조회결과 처리

```
while(rs.next()) {
    String empno = rs.getString(“empno");
    String ename = rs.getString(“ename");
    String job = rs.getString(“job");
}
```

■ JDBC 기본 코드 (1 / 2)

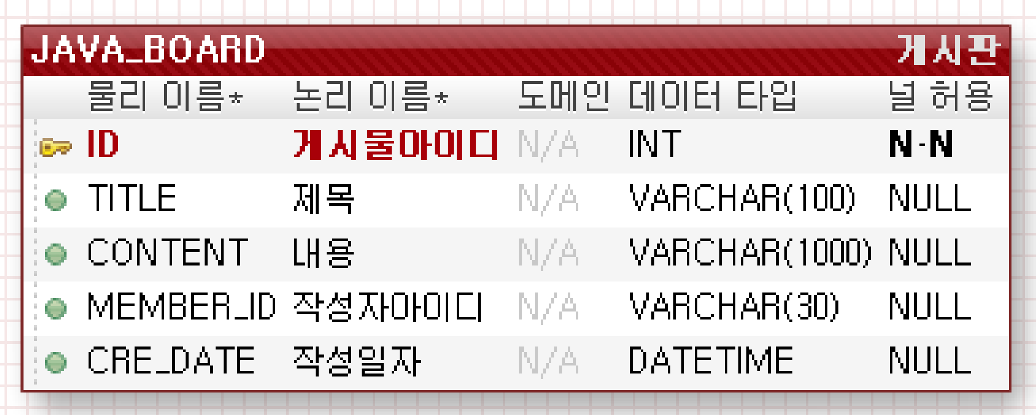
```java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

public class JdbcFrame {
  public static void main(String[] args) {
    String url = "jdbc:mysql://localhost:3306/java";
    String id = "root";
    String pw = "mysql";

    Connection con = null; // DB 연결
    PreparedStatement stmt = null; // Query 실행
    ResultSet rs = null; // 조회 Query 결과 저장

    // 쿼리 작성
    StringBuffer query = new StringBuffer();

    try {
      // 1. 드라이버 로딩
      Class.forName("com.mysql.jdbc.Driver");

      // 2. DB 연결
      con = DriverManager.getConnection(url, id, pw);
```

■ JDBC 기본 코드 (2 / 2)

```
      // 3. Query를 실행할 수 있도록 Statement 객체 준비
      stmt = con.prepareStatement(query.toString());

      rs = stmt.executeQuery(); // 4. Query 실행
      while(rs.next()) { // 5. 조회결과 처리
        // rs.getString();
      }
    } catch (ClassNotFoundException e) {
      e.printStackTrace();
    } catch (SQLException e) {
      e.printStackTrace();
    } finally {
      if(rs != null) {
        try { rs.close(); }
        catch(SQLException e) {}
      }
      if(stmt != null) {
        try { stmt.close(); }
        catch(SQLException e) {}
      }
      if(con != null) {
        try { con.close(); }
        catch(SQLException e) {}
      }
    }
  }
}
```

■ JDBC CRUD 연습

● ERD (Entity Relationship Diagram)



JAVA_BOARD 게시판

물리 이름*	논리 이름*	도메인	데이터 타입	널 허용
ID	게시물아이디	N/A	INT	N-N
TITLE	제목	N/A	VARCHAR(100)	NULL
CONTENT	내용	N/A	VARCHAR(1000)	NULL
MEMBER_ID	작성자아이디	N/A	VARCHAR(30)	NULL
CRE_DATE	작성일자	N/A	DATETIME	NULL

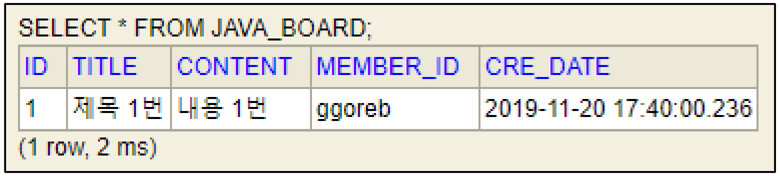
```
CREATE TABLE JAVA_BOARD (
    ID          INT             NOT NULL  COMMENT '게시물아이디',
    TITLE       VARCHAR(100)    NULL      COMMENT '제목',
    CONTENT     VARCHAR(1000)   NULL      COMMENT '내용',
    MEMBER_ID   VARCHAR(30)     NULL      COMMENT '작성자아이디',
    CRE_DATE    DATETIME        NULL      COMMENT '작성일자',
    PRIMARY KEY (ID)
)
```

■ JDBC CRUD 연습 – 1 (1 / 2)

```java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class H2_Sample1 {
  public static void main(String[] args)
      throws ClassNotFoundException, SQLException {
    String url = "jdbc:h2:tcp://localhost/~/testdb";
    String username = "sa";
    String password = "";

    Class.forName("org.h2.Driver");

    StringBuffer query = new StringBuffer();
    query.append("INSERT INTO JAVA_BOARD");
    query.append(" (ID, TITLE, CONTENT, MEMBER_ID, CRE_DATE)");
    query.append(" VALUES ");
    query.append(" (?, ?, ?, ?, CURRENT_TIMESTAMP())");
```

■ JDBC CRUD 연습 – 1 (2 / 2)

```
    Connection con =
        DriverManager.getConnection(url, username, password);
    PreparedStatement stmt =
        con.prepareStatement(query.toString());

    stmt.setInt(1, 1);
    stmt.setString(2, "제목 1번");
    stmt.setString(3, "내용 1번");
    stmt.setString(4, "ggoreb");

    stmt.executeUpdate();

    stmt.close();
    con.close();
  }
}
```



SELECT * FROM JAVA_BOARD;

ID	TITLE	CONTENT	MEMBER_ID	CRE_DATE
1	제목 1번	내용 1번	ggoreb	2019-11-20 17:40:00.236

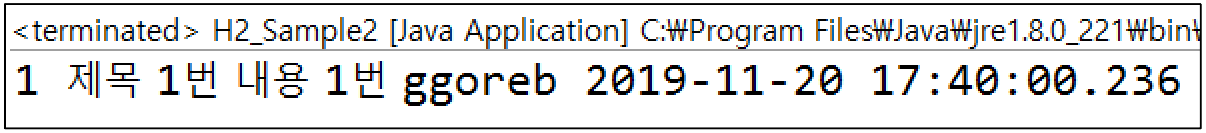
(1 row, 2 ms)

■ JDBC CRUD 연습 – 2 (1 / 2)

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

public class H2_Sample2 {
  public static void main(String[] args)
      throws ClassNotFoundException, SQLException {
    String url = "jdbc:h2:tcp://localhost/~/testdb";
    String username = "sa";
    String password = "";

    Class.forName("org.h2.Driver");

    StringBuffer query = new StringBuffer();
    query.append("SELECT ID, TITLE, CONTENT, MEMBER_ID, CRE_DATE");
    query.append("  FROM JAVA_BOARD");

    Connection con =
        DriverManager.getConnection(url, username, password);
```

■ JDBC CRUD 연습 – 2 (2 / 2)

```
    PreparedStatement stmt =
        con.prepareStatement(query.toString());
    ResultSet rs = stmt.executeQuery();

    while(rs.next()) {
      String id = rs.getString("ID");
      String title = rs.getString("TITLE");
      String content = rs.getString("CONTENT");
      String memberId = rs.getString("MEMBER_ID");
      String creDate = rs.getString("CRE_DATE");
      System.out.printf("%s %s %s %s %s\n",
          id, title, content, memberId, creDate);
    }
    rs.close();
    stmt.close();
    con.close();
  }
}
```



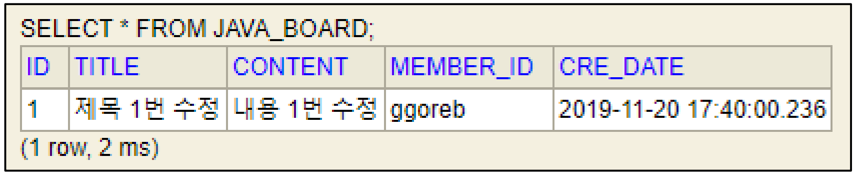
```
<terminated> H2_Sample2 [Java Application] C:₩Program Files₩Java₩jre1.8.0_221₩bin₩
1 제목 1번 내용 1번 ggoreb 2019-11-20 17:40:00.236
```

■ JDBC CRUD 연습 – 3 (1 / 2)

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

public class H2_Sample3 {
  public static void main(String[] args)
      throws ClassNotFoundException, SQLException {
    String url = "jdbc:h2:tcp://localhost/~/testdb";
    String username = "sa";
    String password = "";

    Class.forName("org.h2.Driver");

    StringBuffer query = new StringBuffer();
    query.append("UPDATE JAVA_BOARD SET TITLE = ?, CONTENT = ?");
    query.append(" WHERE ID = ?");

    Connection con =
        DriverManager.getConnection(url, username, password);
```

■ JDBC CRUD 연습 – 3 (2 / 2)

```
    PreparedStatement stmt =
        con.prepareStatement(query.toString());
    stmt.setString(1, "제목 1번 수정");
    stmt.setString(2, "내용 1번 수정");
    stmt.setInt(3, 1);
    stmt.executeUpdate();

    stmt.close();
    con.close();
  }
}
```



SELECT * FROM JAVA_BOARD;

ID	TITLE	CONTENT	MEMBER_ID	CRE_DATE
1	제목 1번 수정	내용 1번 수정	ggoreb	2019-11-20 17:40:00.236

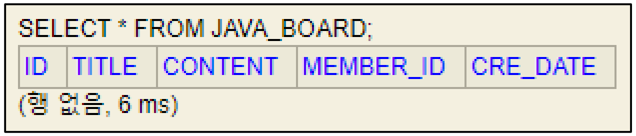
(1 row, 2 ms)

■ JDBC CRUD 연습 – 4 (1 / 2)

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

public class H2_Sample4 {
  public static void main(String[] args)
      throws ClassNotFoundException, SQLException {
    String url = "jdbc:h2:tcp://localhost/~/testdb";
    String username = "sa";
    String password = "";

    Class.forName("org.h2.Driver");

    StringBuffer query = new StringBuffer();
    query.append("DELETE FROM JAVA_BOARD");
    query.append(" WHERE ID = ?");

    Connection con =
        DriverManager.getConnection(url, username, password);
```

■ JDBC CRUD 연습 – 4 (2 / 2)

```
    PreparedStatement stmt =
        con.prepareStatement(query.toString());
    stmt.setInt(1, 1);
    stmt.executeUpdate();

    stmt.close();
    con.close();
  }
}
```



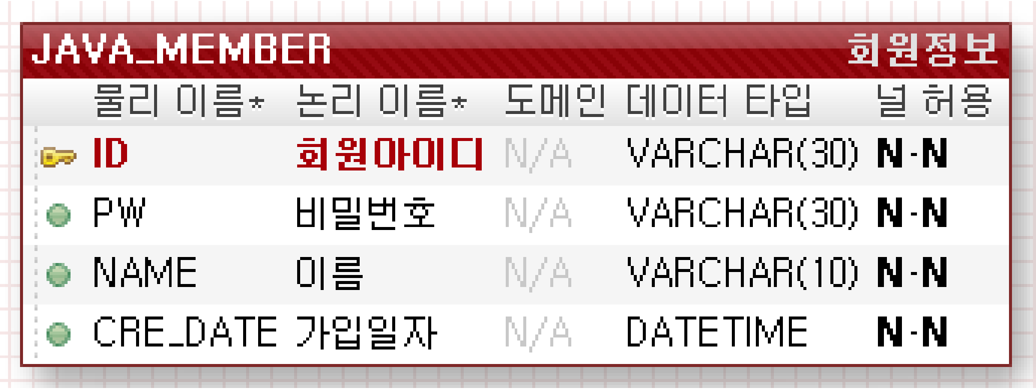
SELECT * FROM JAVA_BOARD;

ID	TITLE	CONTENT	MEMBER_ID	CRE_DATE

(행 없음, 6 ms)

■ 회원 로그인

● ERD (Entity Relationship Diagram)



JAVA_MEMBER 회원정보

	물리 이름*	논리 이름*	도메인	데이터 타입	널 허용
🔑	ID	회원아이디	N/A	VARCHAR(30)	N-N
●	PW	비밀번호	N/A	VARCHAR(30)	N-N
●	NAME	이름	N/A	VARCHAR(10)	N-N
●	CRE_DATE	가입일자	N/A	DATETIME	N-N

```
CREATE TABLE JAVA_MEMBER (
    ID        VARCHAR(30) NOT NULL COMMENT '회원아이디',
    PW        VARCHAR(30) NOT NULL COMMENT '비밀번호',
    NAME      VARCHAR(10) NOT NULL COMMENT '이름',
    CRE_DATE DATETIME    NOT NULL COMMENT '가입일자',
    PRIMARY KEY (ID)
)
```

■ 회원 로그인 (1 / 2)

```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 Login {
  public static void main(String[] args)
      throws ClassNotFoundException, SQLException {
    String url = "jdbc:h2:tcp://localhost/~/testdb";
    String username = "sa";
    String password = "";

    Class.forName("org.h2.Driver");

    StringBuffer query = new StringBuffer();
    query.append("SELECT PW FROM JAVA_MEMBER");
    query.append(" WHERE ID = ?");

    Connection con =
        DriverManager.getConnection(url, username, password);
    PreparedStatement stmt =
        con.prepareStatement(query.toString());
```

■ 회원 로그인 (2 / 2)

```
    Scanner scan = new Scanner(System.in);
    System.out.print("아이디 입력 : ");
    String userId = scan.nextLine();
    System.out.print("비밀번호 입력 : ");
    String userPw = scan.nextLine();

    stmt.setString(1, userId);

    ResultSet rs = stmt.executeQuery();
    if (rs.next()) {
      String dbPw = rs.getString("PW");
      if (dbPw.equals(userPw)) {
        System.out.println("로그인 되었습니다.");
      } else {
        System.out.println("비밀번호를 확인해주세요.");
      }
    } else {
      System.out.println("아이디가 존재하지 않습니다.");
    }

    scan.close();
  }
}
```