

27강. 연결형 데이터베이스 프로그래밍

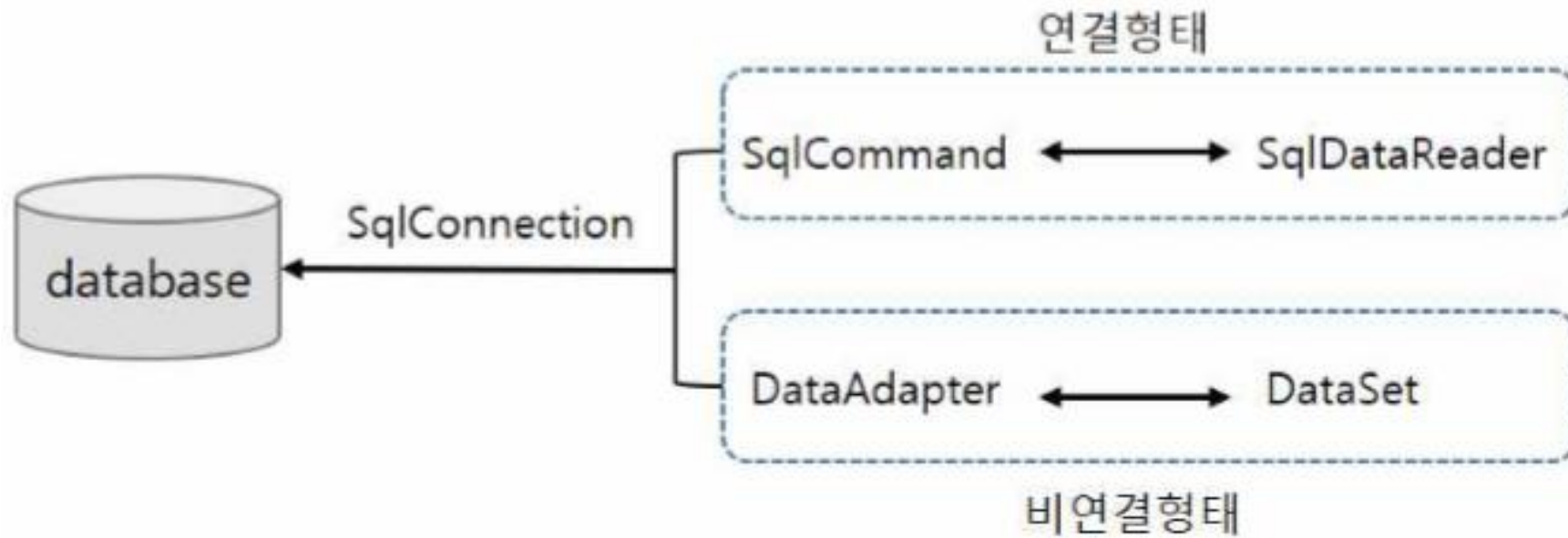
SqlCommand, SqlDataReader 사용



목차

- 개념 정리
- SqlCommand
- SqlDataReader
- 실습

연결형 구조





SqlCommand

- 역할
데이터베이스에 대해 SQL 문을 실행
- 상속 계층 구조

System.Object

System.MarshalByRefObject

System.ComponentModel.Component

System.Data.Common.DbCommand

System.Data.SqlClient.SqlCommand

네임스페이스 필요

SqlCommand 생성자

- 역할
데이터베이스에 SQL문 실행
- SqlCommand SQL문 설정 방법
 - SqlCommand 생성 + CommandText속성에 SQL문 설정
 - SqlCommand 생성할 때 SQL문 설정
- 생성자
 - SqlCommand()
 - SqlCommand(String)
 - SqlCommand(String, SqlConnection)
SqlConnection 접속 → String SQL 실행



SqlCommand 주요속성

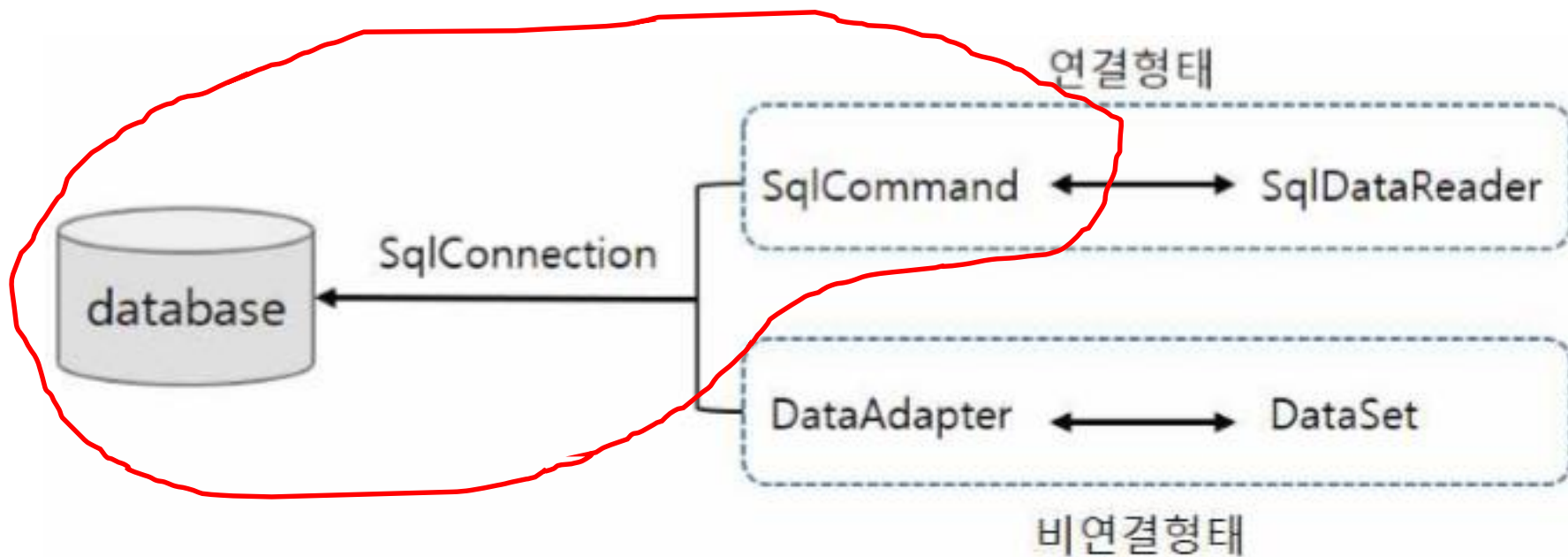
- CommandText
SQL 실행 및 저장 프로시저 읽기/설정
- Connection
SqlConnection 설정 or 가져오기
- CommandType
CommandText 속성을 다른 형식으로 설정 or 가져오기
 - CommandType.Text : SQL문 ← 기본값
 - CommandType.TableDirect : 테이블명령
 - CommandType.StoredProcedure : 프로시저 명령



SqlCommand 주요 메서드

- ExecuteNonQuery()
SQL문을 실행 했을 때 실행된 행의 수를 리턴
- ExecuteReader()
CommandText의 SQL문을 Connection에 보내고
SqlDataReader 를 생성
- 그외에 ExecuteXmlReader 등....

SqlConnection과 SqlCommand 관계





SqlDataReader

- 역할
데이터베이스에서 전용 행 스트림으로 읽는 역할
(삽입, 삭제등은 할 수 없다)
- 상속 계층 구조

`System.Object`

`System.MarshalByRefObject`

`System.Data.Common.DbDataReader`

`System.Data.SqlClient.SqlDataReader`

namespace



SqlDataReader 주요 속성

- Connection
SqlConnection과 연결된 SqlDataReader를 가져옴
- FieldCount
현재 행의 열 수를 가져옴
- HasRows
SqlDataReader로부터 하나 이상의 행 포함 여부를
체크 (true, false)
- RecordsAffected
변경, 삽입 및 삭제 된 행의 개수를 가져옴,
없으면 0

SqlDataReader 주요 메서드

- public override bool Read()

다음 레코드로 이동하는 역할
처음 읽은 위치가 첫 번째 레코드 앞에 있으므로
Read() 후에 데이터 읽으면 됨



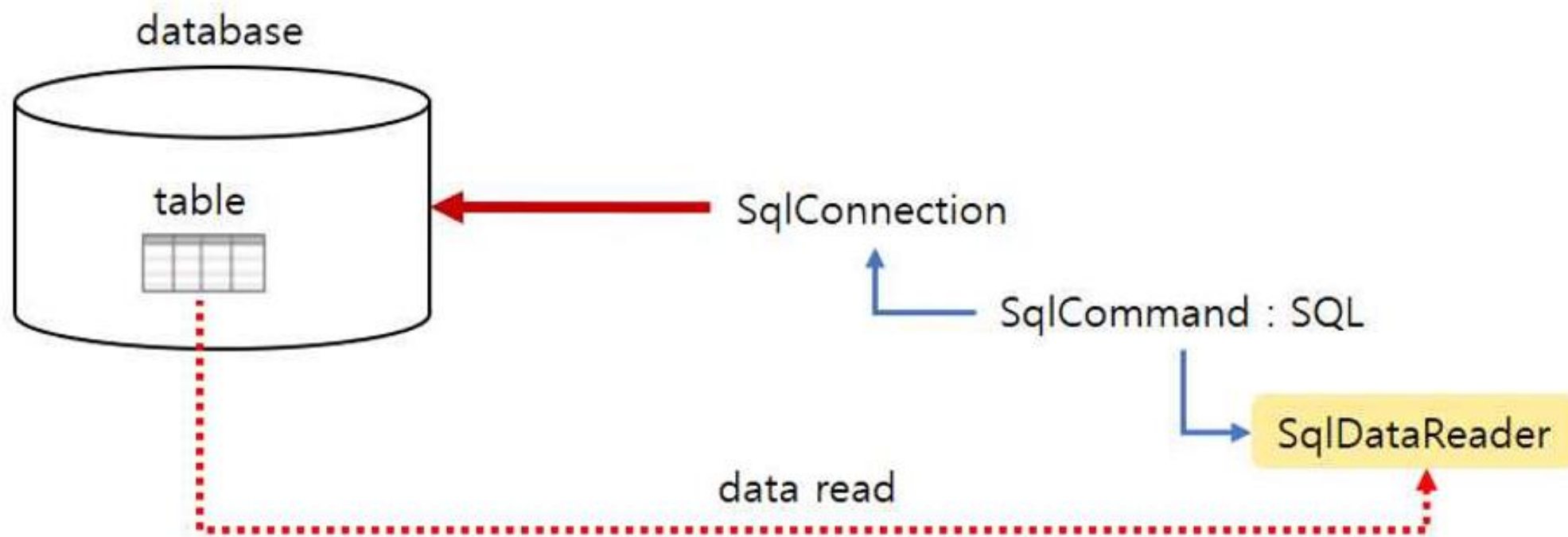
- Reader["Name"] or Reader[0] 형식으로 읽음
- 레코드가 없으면 false 리턴



SqlDataReader 주요 메서드

- Close()
SqlDataReader 객체 닫기

SqlConnection, SqlCommand, SqlDataReader관계



연결형 데이터베이스 실습



SqlConnection, SqlCommand, SqlDataReader



준비 사항



1 단계

- 데이터베이스 만들기
데이터베이스명 → school

2 단계

- MS SQL에 테이블 만들기 ➔ tblStudent

열이름	Name	Age	Male
데이터형식	nvarchar(10)	int	bit

nchar : 고정길이 유니코드 문자열(고정길이),
1 ~ 4000자

nvarchar: 가변길이 유니코드 문자열
(문자열에 따라 가변길이)
1 ~ 4000자 안에서 길이 지정, 2G까지 가능

3 단계

- 데이터 입력하기

	Name	Age	Male
▶	곰돌이	10	True
	곰이	14	False
	이철수	27	True
	콩순이	17	False
	홍길동	34	True



실습 1

- Console에서 출력하기
 - SqlConnection 접속
 - SqlCommand SQL문
 - SqlDataReader 읽기
 - 출력

[소스 보기](#)



실습 2

- SQL 데이터 삽입 및 출력
- SqlCommand 사용한 SQL
- ExecuteNonQuery() : 삽입 실행
- SqlDataReader 사용 : 읽기 ➔ 출력

소스 보기

실습 3

- SQL 문 입력 받아 실행하는 툴 만들기

The screenshot shows a Windows Form titled "Form1" with a standard Windows XP-style title bar. Inside the form, there is a label "SQL문" (SQL Statement) above a single-line text input box. To the right of the input box is a button labeled "실행" (Execute). Below the input box and button is a label "결과 보기" (View Result). Underneath this label is a large rectangular area with a dashed border and small square handles at the corners and midpoints, indicating it is a resizable container for displaying the results of the SQL execution.

소스 보기

Form1

SQL문

```
INSERT INTO tblStudent VALUES('우차차', 8, 1)
```

실행

결과 보기

```
이름:곰돌이 나이:10 성별:True  
이름:김지우 나이:41 성별:False  
이름:콩이 나이:14 성별:False  
이름:우차차 나이:8 성별:True
```