

Module 01

JDBC 소개

NHN Academy



Overview

- JDBC 소개
- JDBC 드라이버 타입
- JDBC 아키텍처
- 연습: Install JDBC Driver

JDBC 소개

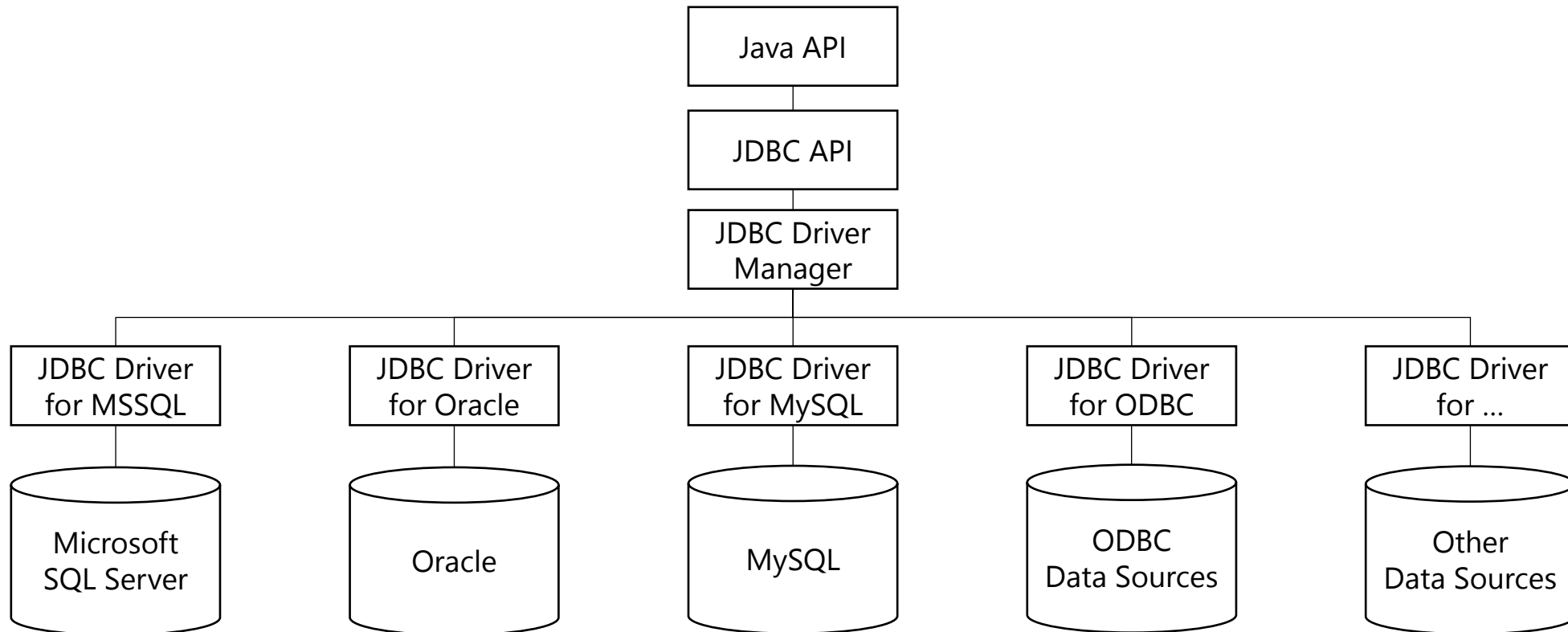
- JDBC
- JDBC 구조
- JDBC 표준
- JDBC 역사

JDBC

- JDBC (Java DataBase Connectivity)
- 관계형 데이터베이스에 저장된 데이터를 접근 및 조작할 수 있게 하는 Java API
 - Java 응용 프로그램이 다양한 DBMS에 대해 일관된 API로 데이터베이스에 연결, 검색, 수정, 관리
 - DBMS의 종류에 관계없이 동일한 방법으로 데이터베이스에 접근

JDBC 구조

- Network상의 데이터베이스에 접속할 수 있도록 하는 Database 연결 기능 제공
- JDBC API, JDBC Driver Manager, JDBC Driver 로 구성



JDBC 구조

구성요소	설명	역할
Java Application	Java 응용 프로그램, Java 웹 응용프로그램 서버 (Tomcat, Weblogic 등)	응용 프로그램 개발자 웹 응용 프로그램 서버 개발사
JDBC Driver Manager	Java 응용 프로그램에서 사용하는 데이터베이스에 맞는 JDBC 드라이버 로드	Java SE 개발사
JDBC API	Java 응용 프로그램에서 데이터베이스에 연결하고 데이터를 제어할 수 있도록 하는 데이터베이스 연결 및 제어를 위한 인터페이스/클래스	Java SE 개발사
JDBC Driver	데이터베이스 개발사에서 만든 데이터베이스 드라이버	데이터베이스 개발사

JDBC 표준

- JDBC 표준 (Oracle)

https://download.oracle.com/otndocs/jcp/jdbc-4_2-mrel2-spec/

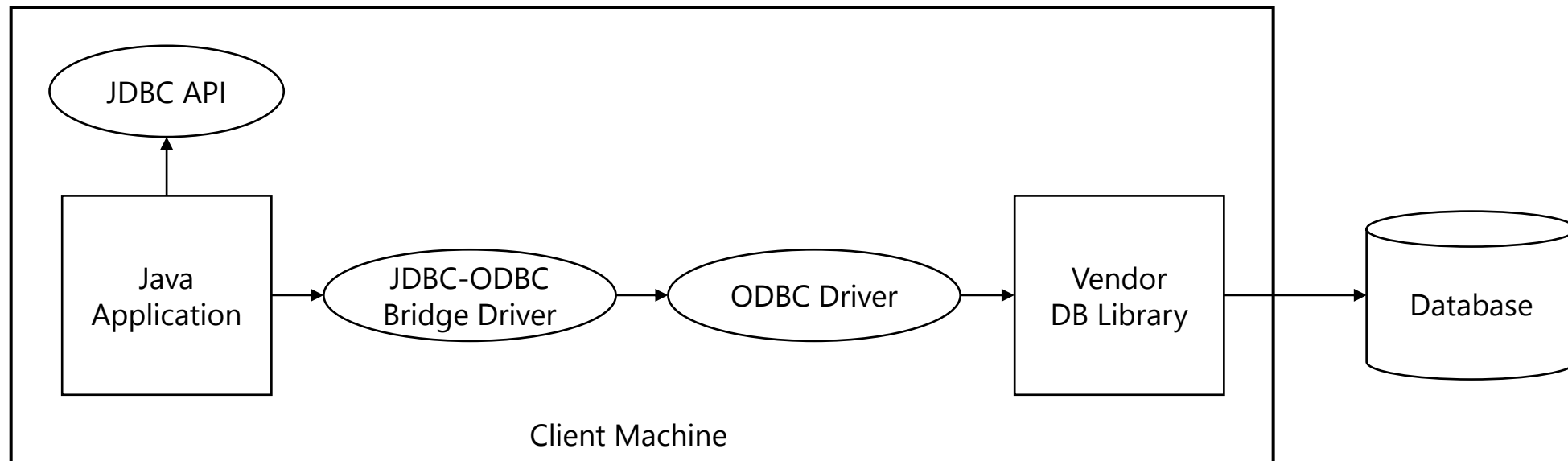
- JDBC는 Java 프로그래밍 언어와 다양한 데이터베이스의 독립적 연결을 위한 표준
- JDBC API는 SQL 기반 데이터베이스 액세스 API를 제공
- JDBC 기술을 사용하면 Java 프로그래밍 언어를 사용하여 엔터프라이즈 데이터에 액세스 해야 하는 응용 프로그램에 대해 WORE 기능을 활용할 수 있음

JDBC Driver Type

- JDBC-ODBC Bridge Type
- Native_API / Partly Java Type
- Net-Protocol / All-Java Type
- Native-Protocol / All-Java Type

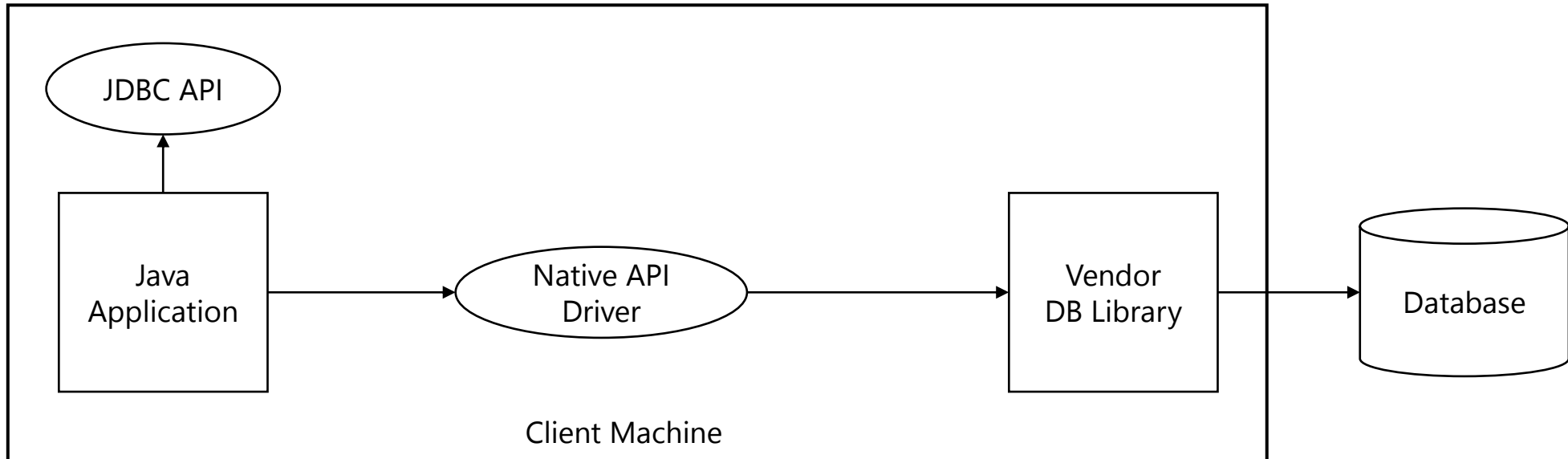
Type 1: JDBC-ODBC Bridge

- JDBC와 ODBC 드라이버 간의 다리 역할 수행
 - ODBC(Open DataBase Connectivity)
 - Microsoft가 만든 DBMS 연결 표준
- JDBC 호출을 ODBC 호출로 변환하고 요청을 ODBC 드라이버에 전송
- 설정이 간편하지만 실행 속도가 느림



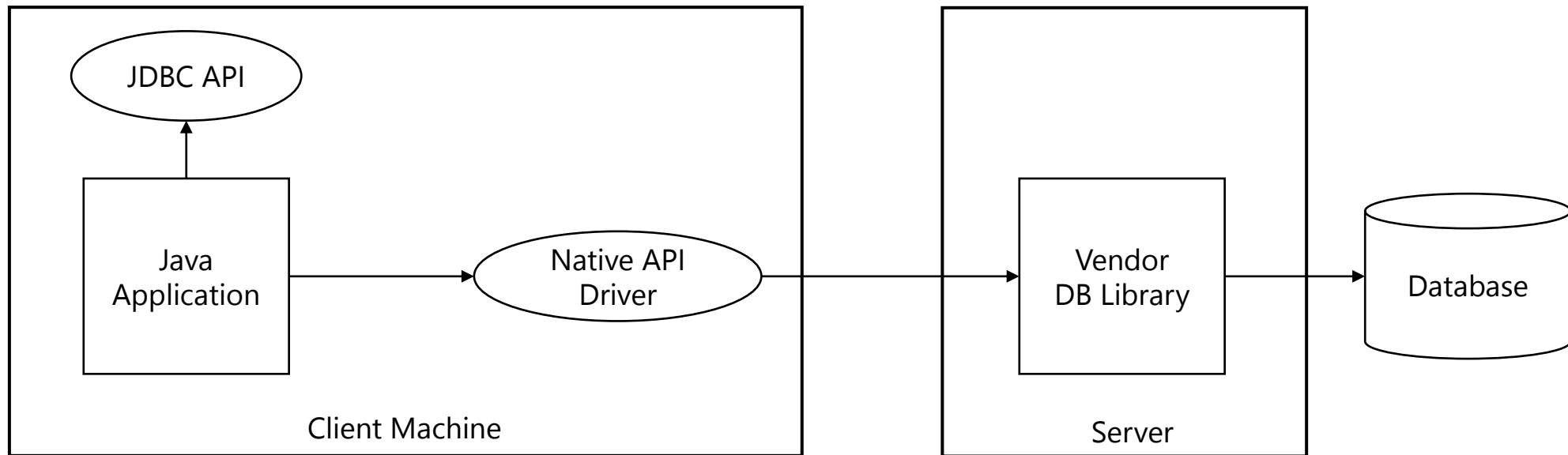
Type 2: Native API / Partly Java

- JNI(Java Native Interface)를 사용해 데이터베이스 전용 클라이언트 호출
- Type 1보다 빠르지만, Native Libraray를 설치해야 함
- 데이터베이스에 종속적



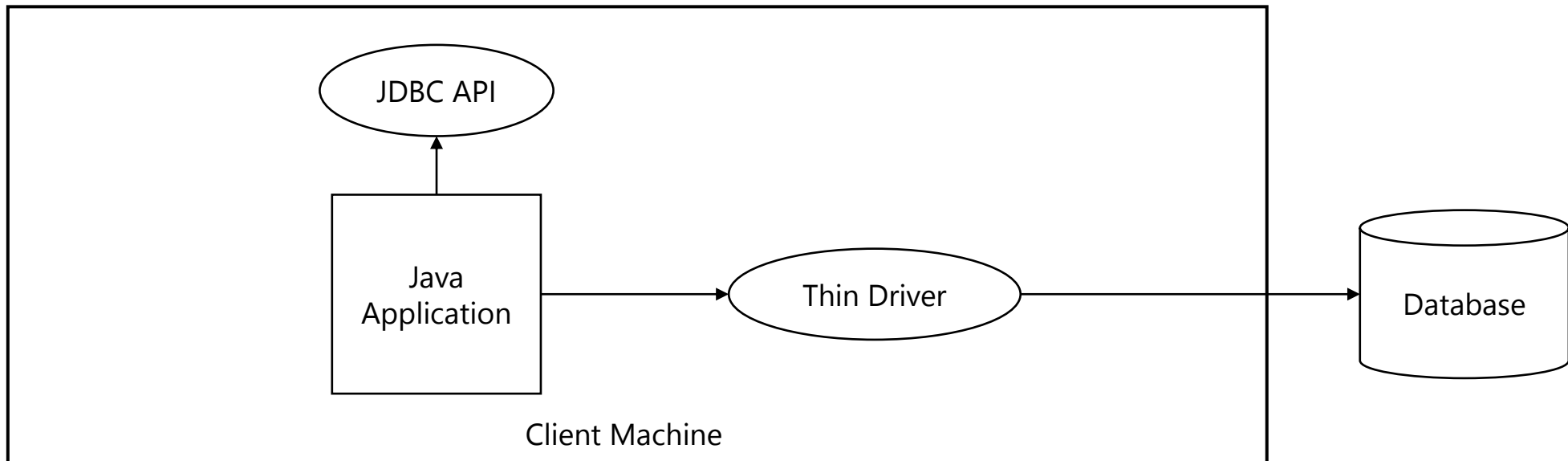
Type 3: Net-Protocol / All-Java

- JDBC 미들웨어 서버와 독점 프로토콜로 통신
- JDBC 미들웨어가 요청된 프로토콜을 데이터베이스 호출로 변환
- 데이터베이스에 독립적
- 많은 네트워크 호출로 인해 상대적으로 느림



Type 4: Native-Protocol / All-Java

- 가장 많이 사용되고 있는 형태
- 직접 데이터베이스와 통신, 순수 Java Driver라고 불림
- Native Library 및 미들웨어 서버가 필요하지 않음
- 가장 좋은 성능

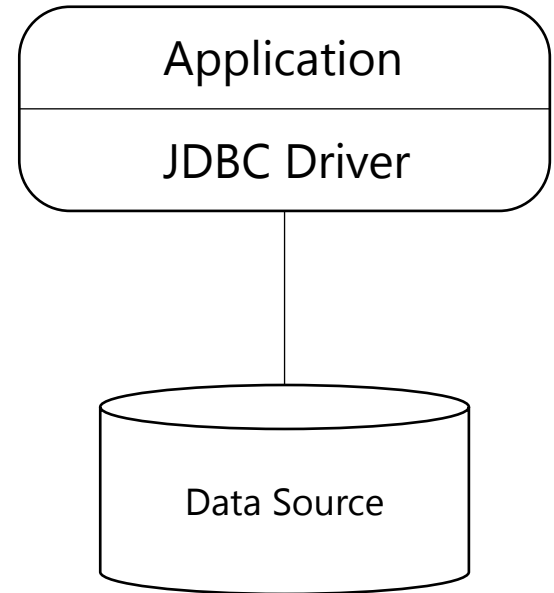


JDBC 아키텍처

- 2-Tier 아키텍처
- 3-Tier 아키텍처

2-Tier 아키텍처

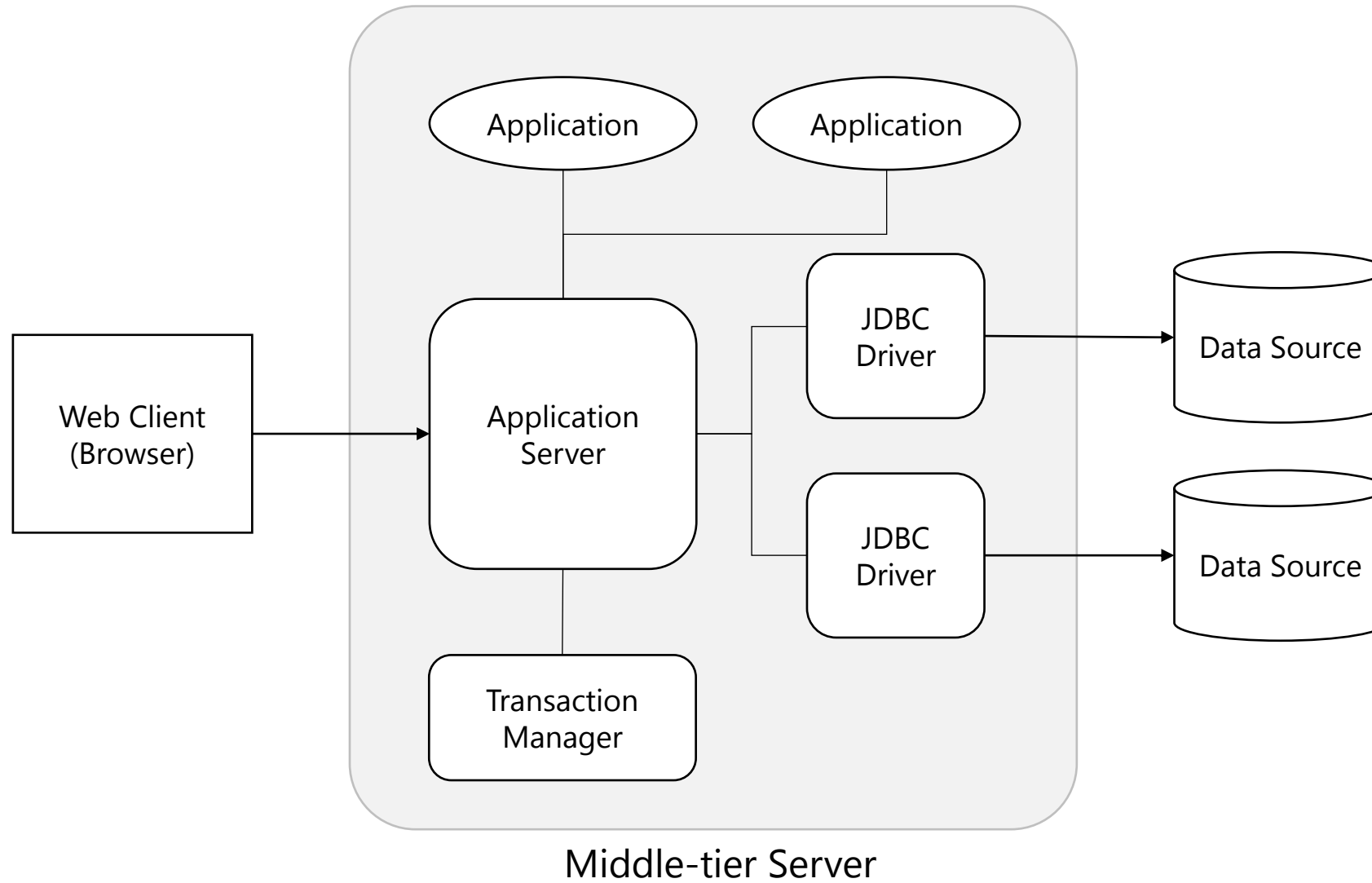
- Client Layer와 Server Layer로 구성
- Java 응용 프로그램이 데이터베이스와 직접 통신하는데 도움이 됨
- JDBC 드라이버를 사용하여 특정 데이터베이스와 통신
- 쿼리 또는 요청은 사용자가 데이터베이스로 전송하고 결과를 사용자에게 의해 수신
- 데이터베이스는 동일한 컴퓨터 또는 네트워크 상의 원격 시스템이 있을 수 있음
- 클라이언트 프로그램이 바로 데이터베이스에 접속하는 구성 – 확장성에 제한
- Client – Server 아키텍처 (또는 구성)



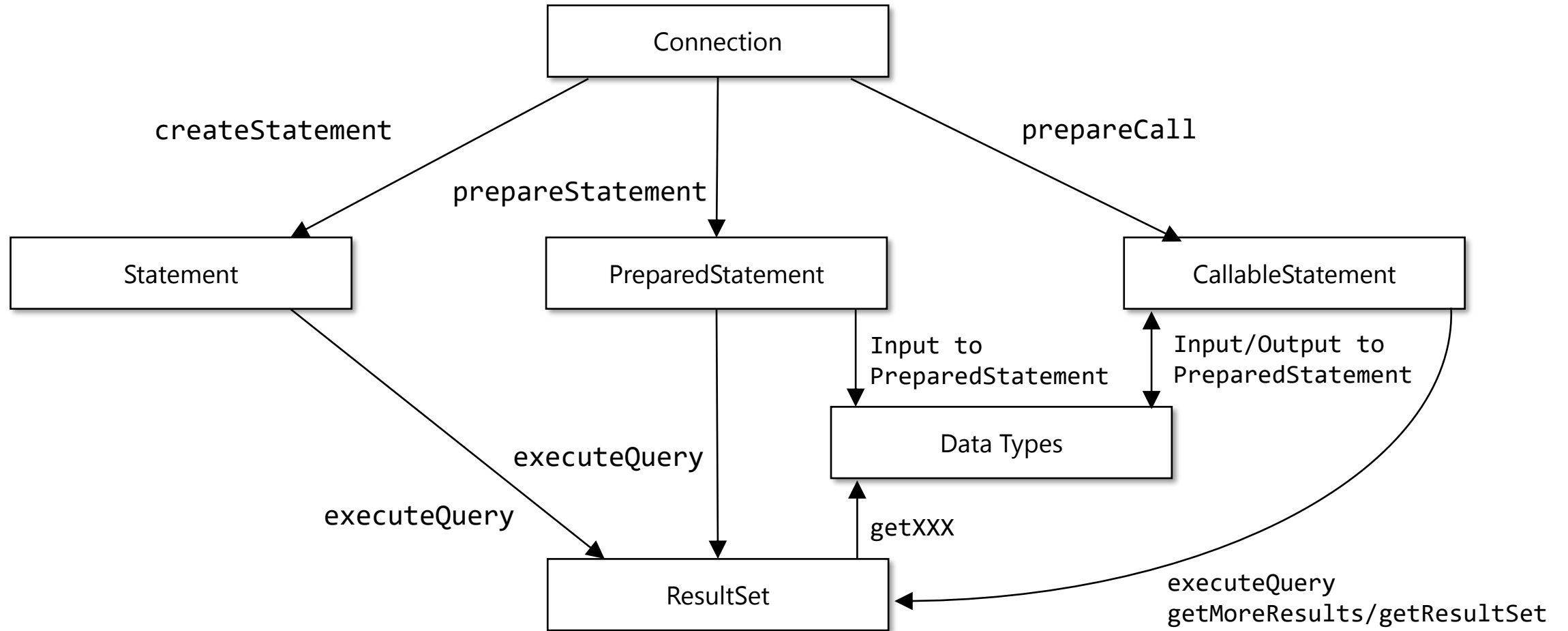
3-Tier 아키텍처

- Business Logic을 포함하는 미들웨어가 추가됨
 - Client에서 데이터베이스로의 직접적인 통신은 없음
- 사용자의 요청은 미들웨어로 전송됨
- 데이터베이스는 요청을 처리하고 결과를 중간 계층으로 보낸 다음 사용자와 통신
- 성능을 높이고 응용 프로그램 배포를 단순화
- Client Tier
 - 사용자와 상호작용하는 화면들로만 구성된 Thin Layer
 - Java 프로그램, 웹 브라우저, PDA, 스마트 폰 등
- Middle Tier Server
 - 응용 프로그램은 Client와 상호작용하는 비즈니스 로직 구현체
 - 응용 프로그램 서버는 다양한 응용 시스템 구현을 위한 기반 기능 제공
 - 일반적으로 Tomcat과 같은 Java EE 서버로, 서버에서 JDBC 드라이버를 직접적으로 사용

3-Tier 아키텍처



java.sql 패키지



JDBC 드라이버

- MySQL

- com.mysql.cj.jdbc.Driver

- Database URL:

`jdbc:mysql://<Server Name or IP>/<DatabaseName>`

- MSSQL

- com.microsoft.sqlserver.jdbc.SQLServerDriver

- Database URL:

`jdbc:sqlserver://<ServerName or IP>;encrypt=true;trustServerCertificate=true;databaseName=<DatabaseName>`

Install JDBC Driver

Install and check JDBC Drivers for MySQL, MSSQL and Oracle



검토

- JDBC 개요
- JDBC Driver Type
- JDBC 아키텍처
- java.sql 패키지