

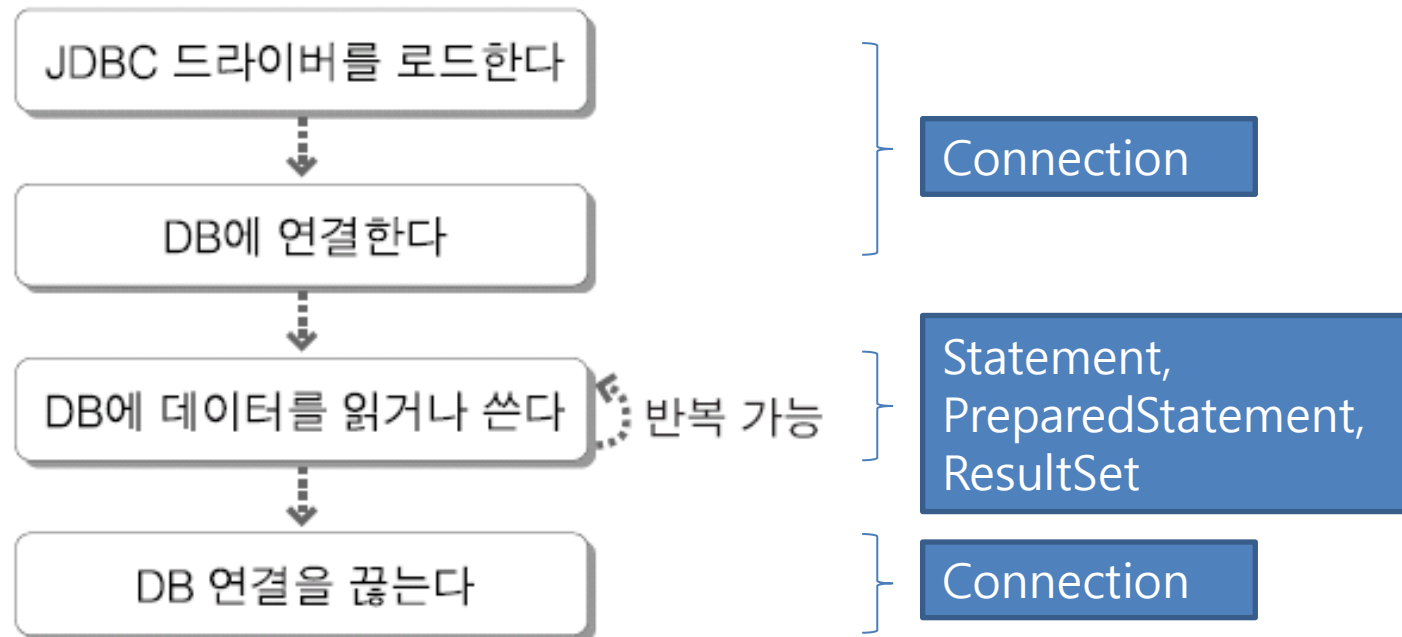
# JDBC 프로그래밍

## - Select -

# SELECT 처리

## ❖ JDBC를 이용한 프로그램의 작성 방법

- 자바 프로그램에서 데이터베이스를 사용하는 4단계의 과정



# SELECT 처리

---

## ❖ 데이터의 조회

- 테이블의 데이터를 읽어오는 방법
  - 1) Connection 객체로부터 Statement 객체 획득

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

# SELECT 처리

---

## ❖ 데이터의 조회

- 테이블의 데이터를 읽어오는 방법
  - 2) SQL문을 작성하여 `executeQuery` 메소드 호출

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

- 반환값 : `ResultSet`
  - SELECT 문의 결과에 대한 참조
  - Iterator 패턴으로 순회하여 데이터 추출

# SELECT 처리

---

## ❖ 데이터의 조회

- 테이블의 데이터를 읽어오는 방법
  - 3) next 메소드 호출
  - 다음 데이터가 있으면 true, 없으면 false

```
boolean exists = rs.next();
```

# SELECT 처리

## ❖ 데이터의 조회

- 테이블의 컬럼 데이터를 읽어오는 방법

4) getInt, getString, getFloat 등 데이터 타입별 메소드를 호출하여 특정 컬럼의 값 추출

```
String code = rs.getString("code");
```

컬럼 값을 읽어서 String 타입으로  
리턴하는 메소드

컬럼 이름

```
int price = rs.getInt("price");
```

컬럼 값을 읽어서 int 타입으로  
리턴하는 메소드

컬럼 이름

# SELECT 처리

---

## ❖ 데이터의 조회

- 테이블의 데이터를 읽어오는 방법

```
while (rs.next()) {  
    String code = rs.getString("code");  
    String name = rs.getString("name");  
    int price = rs.getInt("price");  
    String maker = rs.getString("maker");  
    ...  
}
```

} 더 이상 읽을 행이 없을 때까지  
code, name, price, maker 컬럼의  
값을 읽어오는 일을 반복합니다

# SELECT 처리

---

## ❖ 데이터의 조회

- 테이블의 데이터를 읽어오는 방법
- 5) ResultSet 객체에 대해 close() 메소드 호출  
→ try-with-resource로 처리

rs.close();

↑  
ResultSet을 닫는 메소드



# SELECT 처리

---

## ❖ 데이터의 조회

- 테이블의 데이터를 읽어오는 방법
- 6) Statement 객체에 대해 close 메소드 호출

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



ResultSet을 닫는 메소드

→ try-with-resource로 처리

# SELECT 처리

---

## ❖ 데이터의 조회 : 기본 골격

```
Connection conn = Database.getConnection();
String sql = "select * from employees";

try(Statement stmt = conn.createStatement()) {
    try(ResultSet rs = stmt.executeQuery(sql)) {
        while(rs.next()) {
            // 개별 행 처리
        }
    }
} catch(Exception e) {
    e.printStackTrace();
}

Database.close();
```

---

## ❖ 모델 객체 : Employee

```
@Data
@Builder
public class Employees {
    private long employeeId;
    private String firstName;
    private String lastName;
    private String email;
    private String phoneNumber;
    private Date hireDate;
    private String jobId;
    private double salary;
    private double commissionPct;
    private long managerId;
    private long departmentId;
}
```

## **SELECT 처리**

---

❖ **ResultSet -> 모델 객체 변환**