

# Class 230113

오라클 제거부터 하기

## Hour 1 Take a picture with Signal

mysql



mysql download



All

Videos

Books

News

Images

More

About 122,000,000 results (0.43 seconds)

<https://www.mysql.com> > downloads

## MySQL Downloads

**MySQL** Cluster CGE. **MySQL** Cluster is a real-time open source transactional database designed for fast, always-on access to data under high throughput ...

You've visited this page many times. Last visit: 1/12/23

### MySQL Installer 8.0.31

(mysql-installer-web-community-8.0.31.0.msi) ...

### MySQL Community Server

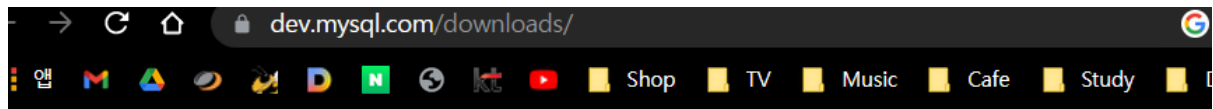
Linux - Generic (glibc 2.12) (x86, 32-bit), Compressed TAR ...

### MySQL Community Downloads

MySQL Community Server · MySQL Cluster · MySQL ...

### MySQL Workbench 8.0.31

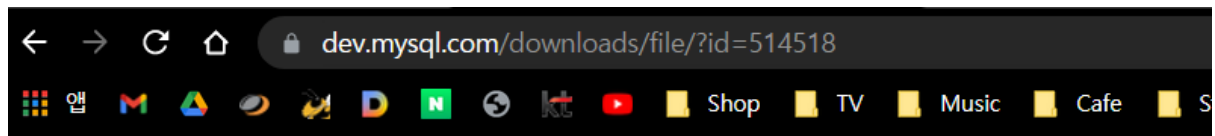
The following LGPL libraries are used by MySQL Workbench and ...



## MySQL Community Downloads

- MySQL Yum Repository
- MySQL APT Repository
- MySQL SUSE Repository
- MySQL Community Server
- MySQL Cluster
- MySQL Router
- MySQL Shell
- MySQL Operator
- MySQL NDB Operator
- MySQL Workbench
- MySQL Installer for Windows
- C API (libmysqlclient)
- Connector/C++
- Connector/J
- Connector/NET
- Connector/Node.js
- Connector/ODBC
- Connector/Python
- MySQL Native Driver for PHP
- MySQL Benchmark Tool
- Time zone description tables
- Download Archives





## MySQL Community Downloads

**Login Now or Sign Up for a free account.**

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

**Login »**  
using my Oracle Web account

**Sign Up »**  
for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can sign up for a free account by clicking the Sign Up link and following the instructions.

**No thanks, just start my download.**



heidisql



All

Images

Videos

Books

News

More

About 349,000 results (0.31 seconds)

<https://www.heidisql.com>

## HeidiSQL - MariaDB, MySQL, MSSQL, PostgreSQL and ...

**HeidiSQL** is free software, and has the aim to be easy to learn. "Heidi" lets you see and edit data and structures from computers running one of the database ...

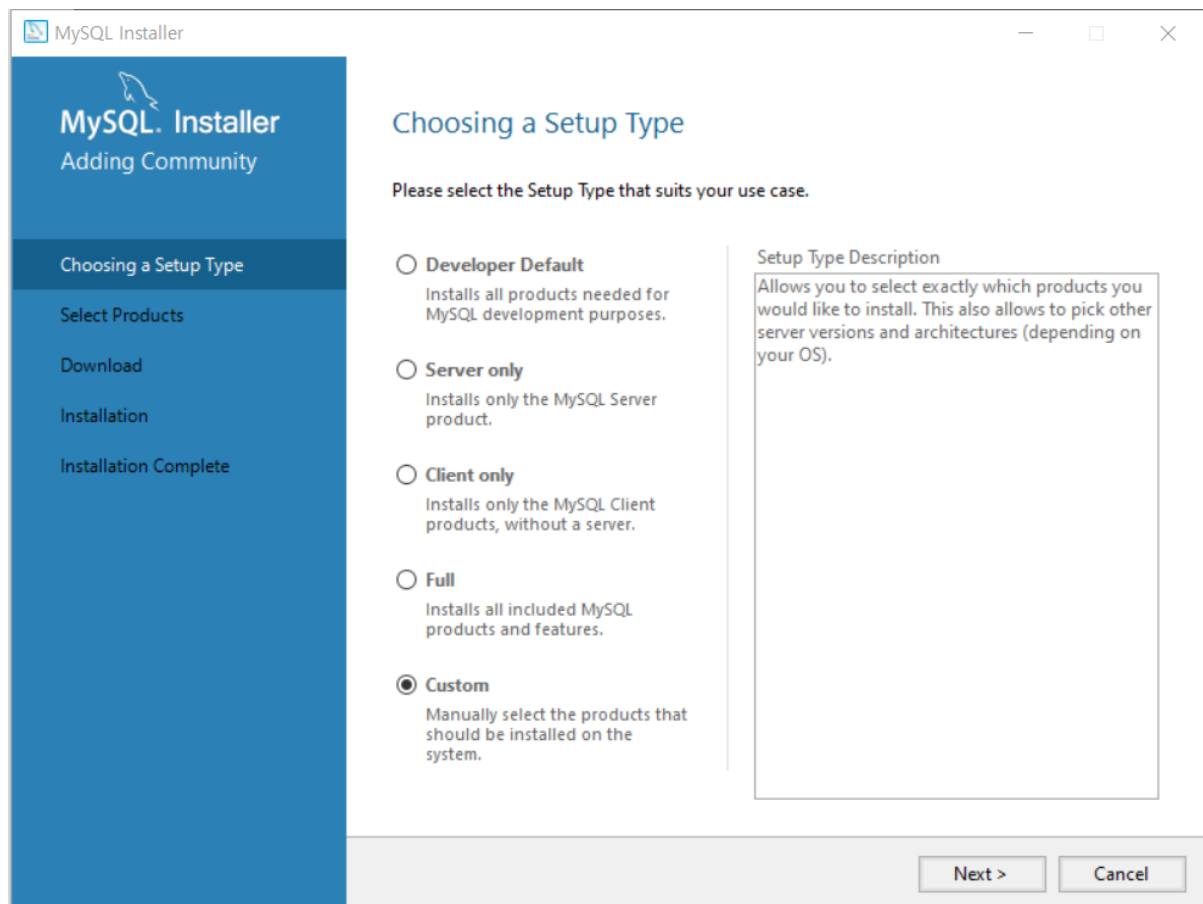
You've visited this page many times. Last visit: 1/12/23

### Downloads

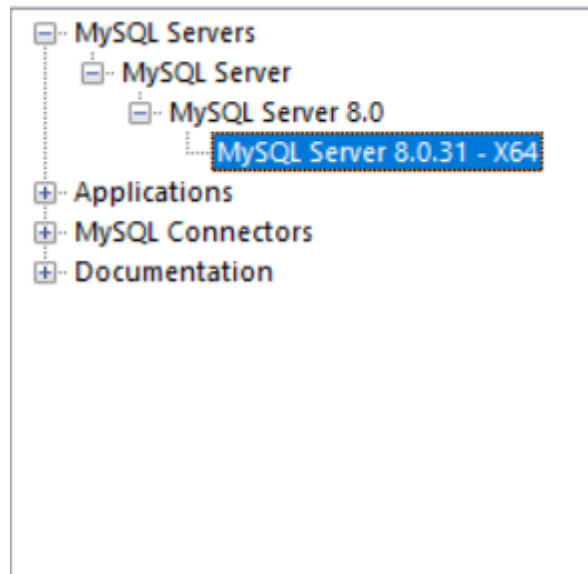
HeidiSQL runs fine on Windows 10 and 11 (and on Windows 7 + 8 ...)

### Basic help on using HeidiSQL

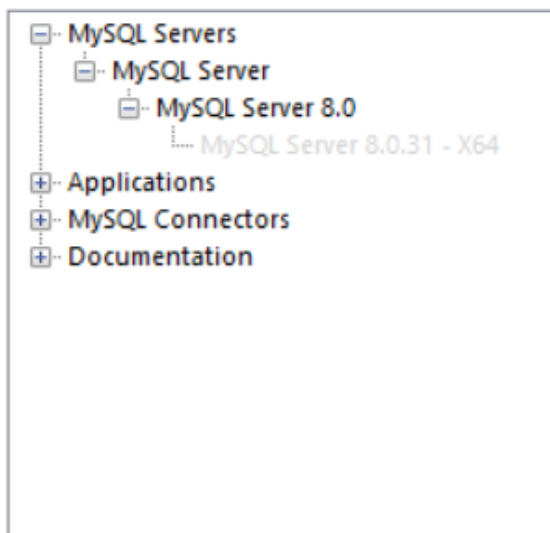
HeidiSQL is a so called client application, only usable when ...



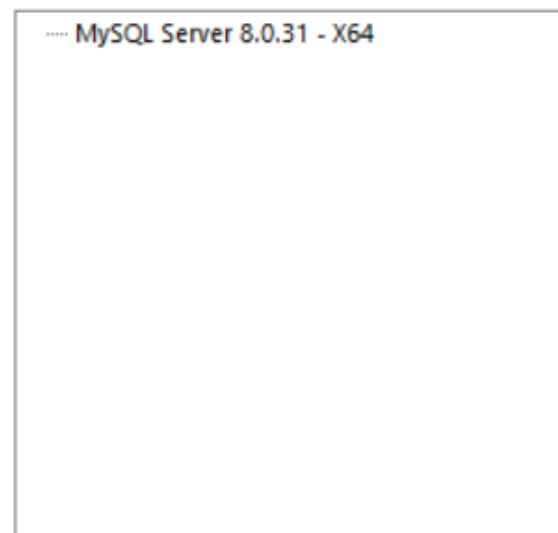
Available Products:



Available Products:

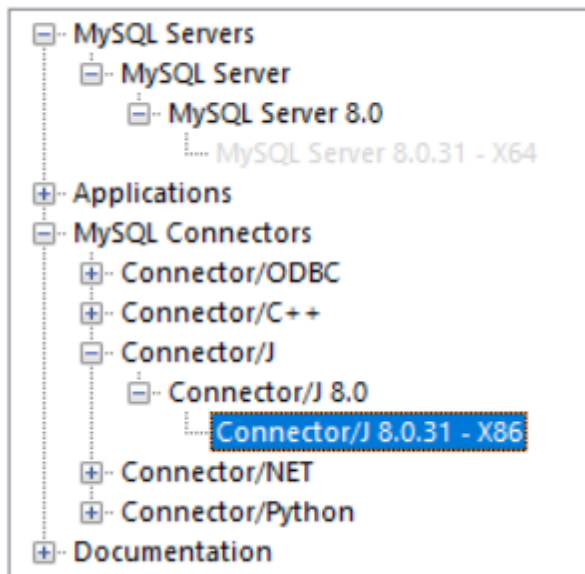


Products to be installed:

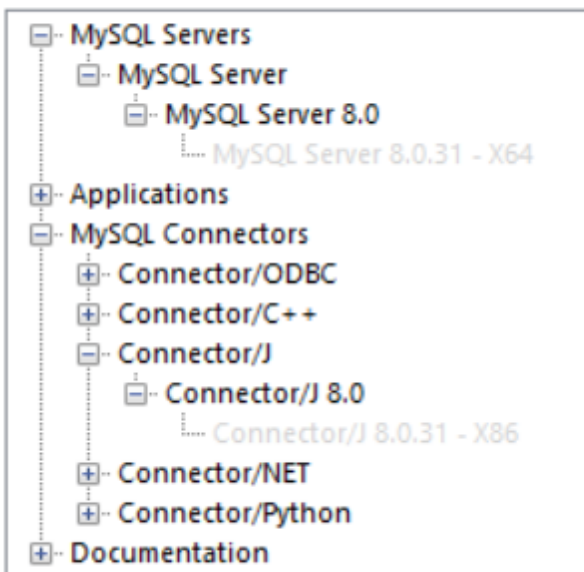




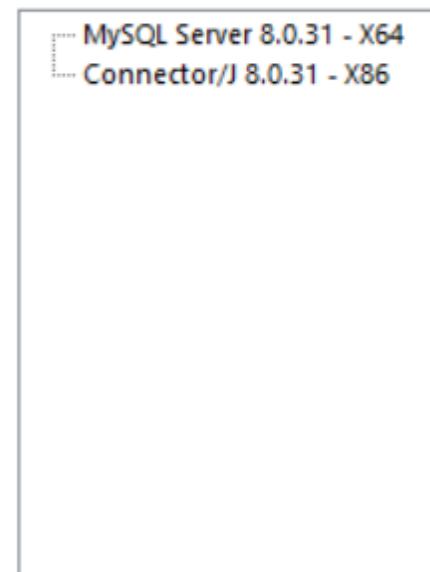
### Available Products:



### Available Products:



### Products To Be Installed:



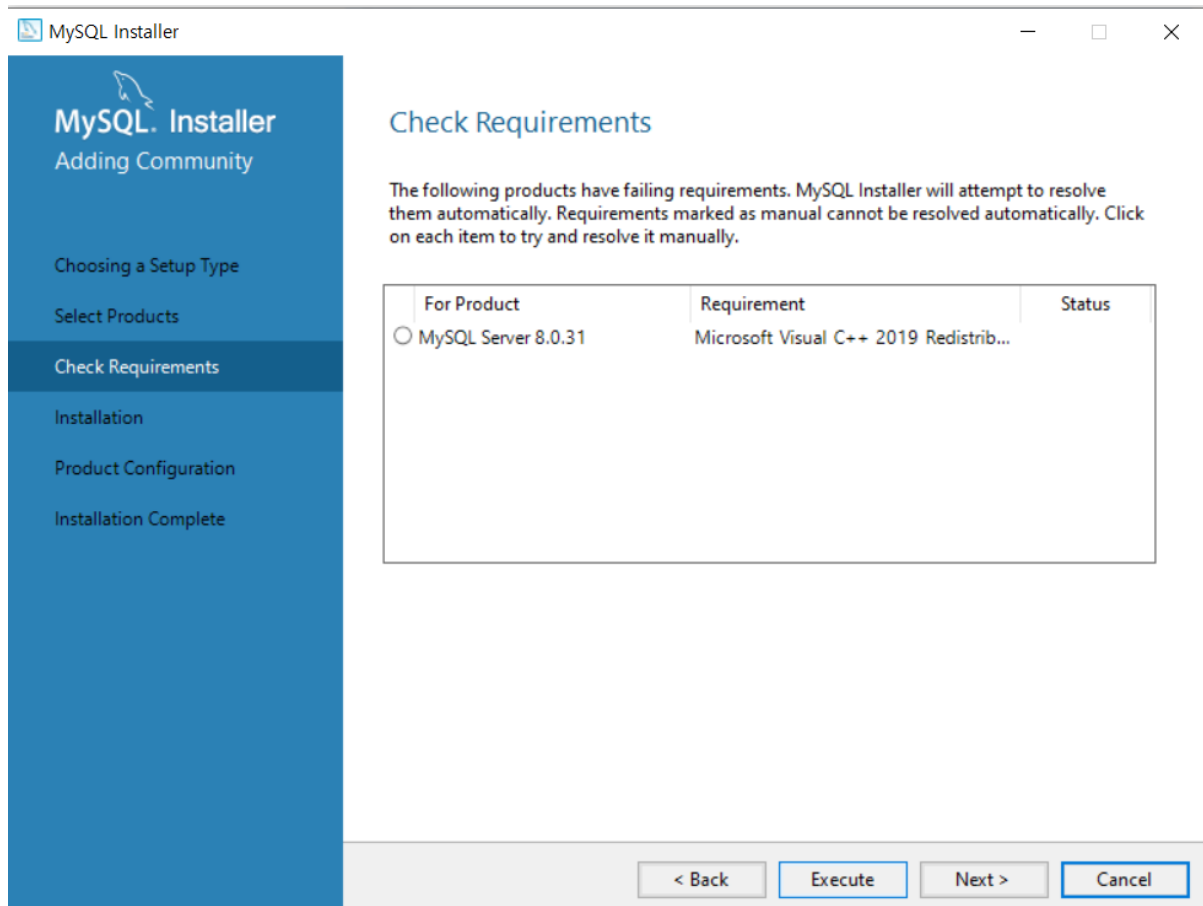
☐ Enable the Select Features pa  
customize product features

Published: 2022년 10월 11일 화요일

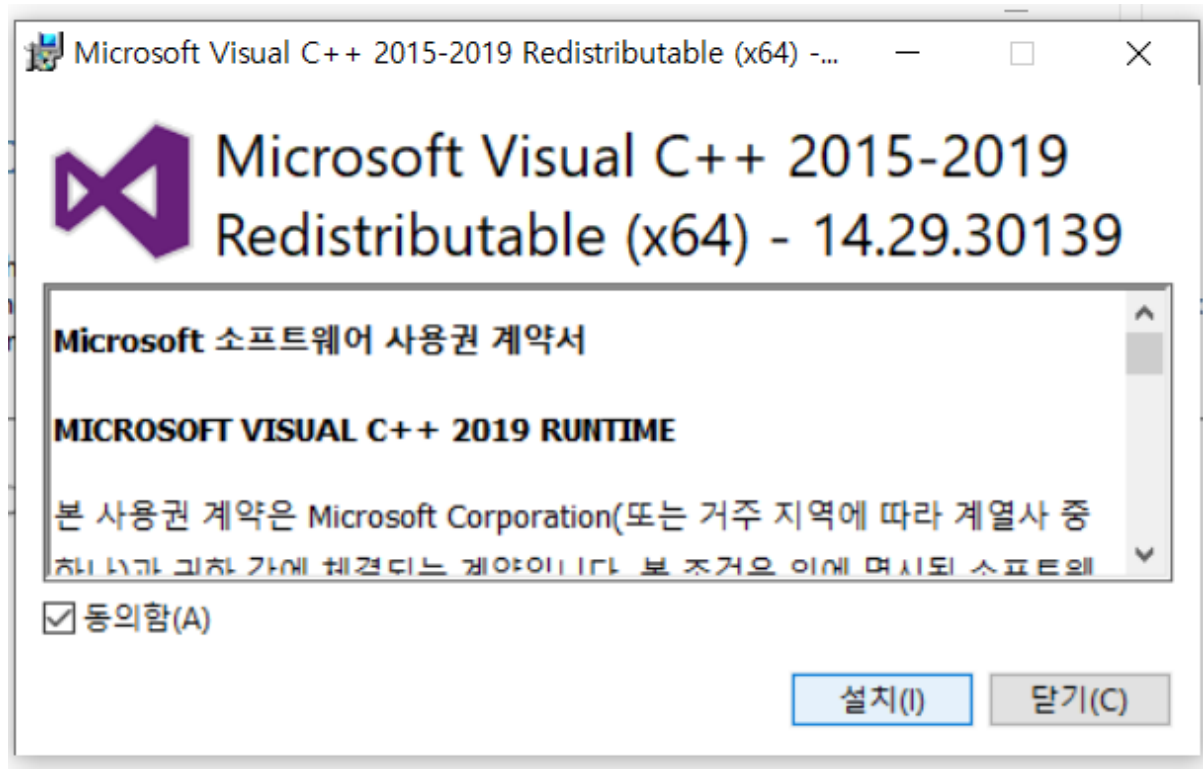
Release Notes: <https://dev.mysql.com/doc/relnotes/mysql-8.0/en/news-8-0-31.html>

< Back

Next >



**Execute Click !!!**



설치 클릭 !!!

## Check Requirements

The following products have failing requirements. MySQL Installer will attempt to resolve them automatically. Requirements marked as manual cannot be resolved automatically. Click on each item to try and resolve it manually.

	For Product	Requirement	Status
✓	MySQL Server 8.0.31	Microsoft Visual C++ 2019 Redistrib...	INSTL DONE

### Requirement Details

MySQL Installer is trying to resolve this requirement automatically.  
There is nothing you need to do.

Requirement:	Microsoft Visual C++ 2019 Redistributable Package (x64) is not installed. Latest binary compatible version will be installed if agreed	^ v
Status:		^ v



< Back

Next >

Cancel

## Installation

The following products will be installed.

Product	Status	Progress	Notes
 MySQL Server 8.0.31	Ready to Install		
 Connector/J 8.0.31	Ready to Install		

Click [Execute] to install the following packages.





< Back

Execute

Cancel

## Installation

The following products will be installed.

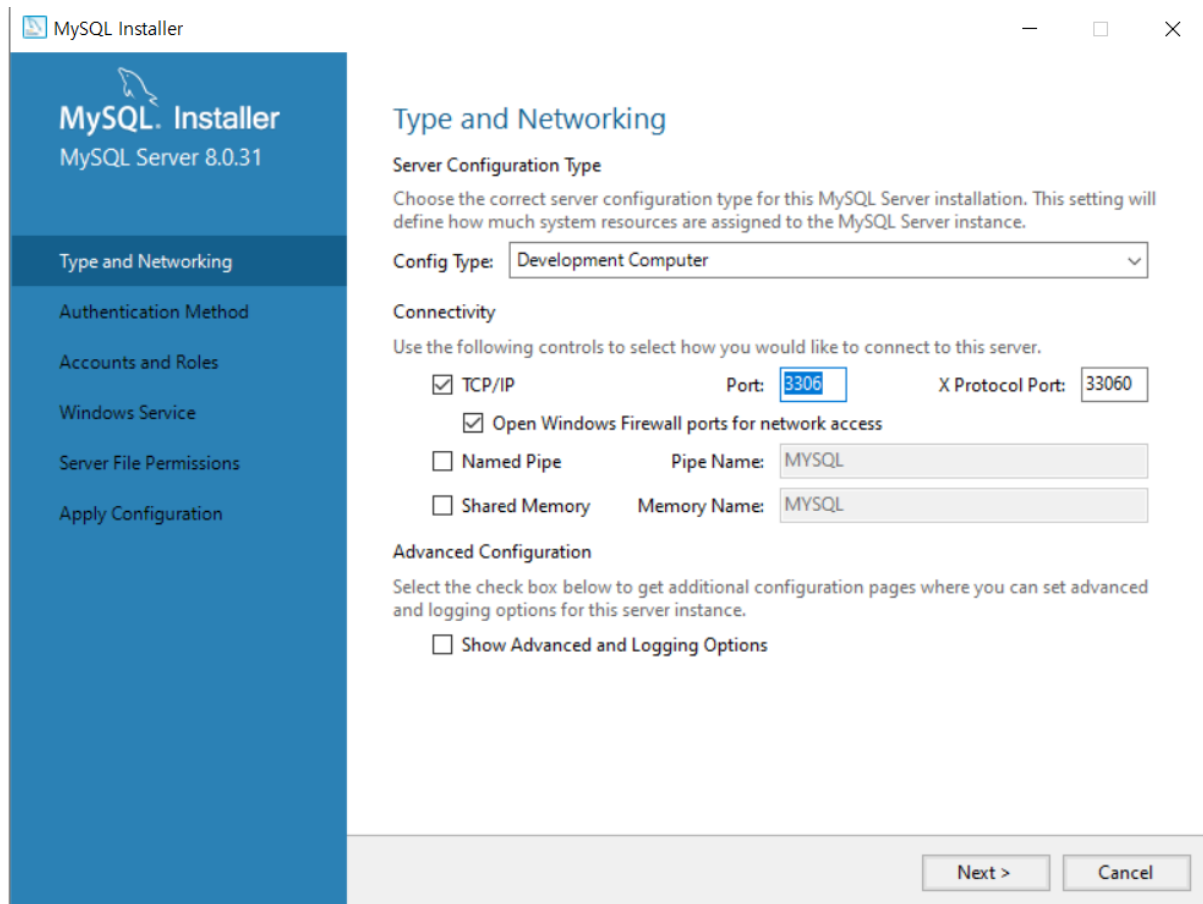
	Product	Status	Progress	Notes
	 MySQL Server 8.0.31	Complete		
	 Connector/J 8.0.31	Complete		

Show Details >

< Back

Next >

Cancel



포트 번호는 반드시 3306을 사용해야만 한다.

MySQL Installer

MySQL Server 8.0.31

Type and Networking

Authentication Method

**Accounts and Roles**

Windows Service

Server File Permissions

Apply Configuration

### Accounts and Roles

**Root Account Password**  
Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

Password strength: **Weak**

**MySQL User Accounts**  
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL User Name	Host	User Role
-----------------	------	-----------

Add User

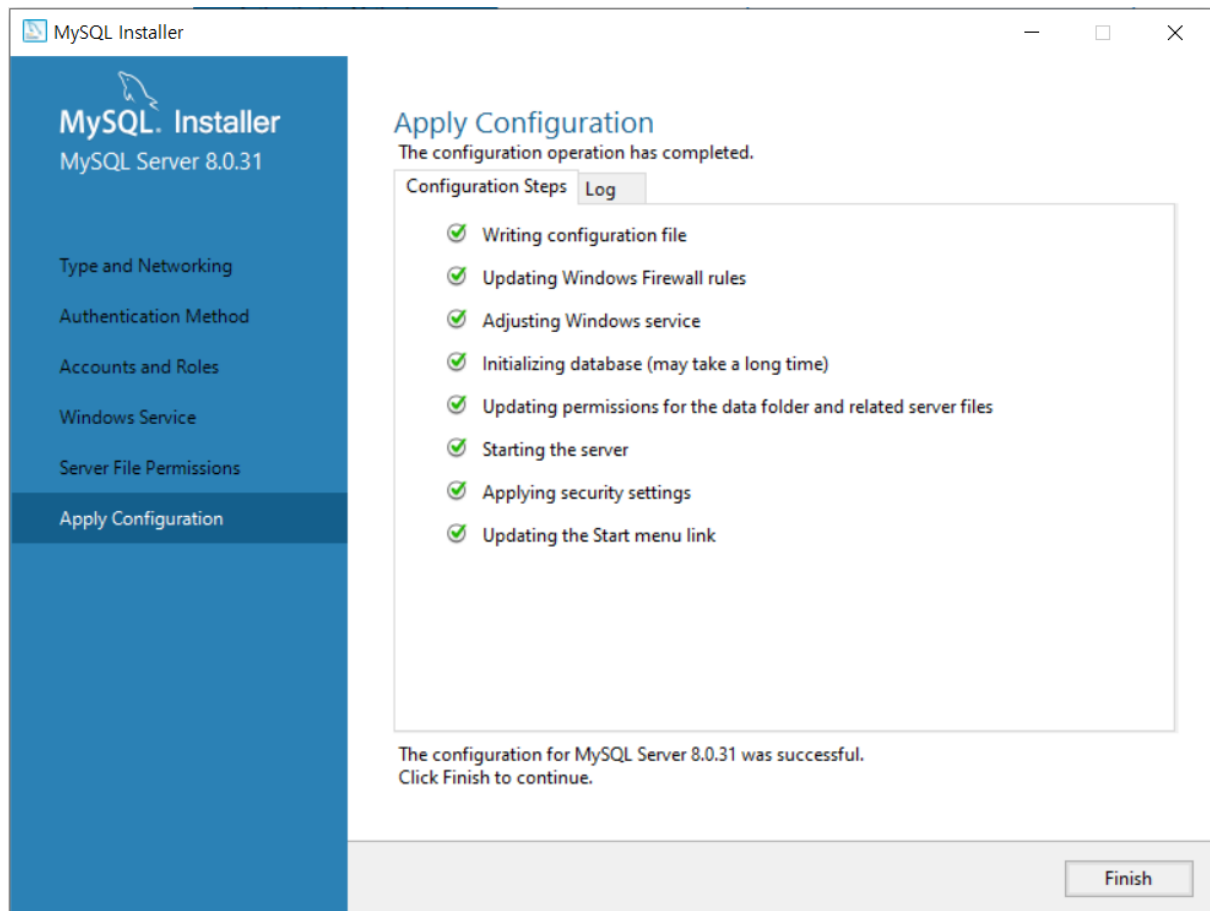
Edit User

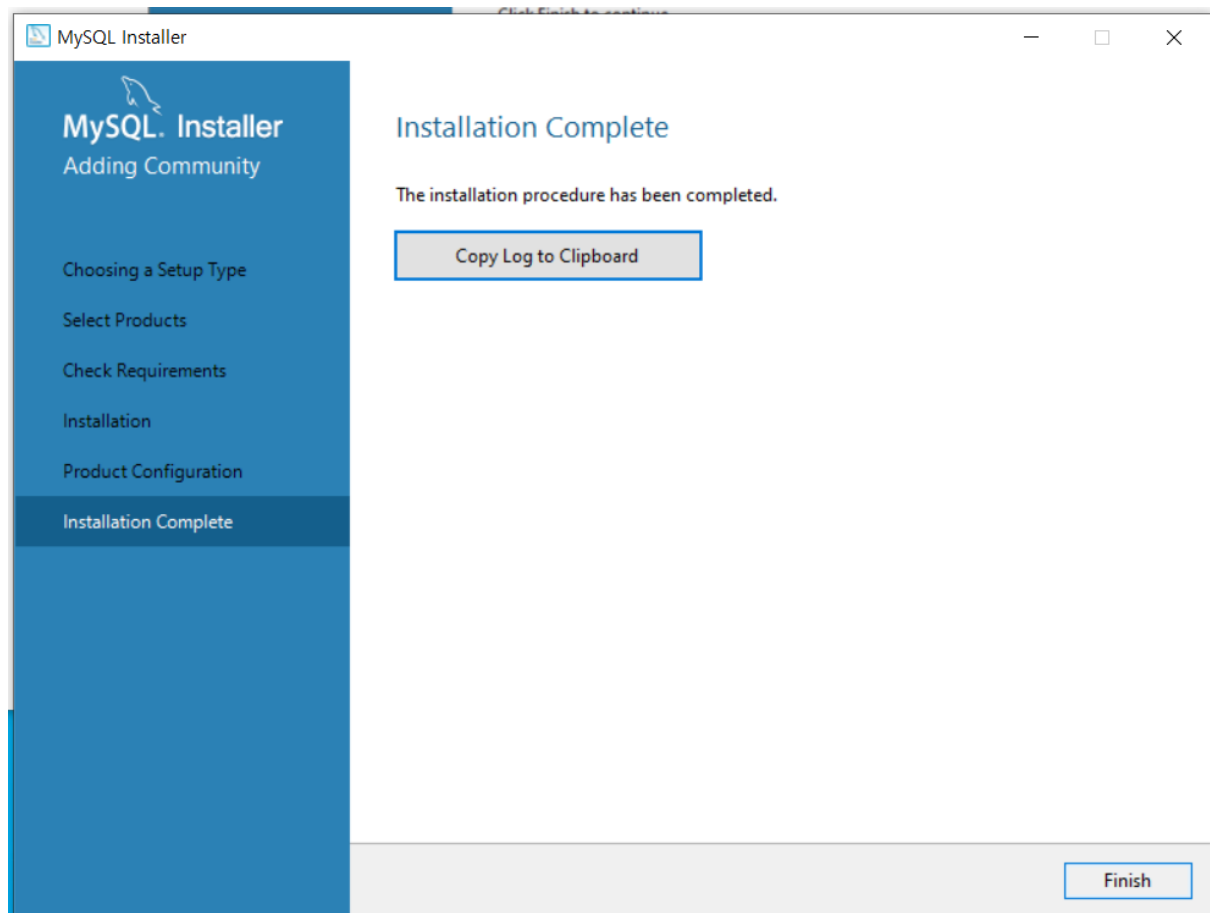
Delete

< Back   Next >   Cancel

암호는 0000으로 통일.







# MySQL 8.0 Command Line Client

Enter password: \*\*\*\*

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 11

Server version: 8.0.31 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;

Database
information_schema
mysql
performance_schema
sys

4 rows in set (0.00 sec)

mysql>

mysql> CREATE DATABASE univ;

Query OK, 1 row affected (0.00 sec)

mysql> SHOW DATABASES;

Database
information_schema
mysql
performance_schema
sys
univ

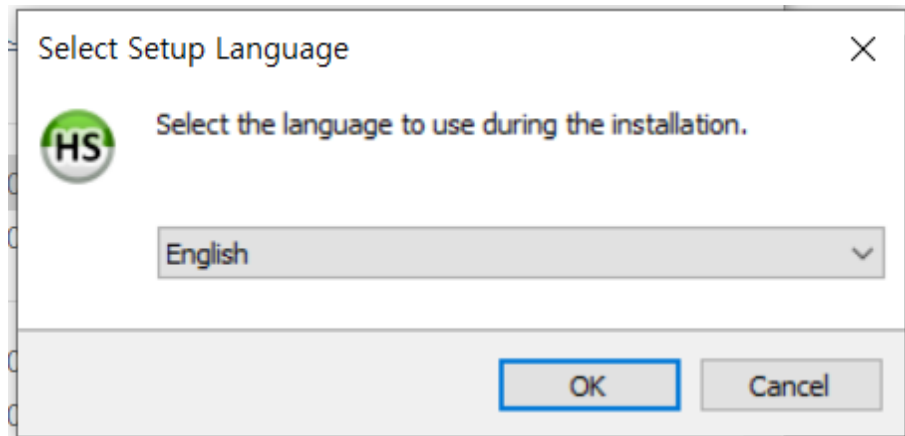
5 rows in set (0.00 sec)

mysql>

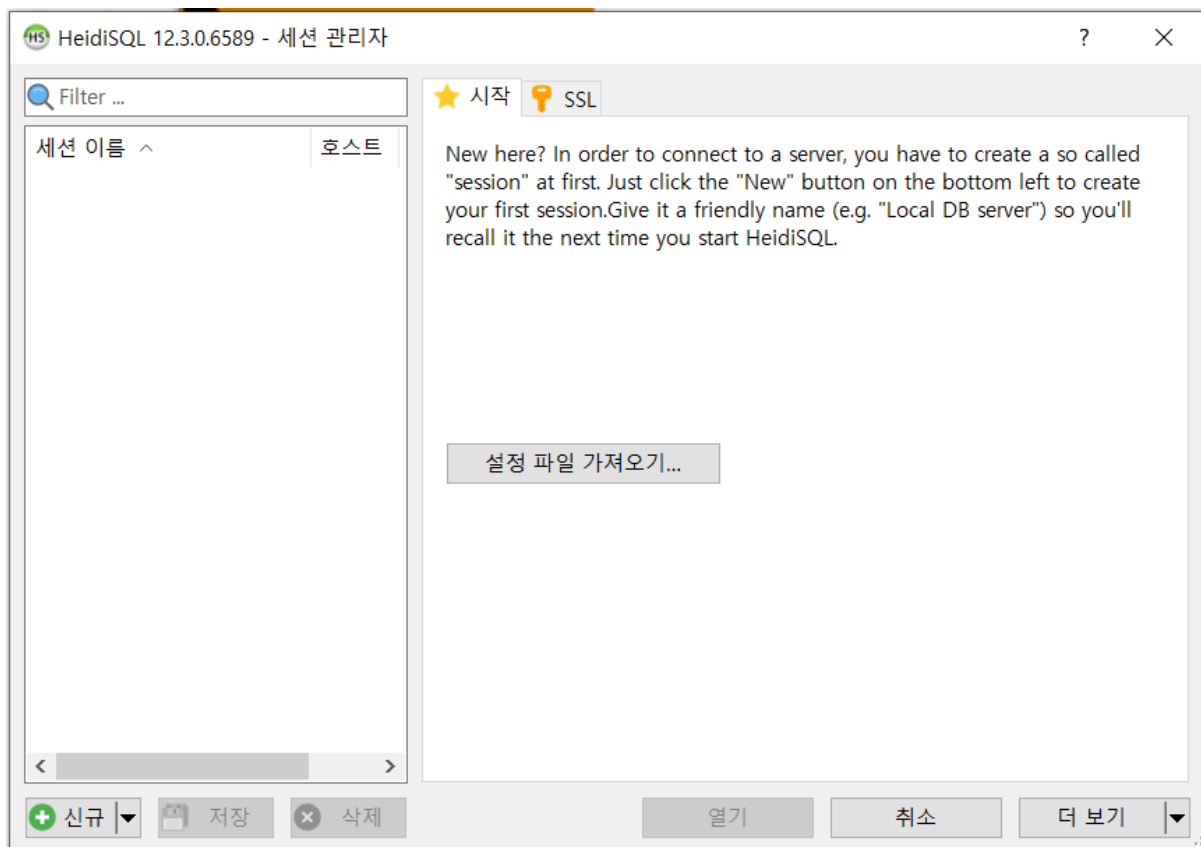
## Hour 2

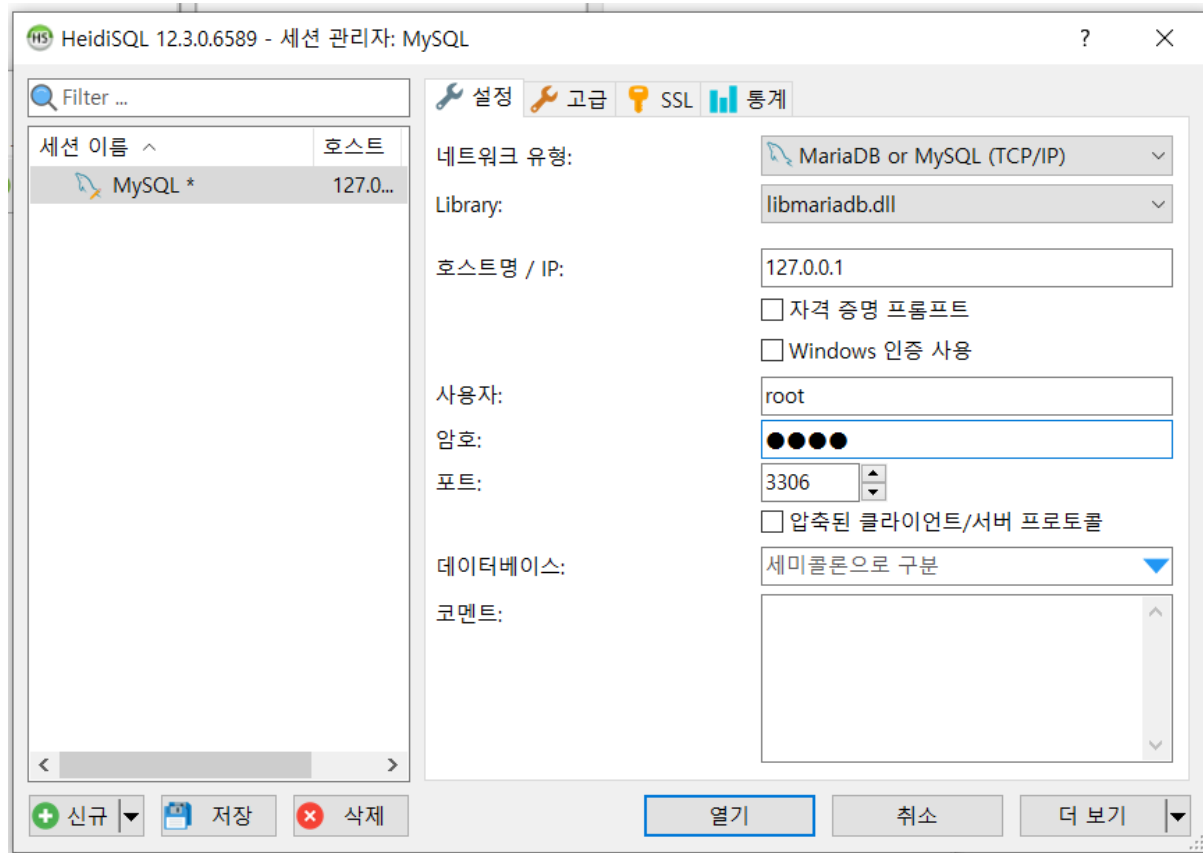
heidiSQL

MySQL 와 MariaDB 를 사용하기 쉽게 도와준다.



설치시 설정이 별로 없음 ,

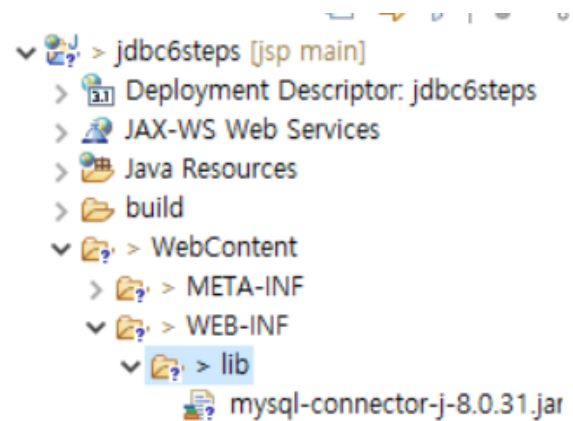
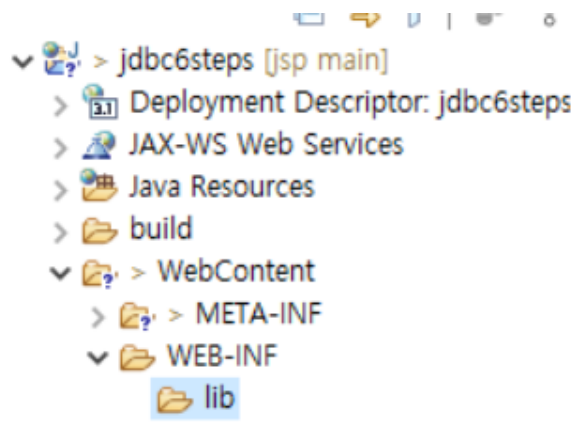




JSP 로 데이터 베이스를 사용하는 4가지 방법

1. 각각의 JSP 파일에서 매번 직접 DB 사용
2. DB 연결 부분만 별도의 File 로 구성
3. DB 연결 부분만 별도의 Class 로 구성
4. **Connection Pool**

<div> <div> <div></div> <div>&gt; 내 PC &gt; 로컬 디스크 (C:) &gt; Program Files (x86) &gt; MySQL &gt; Connector J 8.0</div> <div> <div></div> <div>Connector J 8.0 검색</div> </div> </div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> </div>				
이름	수정된 날짜	유형	크기	
CHANGES	2022. 9. 3. 오후 9:54	파일	271KB	
INFO_BIN	2022. 9. 3. 오후 9:54	파일	1KB	
INFO_SRC	2022. 9. 3. 오후 9:54	파일	1KB	
LICENSE	2022. 9. 3. 오후 9:54	파일	70KB	
mysql-connector-j-8.0.31.jar	2022. 9. 3. 오후 9:54	JAR 파일	2,457KB	
README	2022. 9. 3. 오후 9:54	파일	2KB	



## jdbc basic

6 steps

- Step 1 import SQL Packages

```
<!-- Step 1 import SQL Packages -->
<%@ page import="java.sql.*" %>
```

- Step 2 load JDBC Driver

```

<%
//      2 load JDBC Driver
try {
    Class.forName("com.mysql.jdbc.Driver");
} catch(ClassNotFoundException err) {
    out.print("JDBC Driver loading error<br>" + err.getMessage());
}
// com.mysql.jdbc.Driver    mySQL ☺
// org.maria.jdbc.Driver    maria ☺

%>

```

- Step 3 create Connection Object

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>

<!-- Step 1 import SQL Packages -->
<%@ page import="java.sql.*" %>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<%
// Step 2 load JDBC Driver
try {
    Class.forName("com.mysql.jdbc.Driver");
} catch(ClassNotFoundException err) {
    out.print("JDBC Driver loading error<br>" + err.getMessage());
}

// Step 3 create Connection Object

    Connection conn = null;

    try {
        conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/", "root", "0000");
    } catch(SQLException err) {
        out.print("Connection Object error<br>" + err.getMessage());
    }

%>
</body>
</html>

```



- Step 4 create Statement Object

```
// Step 4 create Statement Object

PreparedStatement pstmt = conn.prepareStatement("CREATE DATABASE test");
```

- Step 5 excute SQL Query

```
// Step 5 excute SQL Query

pstmt.executeUpdate();
```

- Step 6 close Connection ( java 9 부터 생략 가능)

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>

<!-- Step 1 import SQL Packages -->
<%@ page import="java.sql.*" %>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<%
//Step 2 load JDBC Driver
try {
    Class.forName("com.mysql.jdbc.Driver");
}catch(ClassNotFoundException err) {
    out.print("JDBC Driver loading error<br>" + err.getMessage());
}

// Step 3 create Connection Object

Connection conn = null;

try {
    conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/", "root", "0000");
}catch(SQLException err) {
    out.print("Connection Object error<br>" + err.getMessage());
}

// Step 4 create Statement Object
```

```

PreparedStatement pstmt = conn.prepareStatement("CREATE DATABASE test2");

// Step 5 excute SQL Query

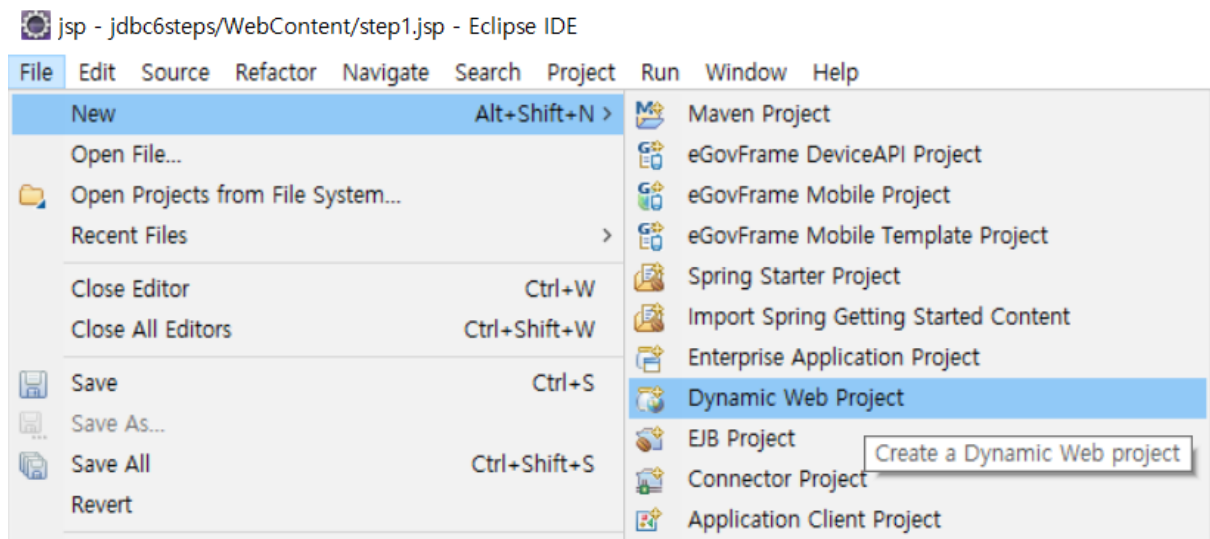
pstmt.executeUpdate();

// Step 6 close Connection

pstmt.close();
conn.close();

%>
</body>
</html>

```



create database

drop database

create table

// Step 4 create Statement Object

```

String sql = "CREATE TABLE student("
+ "hakbun    varchar(10),"
+ "name      varchar(10),"

```

```
+ "dept    varchar(20),"
+ "addr      varchar(30),"
+ "primary key(hakbun))";
```

```
PreparedStatement pstmt = conn.prepareStatement(sql);
```

drop table

## Hour 3

insert table

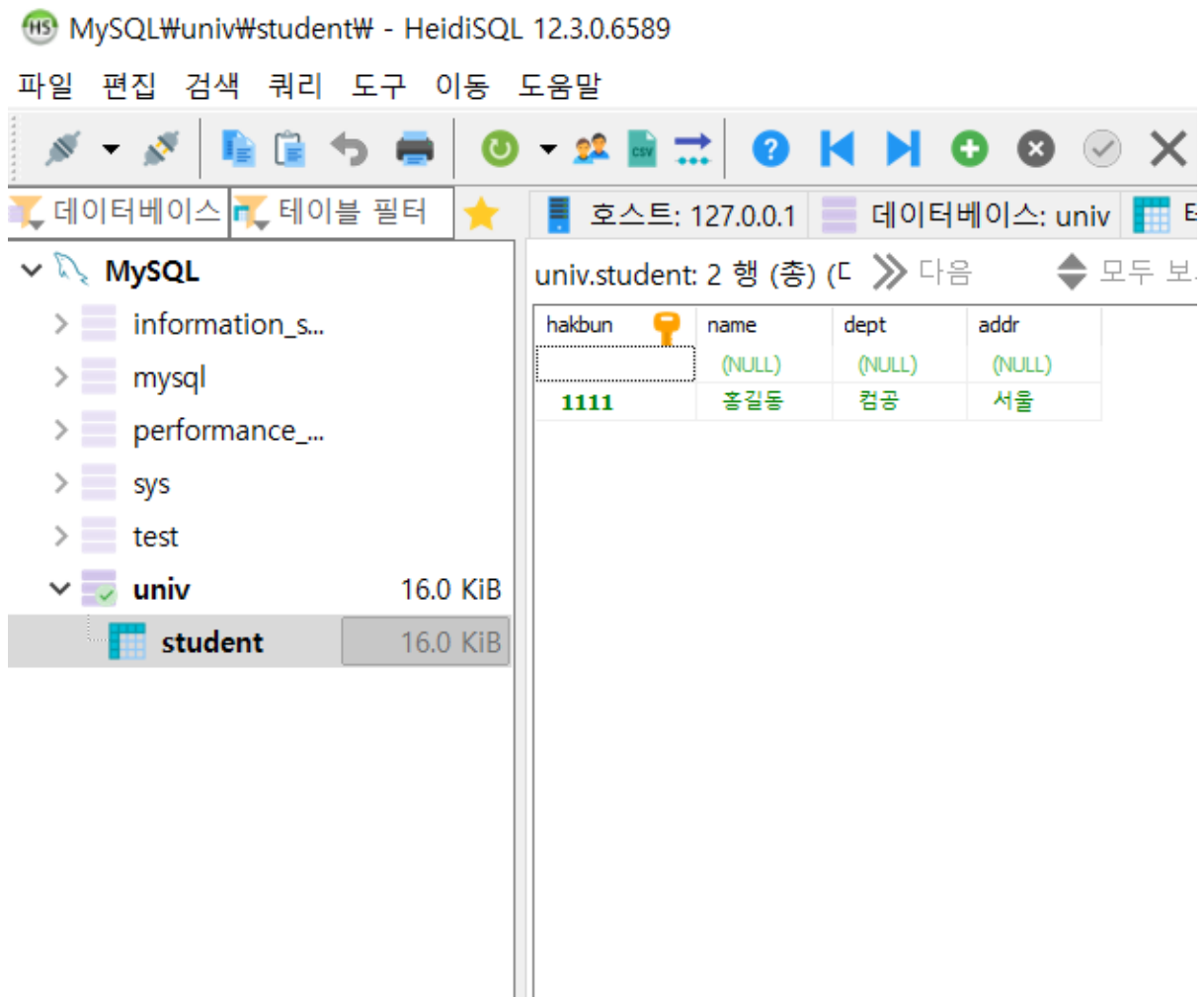
```
// Step 4 create Statement Object

String hakbun = "1111";
String name = "홍길동";
String dept = "컴공";
String addr = "서울";

String sql = "INSERT student VALUES(?, ?, ?, ?)" ;

PreparedStatement pstmt = conn.prepareStatement(sql);
pstmt.setString(1, hakbun);
pstmt.setString(2, name);
pstmt.setString(3, dept);
pstmt.setString(4, addr);
```

## Hour 4



## Lunch

## Hour 5 Take a picture with Signal

TBForm

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
```

```

</head>
<body>
<form action="TBInsert.jsp" method="get">
학번 <input type="text" name="hakbun"> <br><br>
이름 <input type="text" name="name"> <br><br>
전공 <input type="text" name="dept"> <br><br>
주소 <input type="text" name="addr"> <br><br>
<button type="submit">insert</button>
</form>

</body>
</html>

```

```

// Step 4 create Statement Object

/* String hakbun = "1111";
String name = "홍길동";
String dept = "컴공";
String addr = "서울"; */

String hakbun = request.getParameter("hakbun");
String name = request.getParameter("name");
String dept = request.getParameter("dept");
String addr = request.getParameter("addr");

String sql = "INSERT student VALUES(?, ?, ?, ?)" ;

PreparedStatement pstmt = conn.prepareStatement(sql);
pstmt.setString(1, hakbun);
pstmt.setString(2, name);
pstmt.setString(3, dept);
pstmt.setString(4, addr);

```

```

// Step 4 Statement Object

String sql = "SELECT * FROM student";
PreparedStatement pstmt = conn.prepareStatement(sql);

// Step 5 excute SQL Query

/* pstmt.executeUpdate(); */          <---- 리턴되는 결과값이 없는 구문
ResultSet rset = pstmt.executeQuery(); <---- 리턴되는 결과값을 가져오는 구문

```

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>

<!-- Step 1 import SQL Packages -->
<%@ page import="java.sql.*" %>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<%
//Step 2 load JDBC Driver
    try {
        Class.forName("com.mysql.jdbc.Driver");
    }catch(ClassNotFoundException err) {
        out.print("JDBC Driver loading error<br>" + err.getMessage());
    }

// Step 3 create Connection Object

    Connection conn = null;

    try {
        conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/univ","root","0000");
    }catch(SQLException err) {
        out.print("Connection Object error<br>" + err.getMessage());
    }

// Step 4 Statement Object

    String sql = "SELECT * FROM student";
    PreparedStatement pstmt = conn.prepareStatement(sql);

// Step 5 excute SQL Query

    /* pstmt.executeUpdate(); */
    ResultSet rset = pstmt.executeQuery();

    while(rset.next()){
%>

<%=rset.getString("hakbun") %>|
<%=rset.getString("name") %>|
<%=rset.getString("dept") %>|
<%=rset.getString("addr") %><br>

<%

```

```

    }
    //Step 6 Close Connection
    rset.close();
    pstmt.close();
    conn.close();
%>

```

```

</body>
</html>

```

## login in / log out with session

1. 아이디와 비번을 모두 정상적으로 입력한경우 → 로그인 성공
2. 비번이 틀릴 경우 → 로그인 실패 ( 다시 시도하세요)
3. 아이디가 디비에 존재하지 않는 경우 → 로그인 실패 ( 회원 가입 페이지로 리다이렉트)

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>

<!-- Step 1 import SQL Packages -->
<%@ page import="java.sql.*" %>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<%
//Step 2 load JDBC Driver
try {
    Class.forName("com.mysql.jdbc.Driver");
} catch (ClassNotFoundException err) {
    out.print("JDBC Driver loading error<br>" + err.getMessage());
}

// Step 3 create Connection Object

Connection conn = null;

try {
    conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/univ","root","0000");

```

```

} catch (SQLException err) {
    out.print("Connection Object error<br>" + err.getMessage());
}

// Step 4 Statement Object

String hakbun = request.getParameter("hakbun");

String sql = "SELECT * FROM student WHERE hakbun = ?";
PreparedStatement pstmt = conn.prepareStatement(sql);
pstmt.setString(1, hakbun);

// Step 5 excute SQL Query

/* pstmt.executeUpdate(); */
ResultSet rset = pstmt.executeQuery();

//디비에서 학번에 해당하는 자료가 있어서 반환되었는지 확인
if(!rset.isBeforeFirst()) {
    out.print("<script>alert('해당 학번은 존재 하지 않습니다.');"
        + "history.back();"
        + "</script>");
    return;
}

rset.next();
String dbhakbun = rset.getString("hakbun");
String dbname = rset.getString("name");

// dbhakbun hakbun 값이 같으면 세션을 생성하세요
// "hakbun" dbhakbun "name" dbname

/* if(dbHakbun == hakbun) { 문자열 값을 비교할때는 반드시 .equals() */
if(dbhakbun.equals(hakbun)) {

    session.setAttribute("hakbun", dbhakbun);
    session.setAttribute("name", dbname);

    out.print(session.getAttribute("hakbun") + "("
        + session.getAttribute("name")+ ")님 방문을 환영합니다.<br>");
}

//Step 6 Close Connection
rset.close();
pstmt.close();
conn.close();
%>

```



```
</body>
</html>
```

## Hour 6

### module

dbconnect.inc

```
<%
//Step 2 load JDBC Driver
try {
    Class.forName("com.mysql.jdbc.Driver");
}catch(ClassNotFoundException err) {
    out.print("JDBC Driver loading error<br>" + err.getMessage());
}

// Step 3 create Connection Object

Connection conn = null;

try {
    conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/", "root", "0000");
}catch(SQLException err) {
    out.print("Connection Object error<br>" + err.getMessage());
}

%>
```

dbcclose.inc

```
<%
// Step 6 close Connection

pstmt.close();
conn.close();

%>
```

class

```
package jdbc6steps;

import java.sql.*;

public class dbconnclose {

    // DB 연결 공통부분 매서드
    public static Connection getConnection() {
        //Step 2 load JDBC Driver
        try {
            Class.forName("com.mysql.jdbc.Driver");
        } catch (ClassNotFoundException err) {
            System.out.print("JDBC Driver loading error<br>" + err.getMessage());
        }

        // Step 3 create Connection Object

        Connection conn = null;

        try {
            conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/", "root", "0000");
        } catch (SQLException err) {
            System.out.print("Connection Object error<br>" + err.getMessage());
        }

        return conn;
    }

    // DB 해제 공통부분 매서드

    public static void closeConnection(PreparedStatement
    pstmt, Connection conn) {

        // Step 6 close Connection
        try {
            pstmt.close();
            conn.close();
        } catch (SQLException err) {
            System.out.println("Error " + err.getMessage());
        }
    }
}
```

## 적용

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>

<!-- Step 1 import SQL Packages -->
<%@ page import="java.sql.*" %>
<%@ page import="jdbc6steps.*" %>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>

<%
//클래스로 DB 연결
    Connection conn = dbconnclose.getConnection();

// Step 4 create Statement Object

    PreparedStatement pstmt = conn.prepareStatement("CREATE DATABASE test5");

// Step 5 excute SQL Query

    pstmt.executeUpdate();

//클래스로 닫기
    dbconnclose.closeConnection(pstmt, conn);

%>

</body>
</html>
```

## Hour 7

connection pool

## Hour 8 Take a picture with Signal

