

데이터베이스 기초

■ 관계형 데이터베이스 테이블의 예

학교 테이블

학교 번호	이름
S001	한국대학교
S002	대한대학교
S003	미국대학교

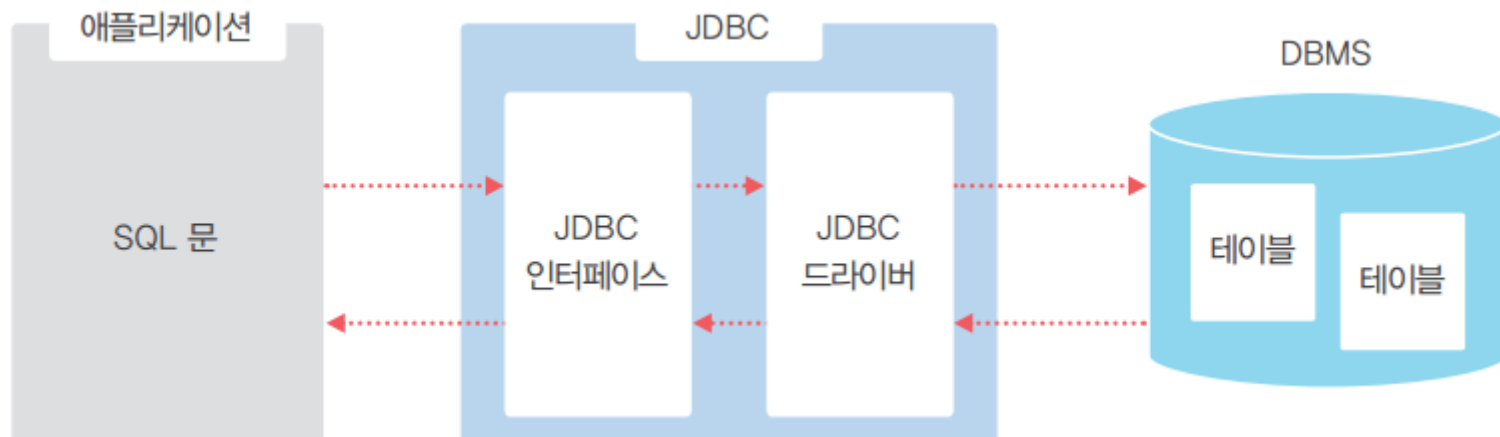
학생 테이블

학교 번호	이름	학교 번호	나이
S001	김선달	S001	32
S002	배장화	S002	20
S003	연홍부	S003	28
S004	임격정	S004	29
S005	홍길동	S005	25

데이터베이스 기초

■ JDBC 드라이버

- DBMS는 종류가 다양하며, 그 구조와 특징도 다 다르다. 따라서 자바 표준 API를 확장해 DBMS에 접근한다는 것은 자바의 기본 개념에 맞지 않을 뿐만 아니라 개발자에게도 고통
- 자바는 모든 DBMS에서 공통으로 사용할 수 있는 인터페이스와 클래스로 구성된 JDBC를 제공
- 각 DBMS 벤더는 JDBC를 구현한 JDBC드라이버를 개발자에게 제공
- JDBC가 데이터베이스에 접근하는 과정



데이터베이스 기초

■ SQL 명령어

- SQL 명령어는 반드시 세미콜론(;)으로 종료

SQL 명령어	설명	
DDL	CREATE	테이블을 생성한다.
	DROP	테이블을 삭제한다.
	ALTER	테이블 구조를 변경한다.
	USE	사용할 데이터베이스를 지정한다.
DML	INSERT	데이터를 추가한다.
	UPDATE	데이터를 갱신한다.
	DELETE	데이터를 삭제한다.
	SELECT	데이터를 조회한다.
DCL	GRANT	테이블에 권한을 부여한다.

MySQL 설치 및 응용

■ MySQL 다운로드 및 설치

- MySQL 다운로드하기(<http://dev.mysql.com/downloads/mysql/>)

The screenshot displays the MySQL download page for Windows. It features two main sections: 'MySQL Community Server 8.0.19' and 'MySQL Installer 8.0.19'. The 'MySQL Installer 8.0.19' section lists two download options for Windows (x86, 32-bit): 'MySQL Installer MSI' (18.6M) and 'MySQL Installer MSI' (398.9M). Both options have a 'Download' button. A red box highlights the 'Download' button for the 18.6M version, with a red circle containing the number '2' next to it. Below the download options, there is a section for 'Login' and 'Sign Up' buttons, with a red box highlighting the 'No thanks, just start my download.' link, marked with a red circle containing the number '3'. A red box with the number '1' highlights the 'Go to Download Page >' button.

General Availability (GA) Releases Archives

MySQL Community Server 8.0.19

Select Operating System:
Microsoft Windows

Recommended Download:

MySQL Installer for Windows

All MySQL Products. For All Windows Platforms. In One Package.

Starting with MySQL 5.6 the MySQL installer package replaces the standalone MSU packages.

Windows (x86, 32 & 64-bit), MySQL Installer MSI

1 Go to Download Page >

2

Operating System	Version	Size	Download
Windows (x86, 32-bit), MSI Installer	8.0.19	18.6M	Download
MD5: 32943776cb2239db451dda86dc0ad51 Signature			
Windows (x86, 32-bit), MSI Installer	8.0.19	398.9M	Download
MD5: 1a882015da71b93f20c4717e63b6817c Signature			

3

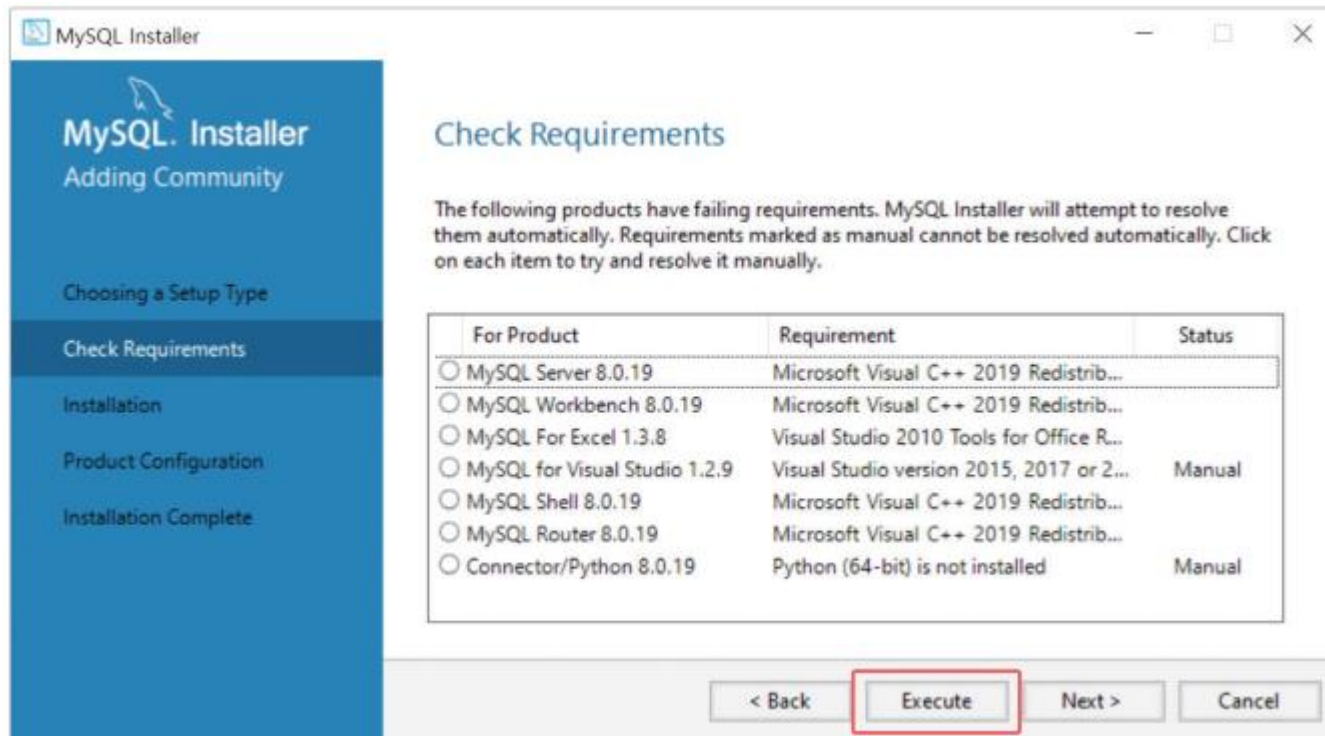
No thanks, just start my download.

- Microsoft Visual C++ 20xx redistributable이 설치 필요

MySQL 설치 및 응용

■ MySQL 다운로드 및 설치

- MySQL 설치하기 : 다양한 설정이 있지만, 여기서는 라이선스에 동의한 후 기본 설정을 그대로 두고 [Execute] 버튼이나 [Next] 버튼을 클릭해 설치하면 큰 무리가 없다. 특히, 다음 화면이 나타나면 [Execute] 버튼을 클릭하여 MySQL 설치에 필요한 소프트웨어 설치 필요



MySQL 설치 및 응용

■ MySQL 다운로드 및 설치

- 사용자 계정 설정 및 설치 완료 : 루트 비밀번호(편의상 'password')를 입력. 계속해서 단계별 [Finish] 과정을 여러 번 반복하게 되며 마지막으로 MySQL Workbench 환영 창이 나타나면 설치가 완료

■ MySQL 드라이버 추출

- 정상적으로 설치하면 'C:\Program Files (x86)\MySQL\Connector J 버전' 폴더에 MySQL 드라이버((mysql-connector-java-버전번호.jar)가 있음
- MySQL 드라이버 파일을 임의 폴더(여기서는 C:\Program Files\MySQL\로 이동

MySQL 설치 및 응용

■ MySQL 드라이버 설정하기

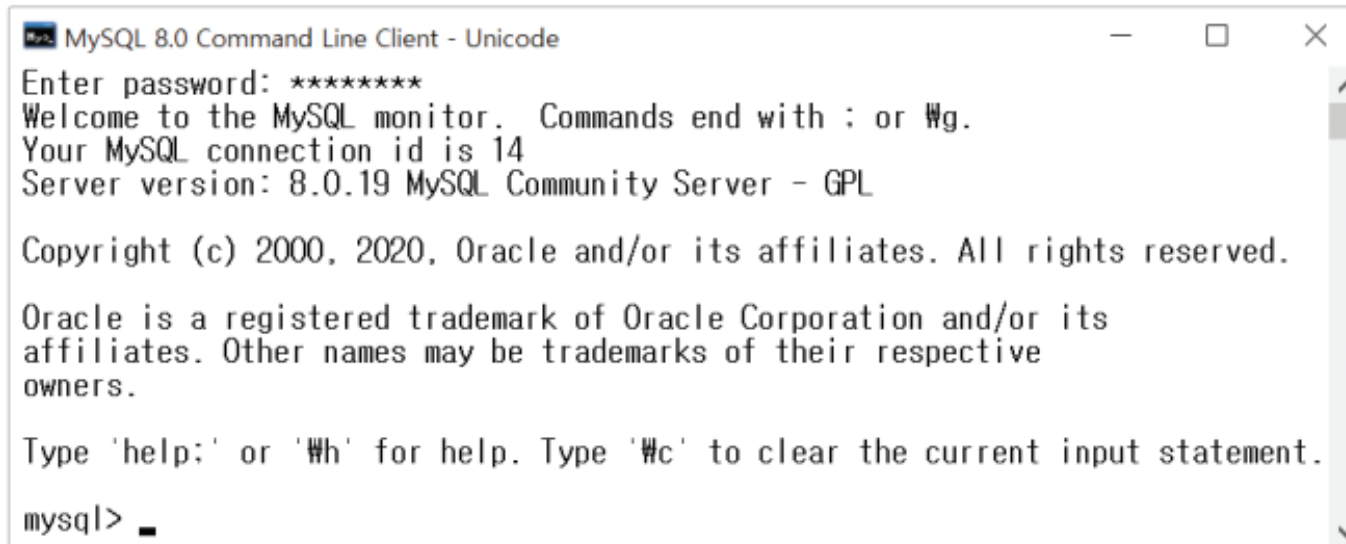
The following steps illustrate the process of adding the MySQL driver to the Eclipse IDE:

1. In the **Properties for chap18** dialog, select **Java Build Path**.
2. Click the **Libraries** tab.
3. Click the **Add External JARs...** button.
4. In the **JAR Selection** dialog, click the **Add External JARs...** button.
5. Select **mysql-connector-java-8.0.19** in the **JAR Selection** dialog.
6. In the **Properties for chap18** dialog, confirm the added JAR under **Modulepath**.
7. Click **Apply and Close** in the **Properties for chap18** dialog.
8. Verify the JAR is listed in the **Referenced Libraries** of the **Package Explorer**.

MySQL 설치 및 응용

■ SQL 명령어

- 대부분의 DBMS는 이런 SQL명령어를 직접 실행할 수 있는 명령 창을 제공



```
MySQL 8.0 Command Line Client - Unicode
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.19 MySQL Community Server - GPL

Copyright (c) 2000, 2020, 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> _
```


MySQL 설치 및 응용

■ SQL 명령어

```
CREATE DATABASE 데이터베이스_이름;
```

```
CREATE TABLE 테이블_이름 (열_이름, 열_타입, ... )
```

```
CREATE DATABASE contacts;
```

contacts 데이터베이스를 생성한다.

```
USE contacts;
```

사용할 데이터베이스로 변경한다.

```
CREATE TABLE person (  
  name VARCHAR(20),  
  phone VARCHAR(20),  
  email VARCHAR(20),  
  PRIMARY KEY(phone));
```

contacts 데이터베이스에
person 테이블을 생성한다.
person 테이블은 name,
phone, email 열로 구성되며
phone을 기본키로 사용한다.

MySQL 설치 및 응용

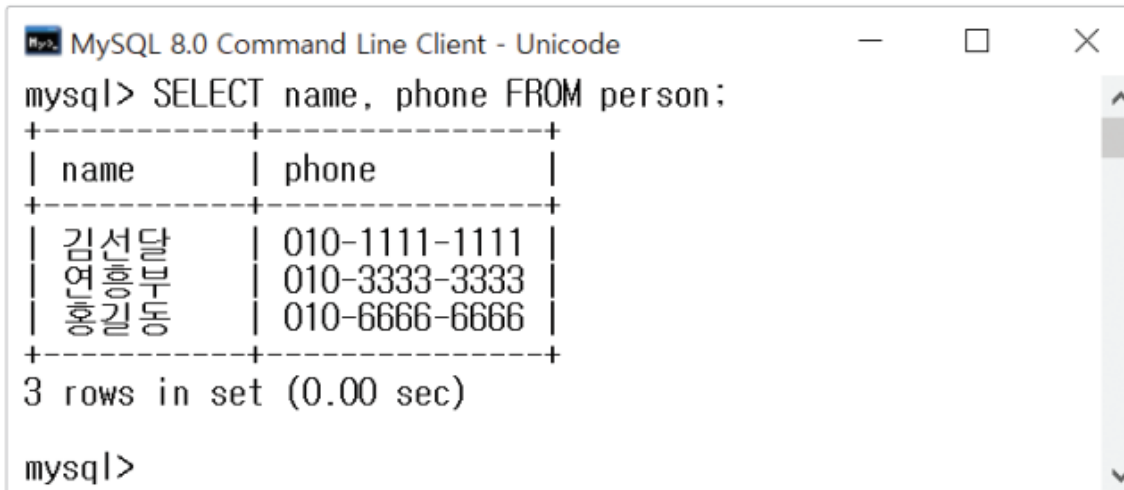
■ SQL 명령어

```
INSERT INTO 테이블_이름 [(열_이름, ...)] VALUES(값, ...);
```

```
INSERT INTO person (name, phone, email) VALUES ('김선달', '010-1111-1111',  
'kim@one.com');
```

```
SELECT 열_이름, ... FROM 테이블_이름 [WHERE 조건] [ORDER BY 정렬 방식];
```

```
SELECT name, phone FROM person;
```



The screenshot shows a terminal window titled "MySQL 8.0 Command Line Client - Unicode". The user has entered the command `mysql> SELECT name, phone FROM person;`. The output is a table with two columns: `name` and `phone`. The table contains three rows of data: `김선달` with phone `010-1111-1111`, `연홍부` with phone `010-3333-3333`, and `홍길동` with phone `010-6666-6666`. Below the table, it says "3 rows in set (0.00 sec)". The prompt `mysql>` is visible at the bottom.

```
MySQL 8.0 Command Line Client - Unicode
mysql> SELECT name, phone FROM person;
+-----+-----+
| name  | phone |
+-----+-----+
| 김선달 | 010-1111-1111 |
| 연홍부 | 010-3333-3333 |
| 홍길동 | 010-6666-6666 |
+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

MySQL 설치 및 응용

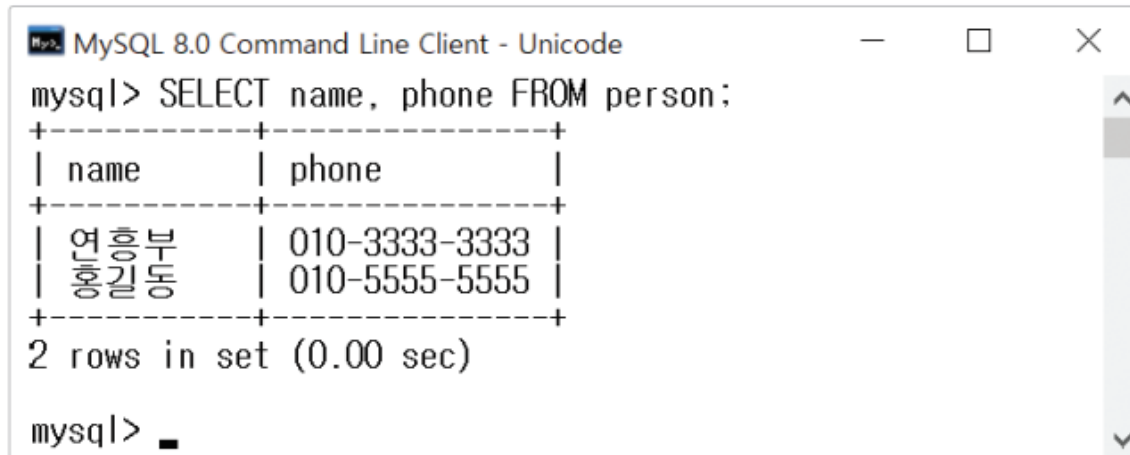
■ SQL 명령어

DELETE FROM 테이블_이름 [WHERE 조건];

DELETE FROM person WHERE name='김선달';

UPDATE 테이블_이름 SET 열_이름=값, ... [WHERE 조건];

UPDATE person SET phone='010-5555-5555' WHERE name='홍길동';



The screenshot shows a terminal window titled "MySQL 8.0 Command Line Client - Unicode". The user has entered the command `mysql> SELECT name, phone FROM person;`. The output is a table with two columns: `name` and `phone`. There are two rows of data: `연홍부` with phone `010-3333-3333`, and `홍길동` with phone `010-5555-5555`. Below the table, it says "2 rows in set (0.00 sec)". The prompt `mysql>` is followed by a cursor.

```
MySQL 8.0 Command Line Client - Unicode
mysql> SELECT name, phone FROM person;
+-----+-----+
| name   | phone   |
+-----+-----+
| 연홍부  | 010-3333-3333 |
| 홍길동  | 010-5555-5555 |
+-----+-----+
2 rows in set (0.00 sec)

mysql> █
```

JDBC 프로그래밍

■ JDBC 프로그래밍 과정

- ① JDBC드라이버 로드 및 데이터베이스를 연결한다.
- ② SQL문을 실행한다.
- ③ ResultSet 객체를 처리한다.
- ④ 자원 객체를 닫는다.

JDBC 프로그래밍

■ JDBC드라이버 로드 및 데이터베이스 연결

- MySQL 드라이버의 경우 다음 실행문으로 드라이버를 메모리에 로드
mysql-connector-java 6.0이전: **Class.forName("com.mysql.jdbc.Driver");**
mysql-connector-java 6.0부터: **Class.forName("com.mysql.cj.jdbc.Driver");**
- 데이터베이스와 연결해 Connection 객체 생성

```
Connection con = DriverManager.getConnection(url, user_id, passwd);
```

- 여기서 url은 다음 형식: jdbc:subprotocol:subname
- mysql-connector-java 6.0부터는 서버 타임 존이 필요
jdbc:mysql://IP주소/DB이름 (mysql-connector-java 6.0이전)
jdbc:mysql://IP주소/DB이름?serverTimezone=Asia/Seoul (mysql-connector-java 6.0부터)

- 예제 : [sec05/DB1Demo](#)

```
데이터베이스 연결 중...
데이터베이스 연결 성공
```

JDBC 프로그래밍

■ SQL 문 실행

- SQL문을 DBMS에 전송할 수 있는 Statement 객체 생성

```
Statement stmt = con.createStatement();
```

- Statement 인터페이스가 제공하는 주요 메서드

메서드	설명
boolean execute(String sql)	SQL 문을 수행한다. 실행 결과가 ResultSet 객체라면 true, 아니면 false를 반환한다.
ResultSet executeQuery(String sql)	SQL 문을 수행한다. ResultSet 객체를 반환한다.
int executeUpdate(String sql)	SQL 문을 수행한다. 영향을 받은 레코드의 개수를 반환한다.

- 예제 : [sec05/DB2Demo](#)

```
-
데이터베이스 연결 중...
데이터베이스 연결 성공
레코드 추가 성공
```

JDBC 프로그래밍

■ SQL 문 실행

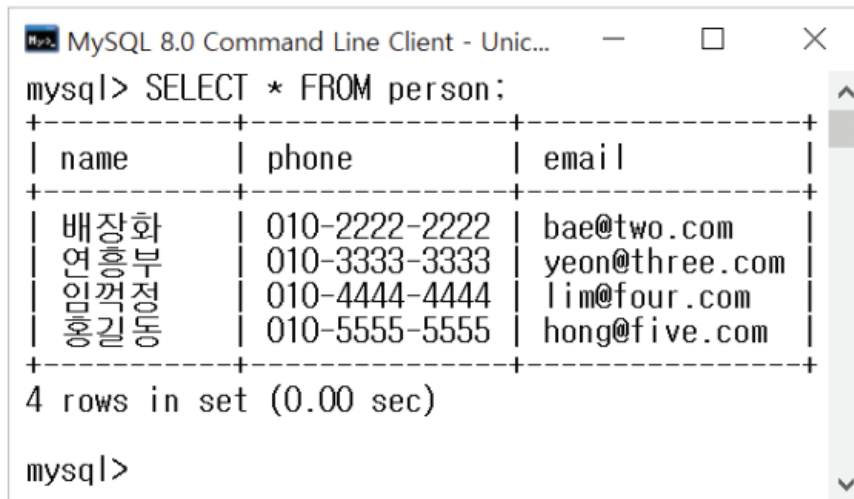
- PreparedStatement 인터페이스
 - Statement의 자식 인터페이스
 - SQL문을 미리 만들어 두고 사용할 수 있기 때문에 효율성이나 유지 보수 면에서 유리

```
PreparedStatement pstmt =  
    con.prepareStatement("UPDATE EMP SET SALARY = ? WHERE ID = ?");  
  
pstmt.setBigDecimal(1, 153833.00)  
  
pstmt.setInt(2, 110592)  
  
pstmt.executeUpdate();
```

JDBC 프로그래밍

■ SQL 문 실행

- PreparedStatement 인터페이스
- 예제 : [sec05/DB3Demo](#)



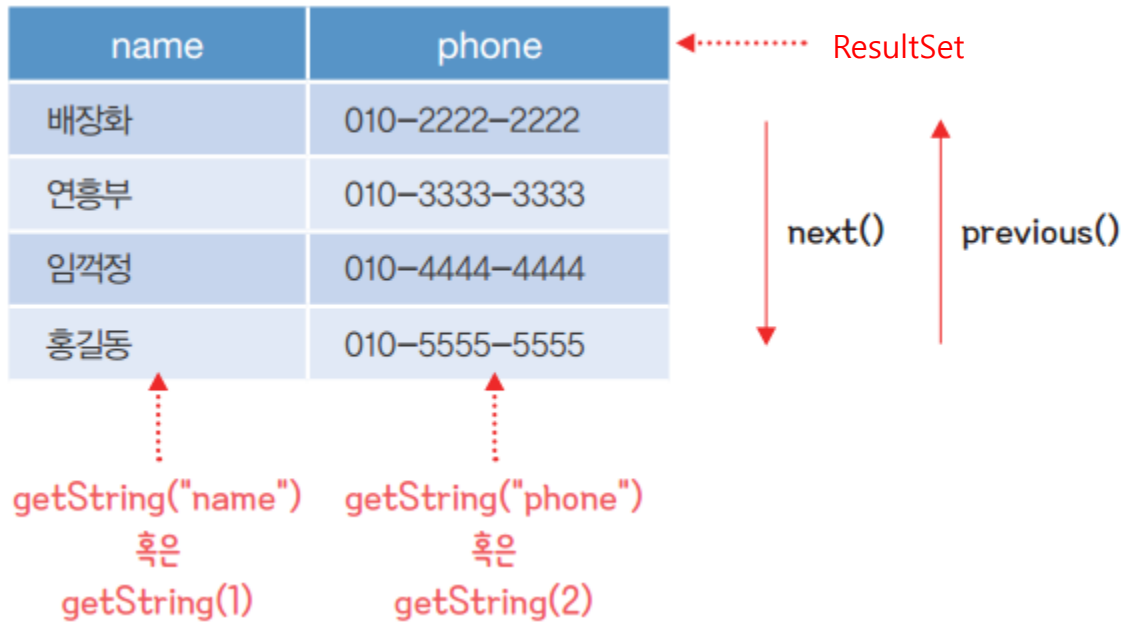
```
MySQL 8.0 Command Line Client - Unic...
mysql> SELECT * FROM person;
+-----+-----+-----+
| name      | phone      | email      |
+-----+-----+-----+
| 배장화     | 010-2222-2222 | bae@two.com |
| 연홍부     | 010-3333-3333 | yeon@three.com |
| 임걱정     | 010-4444-4444 | lim@four.com |
| 홍길동     | 010-5555-5555 | hong@five.com |
+-----+-----+-----+
4 rows in set (0.00 sec)

mysql>
```


JDBC 프로그래밍

■ 결과집합의 처리

- ResultSet은 SQL질의문의 결과물을 추상화한 인터페이스
- 예를 들어 SELECT name, phone FROM person;의 결과집합



JDBC 프로그래밍

■ 결과집합의 처리

- ResultSet 인터페이스가 제공하는 주요 메서드

메서드	설명
void afterLast()	커서를 마지막 행 다음으로 이동한다.
void beforeFirst()	커서를 첫 행 이전으로 이동한다.
void close()	ResultSet 객체를 닫는다.
boolean first()	커서를 첫 행으로 이동한다.
int getInt(int columnIndex) long getLong(String columnLabel)	현재 레코드에서 주어진 열의 값을 int 혹은 long 타입으로 반환한다.
int getRow()	현재 레코드 번호를 반환한다.
String getString(int columnIndex) String getString(String columnLabel)	현재 레코드에서 주어진 열의 값을 String 타입으로 반환한다.
boolean last()	커서를 마지막 행으로 이동한다.
boolean next()	커서를 다음 행으로 이동한다.
boolean previous()	커서를 이전 행으로 이동한다.

JDBC 프로그래밍

■ 결과집합의 처리

- 예제 : [sec05/DB4Demo](#)

데이터베이스 연결 중...

데이터베이스 연결 성공

name : 배장화 phone : 010-2222-2222

name : 연흥부 phone : 010-3333-3333

name : 임꺽정 phone : 010-4444-4444

name : 홍길동 phone : 010-5555-5555