

15 데이터베이스 개발 환경 구축

1

데이터베이스의 개요

2

MySQL 개발 환경 구축

3

통합 개발 환경과 데이터베이스 연동

4

MySQL 기본 명령어

5

[웹 쇼핑몰] 상품 관리 테이블 생성 및 상품 등록하기

학습목표

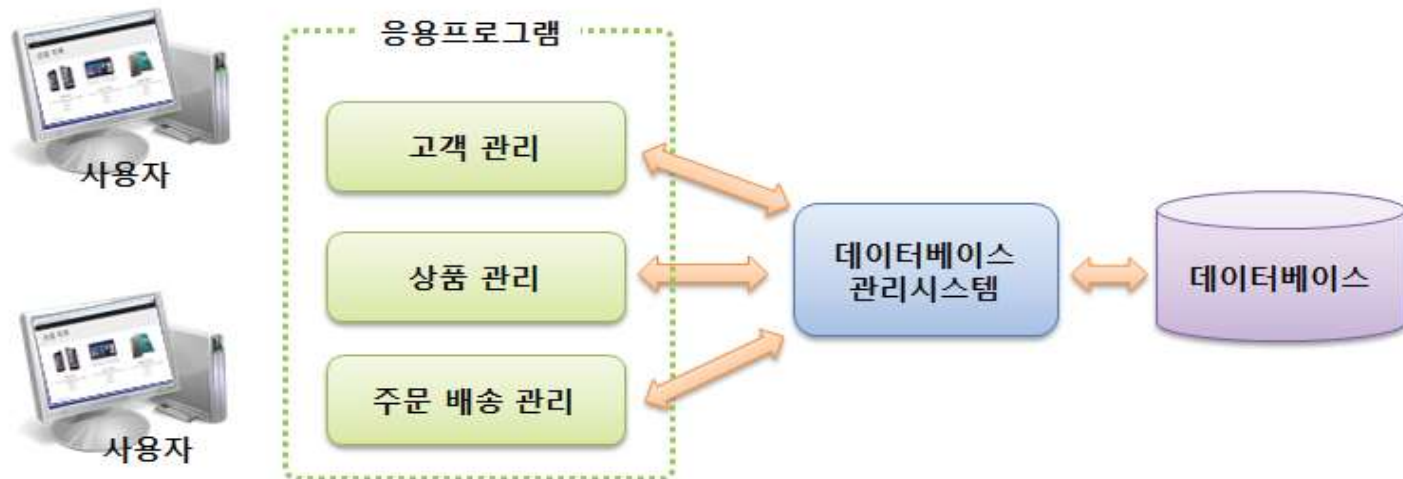
- 데이터베이스의 개념을 이해합니다.
- MySQL 개발 환경을 구축합니다.
- 통합 개발 환경과 데이터베이스 연동 방법을 익힙니다.
- MySQL 기본 명령어 사용법을 익힙니다.
- 웹 쇼핑몰의 상품 관리 테이블을 생성하고 상품을 등록합니다.

1. 데이터베이스의 개요

❖ 데이터베이스

- 여러 자료를 동시에 여러 사람이 공유하여 사용할 수 있도록 체계화한 데이터의 집합
- 몇 개의 데이터 파일을 조직적으로 통합하여 중복을 없애고 구조화한 데이터의 모음이므로 검색과 갱신이 효율적으로 처리

❖ 데이터베이스 시스템의 구조



1. 데이터베이스의 개요

❖ 데이터베이스 특성

- 똑같은 자료를 중복해서 저장하지 않는 통합된 자료
- 컴퓨터가 액세스하여 처리할 수 있는 저장 장치에 수록된 자료
- 어떤 조직의 기능을 수행하는 데 없어서는 안 되는 자료이기 때문에 임시로 모아놓거나 단순한 입출력을 위한 자료가 아님
- 한 조직이 가지고 있는 데이터베이스는 그 조직의 공동 자료로서 사용자는 응용 목적에 따라 각자 다르게 사용할 수 있음

1. 데이터베이스의 개요

❖ 데이터베이스 관리 시스템

- 데이터베이스를 관리하는 소프트웨어
- 다수의 사용자와 데이터베이스 사이에서 사용자의 요구에 따라 정보를 생성하는 역할을 함
- 응용 프로그램들이 데이터베이스를 공유하며 사용할 수 있는 환경을 제공하고, 사용자들이 데이터베이스 안에 데이터를 기록하거나 접근할 수 있게 함
- 사용자나 다른 프로그램의 요구 사항을 관리함으로써, 사용자나 다른 프로그램이 실제로 그 데이터가 저장 매체의 어디에 저장되어 있는지 알지 못해도 다중 사용자 환경의 누구든 데이터를 이용할 수 있게 함
- 사용자의 요구 사항을 처리할 때 데이터의 무결성과 허가된 사용자만 데이터에 접근할 수 있는 보안성을 보장함

1. 데이터베이스의 개요

❖ 데이터베이스 관리 시스템

■ 데이터베이스 관리 시스템의 장단점

장점	단점
<ul style="list-style-type: none">• 데이터의 중복 최소화• 데이터 공유• 일관성, 무결성, 보안성 유지• 최신 데이터 유지• 데이터의 표준화 가능• 데이터의 논리적·물리적 독립성• 용이한 데이터 접근• 데이터 저장 공간 절약	<ul style="list-style-type: none">• 데이터베이스 전문가가 필요함• 비용 부담이 큼• 데이터 백업과 복구가 어려움• 시스템이 복잡함• 대용량 디스크로 액세스가 집중되면 과부하가 발생함

1. 데이터베이스의 개요

❖ MySQL의 개요

- 전 세계적으로 가장 널리 사용되고 있는 오픈 소스 관계형 데이터베이스 관리 시스템
- 관계형 데이터베이스 관리 시스템의 표준화된 사용자 및 프로그래밍 인터페이스인 질의언어 SQL(Structured Query Language)을 사용
- 매우 빠르고 유연하고 사용하기 쉽기 때문에 많은 기업에서 다양한 웹 기반 애플리케이션을 개발하는 데 사용

1. 데이터베이스의 개요

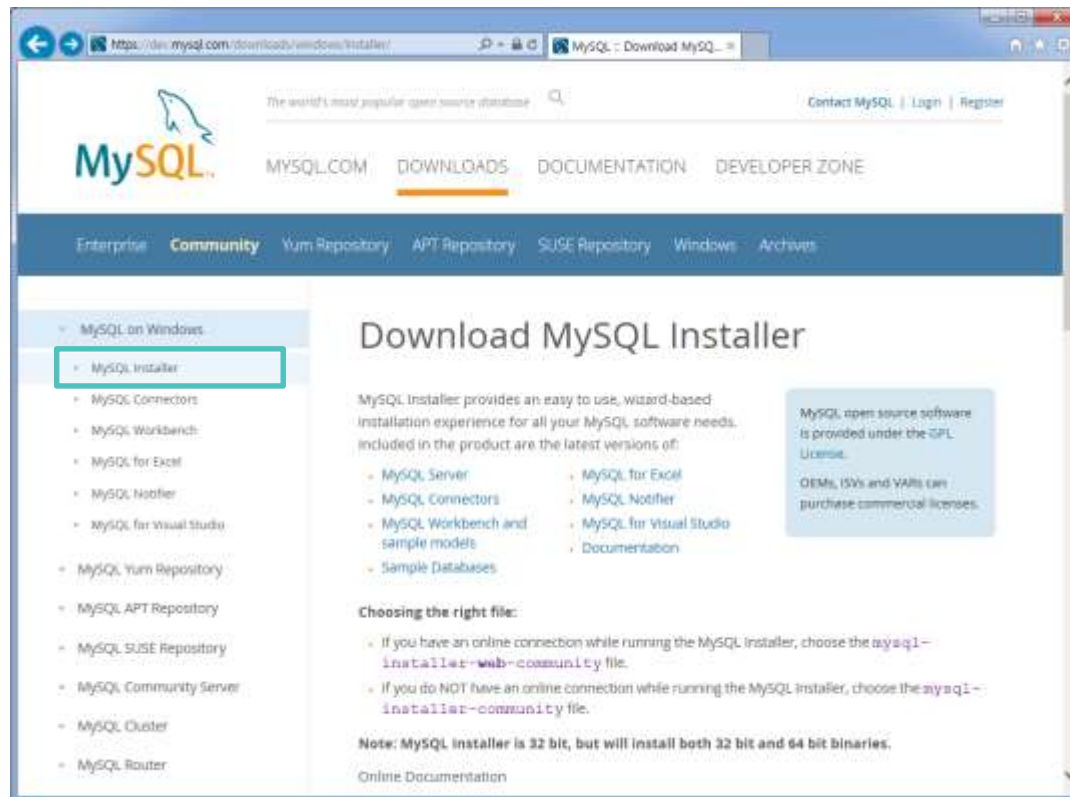
❖ MySQL의 개요

- 오픈 소스 라이선스에 따라 배포되므로 무료로 사용할 수 있음
- 그 자체로 매우 강력한 프로그램으로서 가장 비싸고 강력한 데이터베이스 패키지의 기능 중 상당 부분을 처리함
- 잘 알려진 SQL 데이터 언어의 표준 형식을 사용
- PHP, 펄(Perl), C, C++, 자바 등 많은 언어로 운영되고 있음
- 매우 빠르게 작동하며 대용량 데이터 세트로도 잘 작동함
- 웹 개발에서 가장 높이 평가되는 PHP에 매우 친숙함
- 테이블에서 최대 5,000만 행 이상의 대형 데이터베이스를 지원함
- 테이블의 기본 파일크기 제한은 4GB이지만 운영체제에서 처리하는 경우 이론적인 제한인 800만 TB까지 늘릴 수 있음
- 사용자 정의가 가능함
- 오픈소스 GPL 라이선스를 통해 프로그래머는 자신의 특정 환경에 맞게 MySQL 소프트웨어를 수정할 수 있음

2. MySQL 개발 환경 구축

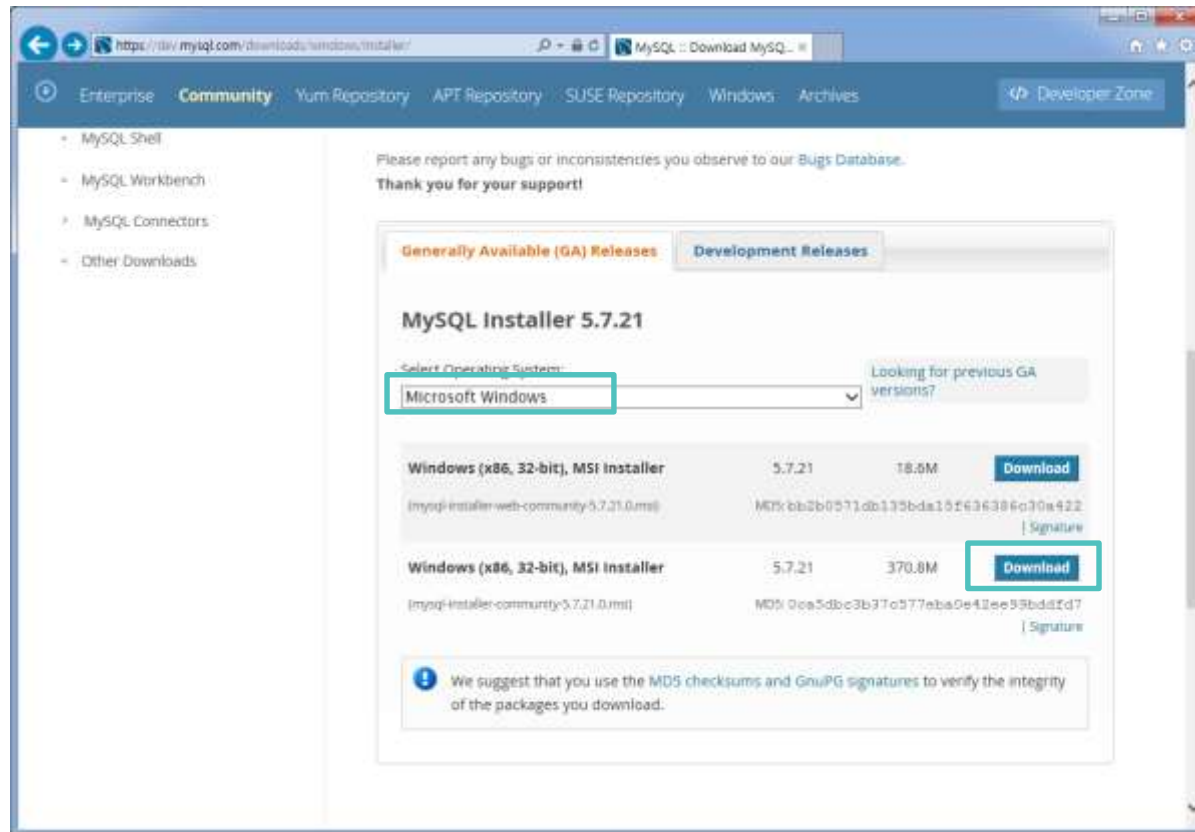
예제 15-1 MySQL 다운로드하기

- MySQL 다운로드 사이트에 접속하기:
 - MySQL 다운로드 사이트(<http://dev.mysql.com/downloads/windows/installer/>)에 접속
 - [MySQL on Windows]-[MySQL Installer] 선택



2. MySQL 개발 환경 구축

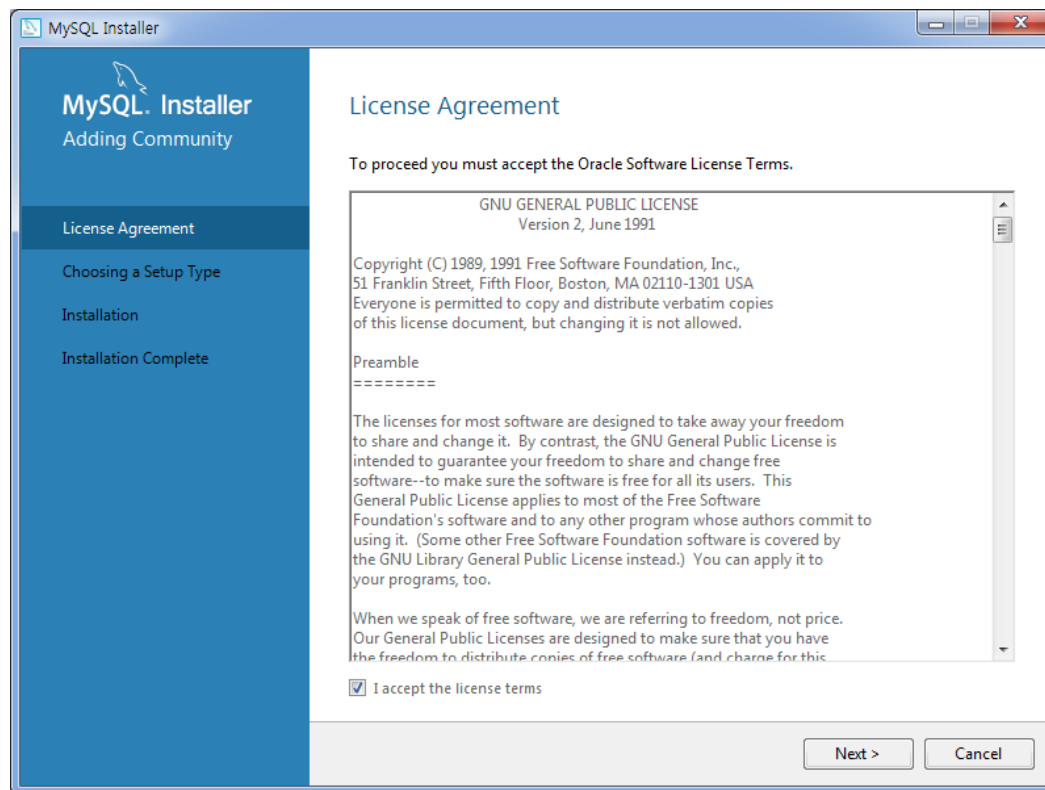
- MySQL 다운로드하기:
 - MySQL Installer 5.7.21의 'Microsoft Windows' 선택한 후 <Download> 클릭



2. MySQL 개발 환경 구축

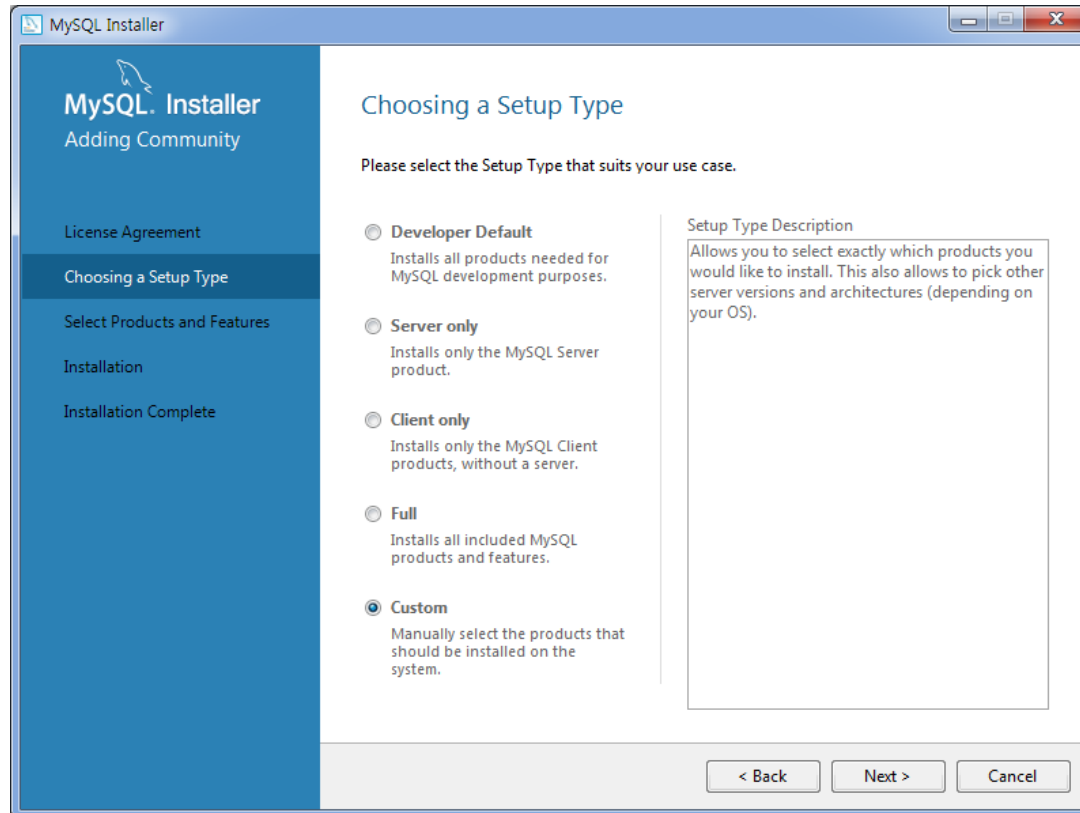
예제 15-2 MySQL 설치하기

- 라이선스에 동의하기:
 - mysql-installer-community-5.7.21.0.msi 설치 패키지 파일을 더블클릭하여 설치 진행
 - 라이선스 규약을 확인하고 <Next> 클릭



2. MySQL 개발 환경 구축

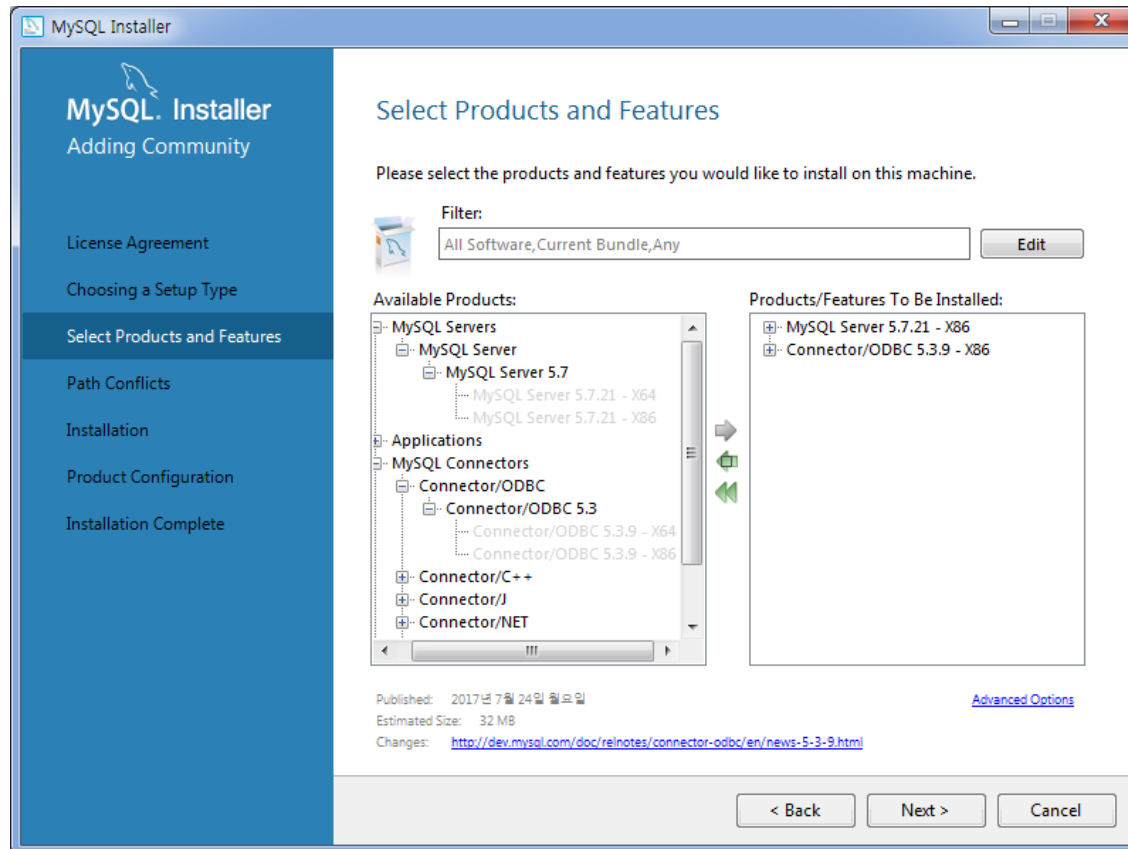
- 설치 유형 선택하기:
 - 사용 목적에 맞게 Setup Type 선택
 - 여기서는 'Custom'을 선택하고 <Next> 클릭



2. MySQL 개발 환경 구축

■ 설치 대상 선택하기:

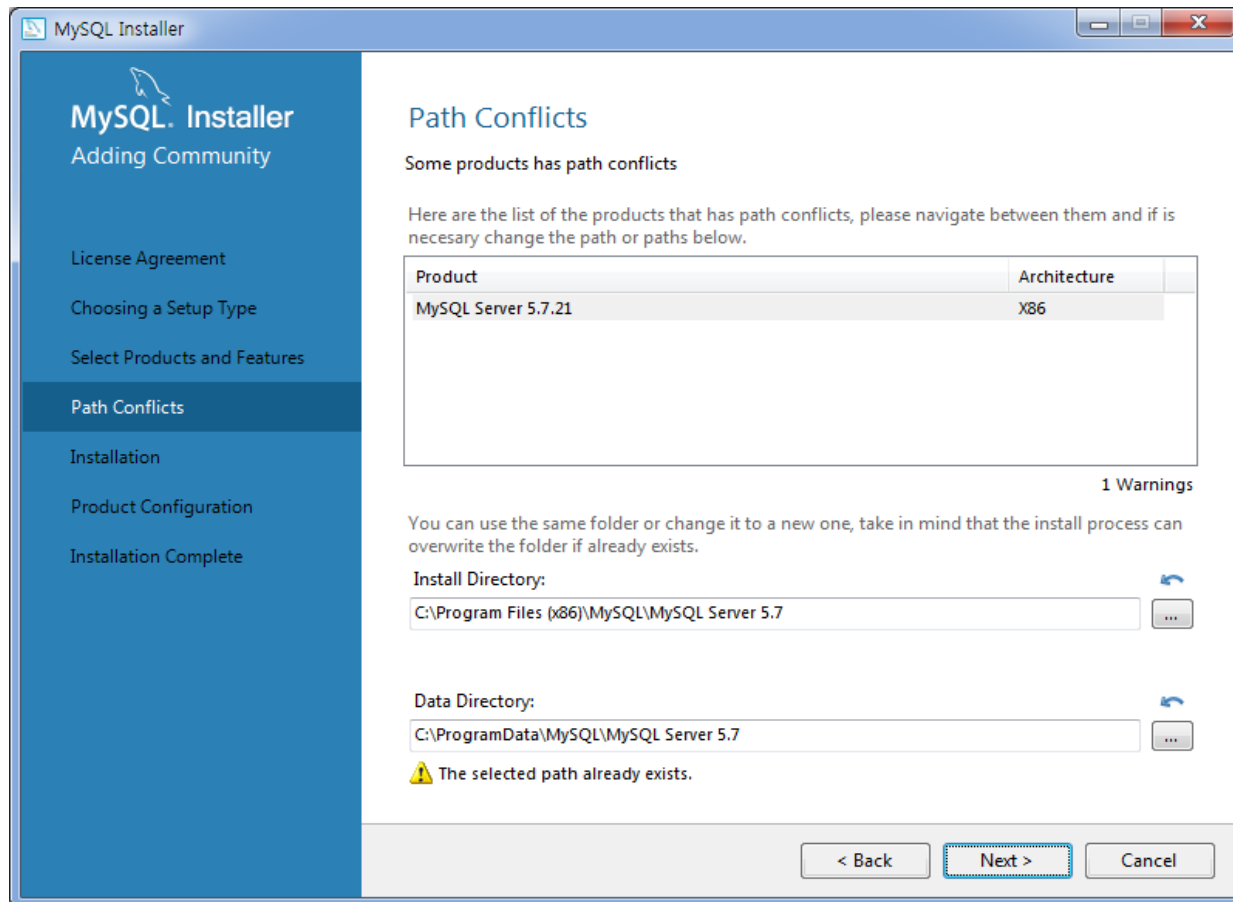
- 설치 대상을 선택하는 화면의 왼쪽 목록에서 MySQL Server의 아래쪽에 있는 'MySQL Server 5.7.21-X64'와 MySQL Connectors의 아래쪽에 있는 'Connection/ODBC5.3.9-X64'를 선택한 후 오른쪽 방향 초록색 화살표를 클릭하여 오른쪽의 목록으로 이동하고 <Next> 클릭



2. MySQL 개발 환경 구축

■ 설치 경로 설정하기:

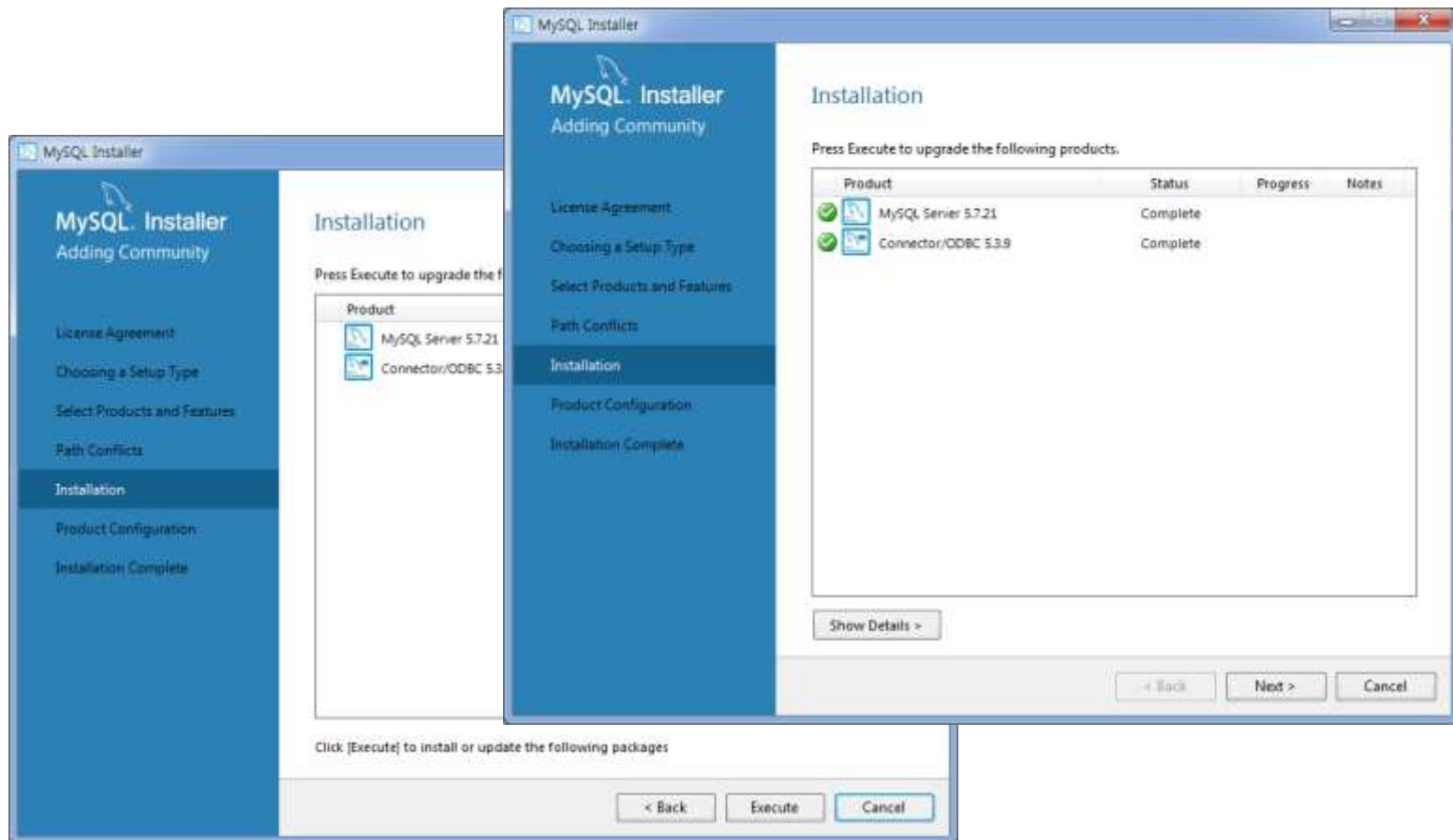
- MySQL을 설치할 디렉터리와 데이터를 설치할 디렉터리를 설정하는 화면에서 디폴트 디렉터리를 변경하지 않아도 상관없음
- 설치 경로를 설정한 후 <Next> 클릭



2. MySQL 개발 환경 구축

■ 설치 대상 설치하기:

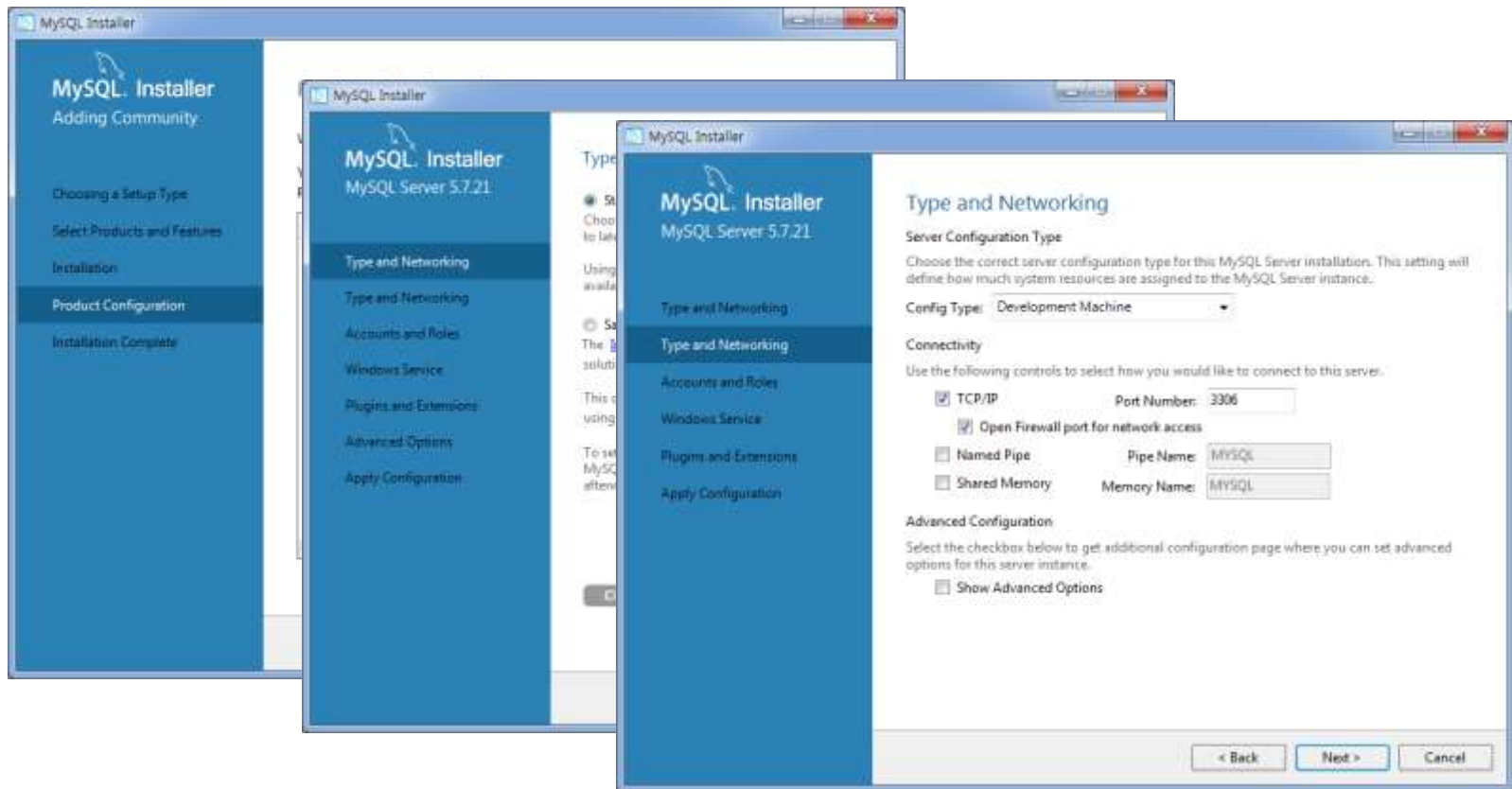
- 설치 확인 화면에서 <Execute>를 클릭하여 설치 시작
- 설치가 완료되어 설치 목록에 초록색 체크가 표시되면 <Next> 클릭



2. MySQL 개발 환경 구축

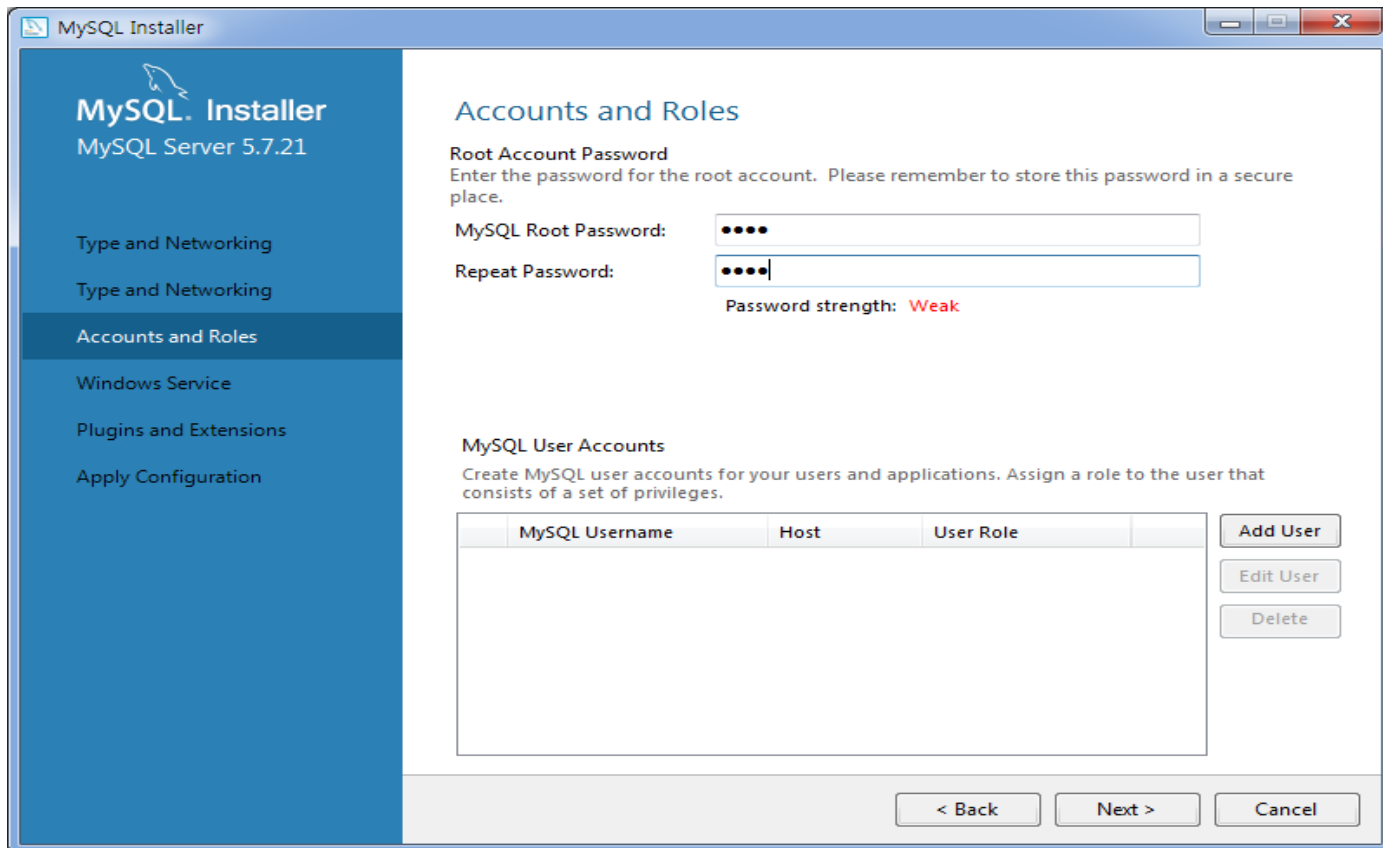
예제 15-3 MySQL 초기 설정하기

- 초기 설정 구성하기:
 - root 계정의 비밀번호 설정 화면이 나올 때까지 모든 기본 설정을 그대로 두고 <Next> 클릭



2. MySQL 개발 환경 구축

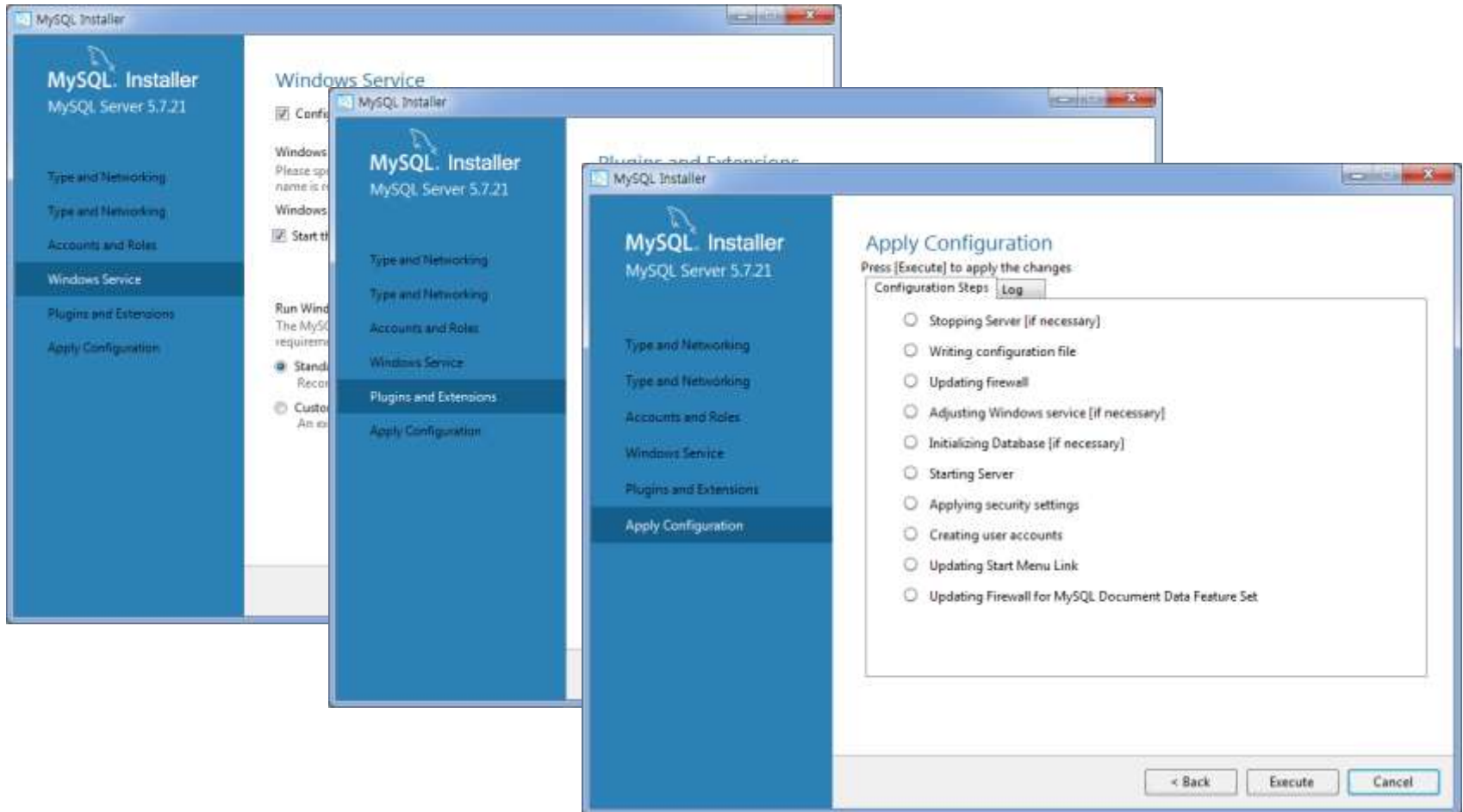
- root 계정의 비밀번호 설정하기:
 - root 계정의 비밀번호 설정과 계정 추가 화면에서 관리자 계정인 root 계정의 비밀번호를 설정하고('1234'를 입력) 계정의 추가는 생략
 - 설정한 후 <Next> 클릭



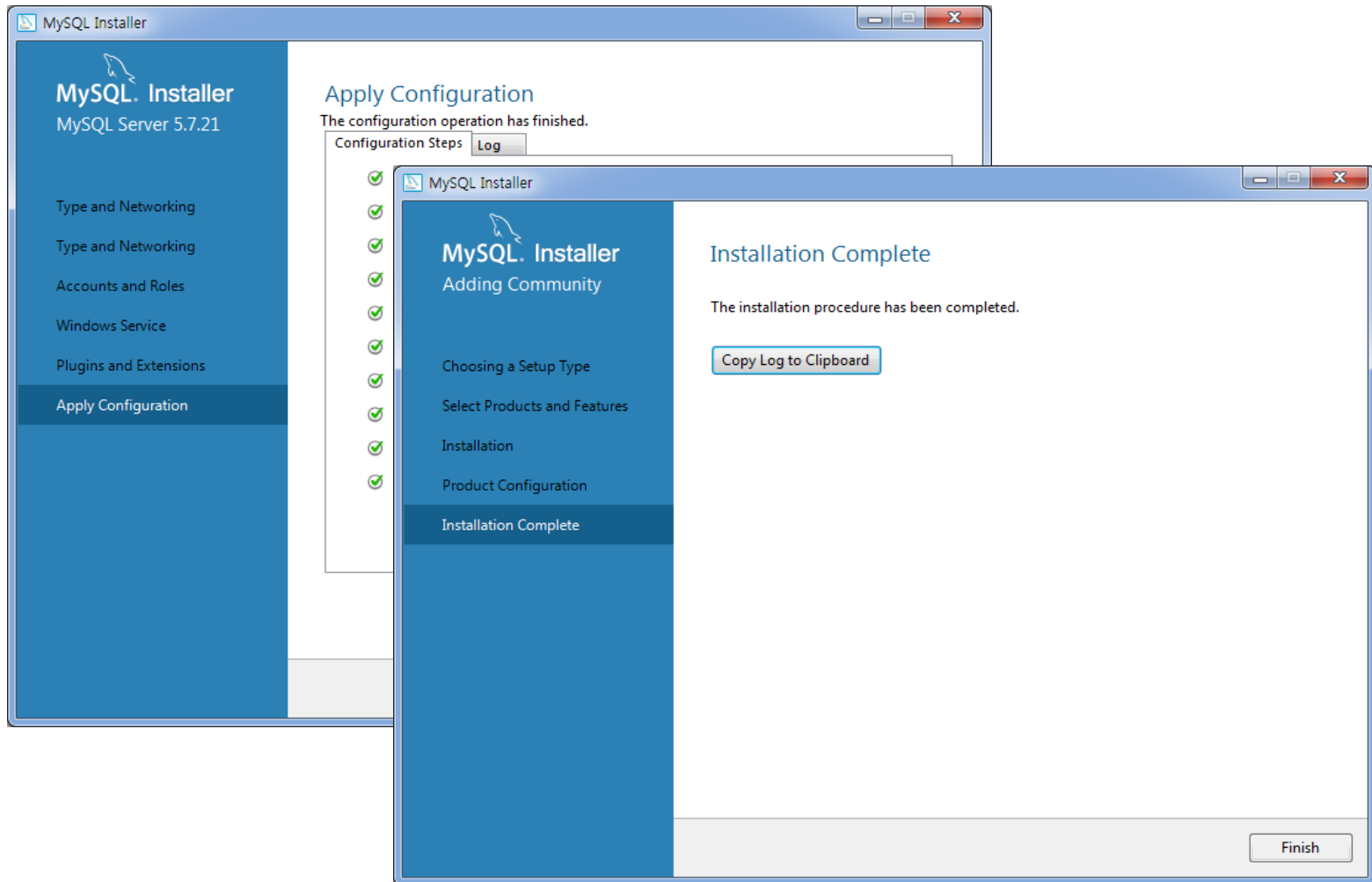
2. MySQL 개발 환경 구축

■ 설치 진행 및 완료하기:

- Windows Service 설정 화면부터는 각 설치 단계마다 모든 기본 설정을 그대로 두고 <Next>나 <Execute> 클릭
- 설치 진행 작업의 완료로 <Finish> 버튼이 나오면 클릭



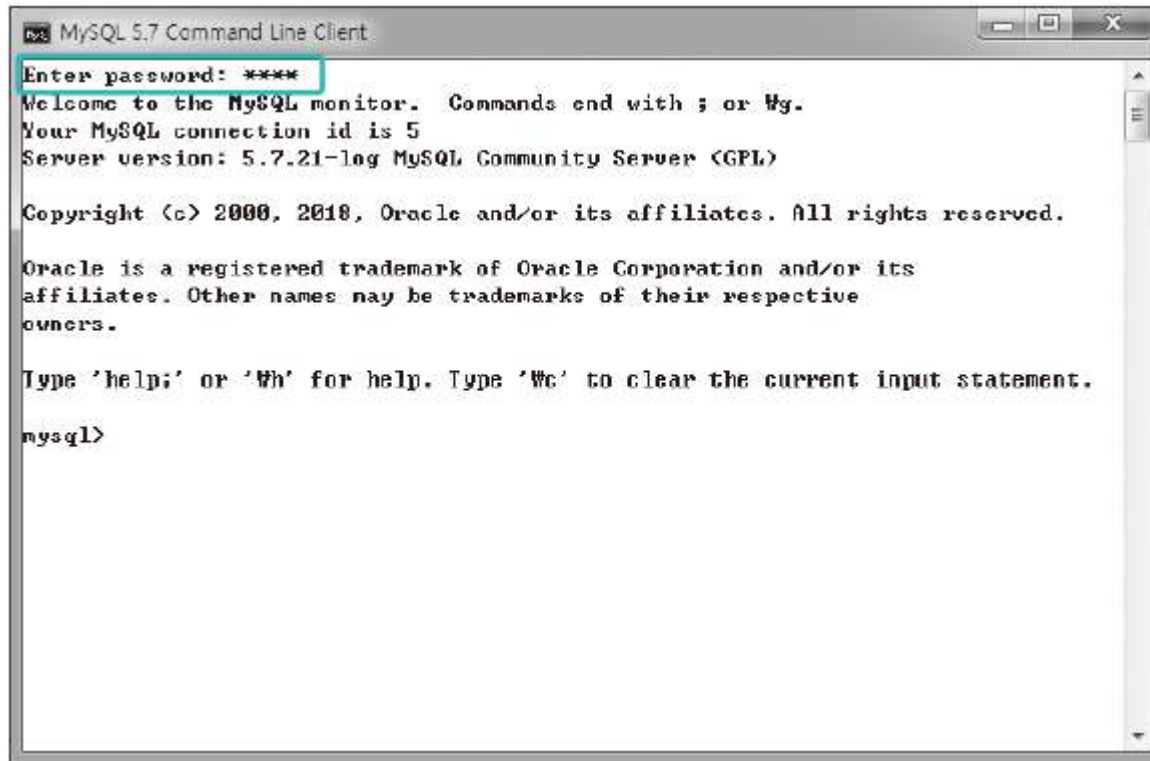
2. MySQL 개발 환경 구축



2. MySQL 개발 환경 구축

예제 15-4 MySQL 설치 확인하기

- 윈도우의 <시작> 버튼을 클릭하여 MySQL 5.7 Command Line Client를 실행한 후 root 계정의 비밀번호를 입력하여 MySQL이 제대로 설치되었는지 확인



```
MySQL 5.7 Command Line Client
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

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

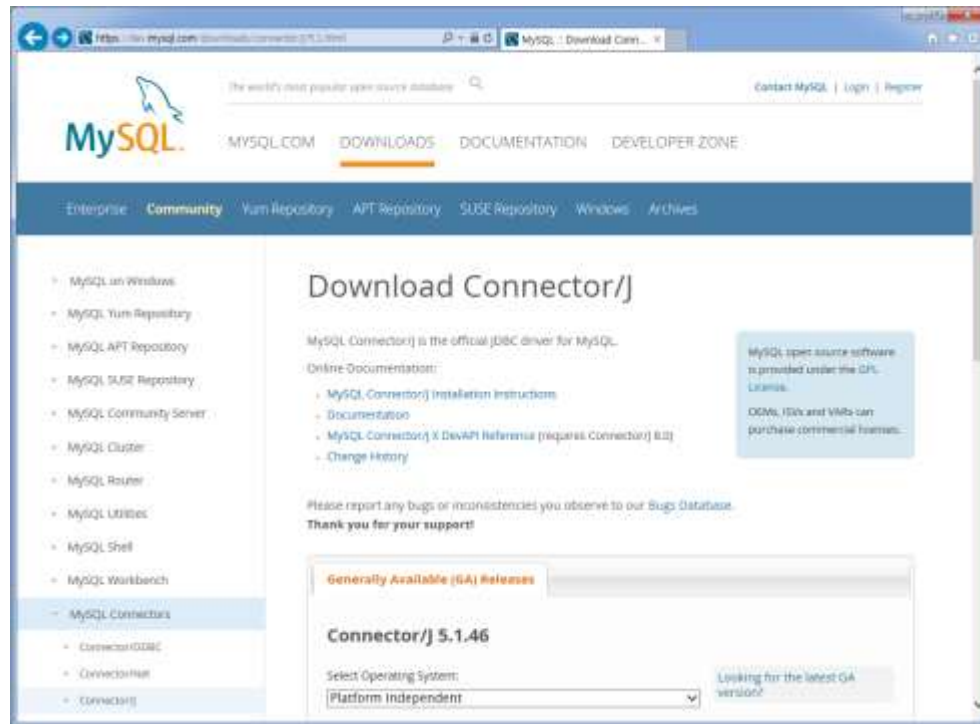
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

2. MySQL 개발 환경 구축

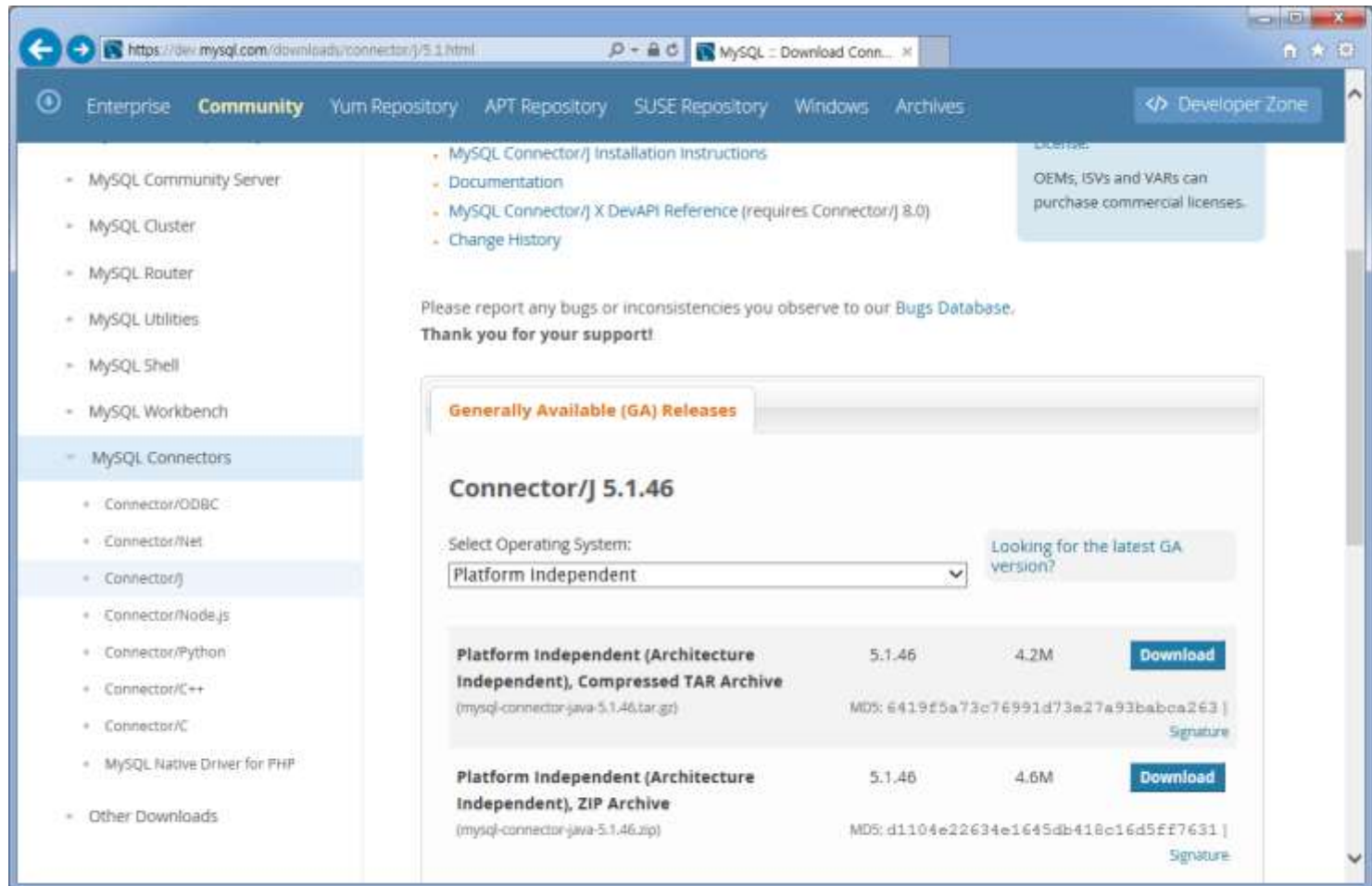
예제 15-5 MySQL 드라이버 다운로드하기

- MySQL 드라이버 다운로드 사이트에 접속하기:
 - 웹 브라우저에서 MySQL 드라이버 다운로드 사이트 (<https://dev.mysql.com/downloads/connector/j/5.1.html>)에 접속하여 [MySQLConnectors]-[Connector/J] 선택



2. MySQL 개발 환경 구축

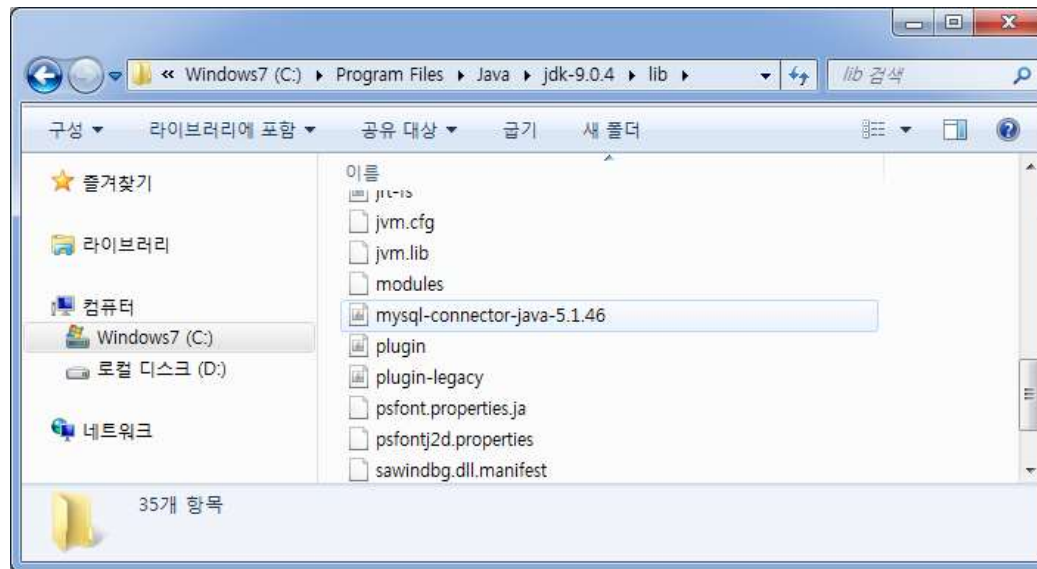
- MySQL 드라이버 다운로드하기:
 - [Platform Independent (Architecture Independent), ZIP Archive]의 <Download>를 클릭하여 설치 파일 다운로드



2. MySQL 개발 환경 구축

예제 15-6 MySQL 드라이버 설치하기

- 다운로드한 커넥트 드라이버 mysql-connector-java-5.1.46 파일의 압축 풀기
- 압축을 푼 폴더에 mysql-connector-java-5.1.46.jar 파일을 복사하고 자바 설치 드라이버 JDK버전\lib(C:\Program Files\Java\jdk-9.0.4\lib) 폴더에 등록



3. 통합 개발 환경과 데이터베이스 연동

❖ 데이터베이스 추가

- 이클립스에서 데이터베이스를 연동하려면 데이터베이스 커넥션을 설정해야 함
- 데이터베이스 커넥션을 설정하기 전에 사용할 데이터베이스가 반드시 생성되어 있어야 함

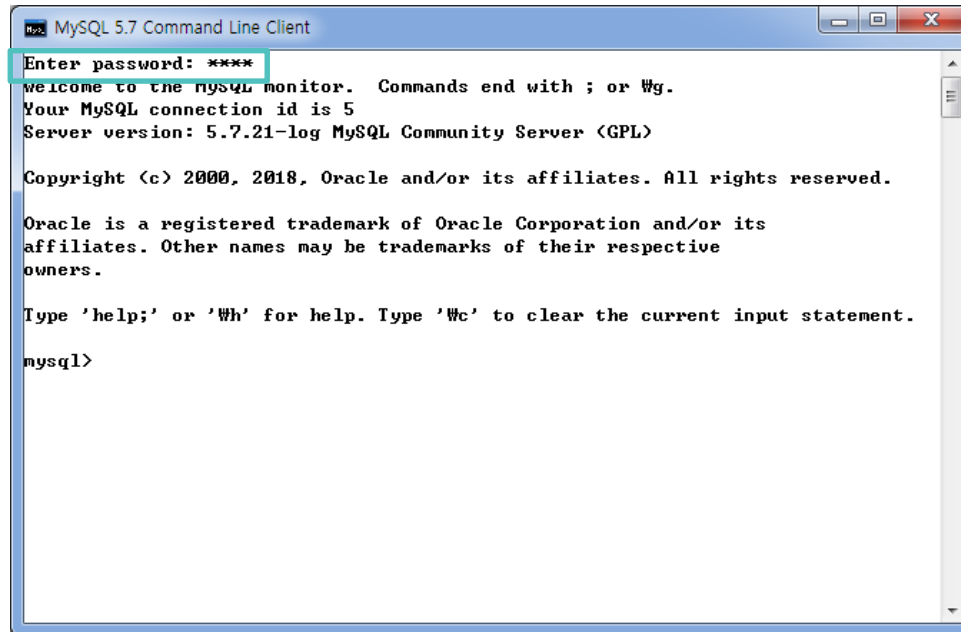
❖ 새로운 데이터베이스 생성

```
CREATE DATABASE 데이터베이스 이름;
```


3. 통합 개발 환경과 데이터베이스 연동

예제 15-7 데이터베이스 생성하기

- MySQL 접속하기:
 - 윈도우의 <시작> 버튼을 클릭하여 MySQL 5.7 Command Line Client를 실행한 후 root 계정의 비밀번호(1234)를 입력하여 MySQL에 접속



```
MySQL 5.7 Command Line Client
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

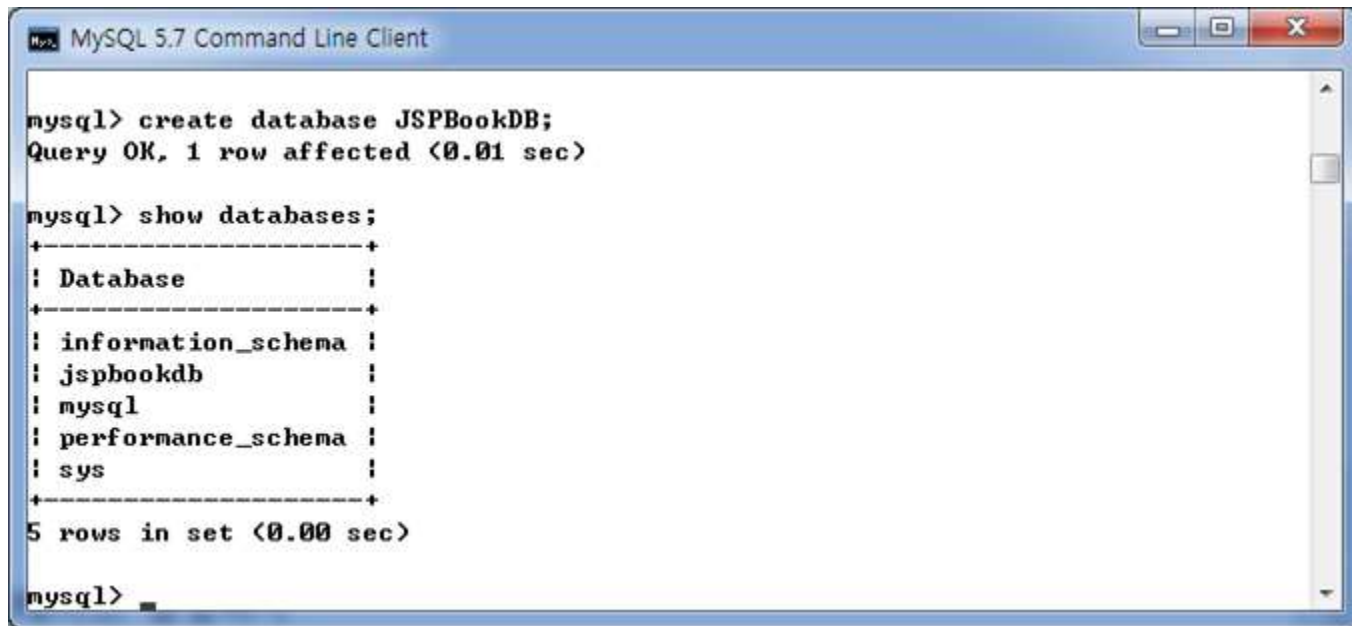
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql>
```

3. 통합 개발 환경과 데이터베이스 연동

- 새로운 데이터베이스 생성하기:
 - CREATE 명령어를 사용하여 JSPBookDB라는 데이터베이스 생성
 - 데이터베이스를 생성한 후 SHOW 명령어를 입력하면 기존의 데이터베이스 목록과 새로 생성한 JSPBookDB를 확인할 수 있음



```
mysql> create database JSPBookDB;
Query OK, 1 row affected (0.01 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| jspbookdb |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql>
```

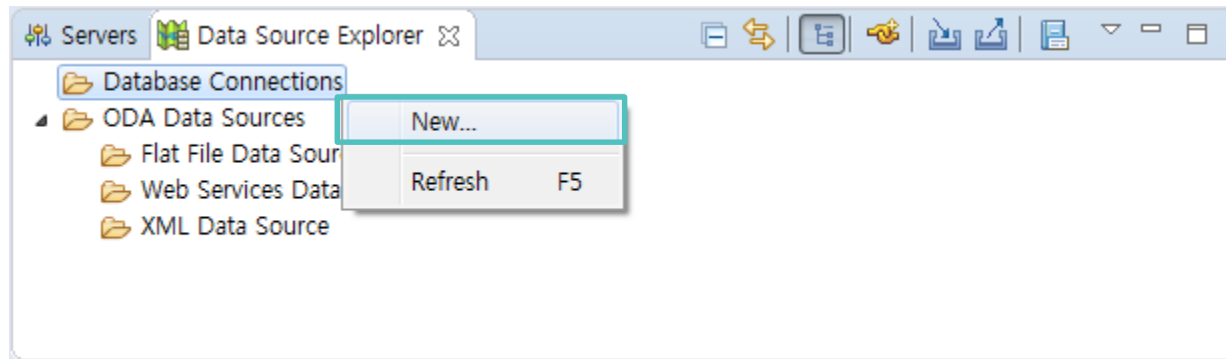
3. 통합 개발 환경과 데이터베이스 연동

❖ 데이터베이스 커넥션 설정

- Data Source Explorer 뷰를 이용하여 이클립스에 데이터베이스 커넥션을 설정

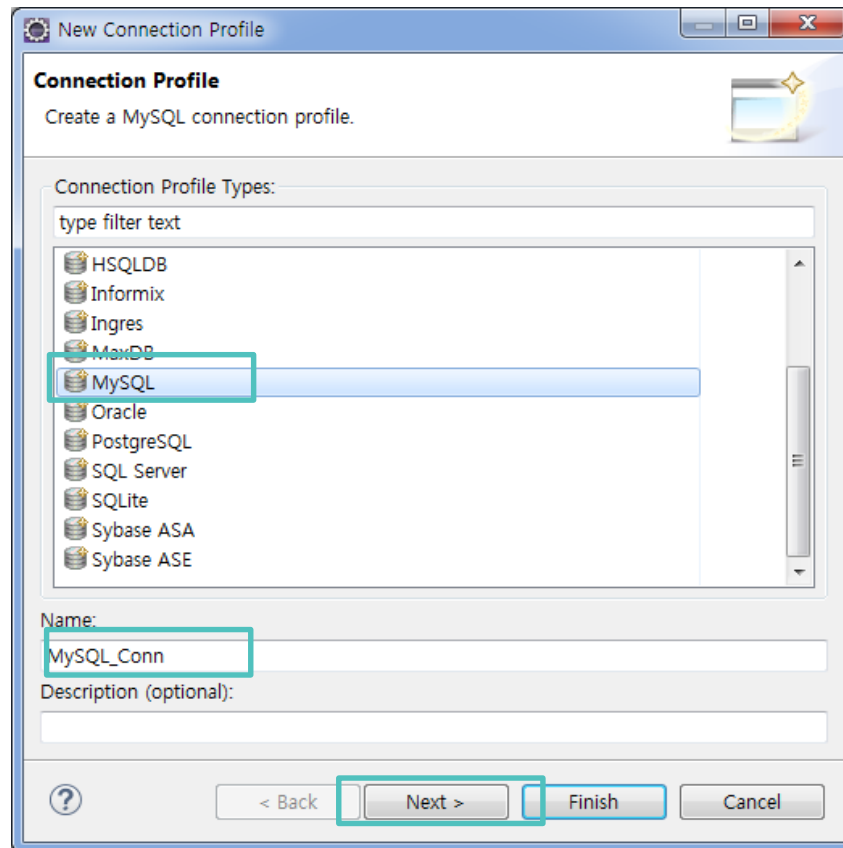
예제 15-8 이클립스에 데이터베이스 커넥션 설정하기

- Data Source Explorer 뷰 열기:
 - 이클립스의 [Window]-[Show View] 메뉴에서 'Data Source Explorer'를 선택하면 이클립스의 뷰에 나타남
 - 뷰에서 'Database Connections'를 선택한 후 마우스 오른쪽 버튼을 눌러 [New] 선택



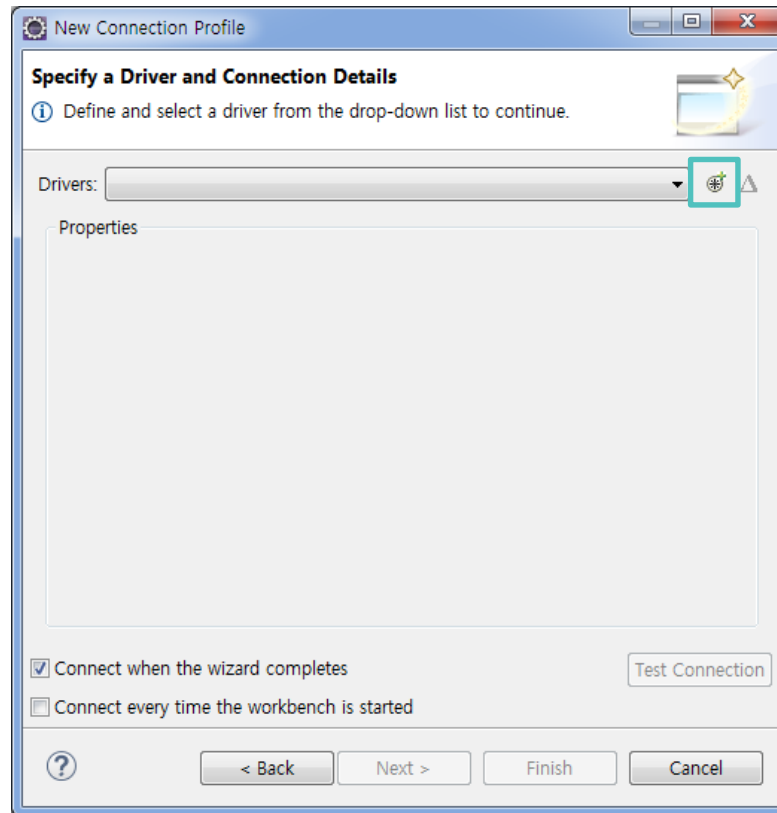
3. 통합 개발 환경과 데이터베이스 연동

- 커넥션 유형 설정하기: [Connection Profile] 화면이 나타나면 Connection Profile Types에서 'MySQL'을 선택하고 Name에 'MySQL_Conn'을 입력한 후 <Next>를 클릭합니다.



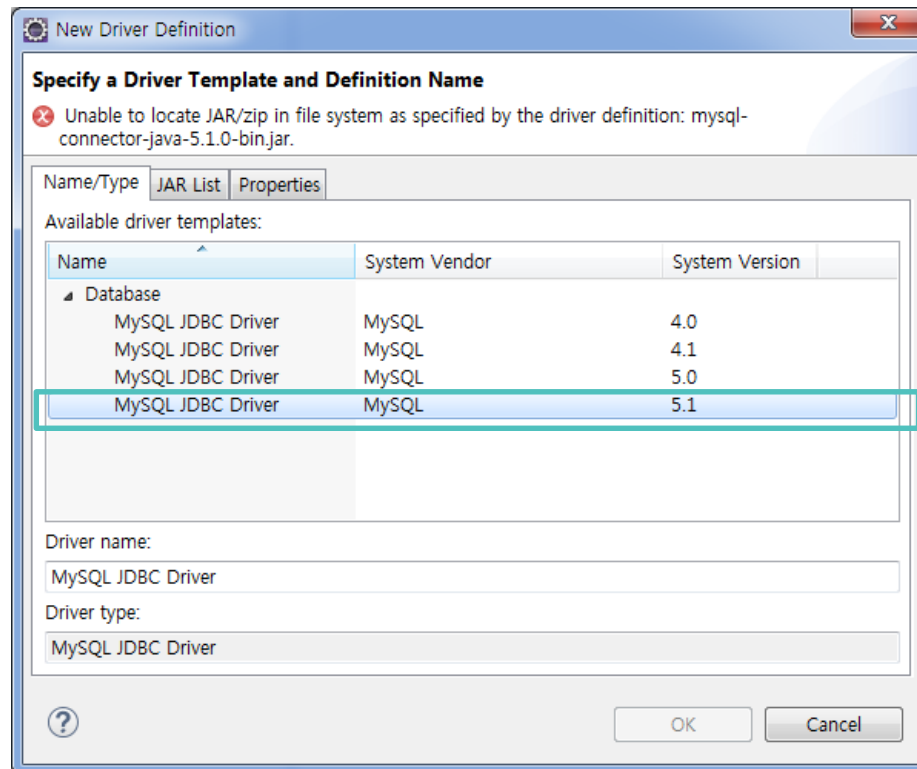
3. 통합 개발 환경과 데이터베이스 연동

- 드라이버와 커넥션 상세 설정하기:
 - [Specify a Driver and Connection Details] 화면이 나타나면 Drivers 항목의 (New Driver Definition) 버튼 클릭

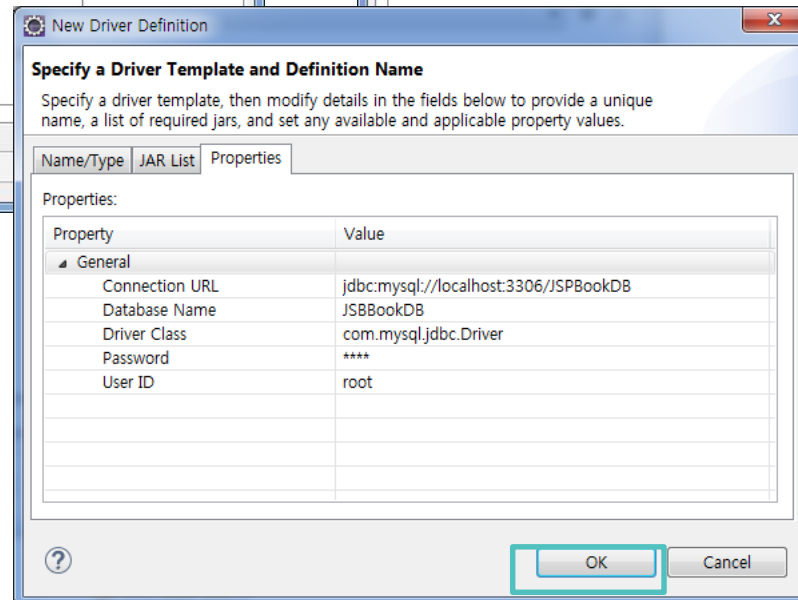
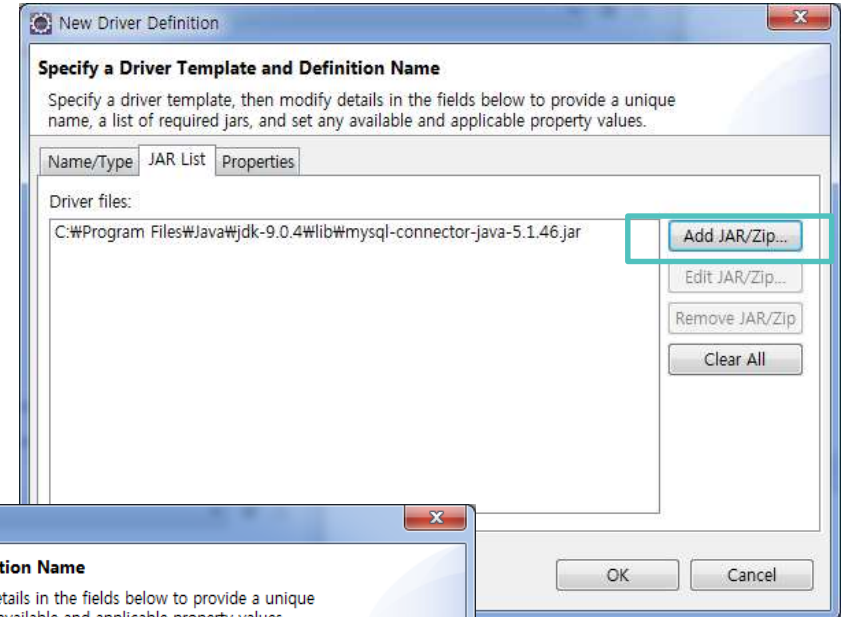
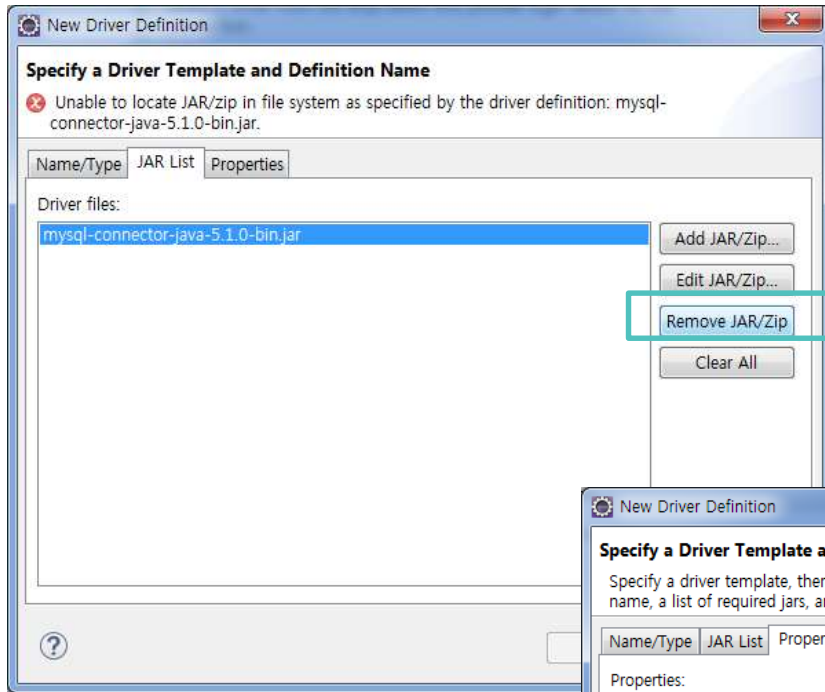


3. 통합 개발 환경과 데이터베이스 연동

- 새로운 드라이버 설정하기:
 - [New Driver Definition] 화면이 나타나면 Name/Type, JAR List, Properties 탭에 차례대로 필요한 설정을 함
 - Name/Type 탭에서는 사용할 JDBC 드라이버와 버전 선택
 - Name/Type 탭에서 MySQL JDBC Driver의 5.1 버전 선택



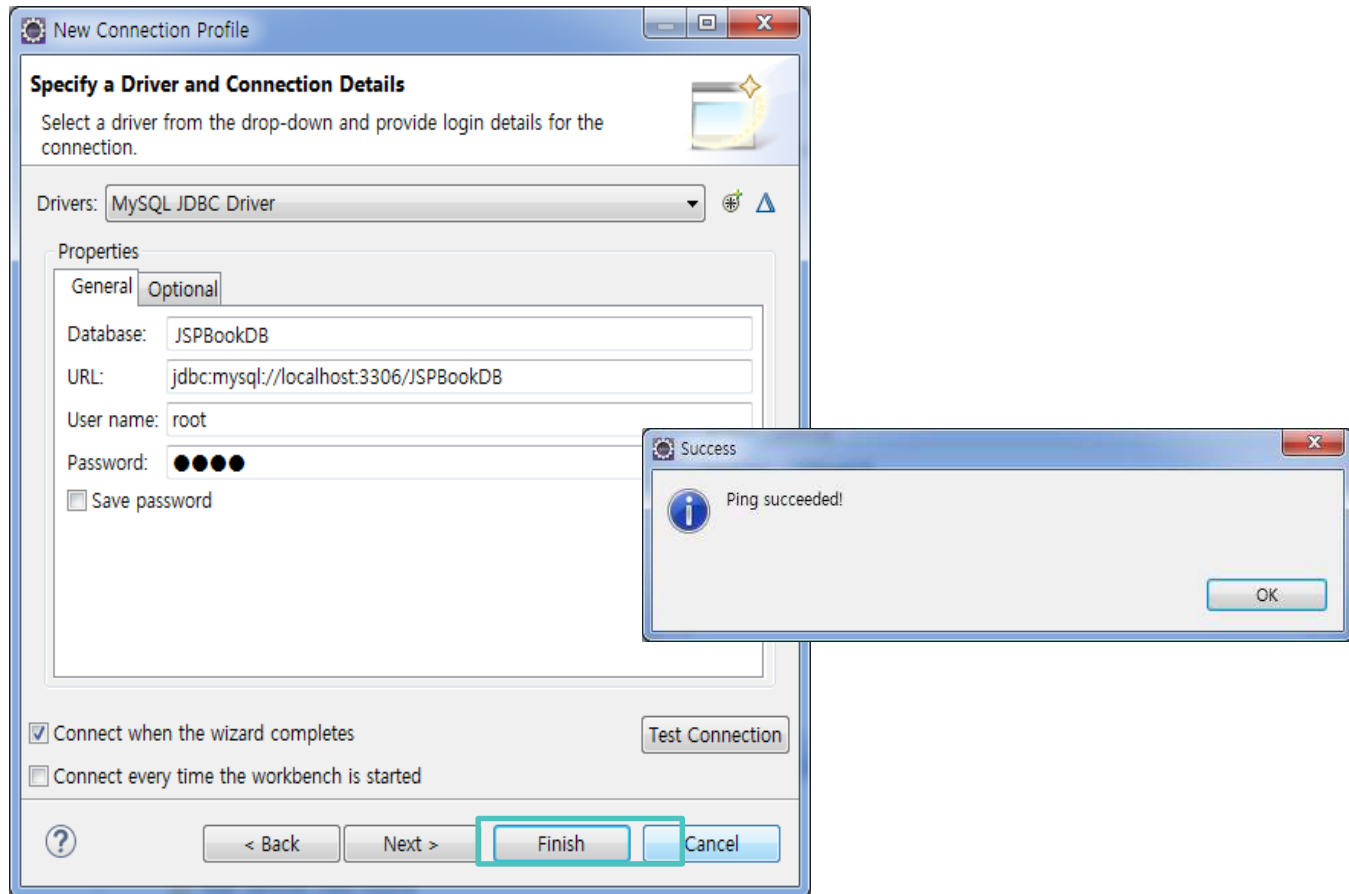
3. 통합 개발 환경과 데이터베이스 연동



3. 통합 개발 환경과 데이터베이스 연동

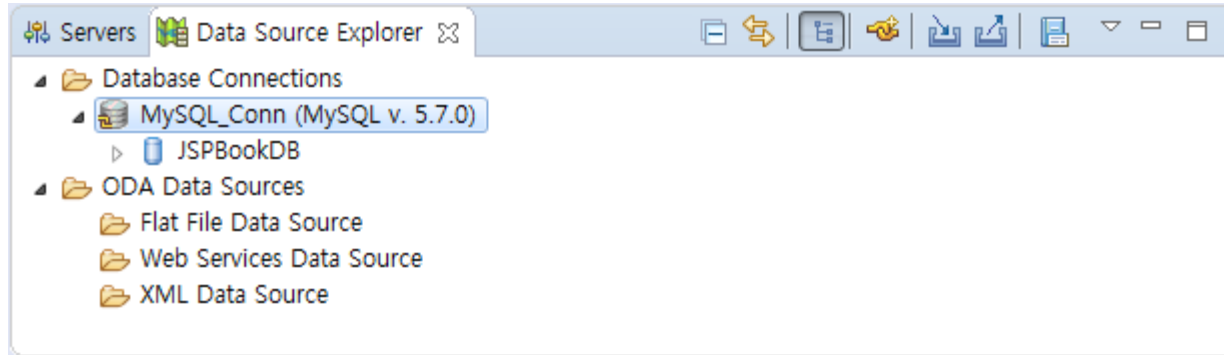
■ 커넥션 연결 설정 끝내기:

- 설정된 JDBC 드라이버와 데이터베이스의 이름이 표시되면 <Test Connection> 클릭
- [Success] 대화상자가 나타나면 <OK> 클릭, <Finish> 클릭하여 모든 설정 완료

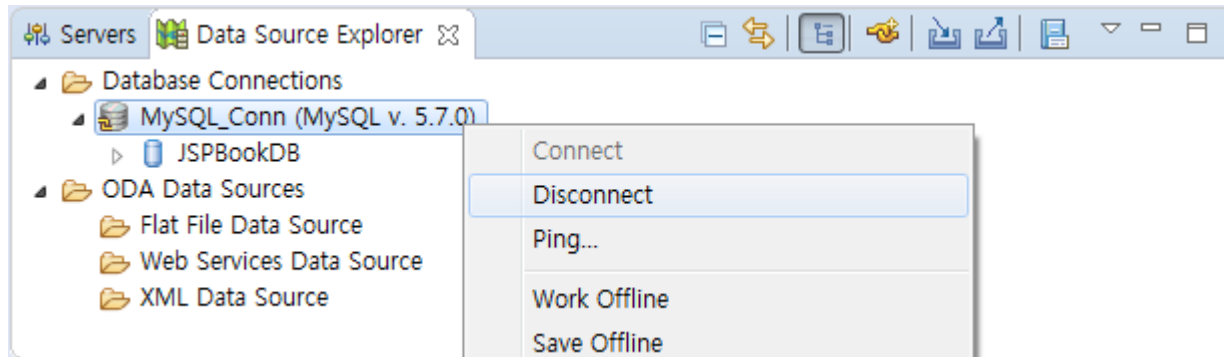


3. 통합 개발 환경과 데이터베이스 연동

- 커넥션 연결 확인하기:
 - MySQL을 직접 제어하는 커넥션이 연결되면 다음과 같이 나타남




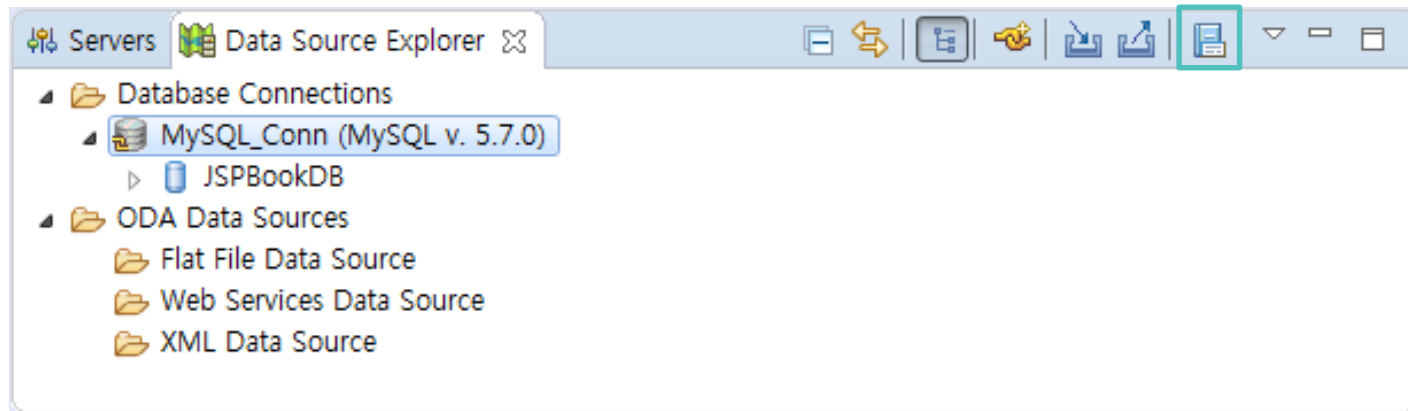
- 앞으로 커넥션 연결은 [Connect] 메뉴를, 커넥션 해제는 [Disconnect] 메뉴를 사용하여 할 수 있음



3. 통합 개발 환경과 데이터베이스 연동

예제 15-9 이클립스에서 데이터베이스 제어하기

- SQL 스크립트 열기:
 - SQL 구문을 편집하는 스크립트를 열기 위해 Data Source Explorer 뷰에서 'MySQL_Conn (MySQL 버전)'을 선택한 후  클릭



3. 통합 개발 환경과 데이터베이스 연동

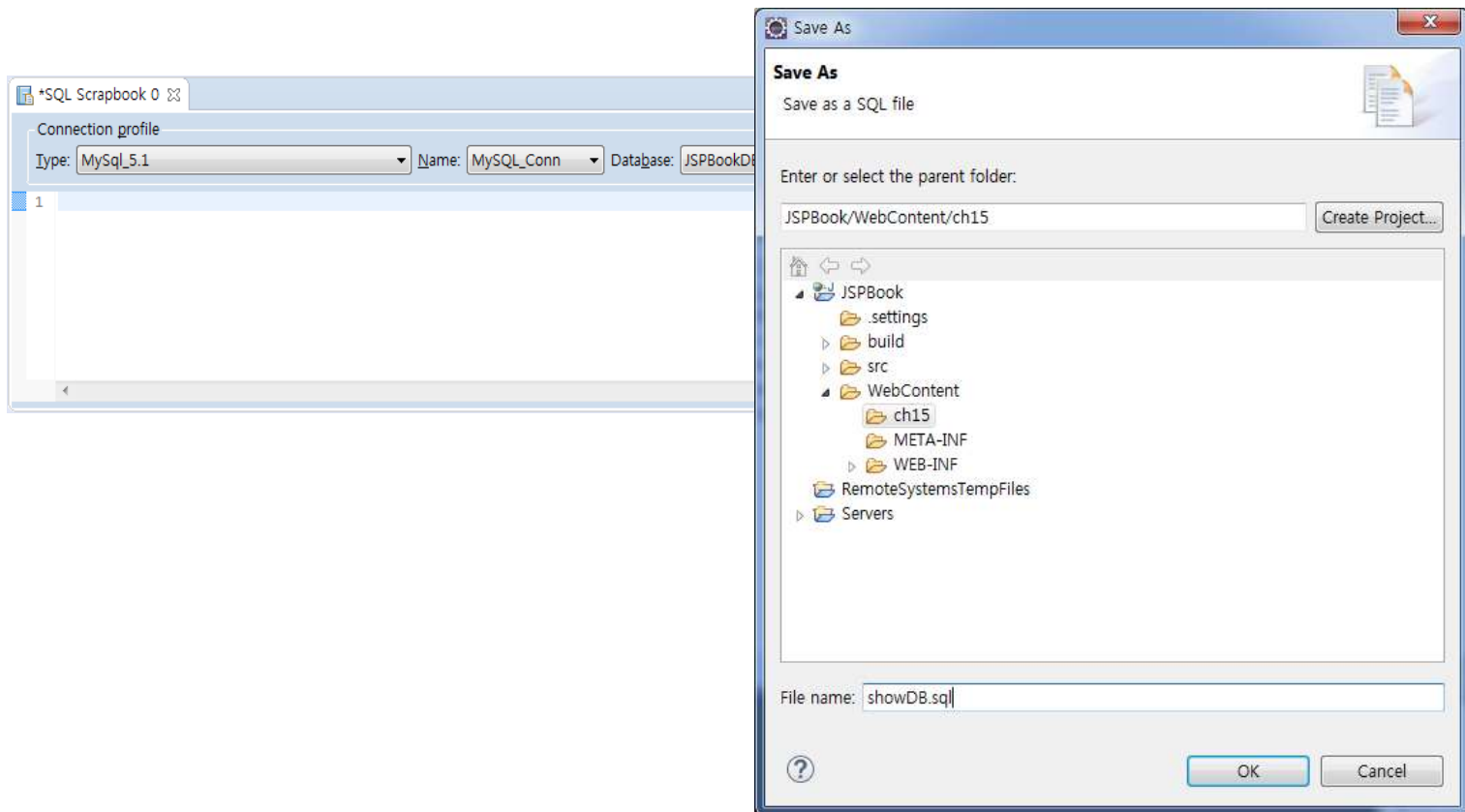
- 설정된 커넥션 선택하기:
 - SQL 스크랩북의 Connection profile에서 Type은 'MySql_5.1', Name은 'MySQL_Conn', Database는 'JSPBookDB' 선택



3. 통합 개발 환경과 데이터베이스 연동

- SQL 파일 저장하기:

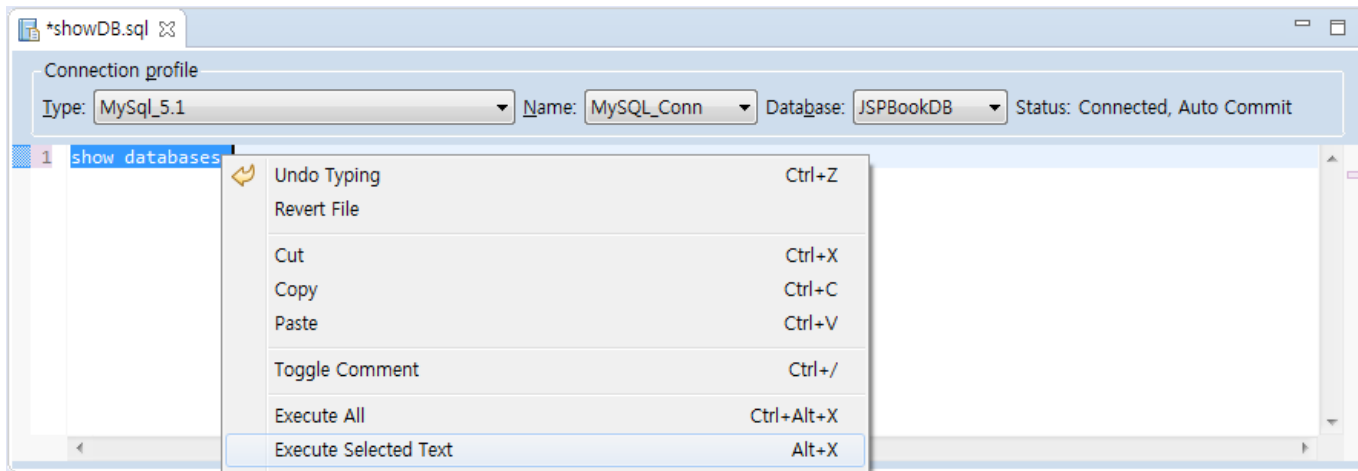
- SQL 스크랩북에 SQL 문장을 입력한 후 이클립스의 [New]-[Save] 메뉴에서 프로젝트 (여기서는 JSPBook/ch15)를 선택하고 File name에 'showDB.sql' 입력, <OK> 클릭



3. 통합 개발 환경과 데이터베이스 연동

- 쿼리문 실행하기:

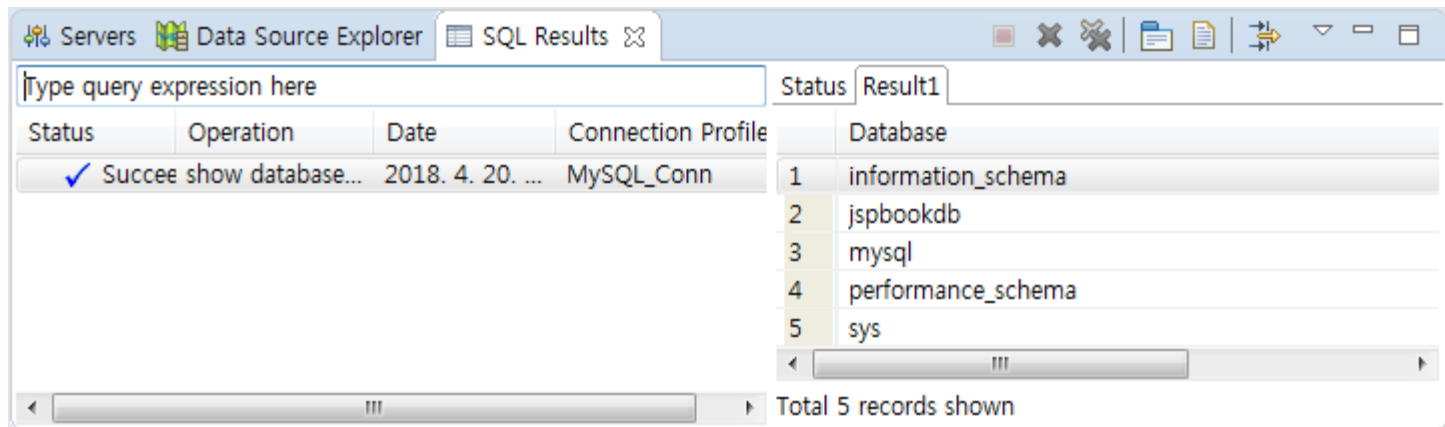
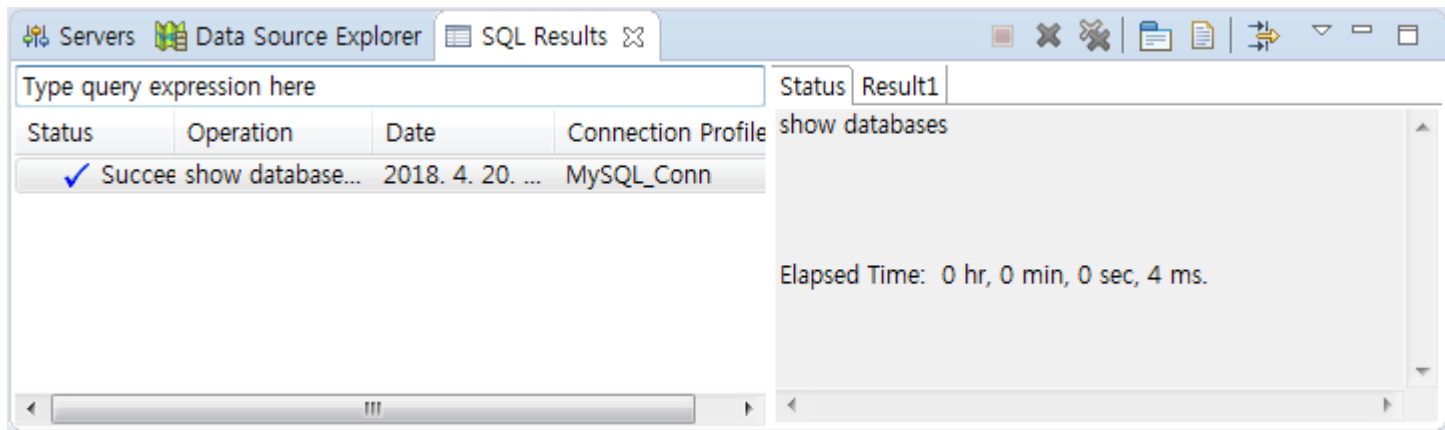
- 입력한 SQL 구문을 마우스로 드래그하여 블록을 설정하고 마우스 오른쪽 버튼을 클릭하여 [Execute Selected Text] 메뉴 선택



3. 통합 개발 환경과 데이터베이스 연동

■ 실행 결과 확인하기:

- 쿼리문을 실행하면 화면에 SQL Results 뷰가 나타남
- SQL Results 뷰의 Status 탭에는 실행한 쿼리문의 성공 여부가 표시되고 Result1 탭에서는 실행 결과를 확인할 수 있음



4. MySQL 기본 명령어

❖ 데이터베이스 언어

- 데이터베이스를 통해 특정 정보를 추출하고 보여주는 언어
- SQL
 - 관계형 데이터베이스를 관리하는 데이터베이스 관리 시스템에서 사용하는 언어
 - 질의 기능을 비롯해 데이터 정의 및 조작 기능을 가짐
 - 개개의 레코드보다 테이블 단위로 연산을 수행
- SQL 문장의 종류

SQL 문장	종류
데이터 정의 언어(data definition language)	CREATE, ALTER, DROP, ...
데이터 조작 언어(data manipulation language)	SELECT, INSERT, DELETE, UPDATE, ...
데이터 제어 언어(data control language)	GRANT, REVOKE, COMMIT, ROLLBACK, ...

4. MySQL 기본 명령어

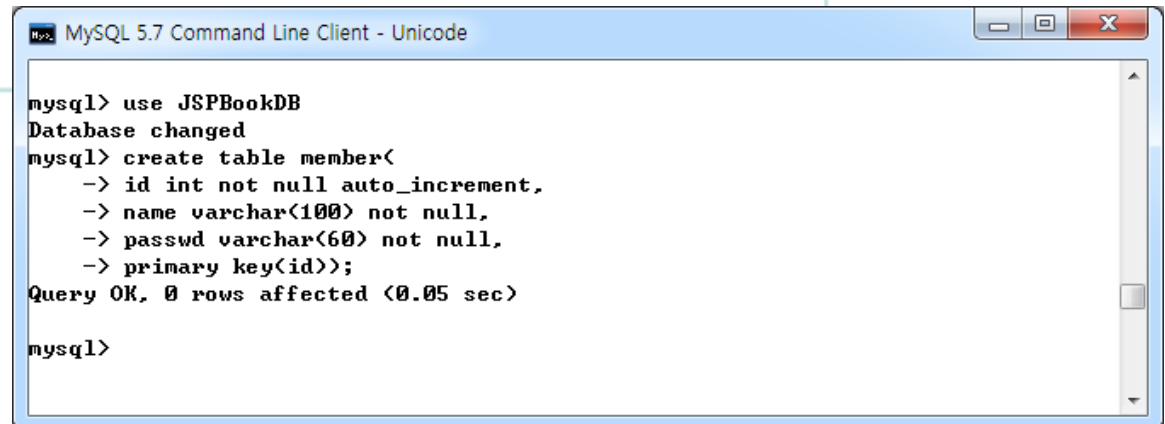
❖ 테이블 관련 명령어

■ 테이블 생성하기

```
CREATE TABLE 테이블 이름(필드 이름1 자료형1[, 필드 이름2 자료형2, ...]);
```

[CREATE 명령어 사용 예]

```
CREATE TABLE member(  
    id int NOT NULL AUTO_INCREMENT,  
  
    name VARCHAR(100) not null,  
    passwd VARCHAR(50) not null,  
    PRIMARY KEY (id)  
);
```



```
MySQL 5.7 Command Line Client - Unicode  
  
mysql> use JSPBookDB  
Database changed  
mysql> create table member(  
    -> id int not null auto_increment,  
    -> name varchar(100) not null,  
    -> passwd varchar(60) not null,  
    -> primary key(id));  
Query OK, 0 rows affected (0.05 sec)  
  
mysql>
```

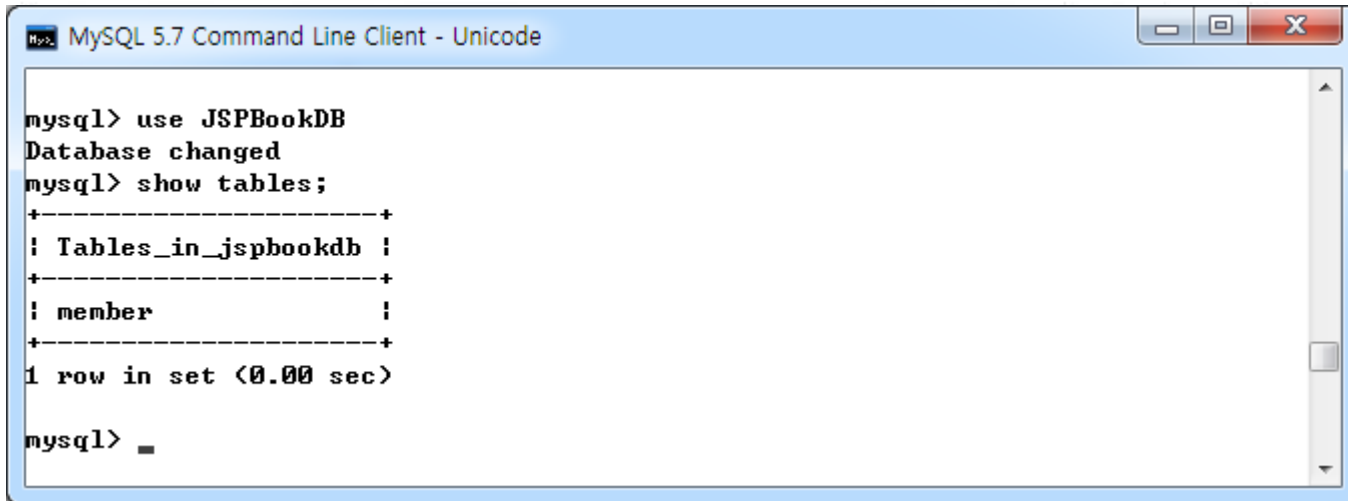

4. MySQL 기본 명령어

❖ 테이블 관련 명령어

- 테이블 조회하기

```
SHOW TABLES;
```

[SHOW 명령어 사용 예]



```
MySQL 5.7 Command Line Client - Unicode

mysql> use JSPBookDB
Database changed
mysql> show tables;
+-----+
| Tables_in_jspbookdb |
+-----+
| member               |
+-----+
1 row in set (0.00 sec)

mysql> _
```

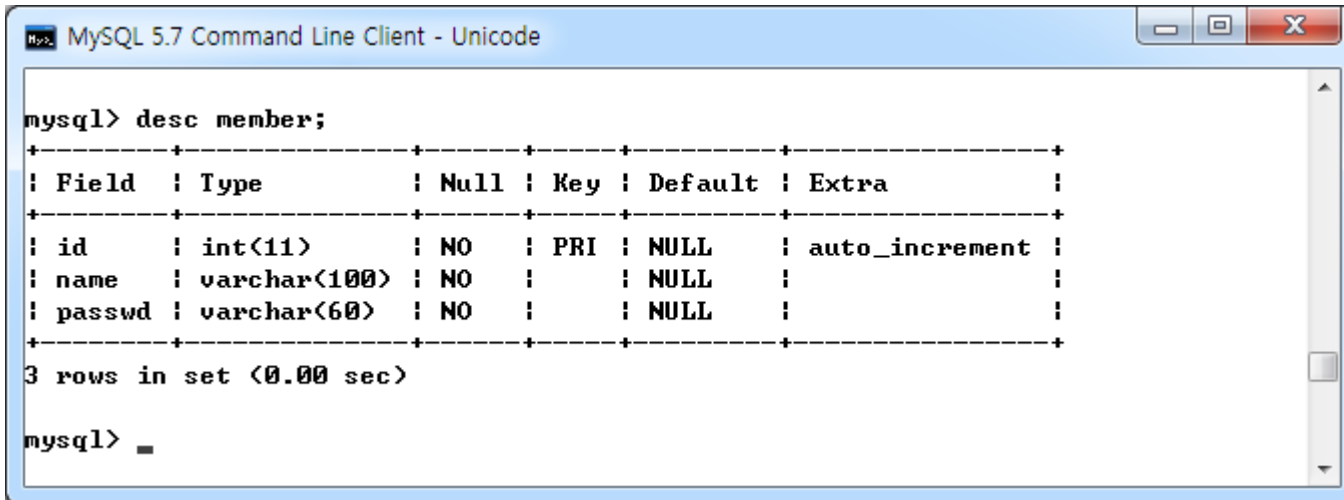
4. MySQL 기본 명령어

❖ 테이블 관련 명령어

- 테이블 세부 조회하기

DESC 테이블 이름;

[DESC 명령어 사용 예]



```
MySQL 5.7 Command Line Client - Unicode

mysql> desc member;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id     | int(11)       | NO   | PRI | NULL    | auto_increment |
| name   | varchar(100)  | NO   |     | NULL    |                |
| passwd | varchar(60)   | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> _
```

4. MySQL 기본 명령어

❖ 테이블 관련 명령어

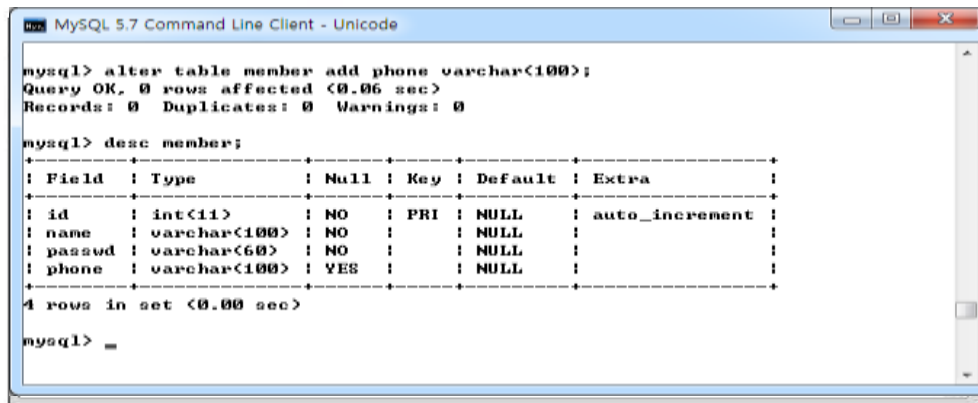
■ 테이블 구조 변경하기

```
ALTER TABLE 테이블 이름  
[ADD 필드 이름 자료형 |  
DROP COLUMN 필드 이름 |  
CHANGE COLUMN 기존 필드 이름 새 필드 이름 자료형];
```

■ 기존 테이블에 열을 추가하는 경우

```
ALTER TABLE Member ADD phone varchar(100);
```

[ALTER 명령어 사용 예: 열 이름 phone 추가]



```
mysql> alter table member add phone varchar(100);  
Query OK, 0 rows affected (0.06 sec)  
Records: 0 Duplicates: 0 Warnings: 0  
  
mysql> desc member;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type          | Null | Key | Default | Extra          |  
+-----+-----+-----+-----+-----+-----+  
| id    | int(11)       | NO   | PRI | NULL    | auto_increment |  
| name  | varchar(100)  | NO   |     | NULL    |                |  
| passwd| varchar(60)   | NO   |     | NULL    |                |  
| phone | varchar(100)  | YES  |     | NULL    |                |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.00 sec)  
  
mysql> _
```

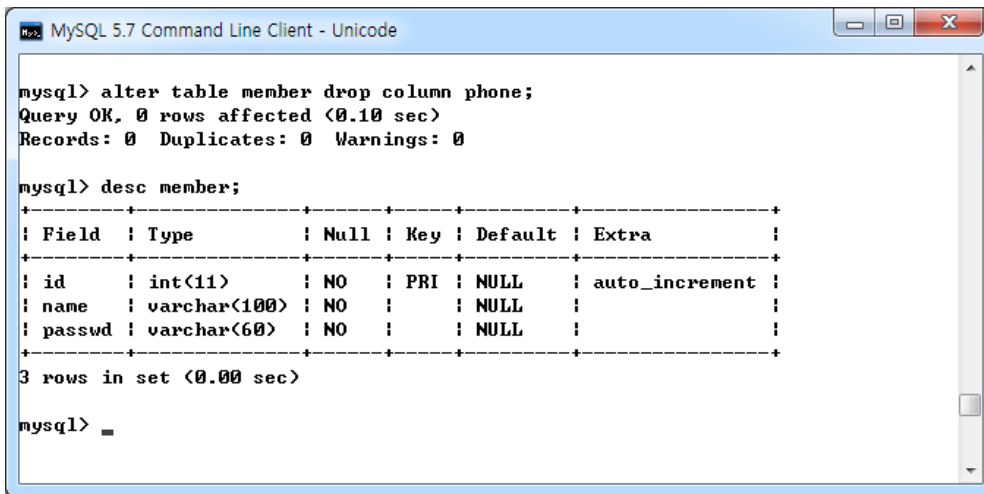
4. MySQL 기본 명령어

❖ 테이블 관련 명령어

- 기존 테이블의 열을 삭제하는 경우

```
ALTER TABLE Member DROP COLUMN phone;
```

[ALTER 명령어 사용 예: 열 이름 phone 삭제]



```
mysql> alter table member drop column phone;
Query OK, 0 rows affected (0.10 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc member;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | NO   | PRI | NULL    | auto_increment |
| name  | varchar(100)  | NO   |     | NULL    |                 |
| passwd| varchar(60)   | NO   |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> _
```

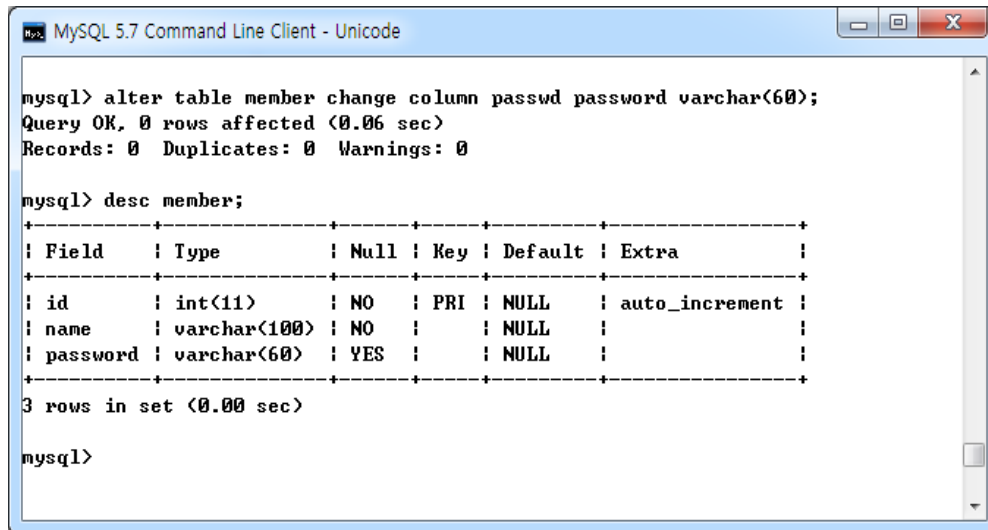
4. MySQL 기본 명령어

❖ 테이블 관련 명령어

- 기존 테이블의 열 이름을 수정하는 경우

```
ALTER TABLE Member |CHANGE COLUMN passwd password varchar(60);
```

[ALTER 명령어 사용 예: 열 이름 passwd를 password로 수정]



```
mysql> alter table member change column passwd password varchar(60);
Query OK, 0 rows affected (0.06 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc member;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | NO   | PRI | NULL    | auto_increment |
| name  | varchar(100)  | NO   |     | NULL    |                 |
| password | varchar(60)  | YES  |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

4. MySQL 기본 명령어

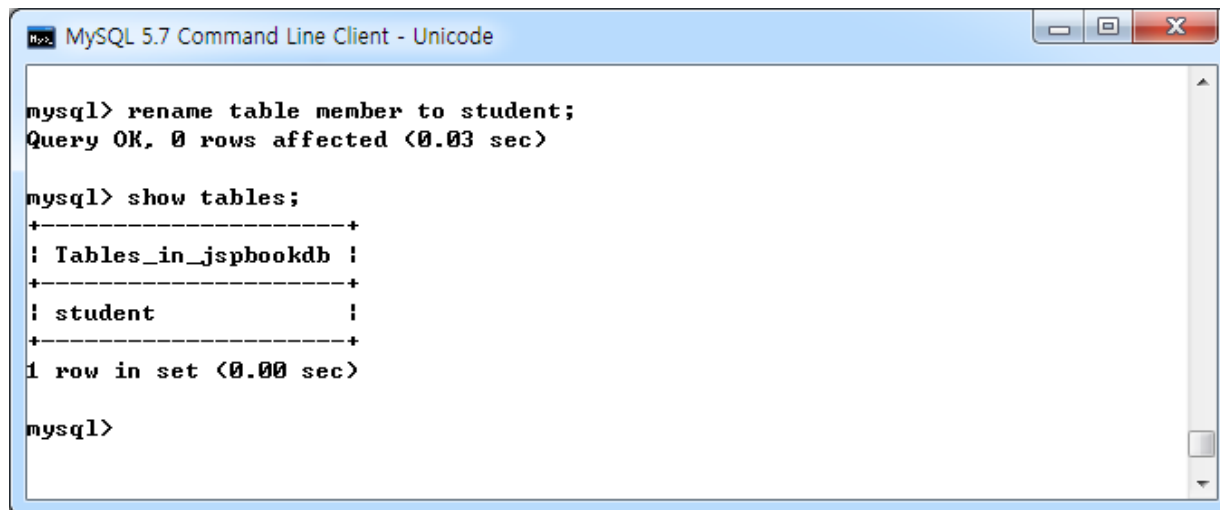
❖ 테이블 관련 명령어

- 테이블 이름 변경하기

```
RENAME TABLE 테이블 이름 TO 새 테이블 이름[,  
테이블 이름 TO 새 테이블 이름, ...];
```

[RENAME 명령어 사용 예]

```
RENAME TABLE Member TO Student;
```



```
MySQL 5.7 Command Line Client - Unicode  
  
mysql> rename table member to student;  
Query OK, 0 rows affected (0.03 sec)  
  
mysql> show tables;  
+-----+  
! Tables_in_jspbookdb !  
+-----+  
! student             !  
+-----+  
1 row in set (0.00 sec)  
  
mysql>
```

4. MySQL 기본 명령어

❖ 데이터 조작 명령어

- 데이터 조작 명령어는 사용자가 적절한 데이터 모델로 구성된 데이터에 접근하거나 데이터를 조작할 수 있도록 하는 언어
- 데이터베이스 내의 데이터 연산을 위한 언어로 데이터베이스 내에서 데이터 검색, 추가, 삭제, 갱신 작업이 가능

명령어	설명	명령어	설명
INSERT	데이터를 등록합니다.	DELETE	데이터를 삭제합니다.
UPDATE	데이터를 수정합니다.	SELECT	데이터를 조회합니다.

4. MySQL 기본 명령어

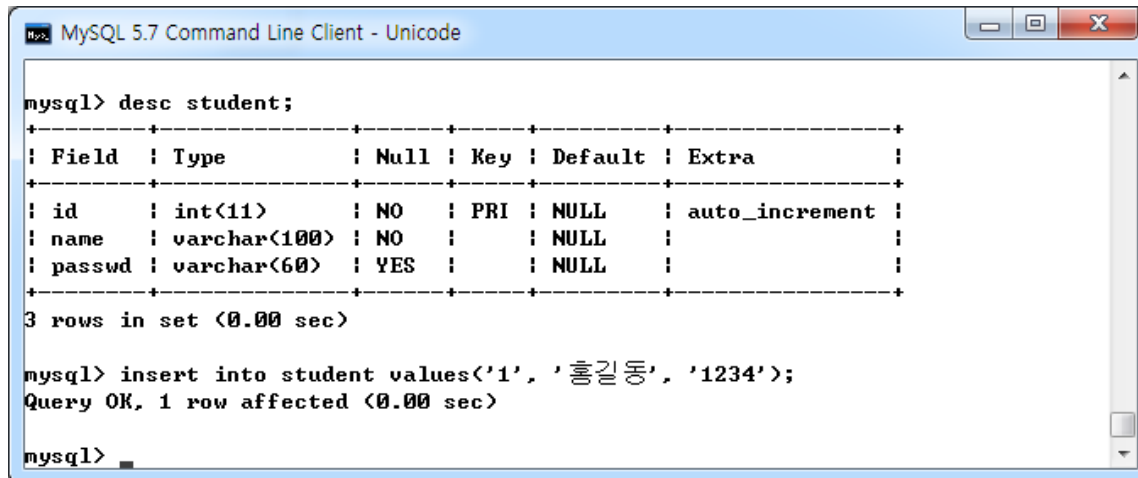
❖ 데이터 조작 명령어

■ 데이터 등록하기

INSERT [INTO] 테이블 이름 [(필드 이름, 필드 이름, ...)] VALUES (필드 값, 필드 값, ...)

[INSERT 명령어 사용 예]

```
INSERT INTO Student VALUES('1', '홍길동', '1234');
```



```
MySQL 5.7 Command Line Client - Unicode

mysql> desc student;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | NO   | PRI | NULL    | auto_increment |
| name  | varchar(100)  | NO   |     | NULL    |                 |
| passwd| varchar(60)   | YES  |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> insert into student values('1', '홍길동', '1234');
Query OK, 1 row affected (0.00 sec)

mysql>
```


4. MySQL 기본 명령어

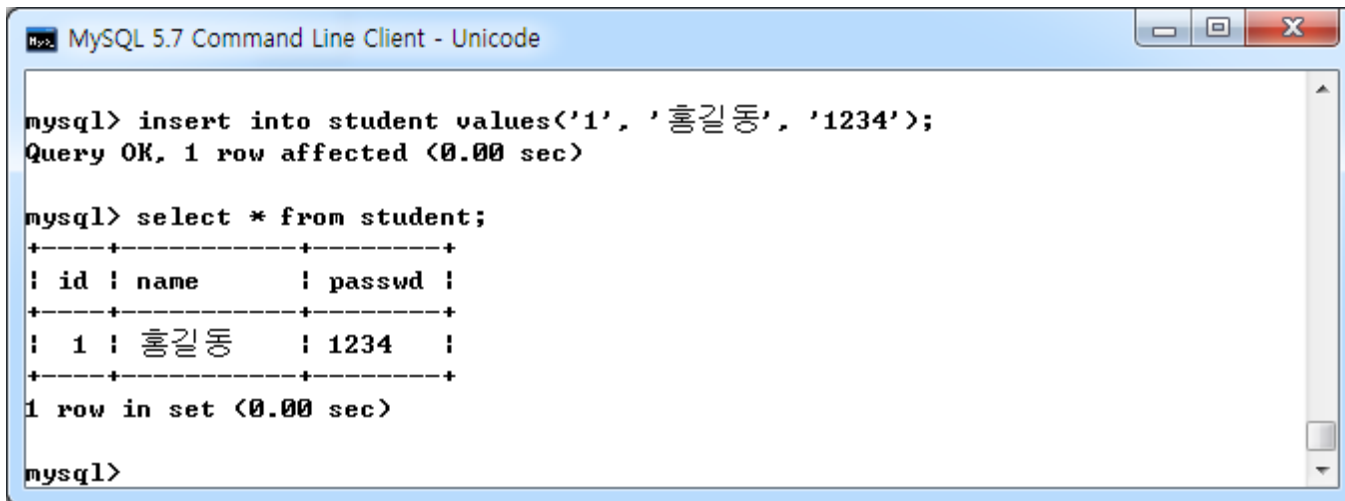
❖ 데이터 조작 명령어

- 데이터 조회하기

```
SELECT 필드 이름[, 필드 이름, ...] FROM 테이블 이름 [WHERE 검색조건] [ORDER BY 필드 이름  
[ASC or DESC]] [GROUP BY 필드 이름[, 필드 이름, ...]] ...
```

[SELECT 명령어 사용 예: 모든 행 검색]

```
SELECT * FROM Student ;
```



The screenshot shows a MySQL 5.7 Command Line Client window with the following text:


```
mysql> insert into student values('1', '홍길동', '1234');  
Query OK, 1 row affected (0.00 sec)  
  
mysql> select * from student;  
+----+-----+-----+  
| id | name   | passwd |  
+----+-----+-----+  
| 1  | 홍길동 | 1234   |  
+----+-----+-----+  
1 row in set (0.00 sec)  
  
mysql>
```

4. MySQL 기본 명령어

[SELECT 명령어 사용 예: name, passwd 필드의 데이터 검색]

```
SELECT name, passwd FROM Student ;
```

[SELECT 명령어 사용 예: id의 내림차순 정렬과 3개 행의 데이터 검색]

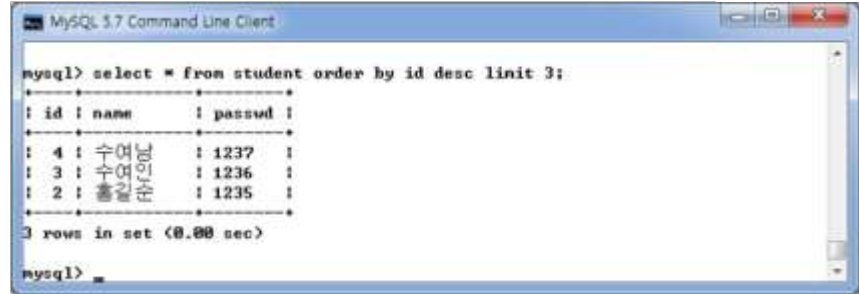


```
mysql> select name, passwd from student;
```

name	passwd
홍길동	1234

```
1 row in set (0.00 sec)
```

```
mysql>
```



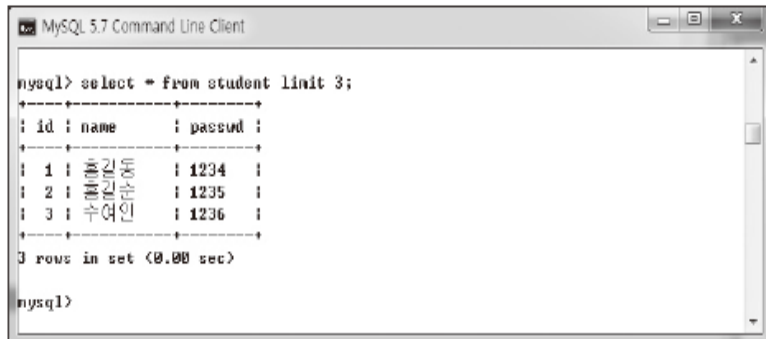
```
mysql> select * from student order by id desc limit 3;
```

id	name	passwd
4	수여남	1237
3	수여인	1236
2	홍길순	1235

```
3 rows in set (0.00 sec)
```

```
mysql>
```

[SELECT 명령어 사용 예: 3개 행의 데이터 검색]



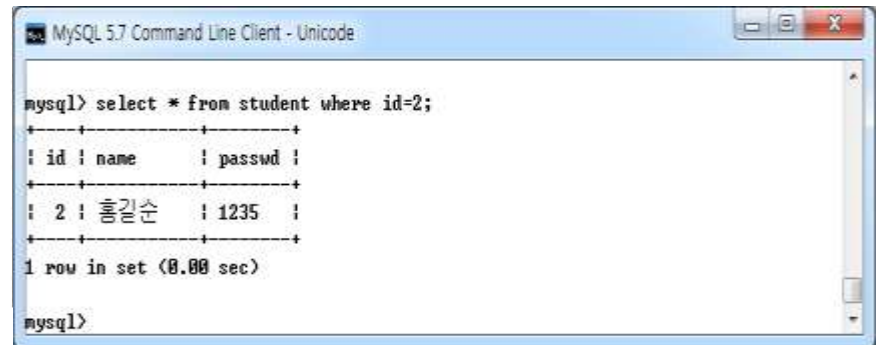
```
mysql> select * from student limit 3;
```

id	name	passwd
1	홍길동	1234
2	홍길순	1235
3	수여인	1236

```
3 rows in set (0.00 sec)
```

```
mysql>
```

[SELECT 명령어 사용 예: id가 2인 데이터 검색]



```
mysql> select * from student where id=2;
```

id	name	passwd
2	홍길순	1235

```
1 row in set (0.00 sec)
```

```
mysql>
```

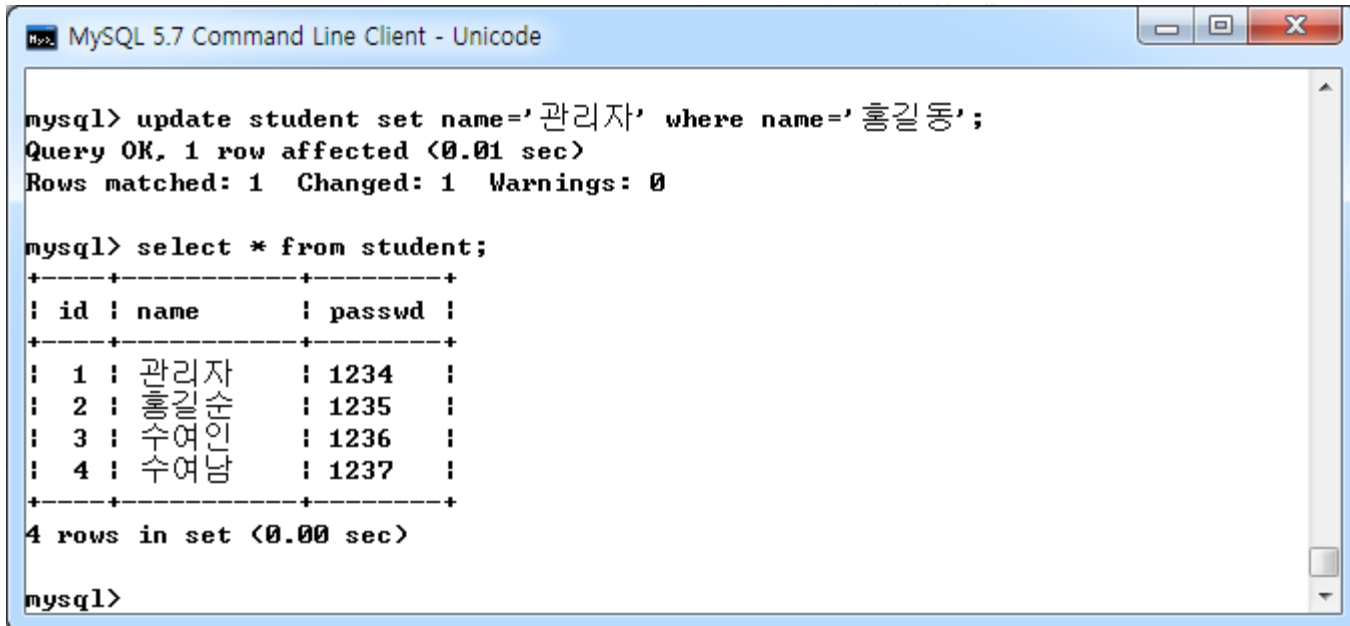
4. MySQL 기본 명령어

❖ 데이터 조작 명령어

- 데이터 수정하기

UPDATE 테이블 이름 SET 필드 이름=필드 값[, 필드 이름=필드 값, ...] [WHERE 검색조건]

[UPDATE 명령어 사용 예: name이 '홍길동'인 것을 '관리자'로 수정]



```
MySQL 5.7 Command Line Client - Unicode

mysql> update student set name='관리자' where name='홍길동';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from student;
+----+-----+-----+
| id | name   | passwd |
+----+-----+-----+
| 1  | 관리자 | 1234   |
| 2  | 홍길순 | 1235   |
| 3  | 수여인 | 1236   |
| 4  | 수여남 | 1237   |
+----+-----+-----+
4 rows in set (0.00 sec)

mysql>
```

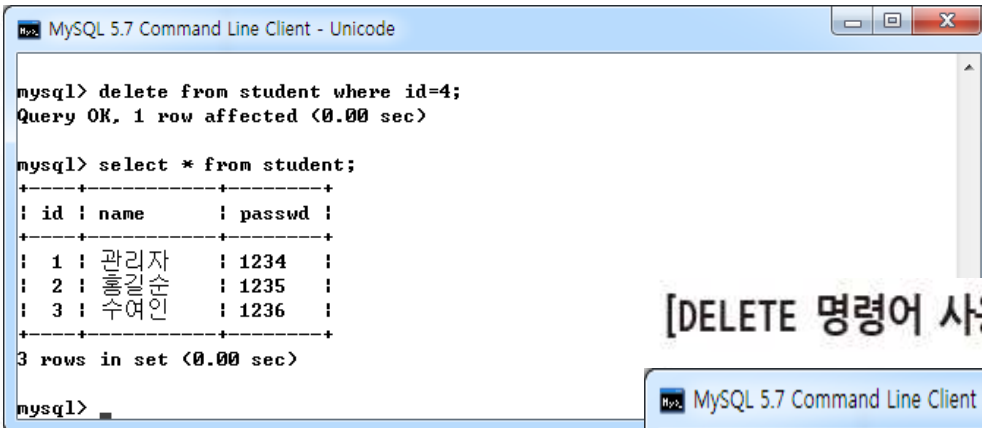
4. MySQL 기본 명령어

❖ 데이터 조작 명령어

- 데이터 삭제하기

DELETE FROM 테이블 이름 [WHERE 검색조건]

[DELETE 명령어 사용 예: id가 4인 행 삭제]

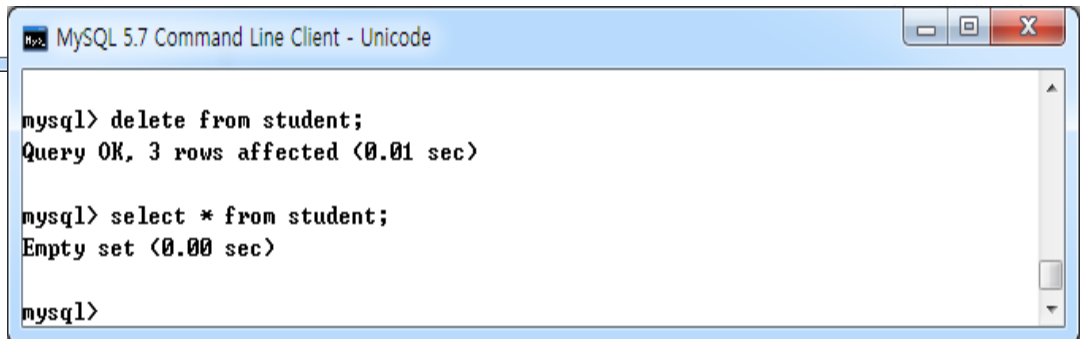


```
mysql> delete from student where id=4;
Query OK, 1 row affected (0.00 sec)

mysql> select * from student;
+----+-----+-----+
| id | name  | passwd |
+----+-----+-----+
| 1  | 관리자 | 1234   |
| 2  | 홍길순 | 1235   |
| 3  | 수여인 | 1236   |
+----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

[DELETE 명령어 사용 예: 모든 행 삭제]



```
mysql> delete from student;
Query OK, 3 rows affected (0.01 sec)

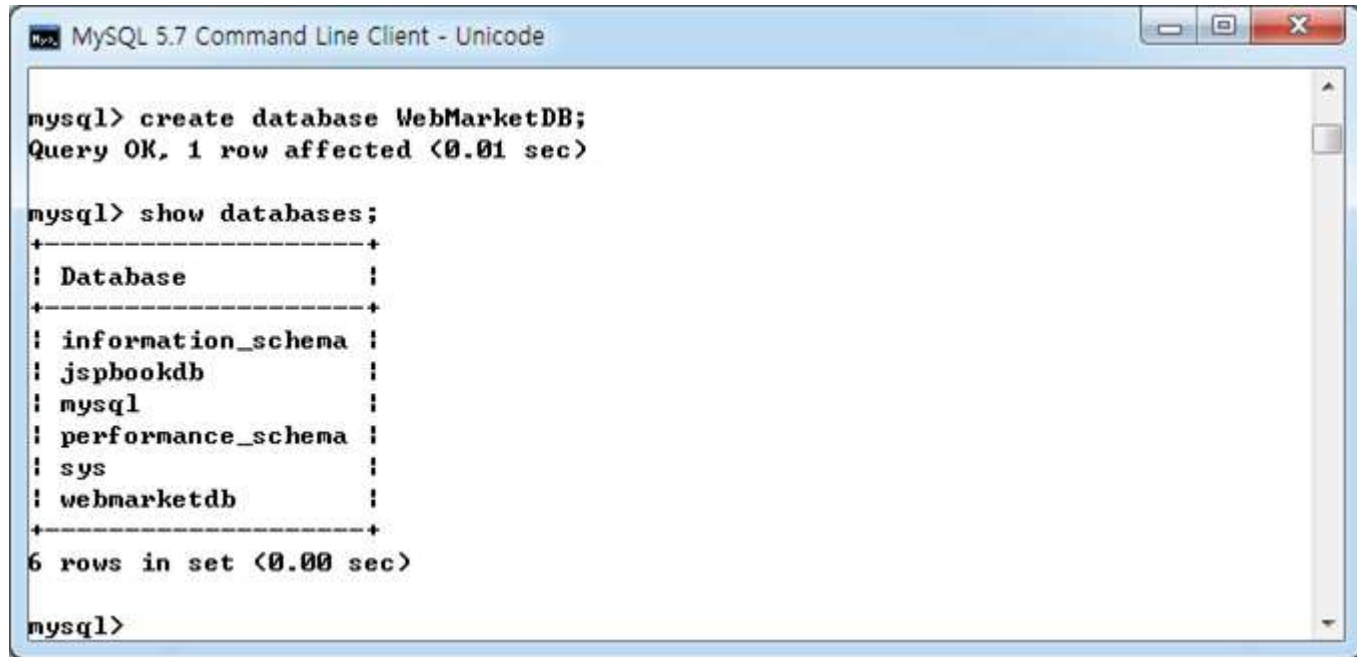
mysql> select * from student;
Empty set (0.00 sec)

mysql>
```

5. [웹 쇼핑몰] 상품 관리 테이블 생성 및 상품 등록하기

예제 15-10 웹 쇼핑몰의 데이터베이스 연동하기

- 데이터베이스 생성하기:
 - 명령 프롬프트로 MySQL에 접속하여 WebMarketDB라는 데이터베이스 생성



```
MySQL 5.7 Command Line Client - Unicode

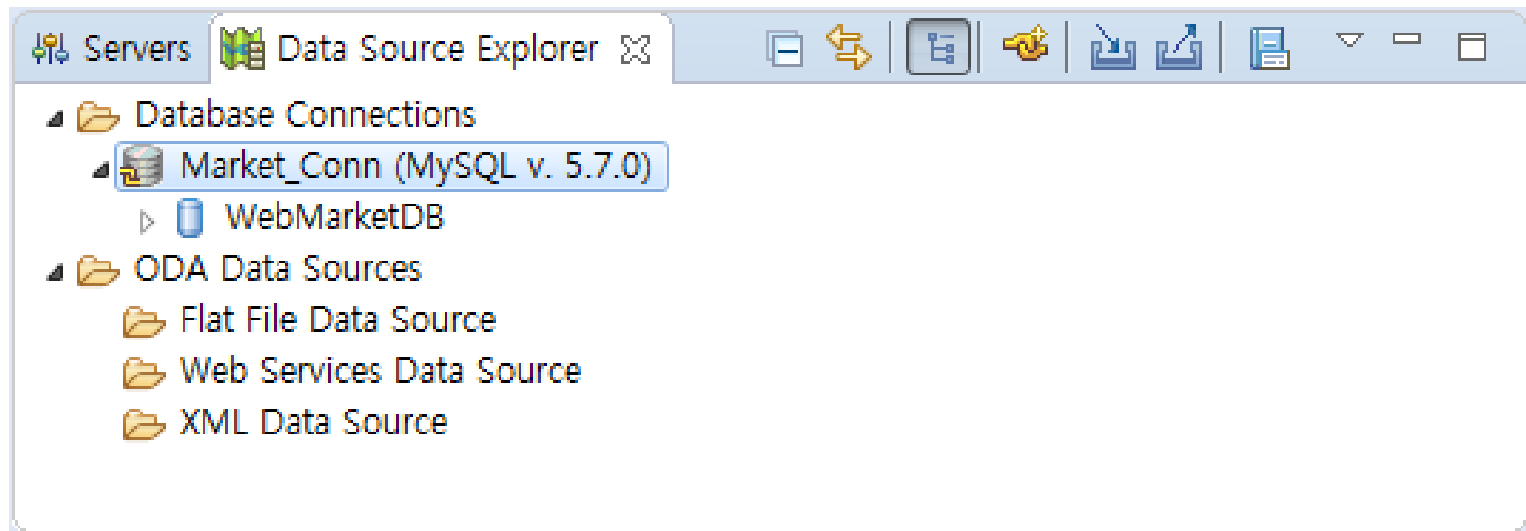
mysql> create database WebMarketDB;
Query OK, 1 row affected (0.01 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| jspbookdb |
| mysql |
| performance_schema |
| sys |
| webmarketdb |
+-----+
6 rows in set (0.00 sec)

mysql>
```

5. [웹 쇼핑몰] 상품 관리 테이블 생성 및 상품 등록하기

- 데이터베이스 커넥션 설정하기:
 - 다음과 같이 커넥션 이름은 Market_Conn,
 - 데이터베이스 이름은 WebMarketDB로 설정



5. [웹 쇼핑몰] 상품 관리 테이블 생성 및 상품 등록하기

예제 15-11 웹 쇼핑몰의 상품 관리 테이블 만들기

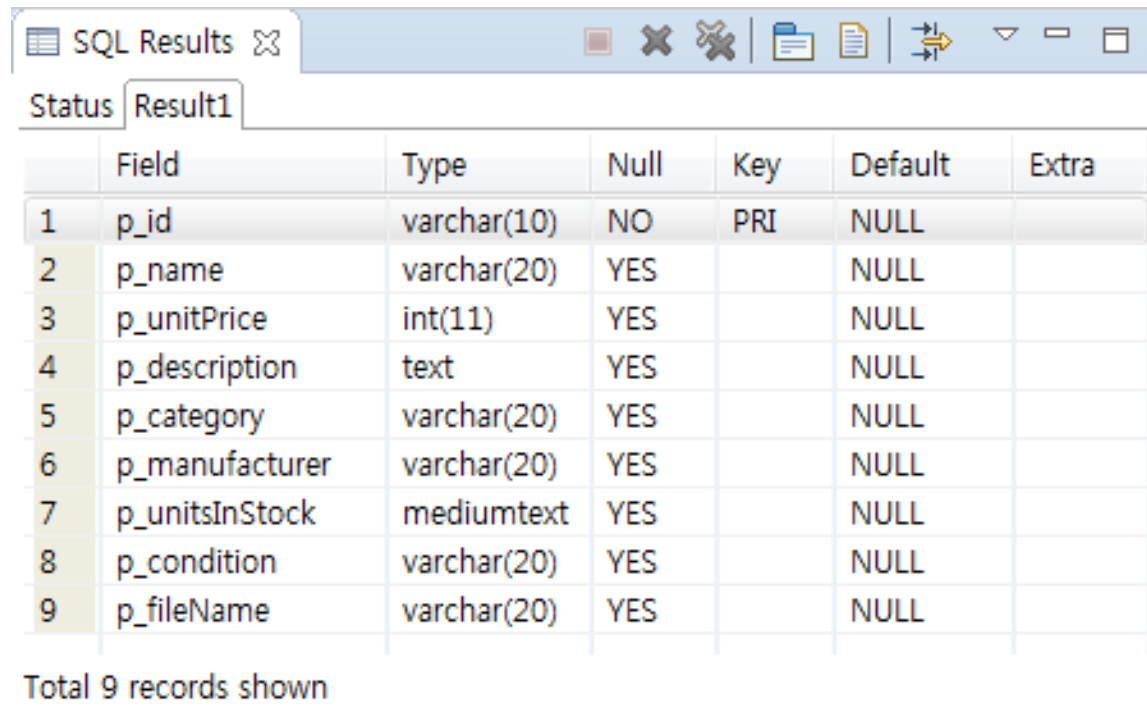
- 상품 관리 테이블 생성하기:
 - /WebMarket/ 폴더에 sql 폴더를 만든 후 이 폴더에 product.sql 파일 생성하고 다음과 같이 작성

WebMarket/WebContent/resources/sql/product.sql

```
01 CREATE TABLE IF NOT EXISTS product(  
02     p_id VARCHAR(10) NOT NULL,  
03     p_name VARCHAR(10),  
04     p_unitPrice INTEGER,  
05     p_description TEXT,  
06     p_category VARCHAR(20),  
07     p_manufacturer VARCHAR(20),  
08     p_unitsInStock LONG,  
09     p_condition VARCHAR(20),  
10     p_fileName VARCHAR(20),  
11     PRIMARY KEY (p_id)  
12 )default CHARSET=utf8;
```

5. [웹 쇼핑몰] 상품 관리 테이블 생성 및 상품 등록하기

- 실행 결과 확인하기:
 - Data Source Explorer 뷰에서 실행 결과 확인



The screenshot shows a 'SQL Results' window with a toolbar and tabs. The 'Result1' tab is active, displaying a table structure with 9 columns: Field, Type, Null, Key, Default, and Extra. The table contains 9 rows of data, each representing a field in the product management table. The fields are p_id, p_name, p_unitPrice, p_description, p_category, p_manufacturer, p_unitsInStock, p_condition, and p_fileName. The p_id field is marked as the primary key (PRI) and is not nullable (NO). All other fields are nullable (YES) and have a default value of NULL. The status bar at the bottom indicates 'Total 9 records shown'.

	Field	Type	Null	Key	Default	Extra
1	p_id	varchar(10)	NO	PRI	NULL	
2	p_name	varchar(20)	YES		NULL	
3	p_unitPrice	int(11)	YES		NULL	
4	p_description	text	YES		NULL	
5	p_category	varchar(20)	YES		NULL	
6	p_manufacturer	varchar(20)	YES		NULL	
7	p_unitsInStock	mediumtext	YES		NULL	
8	p_condition	varchar(20)	YES		NULL	
9	p_fileName	varchar(20)	YES		NULL	

Total 9 records shown

5. [웹 쇼핑몰] 상품 관리 테이블 생성 및 상품 등록하기

예제 15-12 웹 쇼핑몰의 상품 등록하기

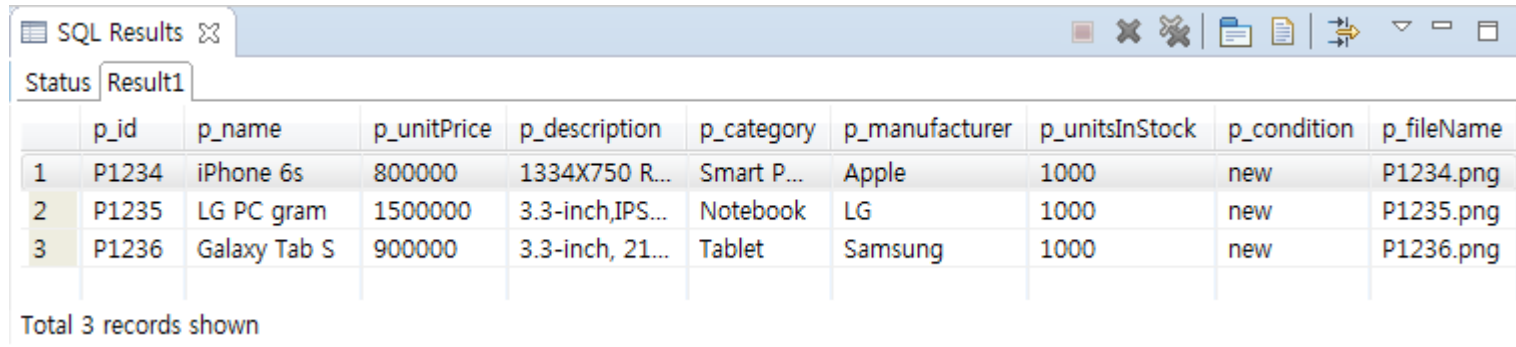
■ 상품 목록 삽입하기

WebMarket/WebContent/resources/sql/insert.sql

```
01 INSERT INTO product VALUES('P1234', 'iPhone 6s', 800000, '1334X750 Renina
    HD display, 8-megapixel iSight
02 Camera','Smart Phone', 'Apple', 1000, 'new', 'P1234.png');
03 INSERT INTO product VALUES('P1235', 'LG PC gram', 1500000, '3.3-inch,IPS
    LED display, 5rd Generation Intel
04 Core processors', 'Notebook', 'LG', 1000, 'new', 'P1235.png');
05 INSERT INTO product VALUES('P1236', 'Galaxy Tab S', 900000, '3.3-inch,
    212.8*125.6*6.6mm, Super AMOLED
06 display, Octa-Core processor', 'Tablet', 'Samsung', 1000, 'new', 'P1236.png');
```

5. [웹 쇼핑몰] 상품 관리 테이블 생성 및 상품 등록하기

■ 실행 결과 확인하기



The screenshot shows a web application window titled "SQL Results". It contains a table with 10 columns: p_id, p_name, p_unitPrice, p_description, p_category, p_manufacturer, p_unitsInStock, p_condition, and p_fileName. There are 3 rows of data. The first row is for an iPhone 6s, the second for an LG PC gram, and the third for a Galaxy Tab S. The status bar at the bottom indicates "Total 3 records shown".

	p_id	p_name	p_unitPrice	p_description	p_category	p_manufacturer	p_unitsInStock	p_condition	p_fileName
1	P1234	iPhone 6s	800000	1334X750 R...	Smart P...	Apple	1000	new	P1234.png
2	P1235	LG PC gram	1500000	3.3-inch,IPS...	Notebook	LG	1000	new	P1235.png
3	P1236	Galaxy Tab S	900000	3.3-inch, 21...	Tablet	Samsung	1000	new	P1236.png

Total 3 records shown

