

Oracle Database

Introduction

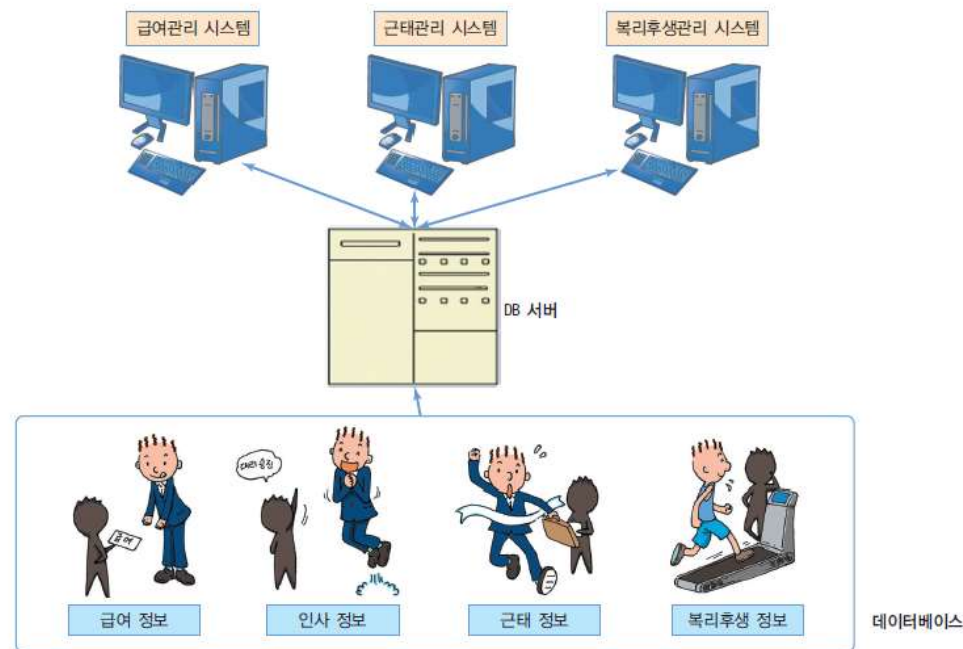
Introduction

데이터베이스와 데이터베이스 관리 시스템

데이터베이스와 데이터베이스 관리 시스템

▶ Database

- ▶ 데이터의 집합 (a Set of Data)
- ▶ 여러 응용 시스템(프로그램)들의 통합된 정보들을 저장하여 운영할 수 있는 공용(Shared) 데이터의 집합
- ▶ 효율적으로 저장, 검색, 갱신할 수 있도록 데이터 집합들끼리 연관시키고 조직화되어야 한다



데이터베이스와 데이터베이스 관리 시스템

▶ 데이터베이스의 특성

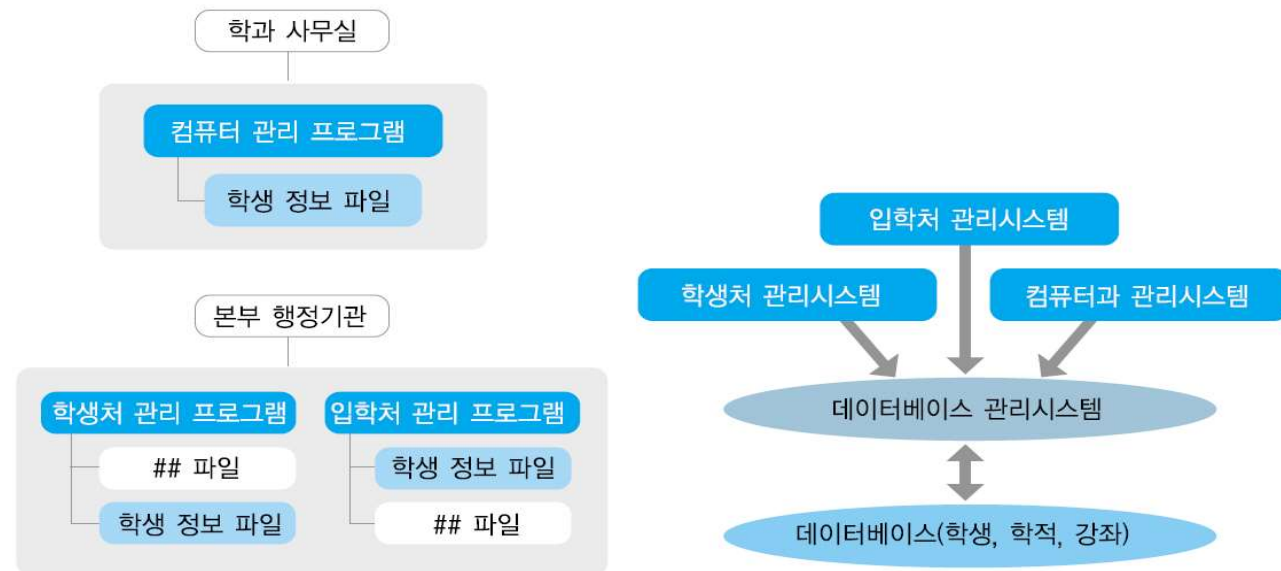
- ▶ 실시간 접근성 (**Real-time Accessibility**)
사용자의 요구를 즉시 처리할 수 있다
- ▶ 계속적인 변화 (**Continuous Evolution**)
정확한 값을 유지하기 위해 삽입, 삭제, 수정 작업 등을 이용하여 데이터를 지속적으로 갱신할 수 있다
- ▶ 동시 공유성 (**Concurrent Sharing**)
사용자마다 서로 다른 목적으로 사용하므로 동시에 여러 사람이 동일한 데이터에 접근하고 이용할 수 있다
- ▶ 내용 참조 (**Contents Reference**)
저장한 데이터 레코드의 위치나 주소가 아닌 사용자가 요구하는 데이터의 내용, 즉 데이터 값에 따라 참조할 수 있어야 한다

데이터베이스와 데이터베이스 관리 시스템

- ▶ 데이터베이스 관리 시스템 (Database Management System = DBMS)
 - ▶ 데이터베이스를 관리하는 소프트웨어
 - ▶ 여러 응용 소프트웨어(프로그램) 또는 시스템이 동시에 데이터베이스에 접근하여 사용할 수 있게 한다
 - ▶ 필수 3기능
 - ▶ 정의 기능 : 데이터베이스의 논리적, 물리적 구조를 정의
 - ▶ 조작 기능 : 데이터를 검색, 삽입, 갱신, 삭제하는 기능
 - ▶ 제어 기능 : 데이터베이스의 내용 정확성과 안전성을 유지하도록 제어하는 기능
 - ▶ Oracle, Microsoft SQL Server, MySQL, DB2 등의 상용 또는 공개 DBMS가 있다

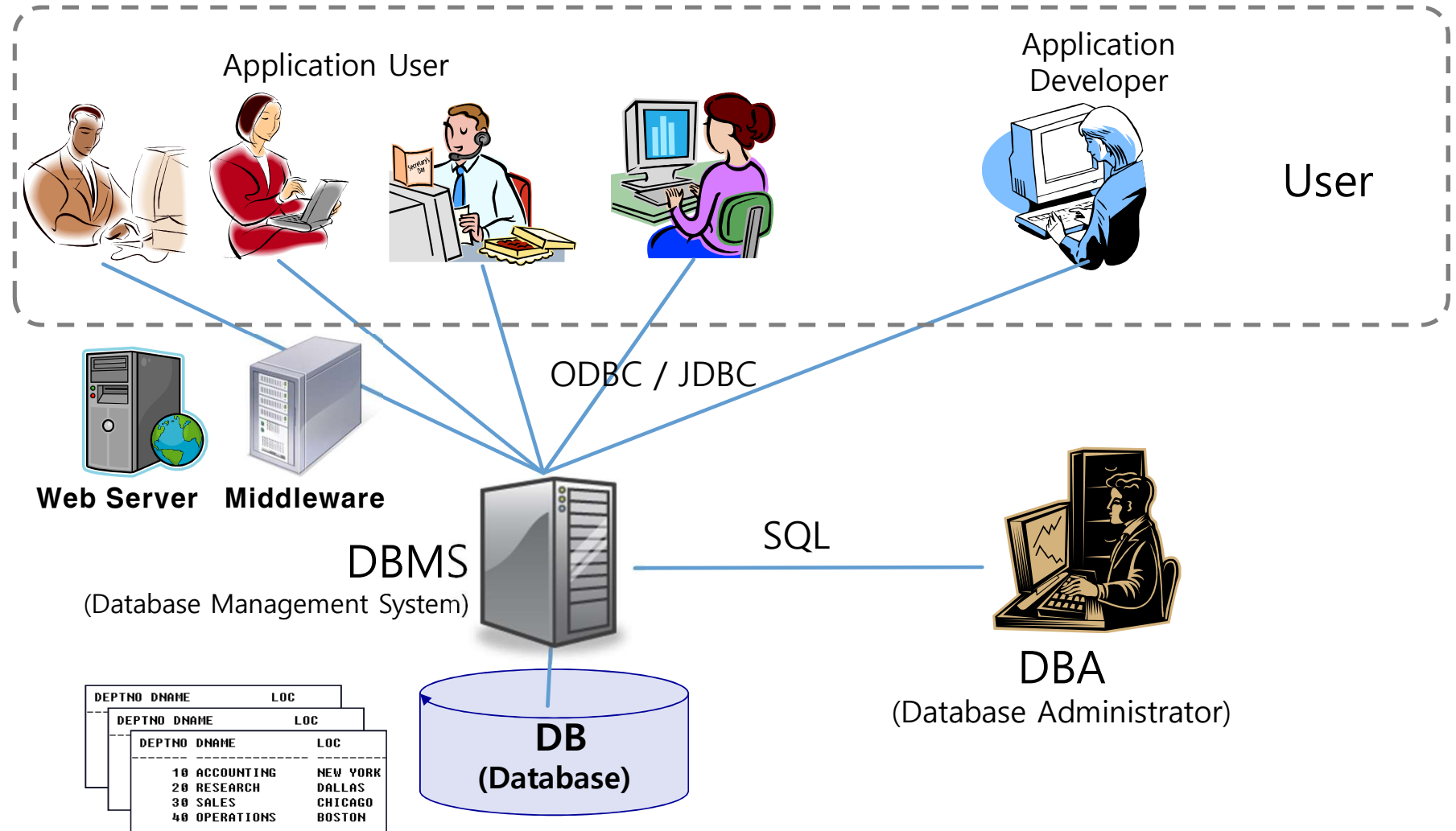
데이터베이스와 데이터베이스 관리 시스템

- ▶ 기존 파일 시스템의 문제
 - ▶ 데이터 종속성으로 인한 문제
 - ▶ 데이터 중복성으로 인한 문제
- ▶ 데이터베이스의 도입
 - ▶ 데이터의 종속성 보완
 - ▶ 중복성 제거



[그림 7-3] 파일관리시스템(좌)과 데이터베이스 관리시스템(우)

데이터베이스와 데이터베이스 관리 시스템



Database Schema

▶ Schema

- ▶ 데이터베이스의 논리적 정의

▶ 3단계 Schema

▶ External Schema (외부 스키마)

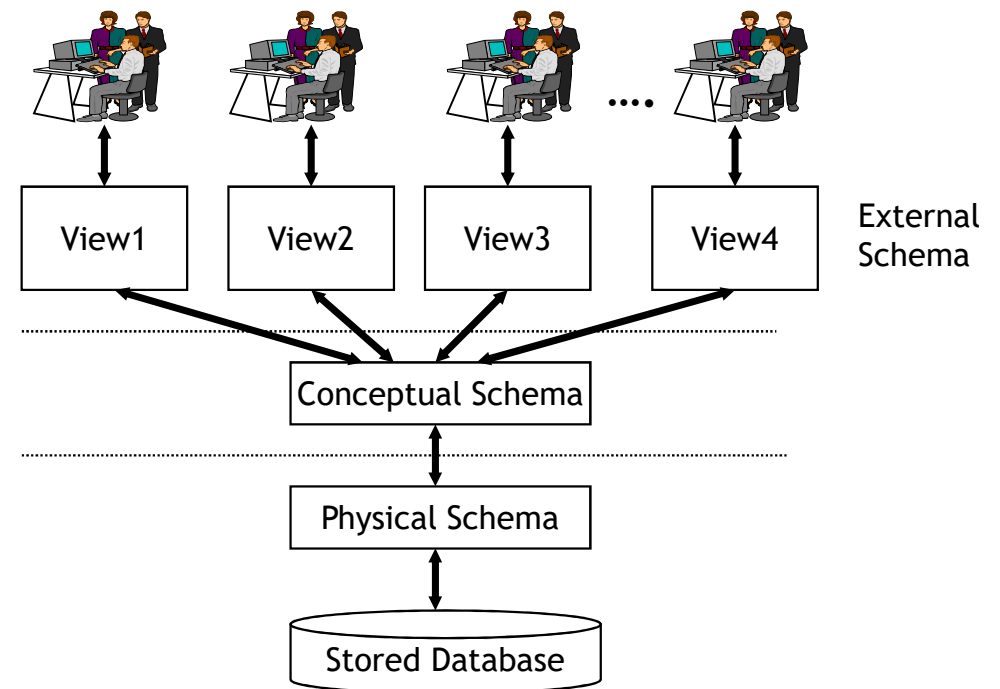
- ▶ 각 사용자 입장에서 본 Database 구조
- ▶ 사용자마다 서로 다른 Schema를 가짐
- ▶ 개념 Schema에 대한 서브 Schema

▶ Conceptual Schema (개념적 스키마)

- ▶ 조직 전체의 입장에서 본 Database 구조
- ▶ 한 개의 Schema만 존재하며, 서로 다른 사용자가 공유
- ▶ Data 객체(개체, 관계), 제약 조건에 대한 명세를 유지

▶ Physical Schema (물리적 스키마)

- ▶ 저장 장치의 입장에서 본 Database 구조
- ▶ 각 data 객체의 저장 구조를 표현
- ▶ 내부 레코드의 형식
- ▶ 저장 data 항목의 표현 방법



데이터베이스와 데이터베이스 관리 시스템

▶ 데이터베이스 관리 시스템의 장점

- ▶ 데이터 독립성 및 중복이 최소화
- ▶ 데이터의 일관성 및 무결성 유지
- ▶ 데이터 보안 보장
- ▶ 표준화되고 일관된 데이터 관리 기능
- ▶ 응용프로그램 개발 시간의 단축
- ▶ 데이터 동시 사용 가능
- ▶ 데이터 회복 가능

▶ 데이터베이스 관리 시스템의 단점

- ▶ 시스템 자원 요구로 운영비 증대
- ▶ 고급 프로그래밍 필요로 자료 처리의 복잡화
- ▶ 부분적 데이터베이스 손실이 전체 시스템을 정지
- ▶ 장애 발생 대비를 위한 복잡한 **Back Up**과 **Recovery** 작업 필요

데이터베이스와 데이터베이스 관리 시스템

▶ 데이터베이스의 종류

▶ 관계형 데이터베이스 (Relational Database = RDB)

- ▶ 1970년 IBM E. F. Codd에 의해 제안되어 수십년 동안 주류 데이터베이스로 성장 확대
- ▶ 키와 값들의 간단한 관계를 테이블화 시킨 매우 간단한 원칙의 개념을 가진 데이터베이스
- ▶ 일련의 정형화된 테이블로 구성된 데이터 항목들의 집합이며 각 테이블은 데이터의 성격에 따라 여러 개의 컬럼(키)이 포함된다
- ▶ 사용자는 SQL이라는 표준 질의어를 통해 데이터를 조작 또는 조회할 수 있다
 - ▶ SQL (Structured Query Language)

▶ 객체지향 데이터베이스 (Object Oriented Database = OODB)

- ▶ 정보를 객체의 형태로 표현하는 데이터베이스
- ▶ 객체 모델이 그대로 데이터베이스에도 적용되어 데이터 모델을 그대로 응용프로그램에 적용, 데이터 변환과 질의 작업이 필요하지 않은 장점이 있다

데이터베이스와 데이터베이스 관리 시스템

▶ 데이터베이스의 종류

▶ 객체관계형 데이터베이스 (Object Relation Database = ORDB)

- ▶ 관계형 데이터베이스에서 사용하는 데이터를 확장
- ▶ 관계형 데이터베이스를 객체 지향 모델링과 데이터를 관리하는 기능을 갖도록 확장한 것

▶ NoSQL

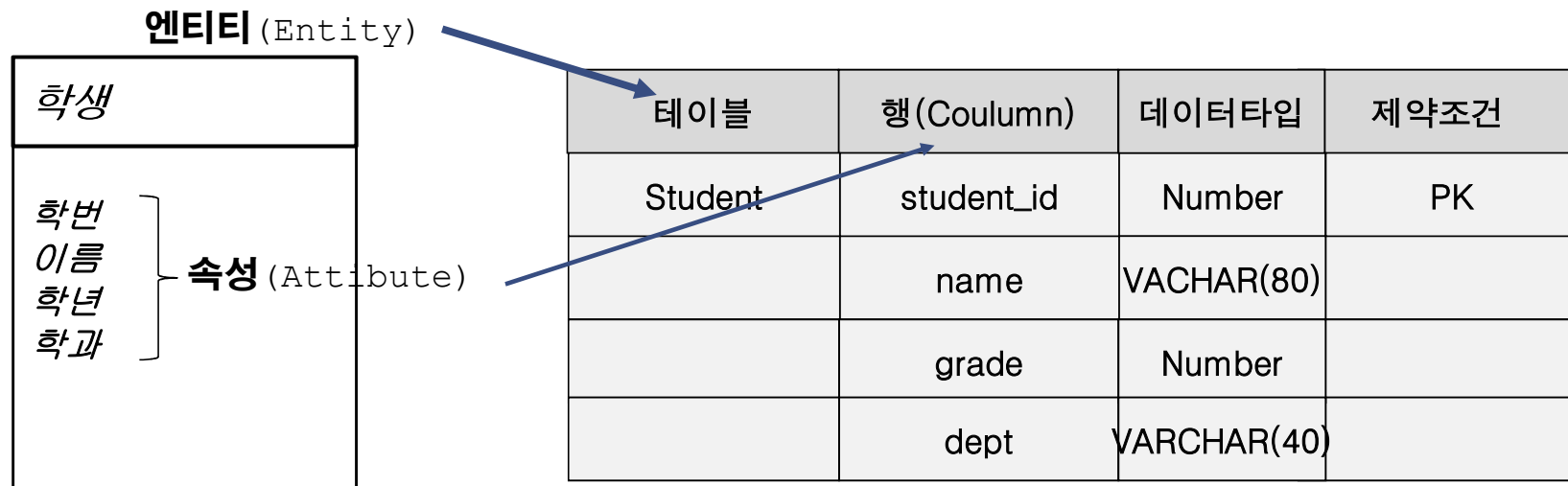
- ▶ 대용량 데이터, 비정형 데이터의 웹 서비스와 SNS, 클라우드 컴퓨팅의 확대 보급과 대중화로 최근 주목받고 있는 데이터베이스 기술

▶ 그 외

- ▶ Hierarchical Database
- ▶ Network Database

관계형 데이터베이스

▶ 논리적(개념적) 데이터 모델링과 물리적인 데이터베이스



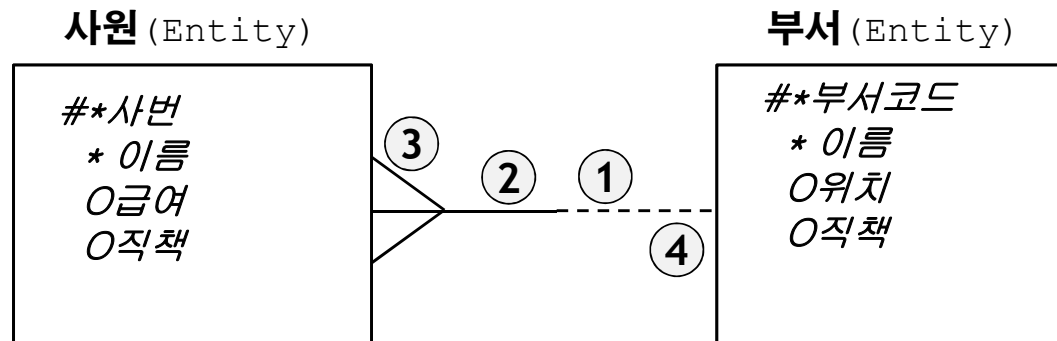
▶ RDBMS의 이해

▶ Relation 또는 2차원 Table을 이용하여 정보 저장

- ▶ Relation = Table 간의 연관
- ▶ Attribute = Column
- ▶ Tuple = Row

관계형 데이터베이스

▶ ERD 예시를 통해 보는 관계



: 대표값 - PK(Primary Key)
• : NOT NULL
O : NUL 가능

1. 어떤 부서는 사원을 배치 받지 않을 수 있다 (점선)
2. 사원은 특정 부서에 소속되어 있다 (실선)
3. 한 부서에는 여러 명의 사원이 소속되어 있다 (다중선)
4. 한 사원은 하나의 부서에만 속한다 (단일선)

관계형 데이터베이스

▶ 테이블

Table Student

column(Attribute)

student_id	name	grade	dept
1	정성진	1	컴퓨터
2	박현진	2	수학
3	홍길동	4	물리
...

field

field

row(record)

테이블 : RDBMS의 기본적 저장 구조. 한 개 이상의 **column**과 0개 이상의 **row**로 구성

열(**Column**) : 테이블상에서의 단일 종류의 데이터를 나타냄. 특정 데이터 타입 및 크기를 가지고 있음

행(**Row**) : **Column** 값들의 조합. 튜플, 레코드라고 함

기본키(**PK**)에 의해 구분된다. 기본키는 중복을 허용하지 않으며 없어서는 안된다

Field : **Row**와 **Column**의 교차점으로 **Field**는 데이터를 포함할 수 있고 없을 때는 **NULL** 값을 가지고 있다

관계형 데이터베이스

: 무결성 제약조건(Integrity Constraint)

▶ 개체 무결성(Entity Integrity)

- ▶ Table은 중복된 ROW를 가질 수 없으며 모든 Table은 각각의 ROW를 유일하게 식별할 수 있는 Column의 집합을 가진다. 이러한 Column의 집합 중에서 대표되는 컬럼을 Primary Key(PK)로 정의한다
- ▶ Primary Key의 값은 항상 유일(Unique)하며 널(Null)을 허용해서는 안된다

▶ 참조 무결성(Referential Integrity)

- ▶ Table들은 Foreign Key(FK)를 통해 서로 연결되어 있다. Foreign Key는 다른 Table 또는 자신 Table의 PK 값을 참조하기 위해 복사하여 가지고 있는 Column을 말한다
- ▶ 참조 무결성이 지켜지기 위해서 FK Column의 값은 참조하는 PK 컬럼 값 중의 하나이거나 Null이어야 한다

▶ 무결성 제약은 DBMS 시스템이 자동으로 수행한다

관계형 데이터베이스

▶ Primary Key (PK) : 기본키

- ▶ 관계(Relation)에서 튜플을 구분하기 위하여 사용하는 기본 키
- ▶ 하나의 애트리뷰트, 또는 애트리뷰트의 집합(복합키) 가능
- ▶ 관리자에 의해 릴레이션 생성시 정의된 (자동으로 Index 생성)
- ▶ 동일한 PK를 지닌 레코드는 존재할 수 없음

▶ 기타

- ▶ **Candidate Key (후보키)** : 튜플을 식별할 수 있는 최소한의 애트리뷰트 집합
 - ▶ 하나의 릴레이션에는 PK가 될 수 있는 키가 여러 개 있을 수 있음
 - ▶ 유일성과 희소성이 있으면 **Candidate Key**가 될 수 있음
- ▶ **Alternative Key (대체키)** : 후보키 중 기본 키가 아닌 것
- ▶ **Composite Key (복합키)** : 둘 이상의 애트리뷰트가 하나의 **Key**를 이루는 것

관계형 데이터베이스

▶ 학생

- ▶ 기본키 : 학번
- ▶ 후보키 : 기본키와 동일
- ▶ 주민등록번호 등이 있다면 후보키가 될 수 있음

학생
(STUDENT)

<u>학번</u> (Sno)	이름 (Sname)	학년 (Year)	학과 (Dept)
100	나 수 영	4	컴퓨터
200	이 찬 수	3	전기
300	정 기 태	1	컴퓨터
400	송 병 길	4	컴퓨터

▶ 등록

- ▶ 기본키 : (학번, 과목번호)
 - ▶ 후보키 : 기본키와 동일
 - ▶ 학번이나 과목번호만으로는 키가 되지 못함
-
- ▶ 등록번호와 같이 별도의 단일키를 추가하여 **PK**로 지정할 수도 있음

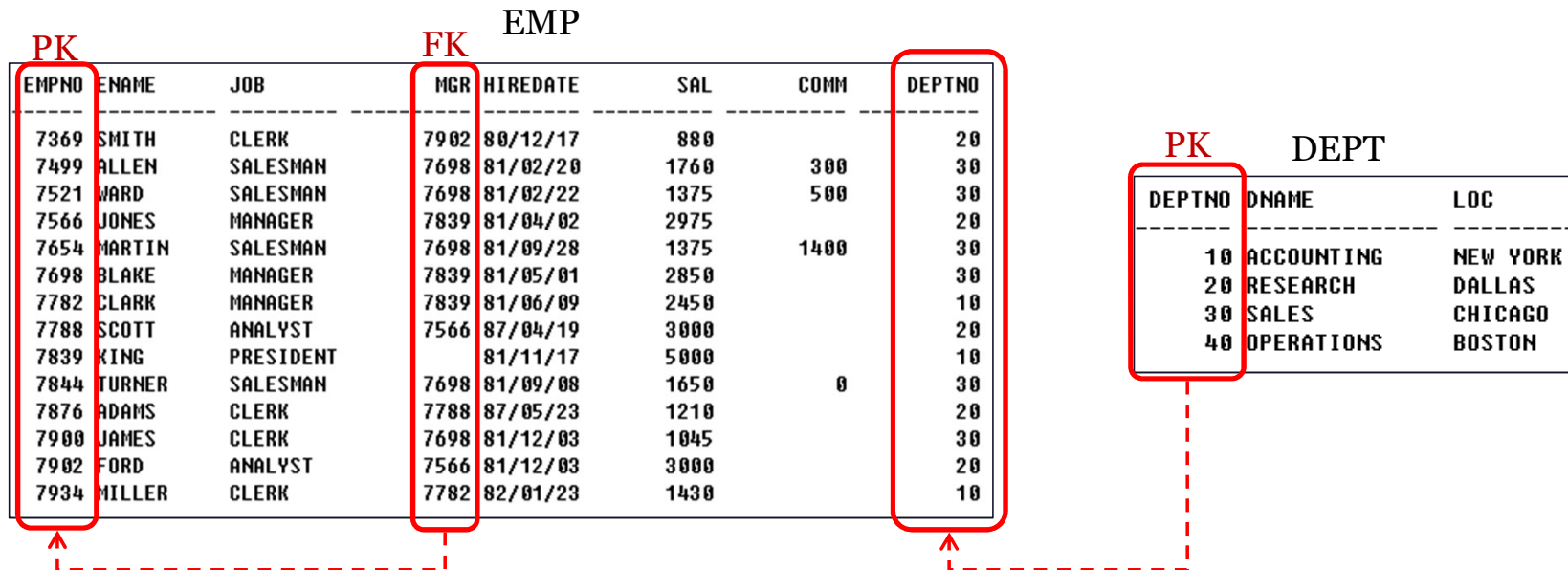
등록
(ENROL)

<u>학번</u> (Sno)	<u>과목번호</u> (Cno)	성적 (Grade)
100	C413	A
100	E412	A
200	C123	B
300	C312	A

관계형 데이터베이스

▶ Foreign Key (FK) : 외래키

- ▶ 기본키를 참조하는 애트리뷰트
- ▶ 다른 릴레이션의 튜플을 대표
- ▶ 릴레이션 간의 관계를 나타내기 위해 사용
- ▶ NULL 가능 (참조되지 않음을 의미)



SQL(Structured Query Language) 개요

- ▶ 데이터베이스 스키마 생성, 자료의 검색, 수정, 그리고 데이터베이스 객체 접근 관리 등을 위해 고안된 언어
 - ▶ 비절차식 언어
 - ▶ 1970년대 IBM의 SYSTEM R 프로젝트를 통해 개발
- ▶ 다수의 데이터베이스 관련 프로그램의 표준 언어
 - ▶ RDBMS에서 사용하기 위해 ANSI에서 책정한 표준 언어
 - ▶ DBMS 제품별로 SQL에 대한 추가 및 확장
- ▶ PL/SQL
 - ▶ 응용프로그램의 logic을 추가하여 SQL을 확장한 Oracle의 절차적 언어
 - ▶ 3세대 언어로 IF 문장이나 LOOP 문장을 통해 프로그램의 흐름을 제어할 수 있으며 SQL 문장을 이용 data 조작 가능
 - ▶ Client가 PL/SQL 블록 실행을 요청하면 Oracle 서버는 Block 전체를 실행한 후 결과를 리턴하므로 한번의 서버 호출로 다량의 SQL과 로직을 구사할 수 있다는 장점
- ▶ SQL*Plus
 - ▶ SQL 및 PL/SQL 문장을 인식하고 실행시켜주는 Oracle의 Tool
 - ▶ SQL, PL/SQL 등을 직접 입력하여 Oracle 서버로 보내 실행한 후 결과를 받아본다

SQL(Structured Query Language) 개요

: SQL 명령어의 종류

▶ DML(Data Manipulation Language)

- ▶ 데이터 처리를 위해 응용프로그램과 데이터베이스 관리 시스템간의 인터페이스를 위한 언어
- ▶ 데이터 처리를 위한 연산의 집합으로 데이터의 검색, 삽입, 수정 및 삭제하기 위한 수단을 제공
- ▶ SELECT, INSERT, UPDATE, DELETE, MERGE

▶ DDL(Data Definition Language)

- ▶ 데이터베이스 구조, 데이터 형식, 접근 방식 등 데이터베이스를 구축, 변경할 목적으로 사용하는 언어
- ▶ DDL 컴파일러가 컴파일한 후 데이터 사전에 저장
- ▶ 데이터의 논리적, 물리적 구조를 생성, 변경, 삭제하는 기능을 제공
- ▶ CREATE, ALTER, DROP, RENAME

▶ DCL(Data Control Language)

- ▶ 보안 및 권한 제어, 무결성, 회복, 병행 제어를 위한 언어
- ▶ : 데이터에 대한 권한 관리 및 트랜잭션 제어
- ▶ GRANT, REVOKE 등

SQL*Plus 기초

: Login

- ▶ Windows 환경에서의 로그인
 - ▶ Windows 메뉴 > Run SQL Command Line
 - ▶ 프롬프트에서 `CONN 계정명/[PASSWORD]`
 - ▶ Windows 커맨드 프롬프트에서
 - ▶ `SQLPLUS 계정명/[PASSWORD]`
- ▶ 접속을 끊고 SQL*Plus를 종료하려면 `EXIT` 명령을 실행

SQL*Plus 기초

- ▶ SQL*Plus에서 SQL 문장 작성시 유의사항
 - ▶ 모든 SQL 문장은 세미콜론(;)으로 끝난다
 - ▶ SQL 문장은 한 줄로 입력하거나 여러 줄로 보기 좋게 나누어 입력한다
 - ▶ SQL 문장은 대소문자를 구분하지 않는다 (Case Insensitive)
 - ▶ data 값은 대소문자를 가린다 (Case Sensitive)
- ▶ SQL*Plus의 결과값 출력
 - ▶ Default Column Heading
 - ▶ Column 명이 대문자로 출력된다
 - ▶ Default Data Justification
 - ▶ Number 값: right-justified
 - ▶ Character, Date 값: left-justified

Installation

OracleXE 11g와 SQL Developer 설치

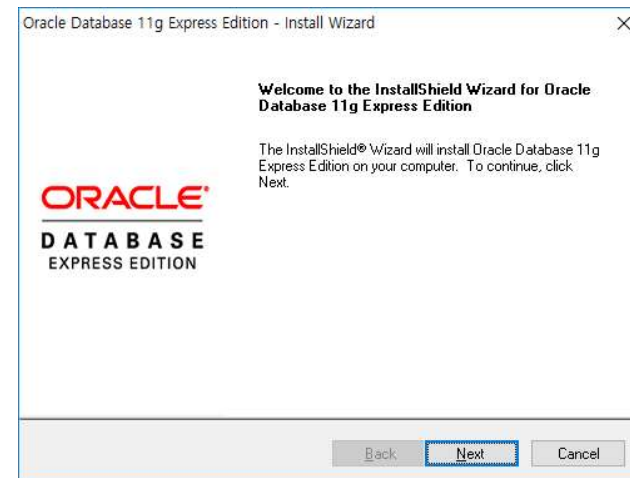
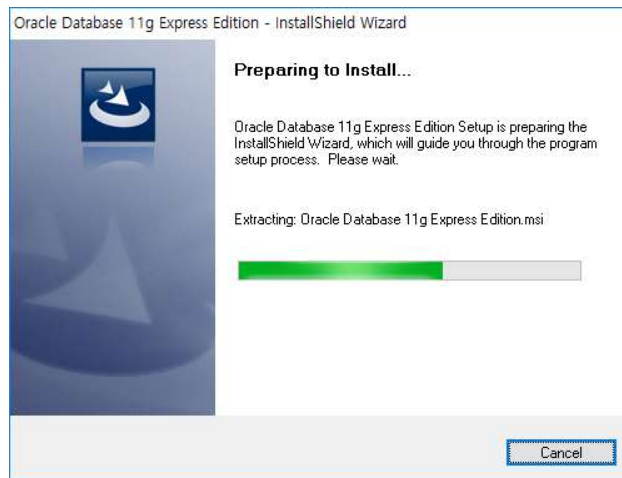
Installation

: OracleXE 11g

- ▶ Oracle 홈페이지에서 오라클 XE 11g 버전을 다운로드
(Oracle Database Express Edition 11g Release 2)
- ▶ 직접 다운로드 경로
<http://www.oracle.com/technetwork/database/database-technologies/express-edition/downloads/index.html>
- ▶ 압축을 풀고 **setup** 파일을 실행

Installation : OracleXE 11g

▶ setup.exe 설치 파일 실행



Next 클릭

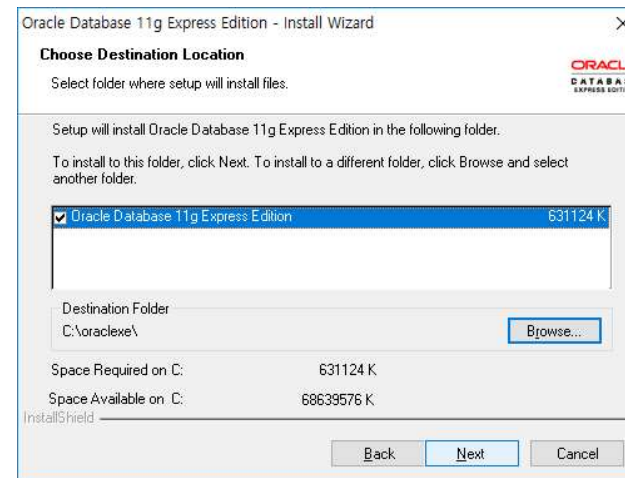
Installation

: OracleXE 11g

▶ 라이선스 동의 및 설치 폴더 지정



Next 클릭

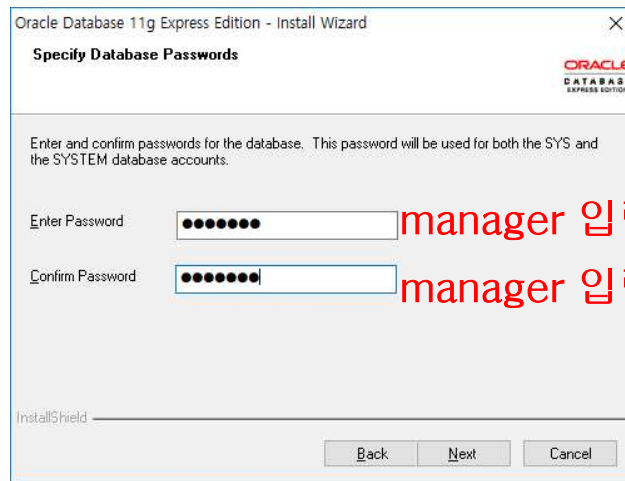


Next 클릭

Installation

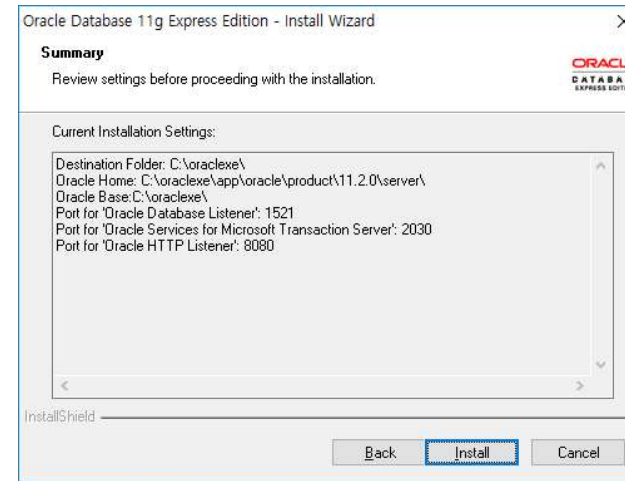
: OracleXE 11g

▶ sys, system 계정 비밀번호 설정



manager 입력
manager 입력

Next 클릭

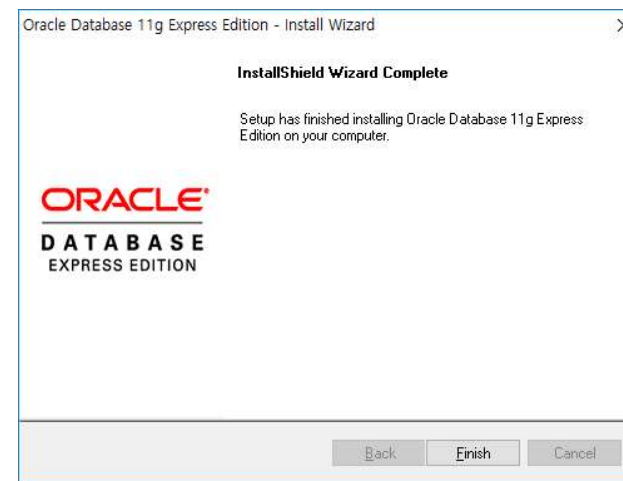
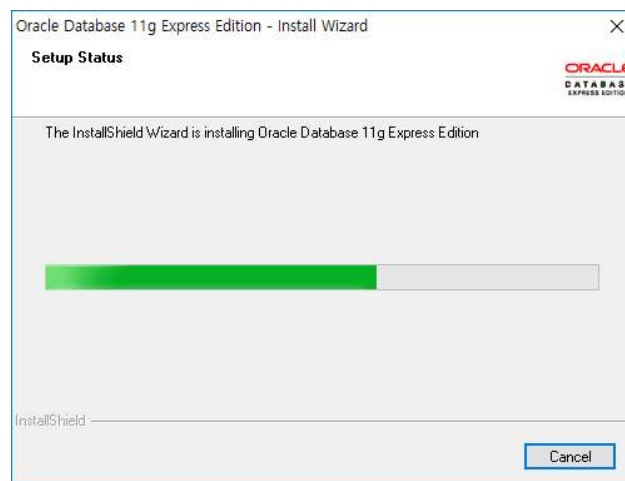


Next 클릭

수업진행을 위해 비밀번호는 반드시 manager 로 설정

Installation : OracleXE 11g

▶ 설치 완료



Install

: Change APEX(Oracle Web Admin) PORT

- ▶ Oracle 11g XE는 기본적으로 APEX라 불리는 Web Admin을 제공
- ▶ 하지만, 기본 포트가 8080으로 되어 있어 Apache Tomcat의 기본 포트와 충돌 발생
- ▶ Apache Tomcat 을 함께 사용하기 위한 환경이라면 이 기능을 사용하지 않거나 포트를 변경해 주어야 함

```
SQL> connect system/manager
Connected.

SQL> EXEC DBMS_XDB.SETHTTPPORT(8088);

PL/SQL procedure successfully completed.

SQL>
```

Installation

: SQL Developer

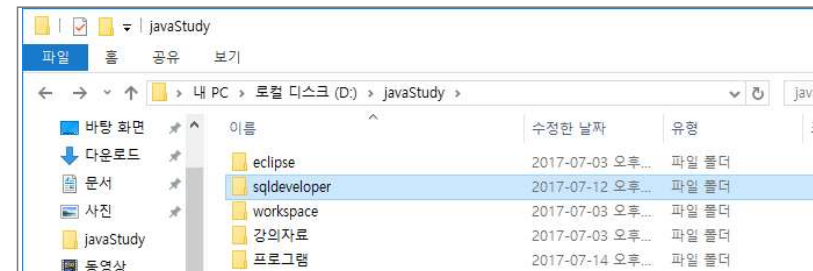
- ▶ 오라클 웹사이트에서 **SQL Developer** 프로그램을 다운
“JDK가 포함되지 않은 버전 “



- ▶ 직접 다운로드 경로

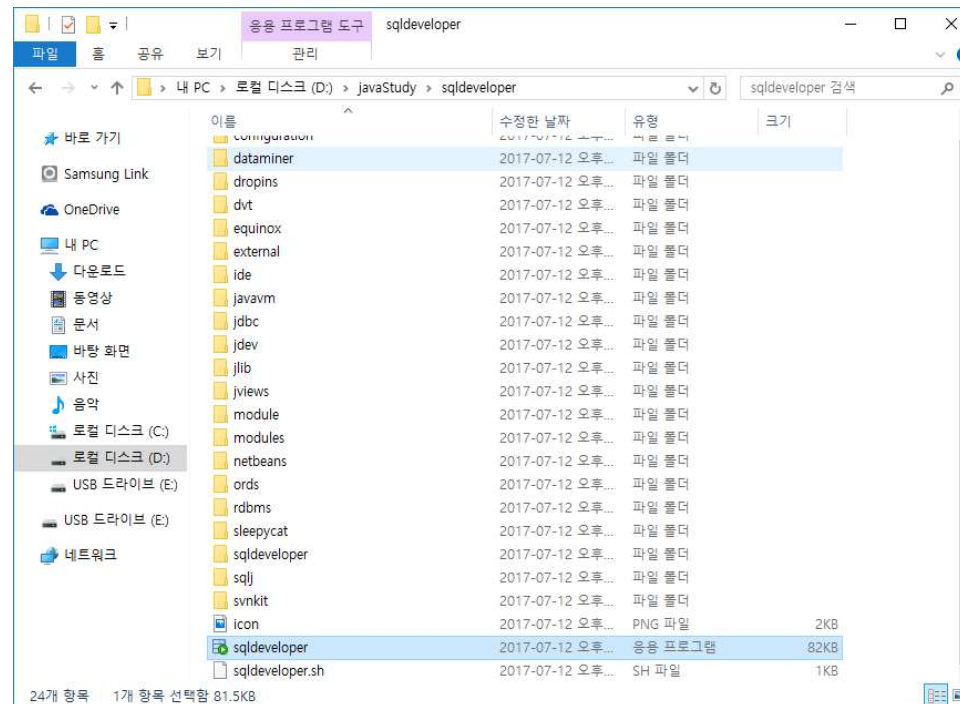
<http://www.oracle.com/technetwork/developer-tools/sql-developer/downloads/index.html>

- ▶ 사용
설치 버전이 아니므로
D:\javaStudy 에 폴더 복사 후 이용



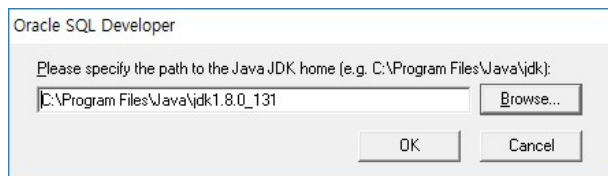
Installation : SQL Developer

- ▶ D:\javaStudy\sqldeveloper\sqldeveloper.exe 파일 실행



Installation : SQL Developer

▶ JDK 연결



자신의 jdk디렉토리 선택 후
OK 클릭

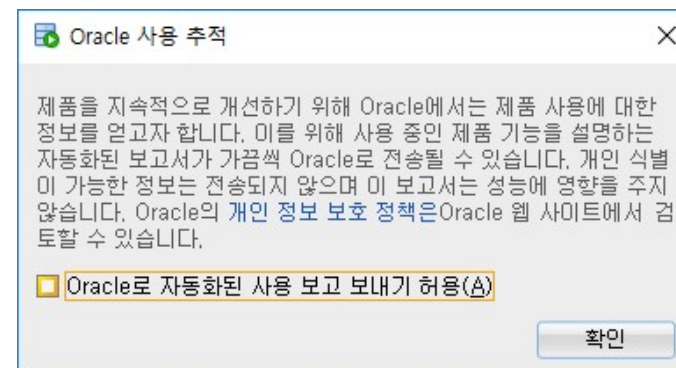


Installation : SQL Developer

▶ 이전 버전 확인 및 사용 추적 옵션



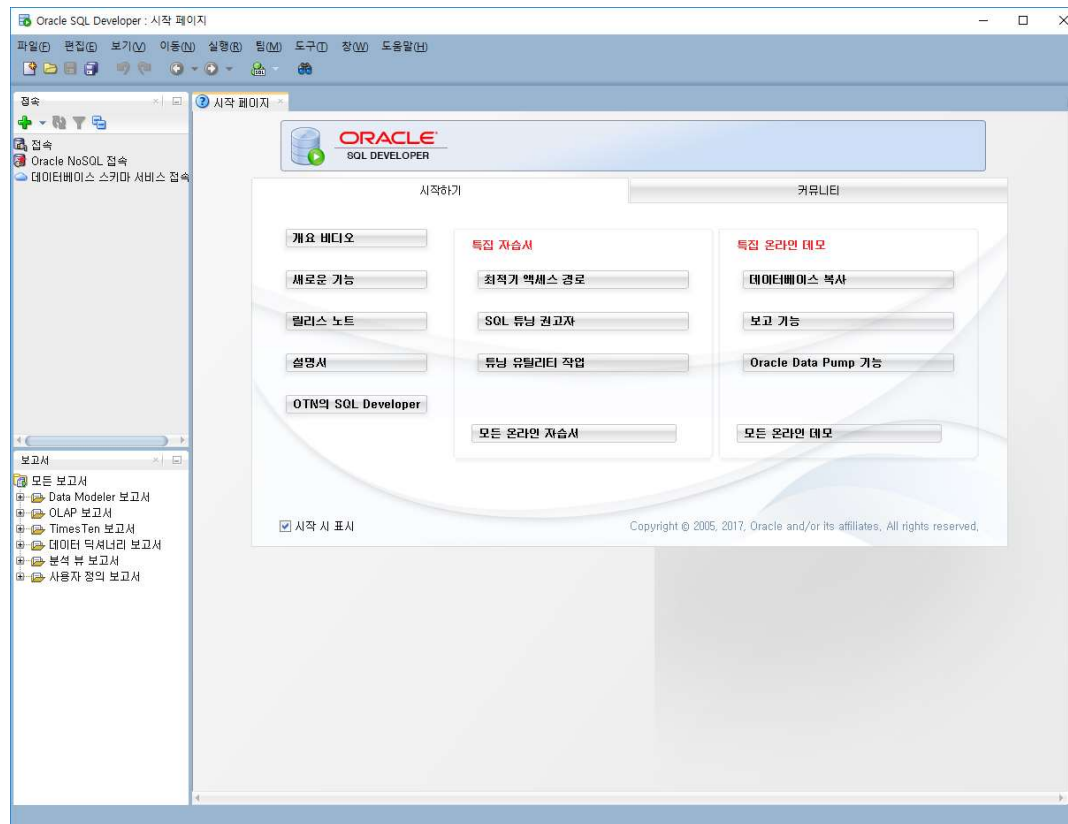
아니오 클릭



확인 클릭

Installation : SQL Developer

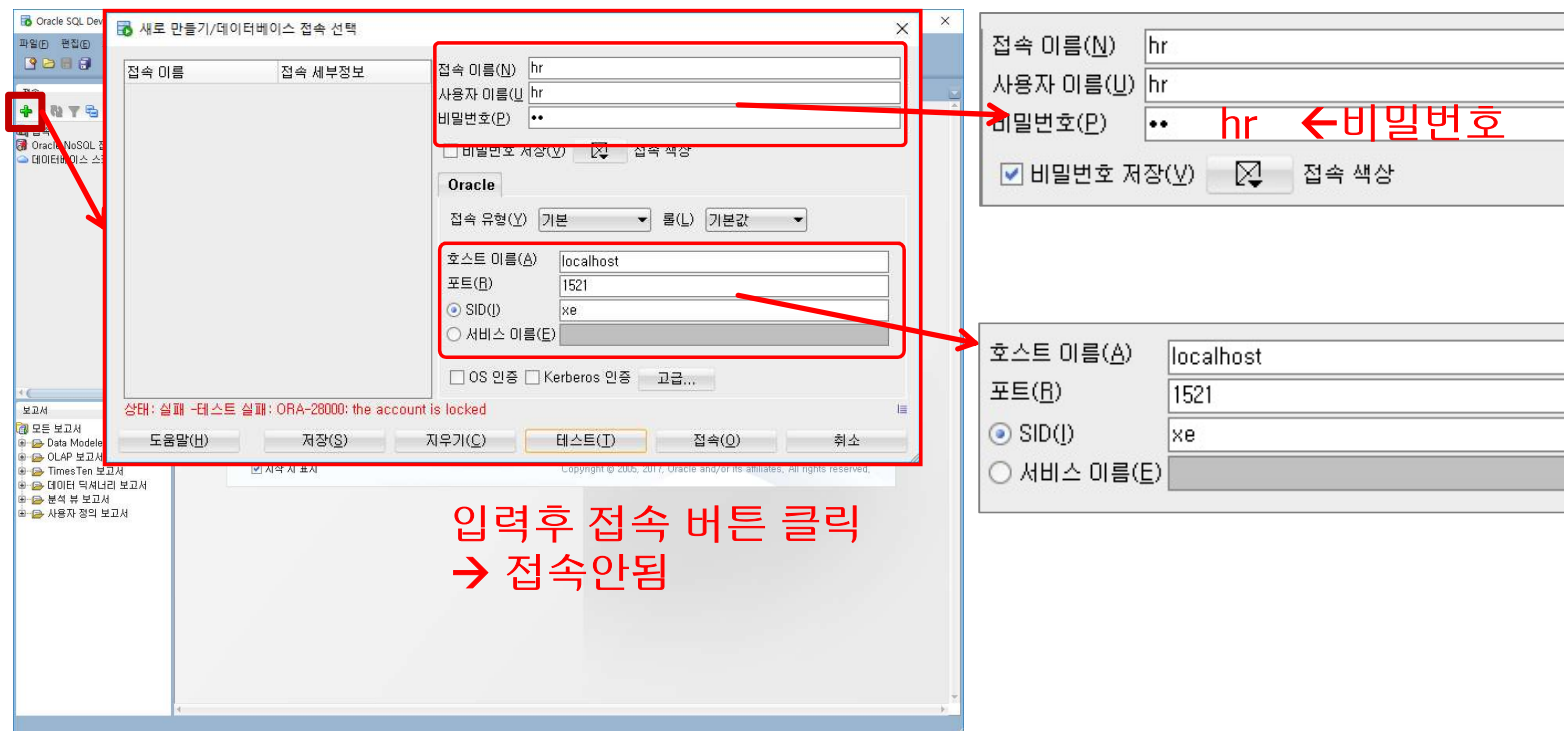
▶ 실행 화면 예시



Installation

: 데이터베이스 접속 시도

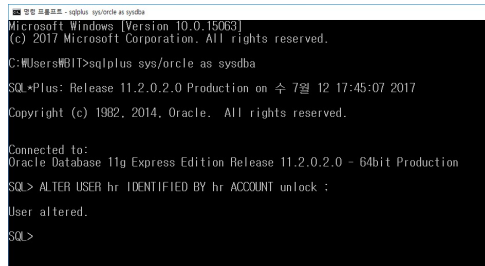
▶ 실행 화면 예시



Installation

: 데이터베이스 접속 시도

▶ hr 계정 풀기



```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\BIT>sqlplus sys/oracle as sysdba

SQL*Plus: Release 11.2.0.2.0 Production on 수 7월 12 17:45:07 2017
Copyright (c) 1982, 2014, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production
SQL> ALTER USER hr IDENTIFIED BY hr ACCOUNT unlock ;
User altered.
SQL>
```

콘솔창에서
빨간색 텍스트 입력 후
실행

```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.
```

```
C:\Users\BIT>sqlplus sys/oracle as sysdba
```

```
SQL*Plus: Release 11.2.0.2.0 Production on 수 7월 12
17:45:07 2017
```

```
Copyright (c) 1982, 2014, Oracle. All rights reserved.
```

```
Connected to:
Oracle Database 11g Express Edition Release 11.2.0.2.0 -
64bit Production
```

```
SQL>ALTER USER hr IDENTIFIED BY hr ACCOUNT unlock ;
```

```
User altered.
```

```
SQL>
```

Installation

: 데이터베이스 접속 시도

▶ hr 계정 접속

새로 만들기/데이터베이스 접속 선택

접속 이름	접속 세부정보
hr	hr@//localhost:1521/xe

접속 이름(N) hr
사용자 이름(U) hr
비밀번호(P) **

☒ 비밀번호 저장(Y) ☐ 접속 색상

Oracle

접속 유형(Y) 기본 롤(L) 기본값

호스트 이름(A) localhost
포트(B) 1521
☒ SID(I) xe
☐ 서비스 이름(E)

☐ OS 인증 ☐ Kerberos 인증 고급...

상태: 성공

도움말(H) 저장(S) 지우기(C) **테스트(T)** 접속(O) 취소

접속 클릭

Installation

: SCOTT 계정 설정

- ▶ 11g XE에는 11g 정식버전에 있는 테스트 계정 SCOTT이 없음
- ▶ C:\를 선택해 XE를 설치했다면 SCOTT 계정 생성 파일 경로는 `C:\oraclexe\app\oracle\product\11.2.0\server\rdbms\admin\scott.sql` (반드시 탐색기에서 확인)
- ▶ sysdba 권한으로 SQL*Plus에 접속
SQL*Plus 접속 상태에서 위 파일을 실행
파일 실행시에는 전체 경로 앞에 @을 붙인다

```
C:\Users\BIT>sqlplus sys/oracle as sysdba

. . .

SQL>@C:\oraclexe\app\oracle\product\11.2.0\server\rdbms\admin\scott.sql;
SQL>
```

Installation

: SCOTT 계정 설정

- ▶ 스크립트 실행 후 성공하면 에러 없이 커서가 바로 떨어짐
- ▶ 이 상태에서 SQL*Plus에 접속된 사용자를 확인

```
SQL> show user  
USER is 'SCOTT'  
SQL>
```

- ▶ SCOTT 계정의 비밀번호를 tiger로 변경한다

```
SQL> alter user scott identified by tiger;  
USER is 'SCOTT'  
SQL>
```

- ▶ SQL*Plus에서 빠져나오려면 exit를 사용

```
SQL> exit  
Disconnected from Oracle Database 11g Express Edition Release  
11.2.0.2.0 - 64bit Production  
C:\Users\BIT>
```

- ▶ 이 과정으로 SCOTT 계정이 만들어지고 SCOTT 계정 내의 테스트용 테이블들이 만들어짐

Installation

: SCOTT 계정 설정

▶ [실습] SQL*Plus를 이용, SCOTT 계정으로 접속하여 SCOTT 계정 테이블을 확인해 봅시다

▶ 계정 내 테이블 확인:

▶ `SELECT * from tab;`

```
C:\Users\BIT> sqlplus scott/tiger
```

```
Connected to: Oracle Database 11g Express Edition  
Release 11.2.0.2.0 - 64bit Production
```

```
SQL> select * from tab;
```

TNAME	TABTYPE
BONUS	TABLE
DEPT	TABLE
EMP	TABLE
SALGRADE	TABLE

```
SQL>
```


Installation

: SCOTT 계정 설정

- ▶ [실습] SCOTT 계정 내 DEPT 테이블 구조를 확인해 봅시다

```
SQL> DESC DEPT;
```

Name	Null?	Type
DEPTNO	NOT NULL	NUMBER(2)
DNAME		VARCHAR2(14)
LOC		VARCHAR2(13)

```
SQL>
```

- ▶ 테이블 구조 확인
 - ▶ DESC <테이블명>