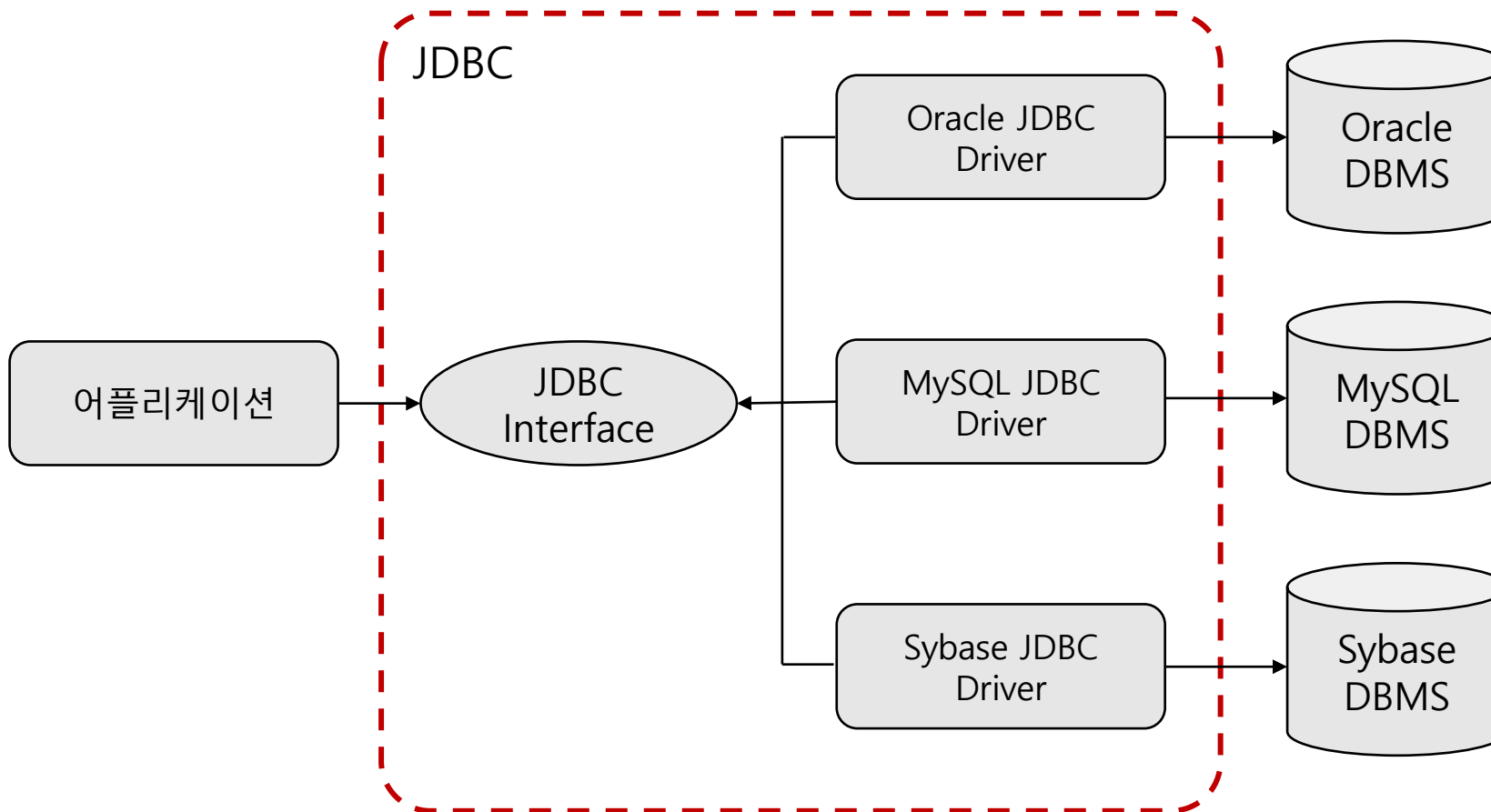




JDBC

▶ JDBC(Java DataBase Connectivity)

자바에서 데이터베이스에 접근할 수 있게 해주는 Programming API







▶ OJDBC 다운로드

✓ 홈페이지 다운로드

<https://www.oracle.com/kr/database/technologies/appdev/jdbc-downloads.html>

Oracle Database 23ai (23.4.0.24.05) JDBC Driver & UCP Downloads

Supports Oracle Database versions - 23ai, 21c, and 19c Refer to [JDBC-UCP-ReleaseNotes-23ai.txt](#) and [Bugs-fixed-in-23ai.txt](#)

Name	Download	JDK Supported	Description
Oracle JDBC driver	 ojdbc11.jar	Implements JDBC 4.3 spec and certified with JDK11, JDK17, JDK19, and JDK21	Oracle JDBC driver except classes for NLS support in Oracle Object and Collection types. (7,170,864 bytes) - (SHA1: 239c6f65f74ebdf08bf0da234815fc757a13224c)
Oracle JDBC Driver	 ojdbc8.jar	Implements JDBC 4.2 spec and certified with JDK8 and JDK11	Oracle JDBC driver except classes for NLS support in Oracle Object and Collection types. (7,056,235 bytes) - (SHA1: 7b80db96f6e8d5d26a7a13bcf282a5433e898348)
Universal Connection Pool (UCP) - ucp11.jar	 ucp11.jar	Certified with JDK11, JDK17, JDK19, and JDK21	Universal Connection Pool (UCP) to be used with ojdbc11.jar (1,533,329 bytes) - (SHA1: eacebf2a63a742c3dffaf169e0f4b75bb66c828e)
Universal Connection Pool (UCP)	 ucp.jar	Certified with JDK8	Universal Connection Pool (UCP) to be used with ojdbc8.jar (1,490,576 bytes) - (SHA1: 28da2325ab811ba61c764bcaabafb355142eef26)
			This archive contains ojdbc11.jar , ucp11.jar , Reactive Streams Ingest (rsi.jar), companion

▶ OJDBC 다운로드

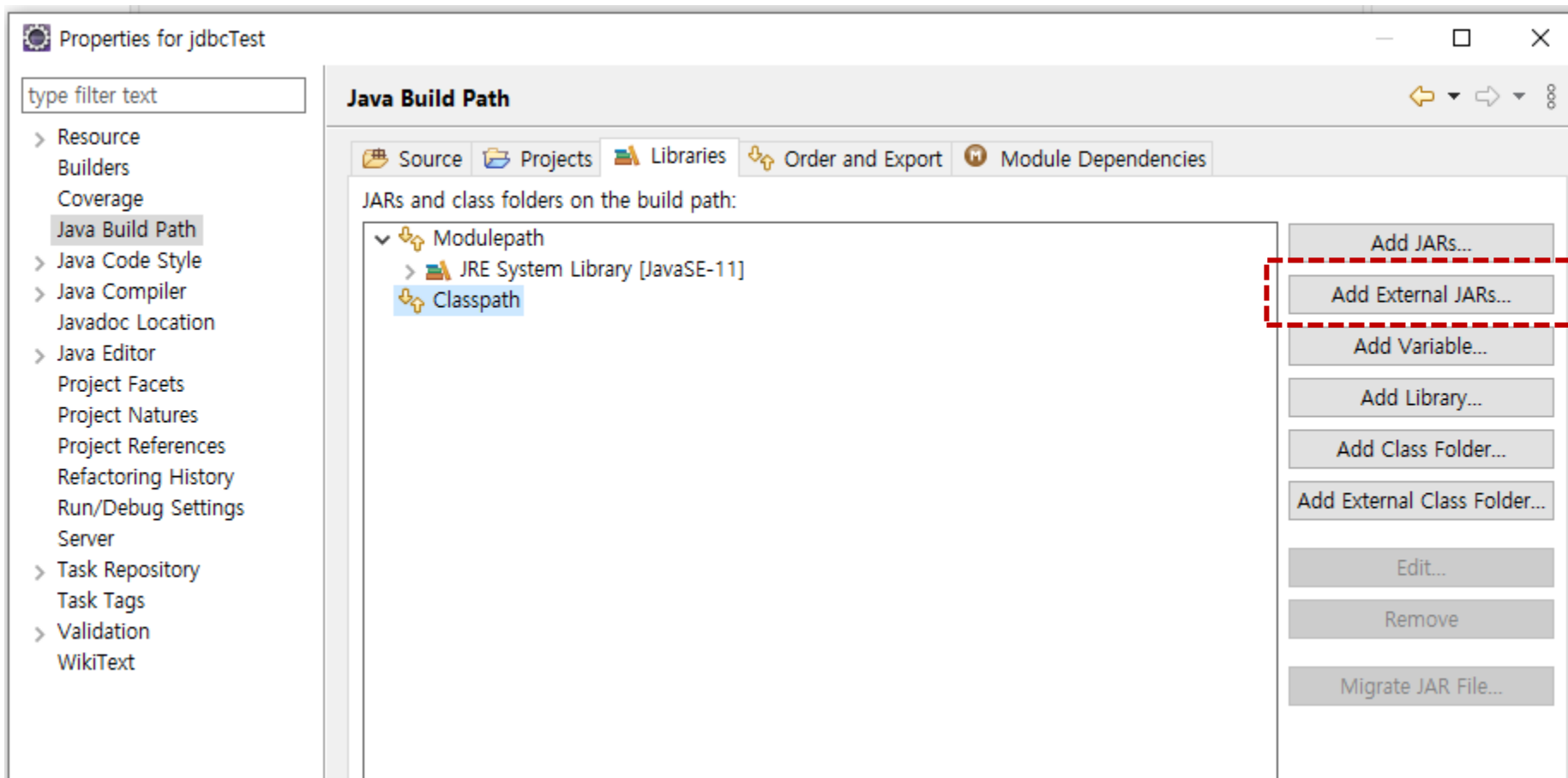
✓ 오라클 폴더에서 복사

- Express 버전

C:\app\사용자계정\product\21c\dbhomeXE\jdbc\lib\ojdbc8.jar

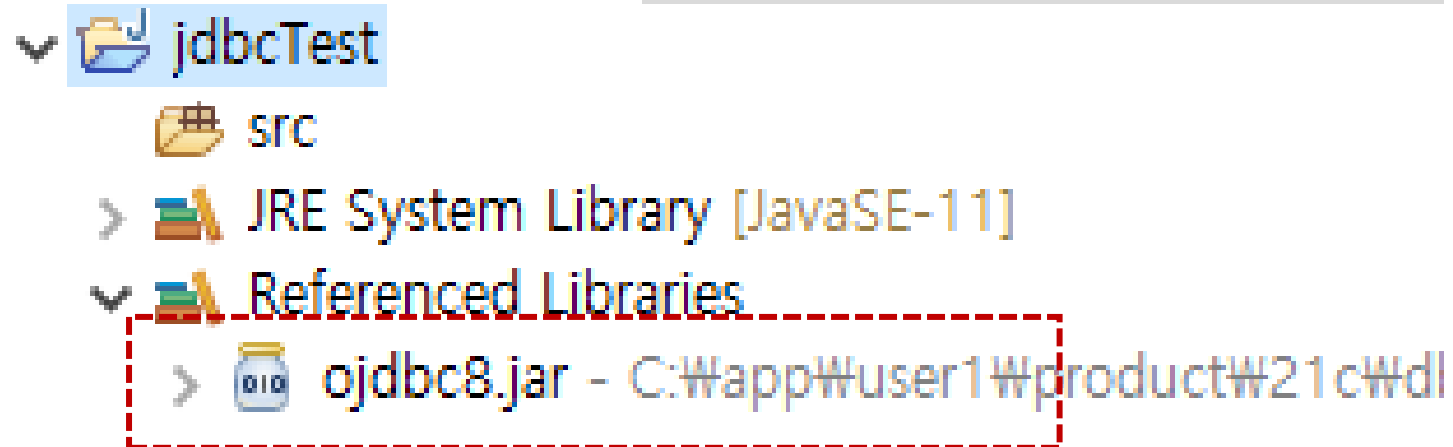
▶ Library 등록

✓ Eclipse Library로 등록



▶ Library 등록

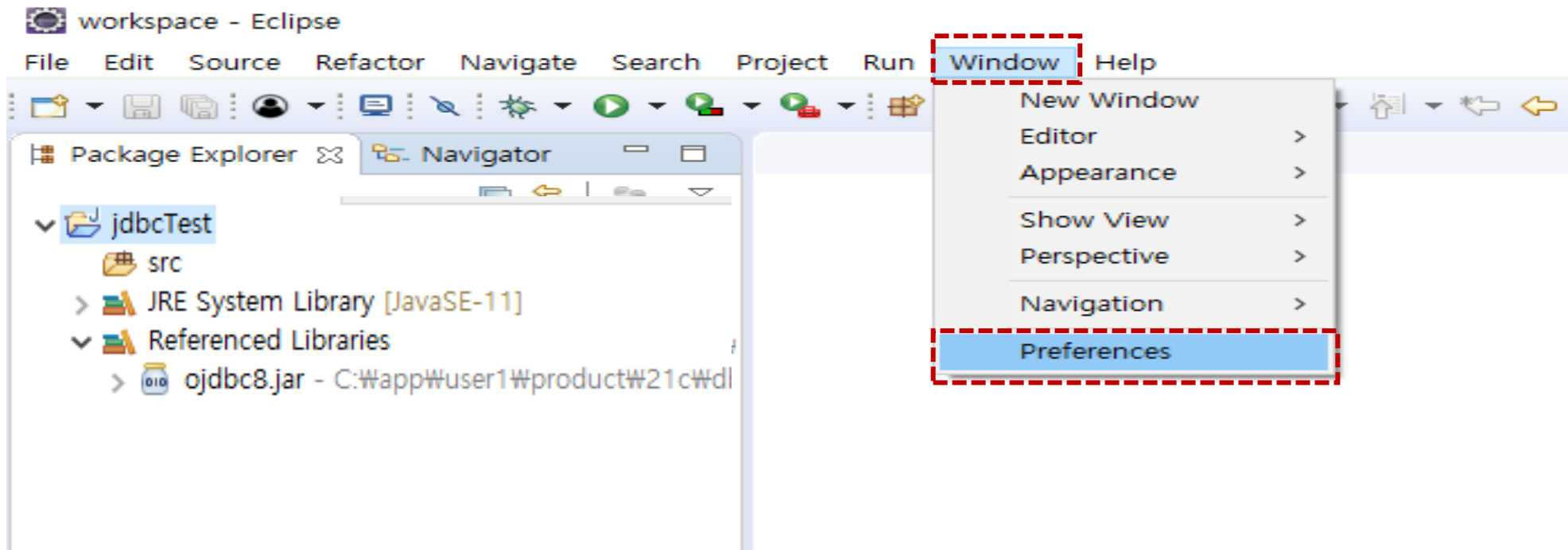
✓ Eclipse Library로 등록



* package Explorer에 Referenced Libraries가 생기고
ojdbc8.jar가 생성됨

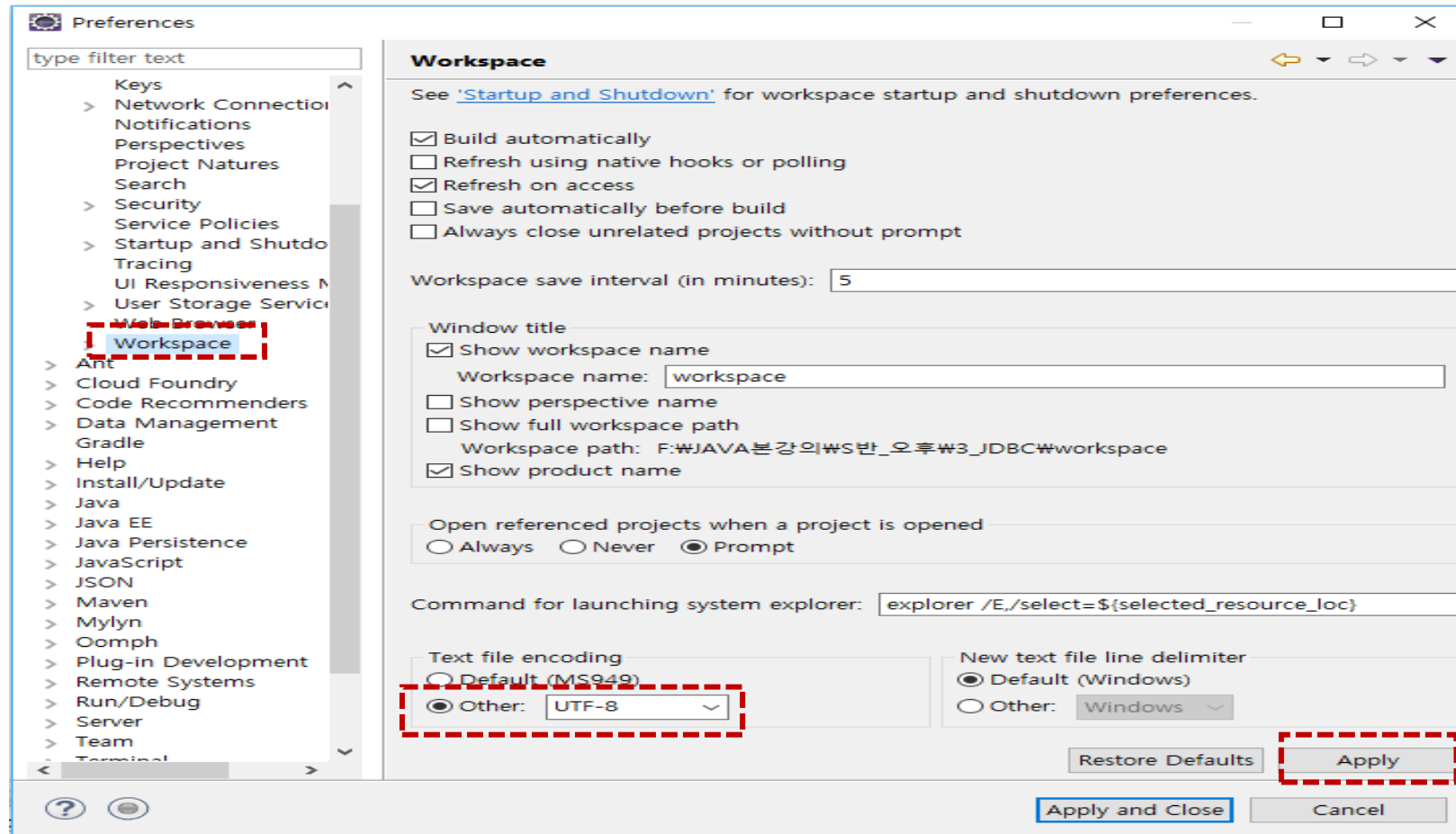
▶ Character Set 설정

문자 인코딩 방식이 맞지 않으면 해당 문자가 제대로 출력되지 않아
이클립스 작업 파일에 대한 문자셋을 UTF-8로 일치시켜야 함



▶ Character Set 설정

✓ 기본 문자 인코딩



* General – Workspace – Text file encoding – other에서 UTF-8 선택

▶ JDBC 사용 객체

✓ DriverManager

데이터 원본에 JDBC드라이버를 통하여 커넥션을 만드는 역할

Class.forName() 메소드를 통해 생성되며 반드시 예외처리를 해야 함

직접 객체 생성이 불가능하고 getConnection() 메소드를 사용하여 객체 생성 가능

✓ Connection

특정 데이터 원본과 연결된 커넥션을 나타내며 Statement객체를 생성할 때도

Connection객체를 사용하여 createStatement() 메소드를 호출하여 생성

SQL문장을 실행시키기 전에 우선 Connection객체가 있어야 함

▶ JDBC 사용 객체

✓ Statement

Connection객체에 의해 프로그램에 리턴되는 객체에 의해 구현되는 일종의 메소드 집합 정의
Connection클래스의 createStatement() 메소드를 호출하여 얻어지며
생성된 Statement객체로 질의문장을 String객체에 담아 인자로 전달하여
executeQuery() 메소드를 호출하여 SQL질의 수행

✓ 예시

```
try{
    String query = "SELECT ID, LAST_NAME FROM EMP";
    stmt = conn.createStatement();
    rset = stmt.executeQuery(query);
} catch(SQLException e){
    e.printStackTrace();
}
```

▶ JDBC 사용 객체

✓ PreparedStatement

Connection객체의 `prepareStatement()` 메소드를 사용하여 객체 생성
SQL문장이 미리 컴파일 되고 실행 시간동안 인수 값을 위한 공간을 확보한다는 점에서
Statement와 다름
각각의 인수에 대해 위치홀더(?)를 사용하여 SQL문장을 정의할 수 있게 함

✓ 예시

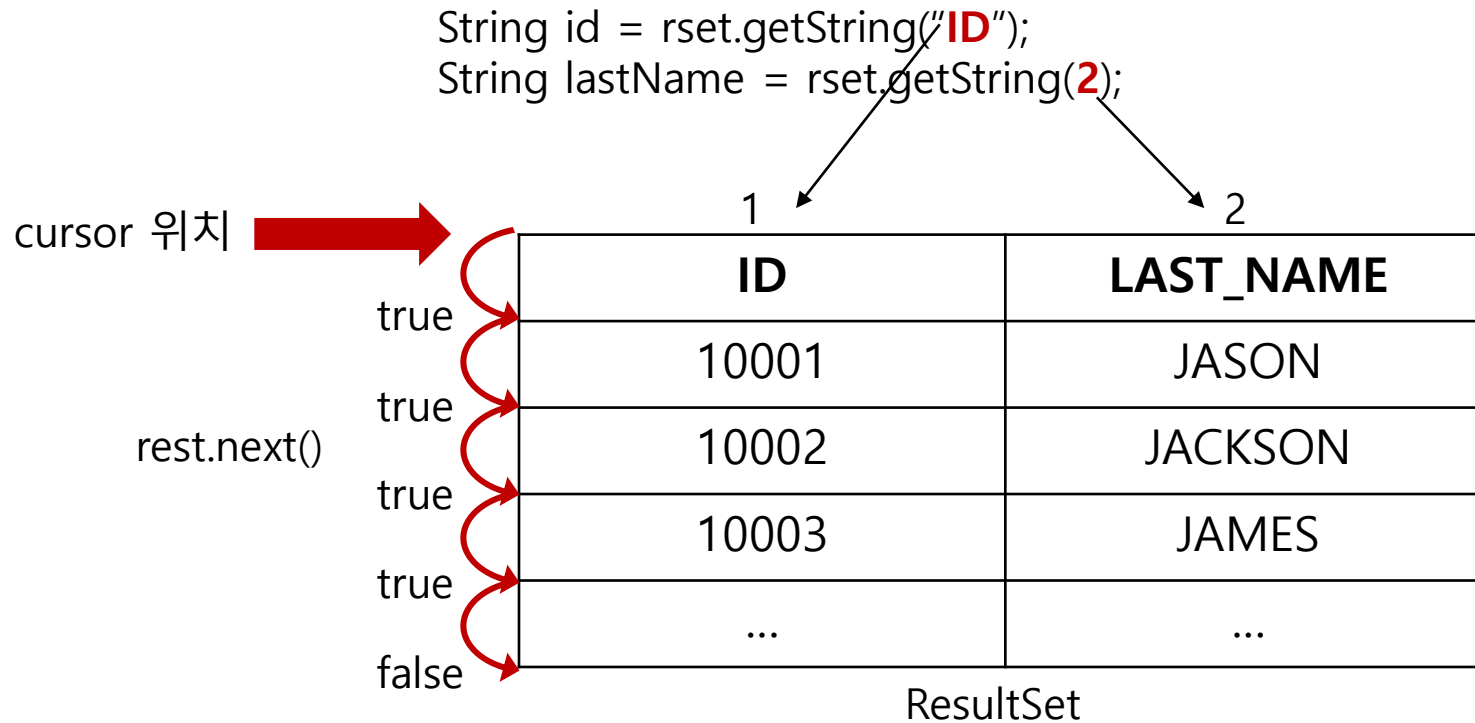
```
try{
    String query = "INSERT INTO MEMBER VALUES(?, ?)";
    pstmt = conn.prepareStatement(query);
    pstmt.setString(1, id);
    pstmt.setString(2, password);
} catch(SQLException e){
    e.printStackTrace();
}
```

▶ JDBC 사용 객체

✓ ResultSet

SELECT문을 사용한 질의 성공 시 Result Set 반환

SQL질의에 의해 생성된 테이블을 담고 있으며 커서(cursor)로 특정 행에 대한 참조 조작



▶ JDBC 코딩 절차

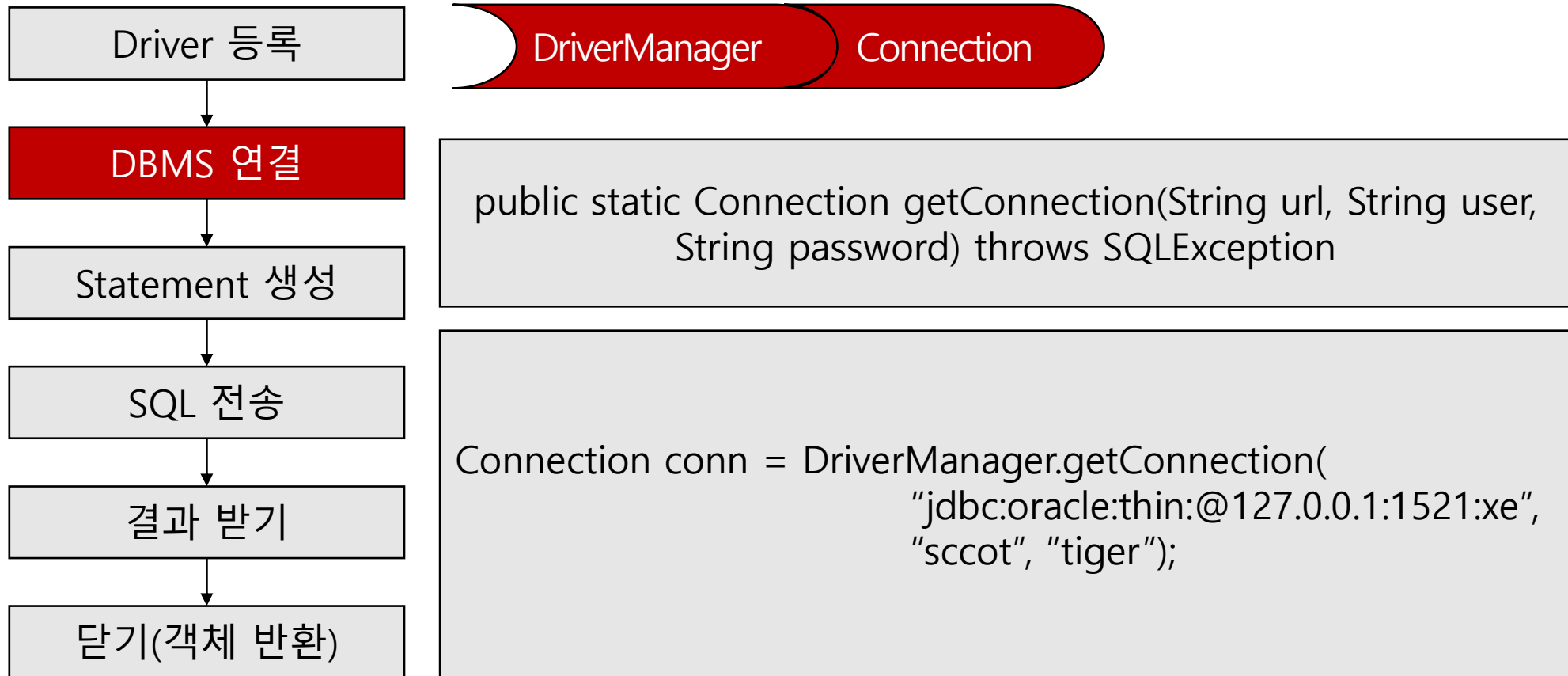
✓ DriverManager에 해당 DBMS Driver 등록



* 반드시 ClassNotFoundException 처리를 해야 함

▶ JDBC 코딩 절차

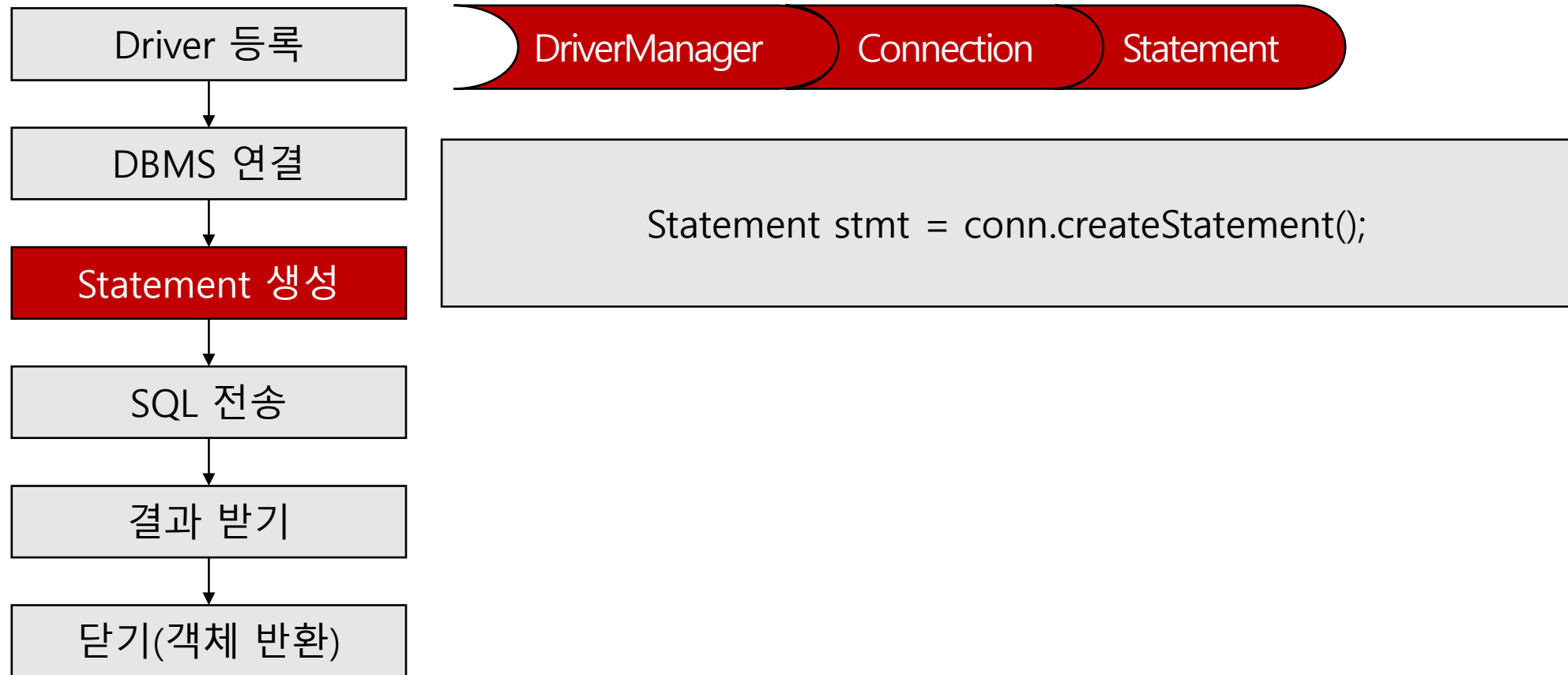
✓ 해당 Driver로부터 Connection instance 획득



* 반드시 SQLException 처리를 해야 함

▶ JDBC 코딩 절차

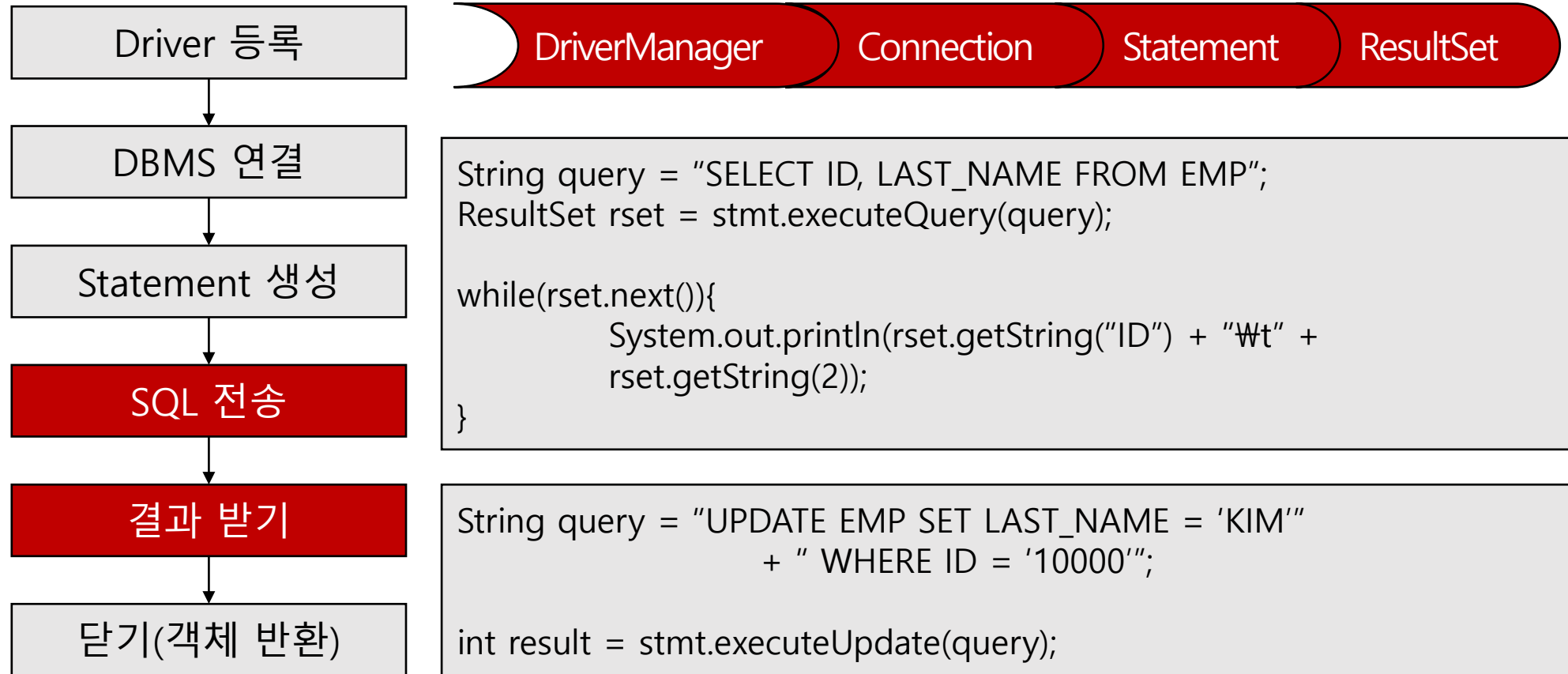
- ✓ Connection instance로부터 Statement instance 획득



* 반드시 SQLException 처리를 해야 함

▶ JDBC 코딩 절차

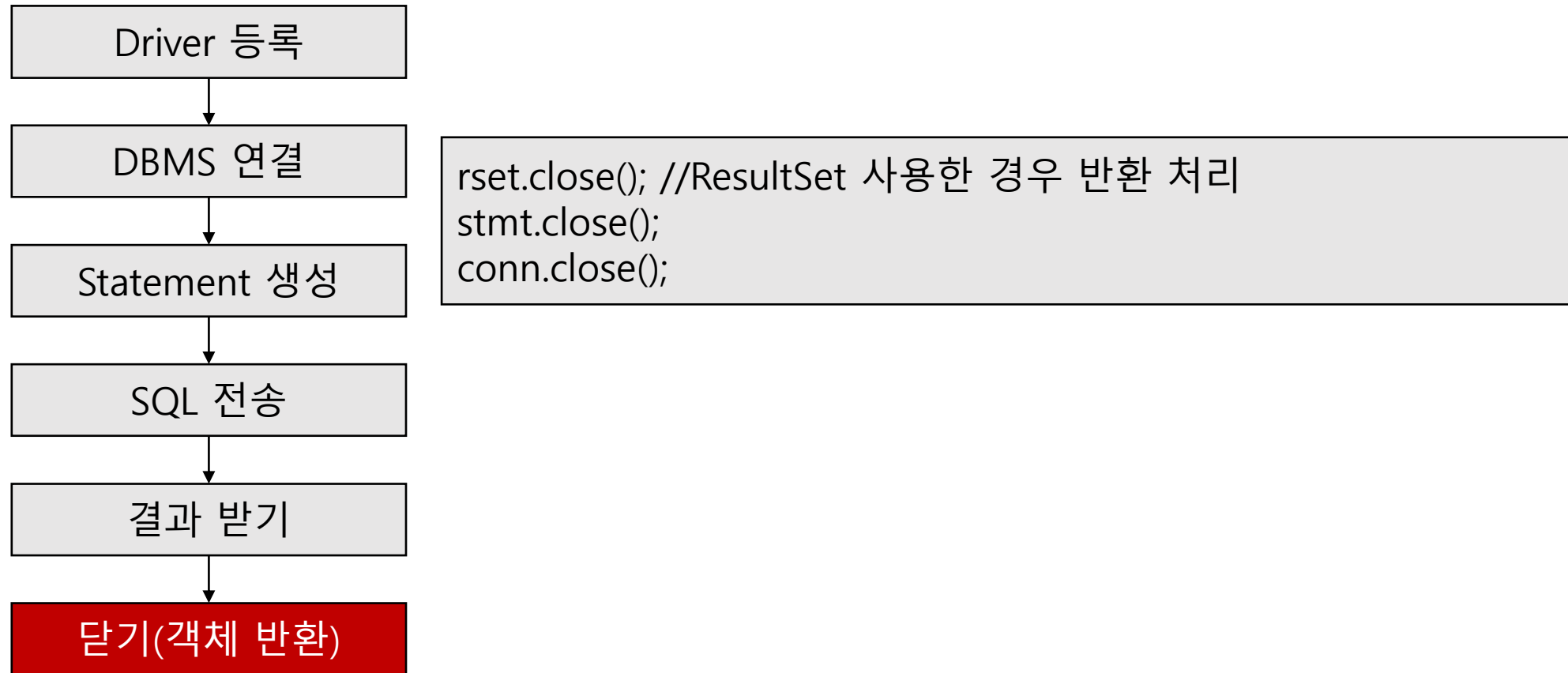
- ✓ Statement method를 이용하여 SQL문 실행
- ✓ 실행결과를 ResultSet(Select) 혹은 int형 변수(DML)로 받아서 처리



* 반드시 SQLException 처리를 해야 함

▶ JDBC 코딩 절차

- ✓ DB로 부터 획득한 instance 들을 획득한 역순으로 반환



* 반드시 SQLException 처리를 해야 함