



# Chapter11

## JDBC로 데이터베이스 연동



# 개요

1

## 학습목표

- ✓ 프로그램에서 데이터베이스를 연동하기 위한 자바의 기술인 **JDBC**에 대해서 알아본다.

2

## 학습내용

- ✓ **JDBC**
- ✓ 데이터베이스 조작을 위한 자바 라이브러리
- ✓ **JSP**와 데이터베이스 연동
- ✓ **ConnectionPool**을 사용한 데이터베이스 연결 기능 향상



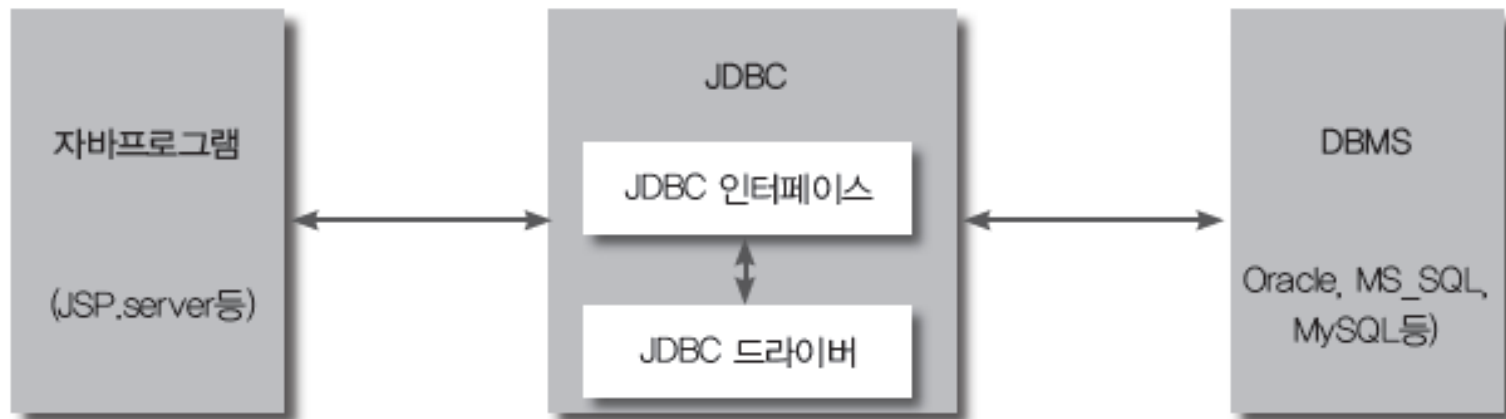
## ❖ JDBC란?

- 데이터베이스를 다루기 위한 자바 API(Application Programming Interface)
- JDBC를 통해서 DBMS의 종류와 관계없이 질의문을 던져서 데이터를 수신
- 각 데이터베이스의 접속에 대한 상세한 정보가 불필요



## ❖ JDBC

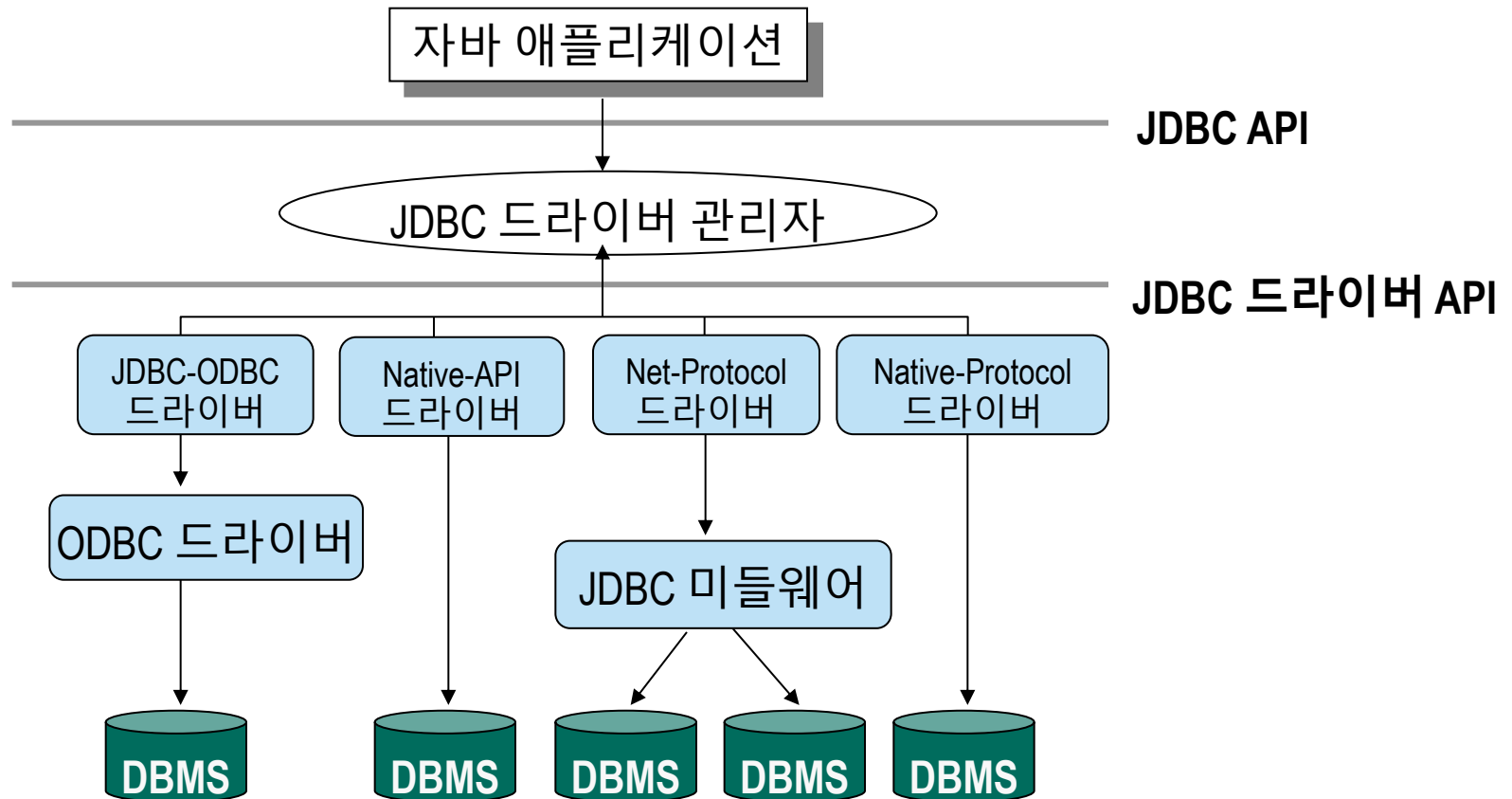
- JDBC 인터페이스 : 프로그래머에게 쉬운 데이터베이스와 연동되는 프로그램을 작성할 수 있게 하는 도구
- JDBC 드라이버 : JDBC 인터페이스를 구현하여 실제로 DBMS를 작동시켜서 질의를 던지고 결과를 받음



▲ [그림 11-1] JDBC 구성



## ❖ JDBC Driver Type







## ❖ JDBC Driver 설치

- MySQL 사이트  
(<http://www.mysql.com/downloads>)
- MySQL Connector/J 5.1.45 ZIP  
for portable Java binaries 다운로드
- 파일 압축 해제 후 이클립스 라이브러리 추가



## ❖ JDBC 를 통한 MySQL과의 연동 테스트

- [source/ch10/DriverTest.java](#)

결과

-----

Success



## ❖ JDBC API (java.sql 패키지)

- Driver : 모든 드라이버 클래스들이 구현해야 하는 인터페이스
- DriverManager : 드라이버를 로드하고 데이터베이스에 연결
- Connection : 특정 데이터베이스와의 연결
- Statement : SQL문을 실행해 작성된 결과를 반환
- PreparedStatement : 사전에 컴파일 된 SQL문을 실행
- ResultSet : SQL문에 대한 결과를 얻어냄





## ❖ JDBC 프로그래밍 단계

### ❶ JDBC 드라이버의 인스턴스 생성

```
Class.forName("Driver_Name");
```



### ❷ JDBC 드라이버 인스턴스를 통해 DBMS에 대한 연결 생성

```
Connection con = DriverManager.getConnection("DBURL","Account ID","Account PW");
```



### ❸ Statement 생성

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



### ❹ 질의문 실행/ResultSet으로 결과 받음

```
ResultSet rs = stmt.executeQuery("select * from ...");
```





## ❖ JDBC 프로그래밍 단계

### ⑤ ResultSet 해지

```
rs.close();
```



### ⑥ Statement 해지

```
stmt.close();
```

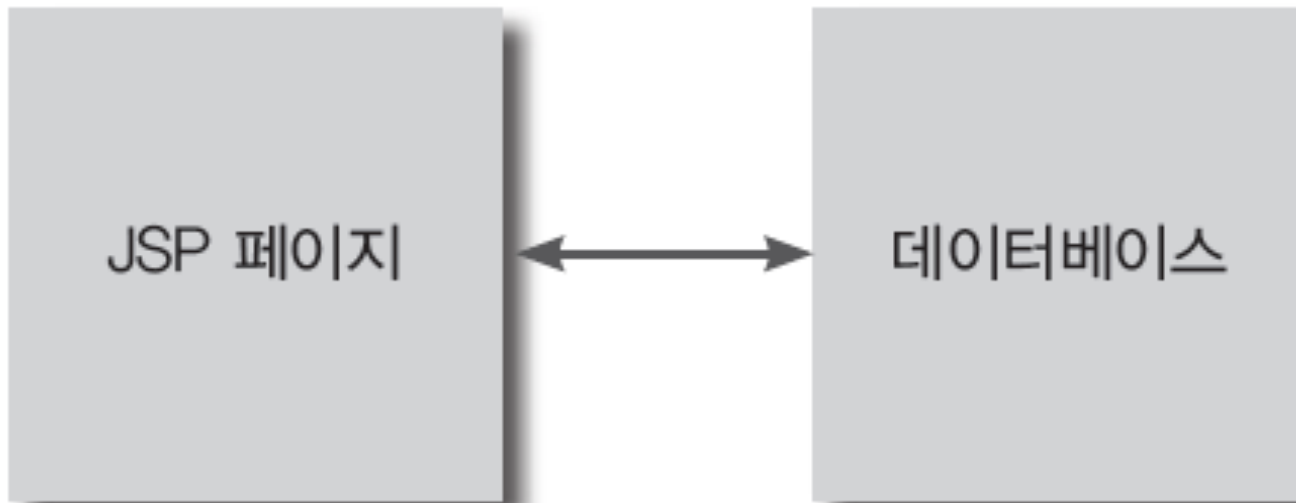


### ⑦ 데이터베이스와 연결 해지

```
conn.close();
```



- ❖ **JSP** 페이지 내에서 직접 데이터베이스 관련 코드가 혼재되어 있어 추후 페이지 변경 등의 작업이 있을 때 유리하지 못함



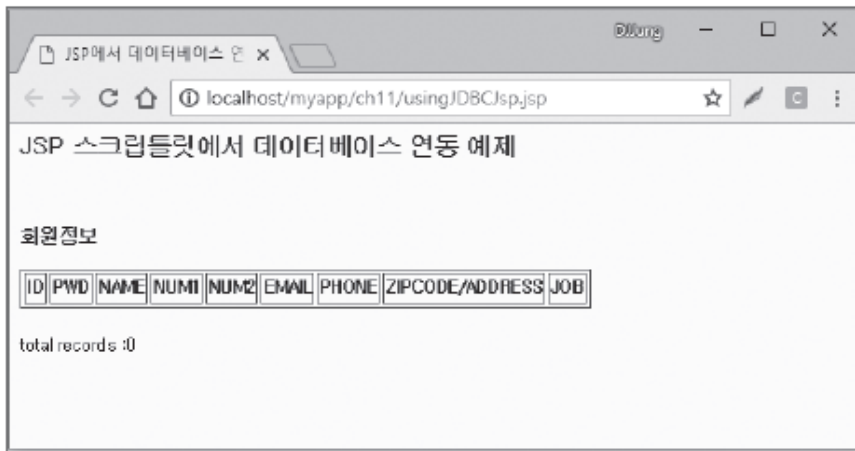
▲ [그림 11-7] JSP와 데이터베이스의 직접적인 연동



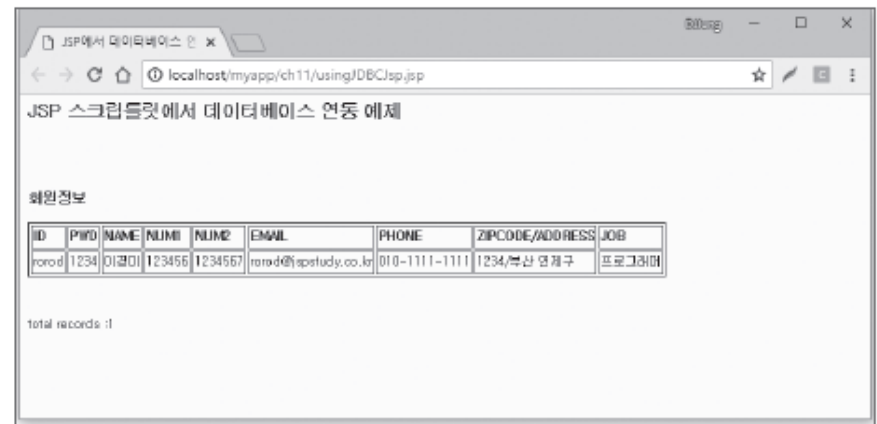
# 데이터베이스 조작을 위한 자바 라이브러리

## ❖ JSP 스크립트릿으로 데이터베이스를 연동하여 member 테이블에 있는 레코드를 조회

- [source/ch11/usingJDBCJsp.jsp](http://source/ch11/usingJDBCJsp.jsp)



▲ [그림 11-8] 테이블에 레코드가 없는 경우의 실행 결과

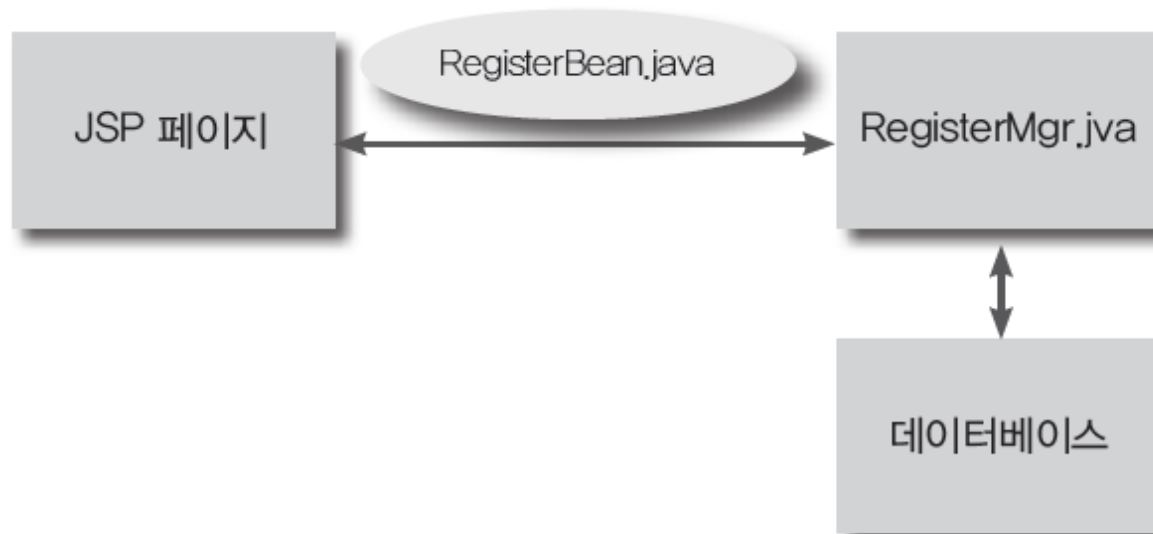


▲ [그림 11-9] 테이블에 레코드가 있는 경우 실행결과



# 데이터베이스 조작을 위한 자바 라이브러리

- ❖ 데이터베이스에 관련된 코드를 JSP 페이지로부터 분리한 예제
- ❖ JDBC 코드들 -> [source/ch11/RegisterMgr.java](#)
- ❖ 레코드들 -> [source/ch11/RegisterBean.java](#)
- ❖ JSP 페이지-> [source/ch11/usingJDBCBean.jsp](#)



▲ [그림 11-11] 자바빈즈를 사용하여 데이터베이스와 연동





# 데이터베이스 조작을 위한 자바 라이브러리

## ❖ 예제

Browser window showing a JSP page titled "Bean를 사용한 데이터베이스 연동 예제". The page displays a table of member information (회원정보).

ID	PASSWD	NAME	NUM1	NUM2	EMAIL	PHONE	ZIPCODE/ADDRESS	JOB
rorod	1234	이경미	123456	1234567	rorod@jspstudy.co.kr	010-1111-1111	1234/부산 연제구	프로그래머

total records : 1



## ❖ ConnectionPool이란?

- 웹 프로그램은 실질적으로 데이터베이스에 연결하고, 질의를 던지고, 결과를 받아오는 부분에서 많은 시간을 소요
- 이러한 과정이 서버에 많은 부하를 준다.
- ConnectionPool 을 사용해서 보다 효율적으로 데이터베이스에 연동



## ❖ Pooling 기법

- 효율적으로 복수의 사용자에게 서비스하기 위해 미리 데이터베이스 연결을 위한 객체들을 생성
- Connection 객체의 재사용
  - 데이터베이스 연결 객체를 매번 생성, 사용, 해제하지 않고 처음 만들어둔 데이터베이스 연결 객체를 계속 사용
- 사용자에게 필요한 응답을 주는데 걸리는 시간을 단축하고 시스템 부하를 줄임



## ❖ Pooling 기법의 DBConnectionMgr

JSP 페이지

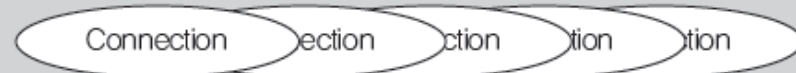
미리 연결된 커넥션을 사용하여 데이터베이스에 연동함.  
jsp 전용 태그 이외에 데이터베이스와 연동하기 위한 자바 코드가 혼재



미리 연결된 커넥션 객체를 사용

DBConnectionMgr

미리 연결된 일정 개수의 커넥션들을 가지고 있다.



DBConnctionMgr에서 생성됨. 미리 연결된 커넥션을 사용한 데이터베이스 연결

데이터베이스

▲ [그림 11-12] jsp 페이지와 DBConnectionMgr를 통한 데이터베이스 연동



## ❖ ConnectionPool을 이용한 데이터베이스 프로그램

- 공개용 풀 관리자

<source/ch11/DBConnectionMgr.java>

- <source/ch11/usingJDBCPOOL.jsp>

JSP 스크립틀릿에서 Connection Pool을 이용한 데이터베이스 연동 예제

회원정보

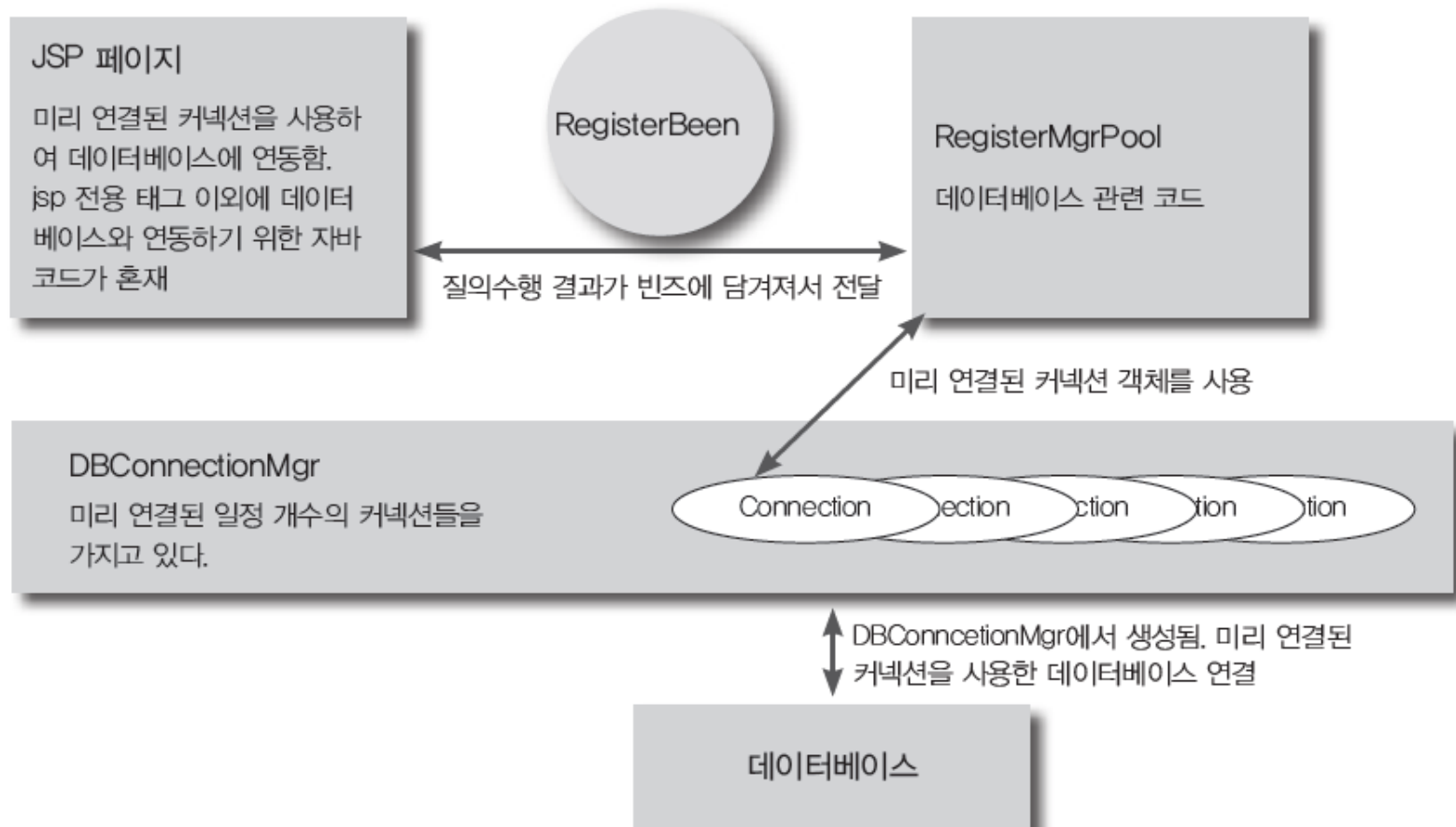
ID	PWD	NAME	NUM1	NUM2	EMAIL	PHONE	ZIPCODE/ADDRESS	JOB
rorod	1234	이경미	123456	1234567	rorod@jspstudy.co.kr	010-1111-1111	1234/부산 연제구	프로그래머

total records : 1





## ❖ ConnectionPool과 Bean을 이용한 데이터베이스 연결





## ❖ ConnectionPool과 Bean을 이용한 데이터베이스 연결

- <source/ch11/RegisterMgrPool.java>
- <source/ch11/usingJDBCPOOLBean.jsp>

Bean과 커넥션 풀을 사용한 데이터베이스 연동 예제

회원정보

ID	PASSWD	NAME	NUM1	NUM2	EMAIL	PHONE	ZIPCODE/ADDRESS	JOB
rorod	1234	이경미	123456	1234567	rorod@jspstudy.co.kr	010-1111-1111	1234/부산 연제구	프로그래머

total records : 1



**Thank You !**