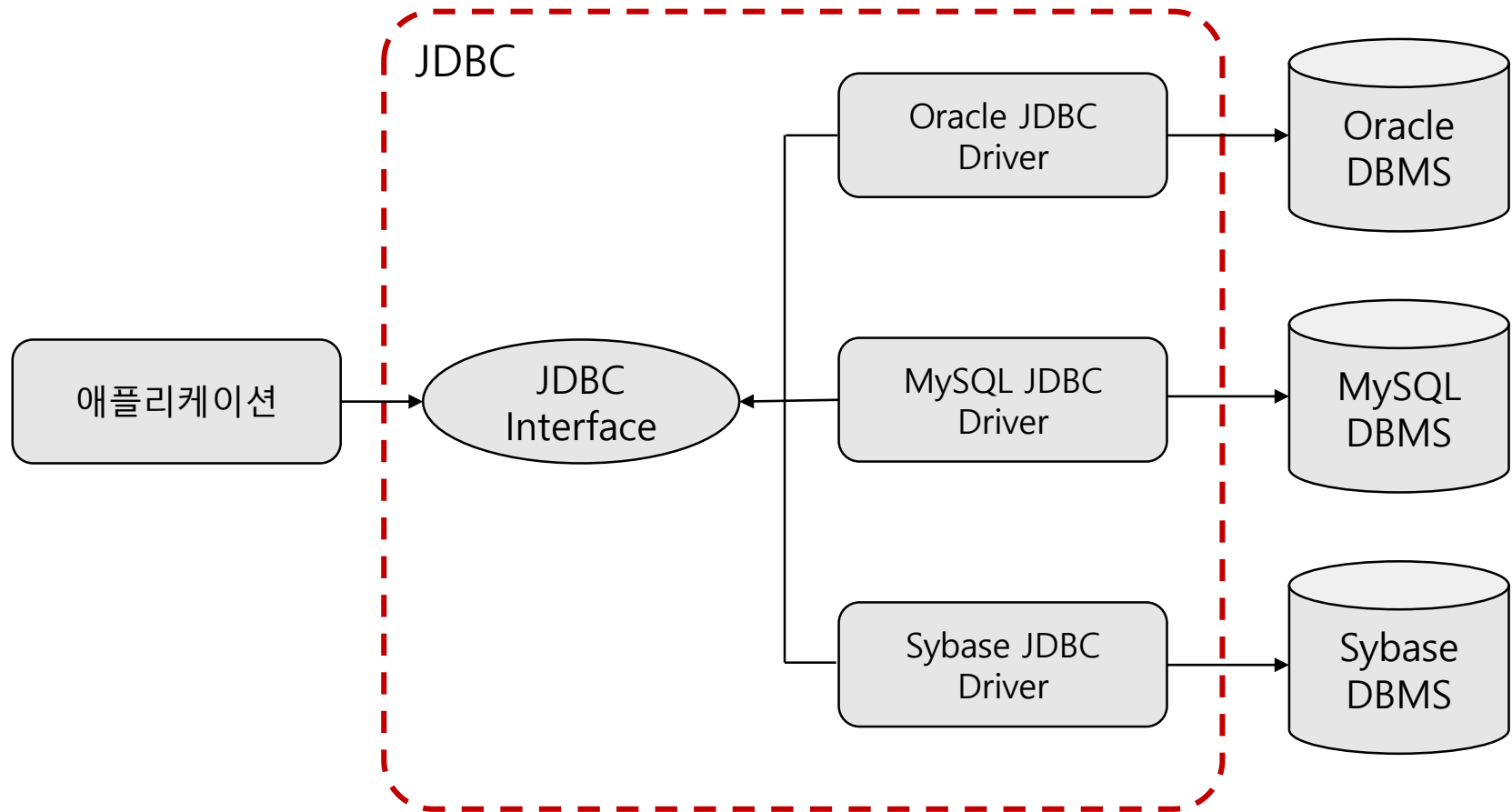


JDBC (Oracle 18c)

▶ JDBC(Java DataBase Connectivity)

자바에서 데이터베이스에 접근할 수 있게 해주는 Programming API



▶ JDBC(Java DataBase Connectivity)

✓ java.sql 패키지

The screenshot shows the Java API documentation for the `java.sql` package. The left sidebar lists various Java packages, with `java.sql` highlighted by a red dashed box. The main content area displays the package overview for `java.sql`, including navigation tabs (OVERVIEW, PACKAGE, CLASS, USE, TREE, D), navigation links (PREV CLASS, NEXT CLASS, FRAMES, NC), and a summary of nested classes, fields, constructors, and methods. The `compact2` and `compact3` classes are listed. The `Interface Connection` is highlighted, and the `All SuperInterfaces:` section lists `AutoCloseable` and `Wrapper`. The `public interface Connection` is shown extending `Wrapper` and `AutoCloseable`.

file:///F:/Storage/1_KH/Dev/Java/jdk-8u65-docs-all/docs/api/

java.rmi.registry
java.rmi.server
java.security
java.security.acl
java.security.cert
java.security.interfaces
java.security.spec
java.sql
java.text
java.text.spi
java.time
java.time.chrono
java.time.format
java.time.temporal
java.time.zone
java.util
java.util.concurrent

OVERVIEW PACKAGE **CLASS** USE TREE D

PREV CLASS NEXT CLASS FRAMES NC

SUMMARY: NESTED | FIELD | CONSTR | METHOD

compact2, compact3
java.sql

Interface Connection

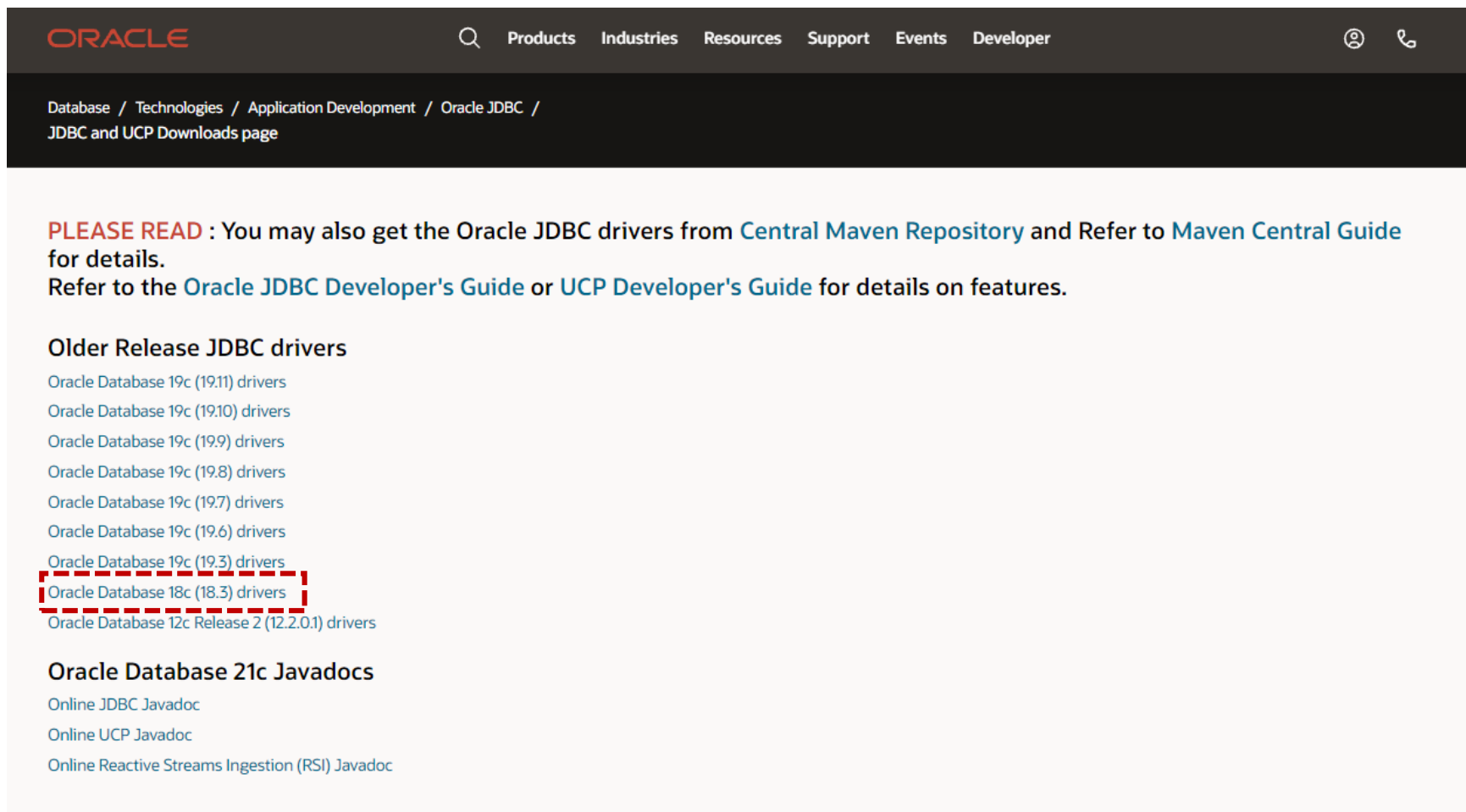
All SuperInterfaces:
AutoCloseable, Wrapper

public interface **Connection**
extends Wrapper, AutoCloseable

▶ OJDBC 다운로드

✓ 홈페이지 다운로드

<https://www.oracle.com/database/technologies/appdev/jdbc-downloads-2.html>



The screenshot shows the Oracle website's JDBC Downloads page. The header includes the Oracle logo and navigation links: Products, Industries, Resources, Support, Events, and Developer. A breadcrumb trail reads: Database / Technologies / Application Development / Oracle JDBC / JDBC and UCP Downloads page. The main content area features a 'PLEASE READ' notice, a list of older JDBC drivers with 'Oracle Database 18c (18.3) drivers' highlighted by a red dashed box, and a section for Oracle Database 21c Javadocs.

ORACLE Products Industries Resources Support Events Developer

Database / Technologies / Application Development / Oracle JDBC /
JDBC and UCP Downloads page

PLEASE READ : You may also get the Oracle JDBC drivers from [Central Maven Repository](#) and Refer to [Maven Central Guide](#) for details.
Refer to the [Oracle JDBC Developer's Guide](#) or [UCP Developer's Guide](#) for details on features.

Older Release JDBC drivers

- [Oracle Database 19c \(19.11\) drivers](#)
- [Oracle Database 19c \(19.10\) drivers](#)
- [Oracle Database 19c \(19.9\) drivers](#)
- [Oracle Database 19c \(19.8\) drivers](#)
- [Oracle Database 19c \(19.7\) drivers](#)
- [Oracle Database 19c \(19.6\) drivers](#)
- [Oracle Database 19c \(19.3\) drivers](#)
- [Oracle Database 18c \(18.3\) drivers](#)**
- [Oracle Database 12c Release 2 \(12.2.0.1\) drivers](#)

Oracle Database 21c Javadocs

- [Online JDBC Javadoc](#)
- [Online UCP Javadoc](#)
- [Online Reactive Streams Ingestion \(RSI\) Javadoc](#)

▶ OJDBC 다운로드

✓ 오라클에서 복사

- Express 버전

C:\app\user1\product\18.0.0\dbhomeXE\jdbc\lib
ojdbc8

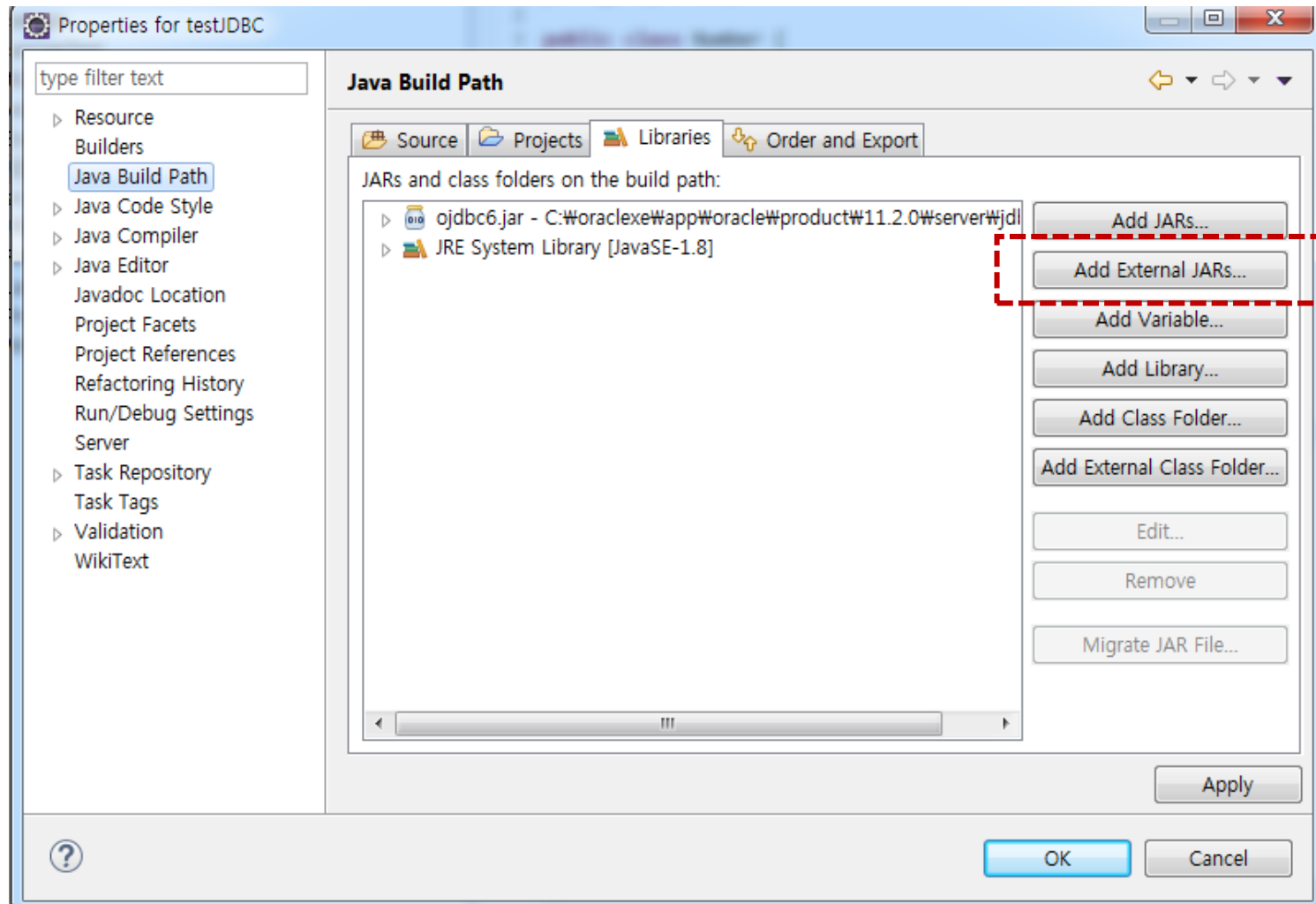
▶ JDBC 설치

다운 받거나 복사한 ojdbc파일을 dev폴더에 넣어주기

D:\dev\

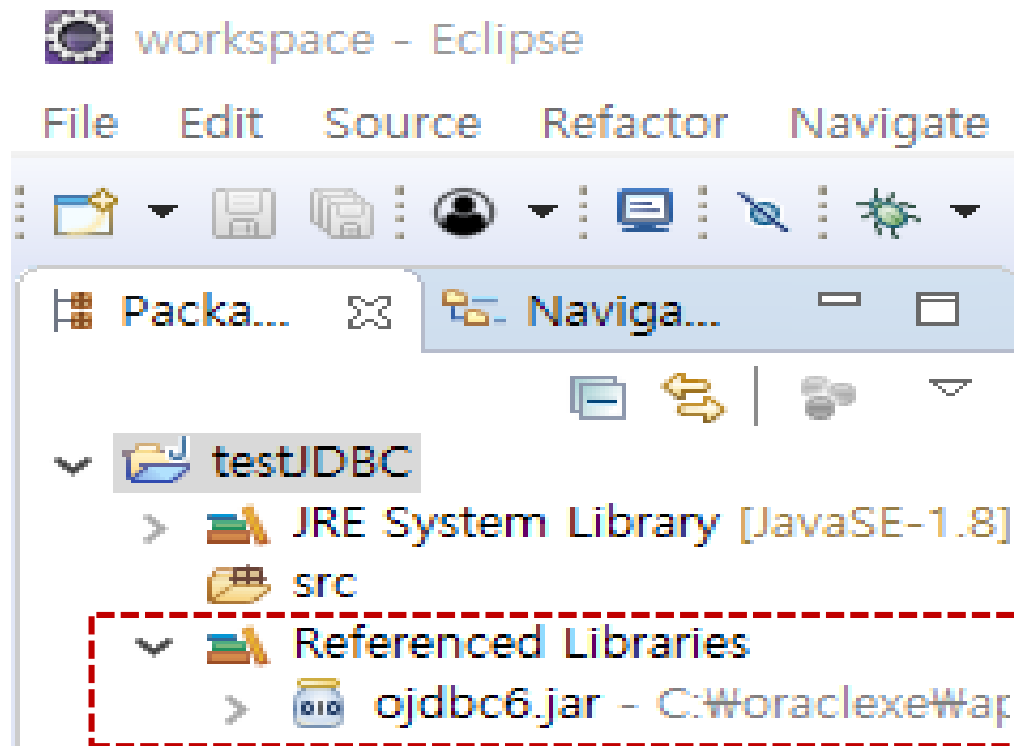
▶ Library 등록

✓ Properties로 등록



▶ Library 등록

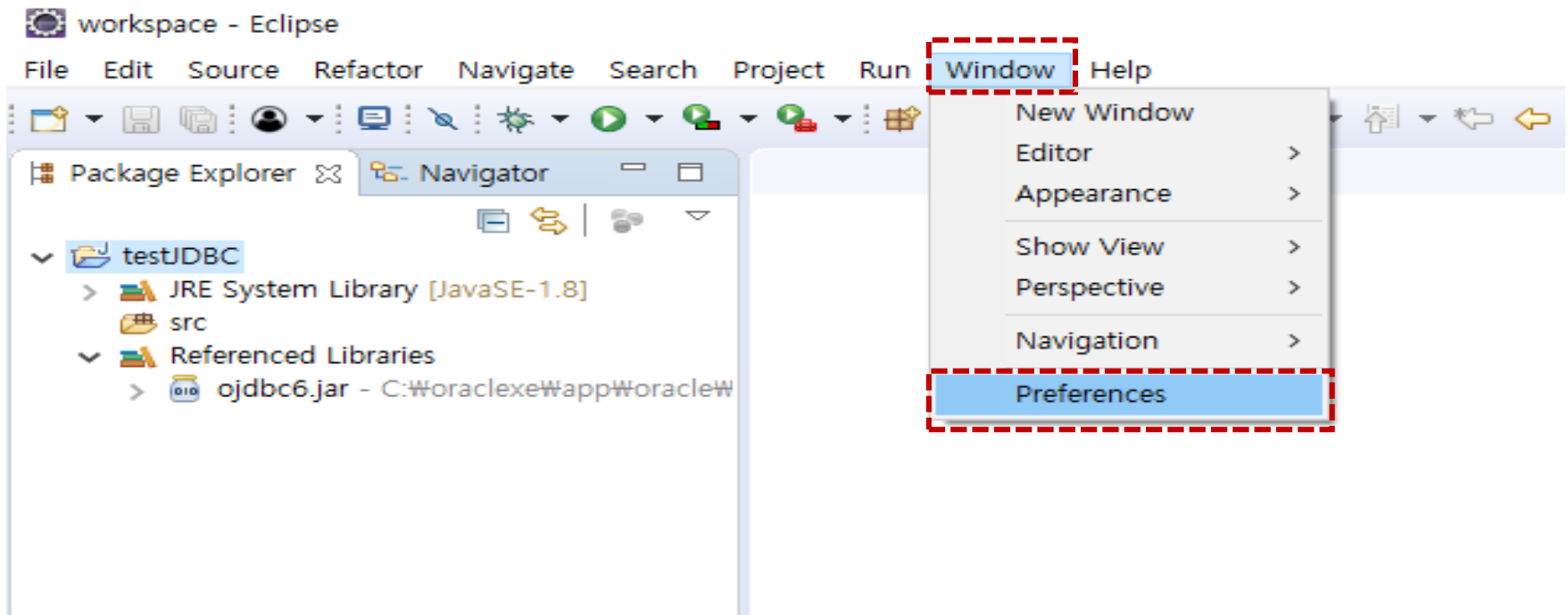
✓ Eclipse Library로 등록



* package Explorer에 Referenced Libraries가 생기고
ojdbc6.jar가 생성됨

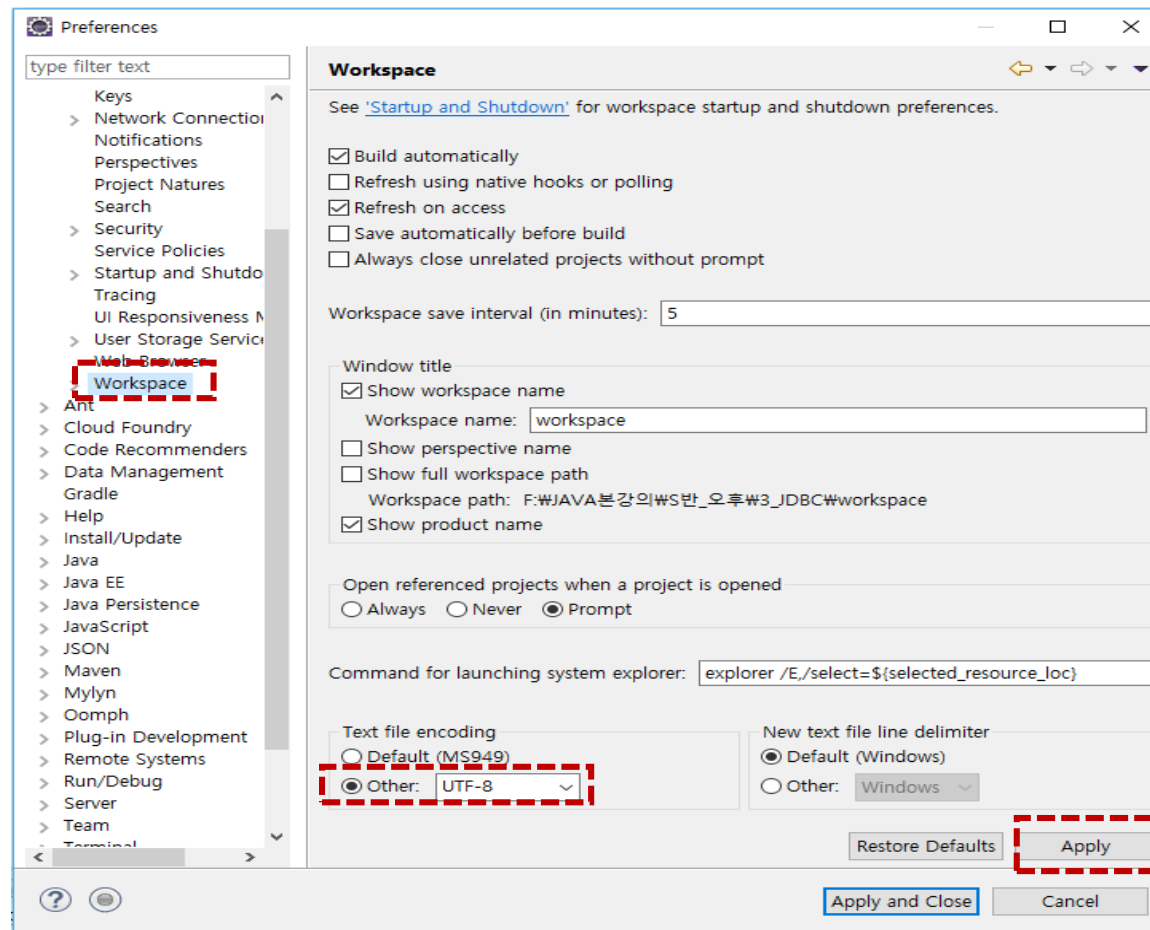
▶ Character Set 설정

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



▶ Character Set 설정

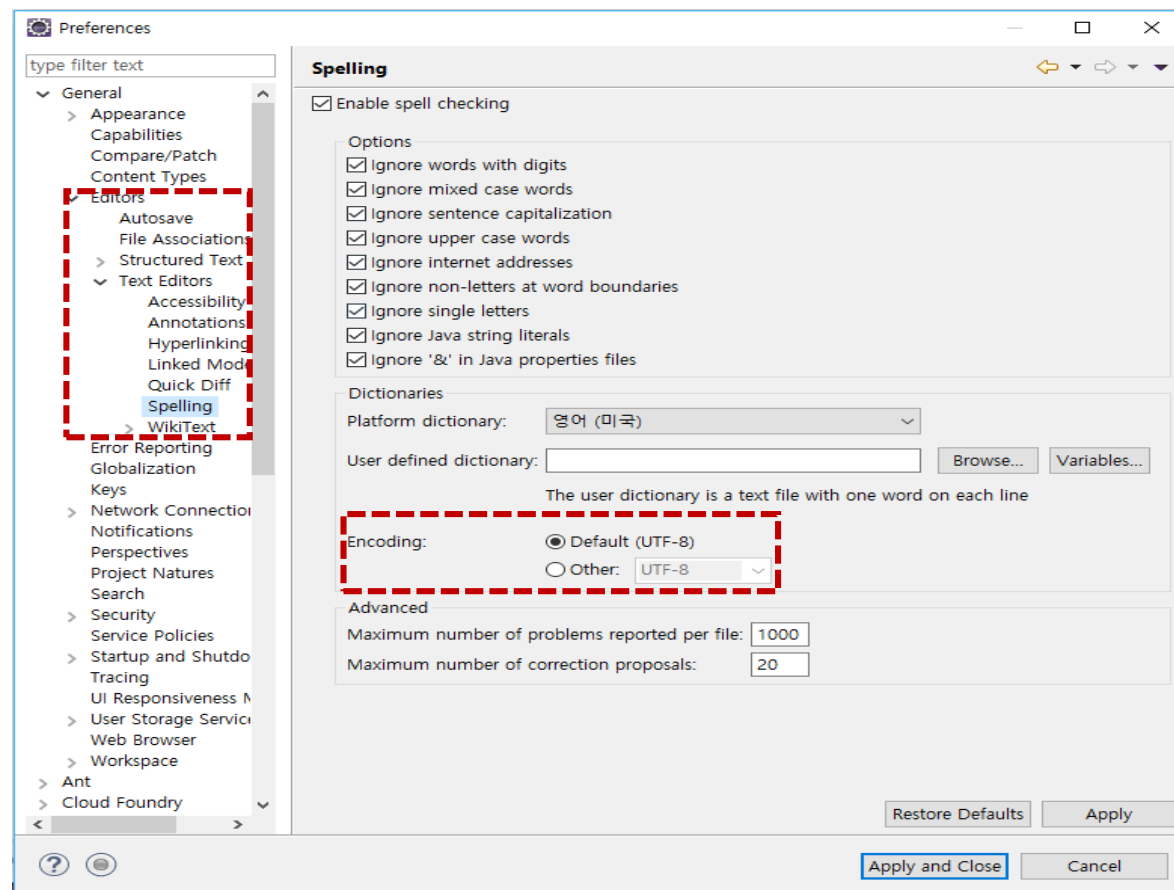
✓ 기본 문자 인코딩



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

▶ Character Set 설정

✓ 기본 문자 인코딩



* General – Editors – Text Editors – Spelling UTF-8 변경

▶ 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 Connection 객체가 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 SQL 질의 수행

▶ JDBC 사용 객체

✓ 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 시간 동안 인수 값을 위한 공간을 확보한다는 점에서 Statement와 다름
각 인수에 대해 위치홀더(?)를 사용해 SQL문장을 정의할 수 있게 함

▶ JDBC 사용 객체

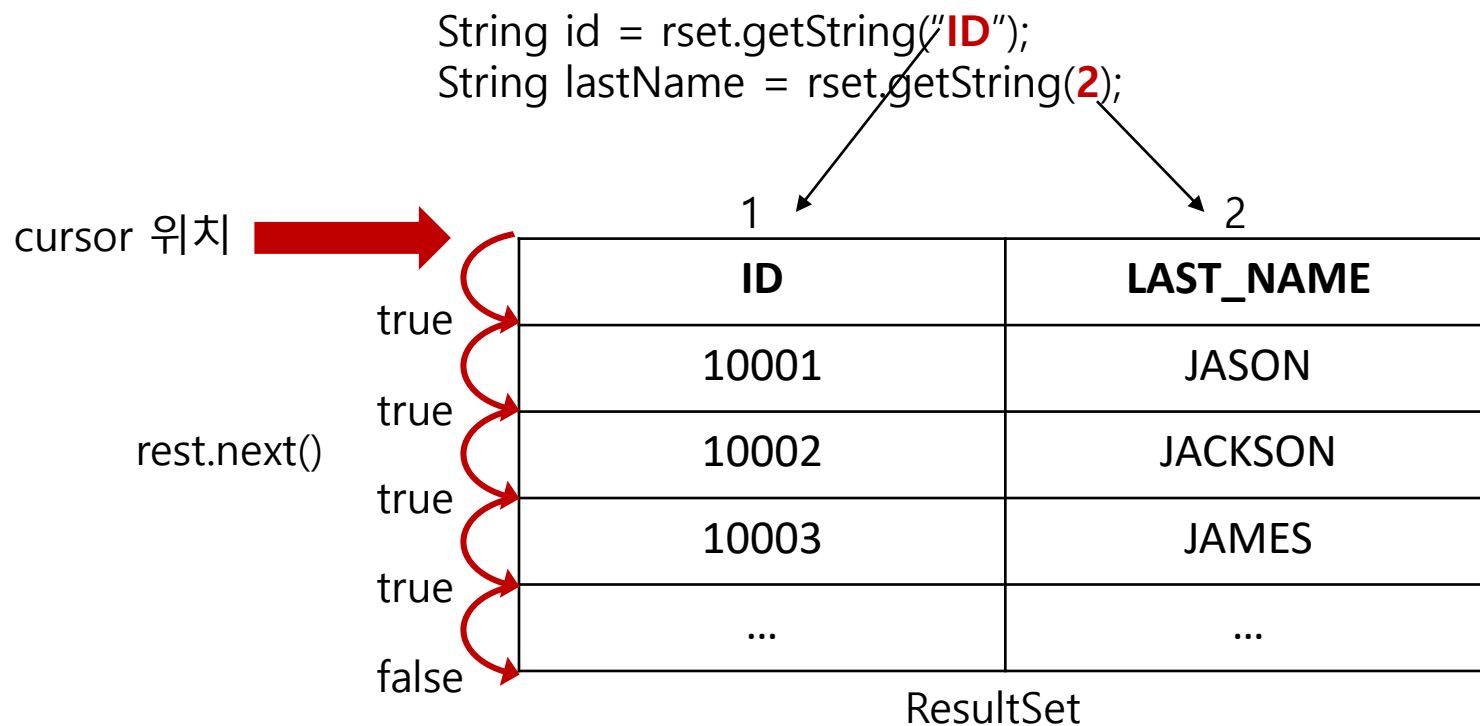
✓ 예시

```
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

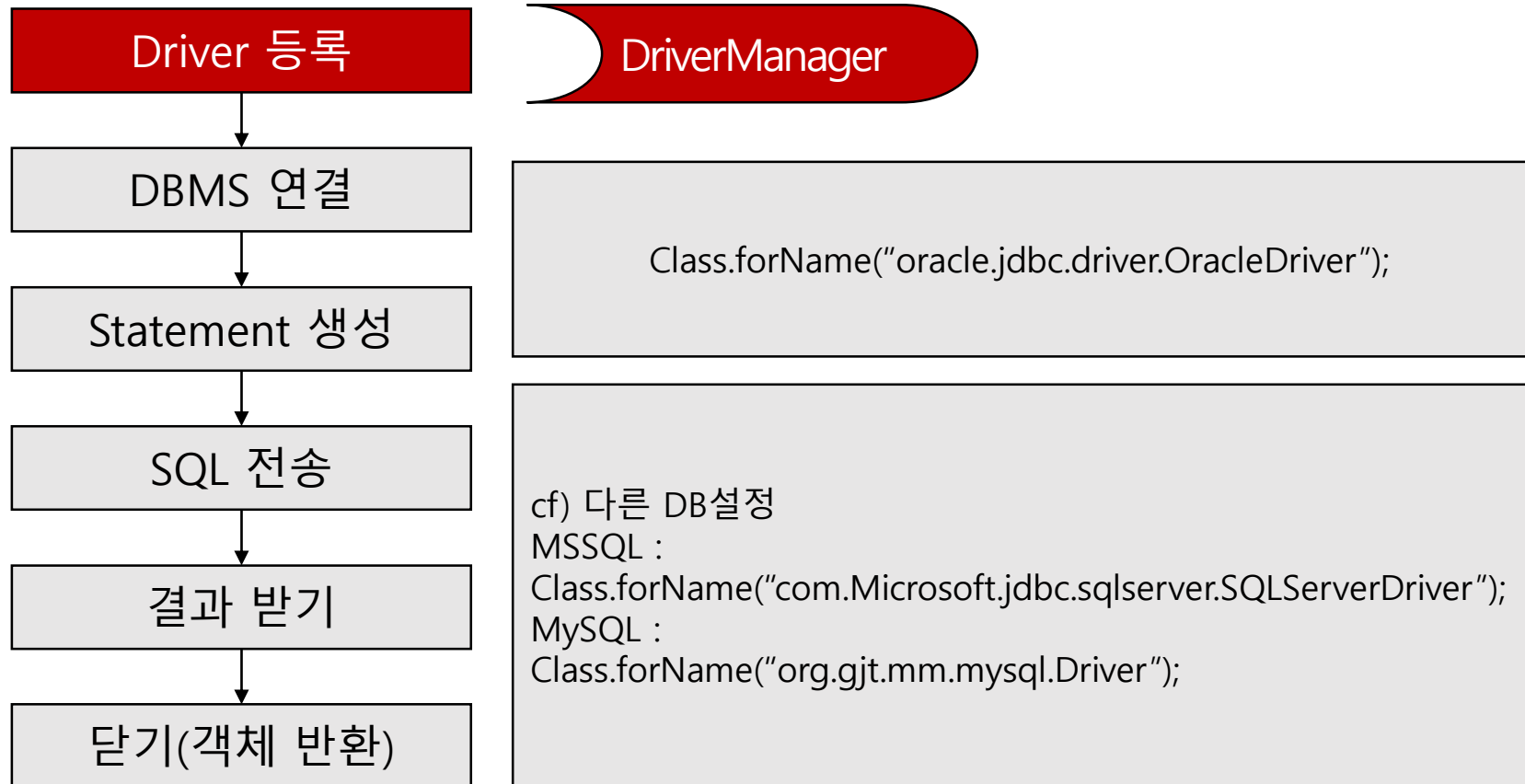
✓ ResultSet

SELECT문을 사용한 질의 성공 시 ResultSet반환
SQL질의에 의해 생성된 테이블을 담고 있으며
커서(cursor)로 특정 행에 대한 참조 조작



▶ JDBC 코딩 절차

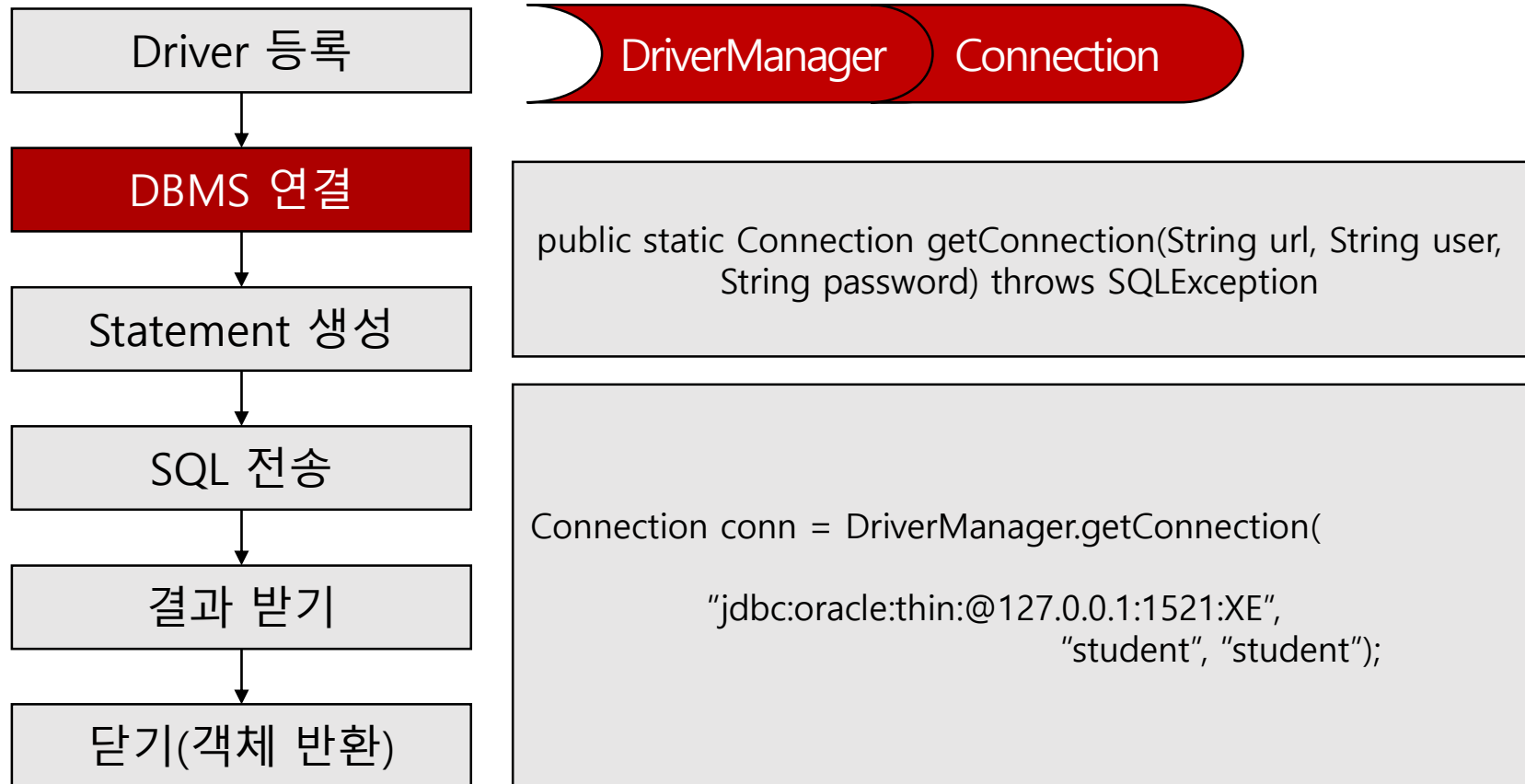
✓ DriverManager에 해당 DBMS Driver 등록



* 반드시 ClassNotFoundException 처리를 해야 함

▶ JDBC 코딩 절차

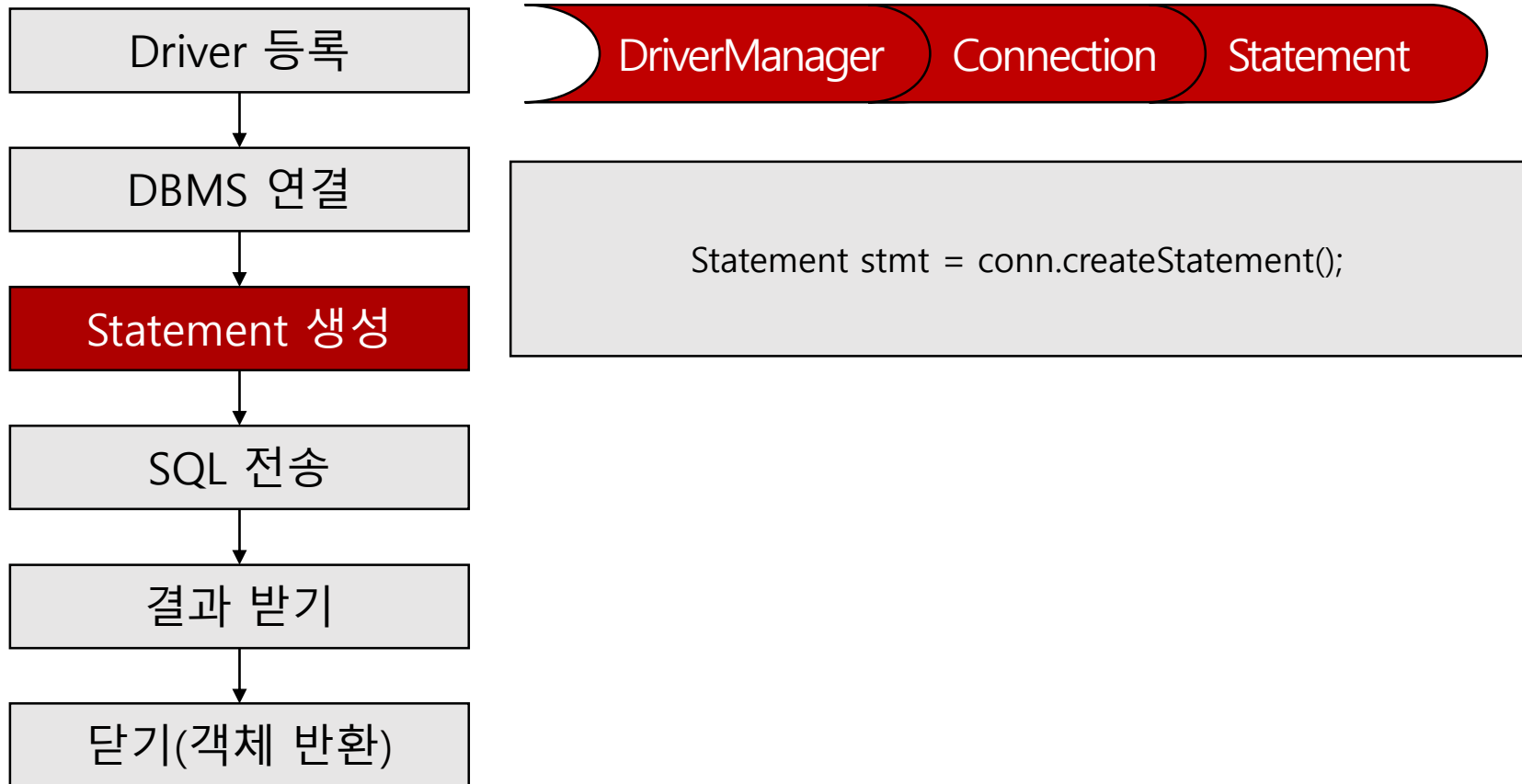
✓ 해당 Driver로부터 Connection instance 획득



* 반드시 SQLException처리를 해야 함

▶ JDBC 코딩 절차

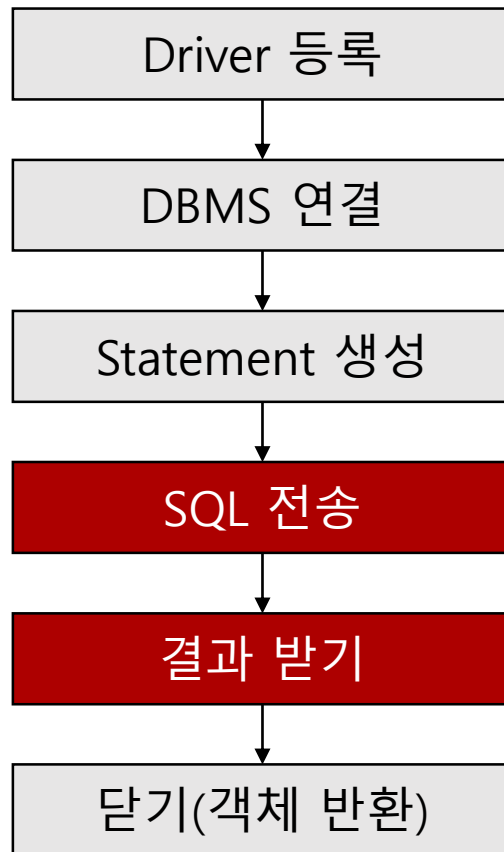
✓ 해당 Driver로부터 Connection instance 획득



* 반드시 SQLException처리를 해야 함

▶ JDBC 코딩 절차

- ✓ Statement method를 이용하여 SQL문 실행
- ✓ 실행결과를 ResultSet 혹은 int형 변수로 받아서 처리



DriverManager Connection Statement ResultSet

```
String query = "SELECT ID, LAST_NAME FROM EMP";
ResultSet rset = stmt.executeQuery(query);

while(rset.next()){
    System.out.println(rset.getString("ID") + "\t" +
        rset.getString(2));
}
```

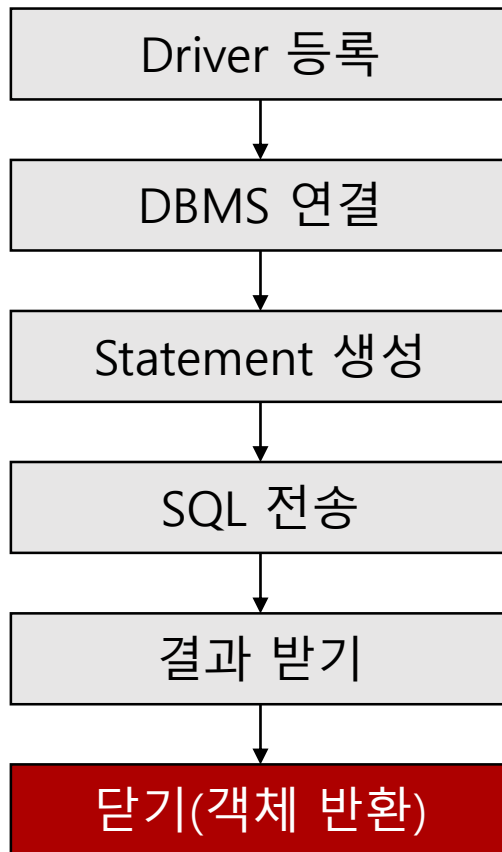
```
String query = "UPDATE EMP SET LAST_NAME = 'KIM'"
    + " WHERE ID = '10000'";

int result = stmt.executeUpdate(query);
```

* 반드시 SQLException처리를 해야 함

▶ JDBC 코딩 절차

- ✓ Statement method를 이용하여 SQL문 실행
- ✓ 실행결과를 ResultSet 혹은 int형 변수로 받아서 처리



DriverManager Connection Statement ResultSet

```
rset.close(); //ResultSet 사용한 경우 반납처리  
stmt.close();  
conn.close();
```

* 반드시 SQLException처리를 해야 함