

인문정보 데이터베이스

제4강: SQL의 이해

김 현

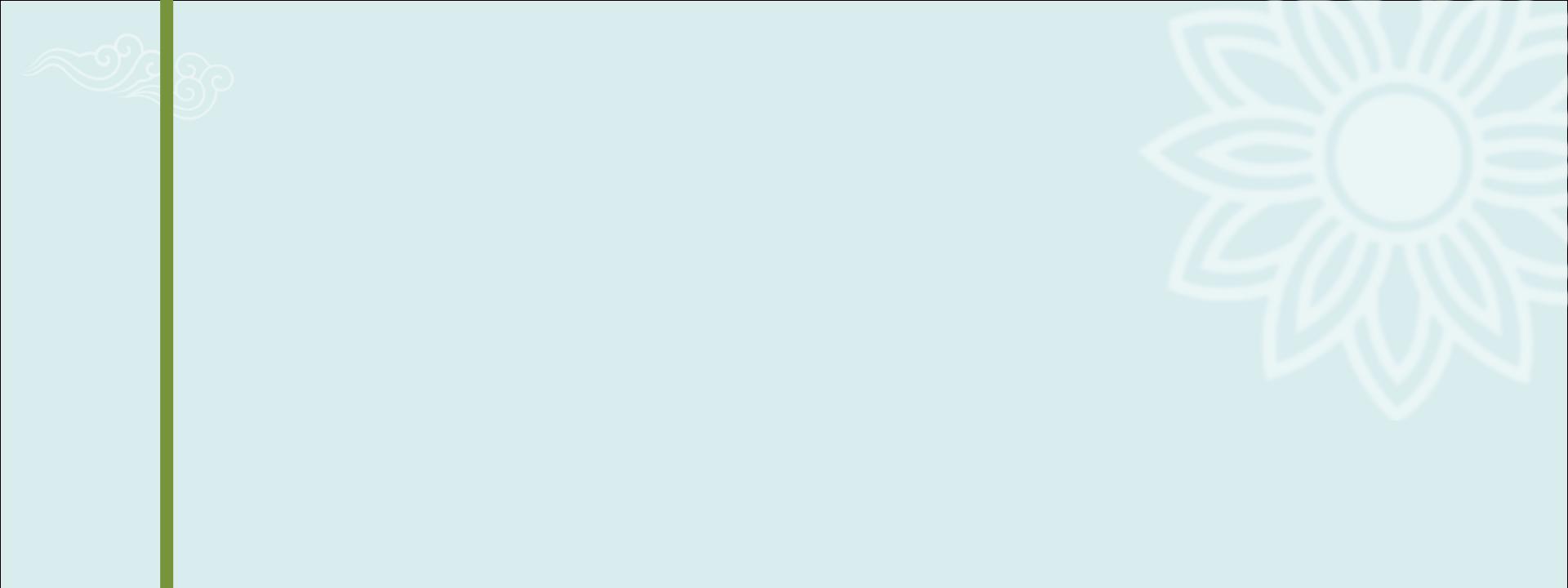
한국학중앙연구원 인문정보학고실

hyeon@aks.ac.kr



이 저작물(PPT)의 인용 표시 방법:

김현, 「인문정보 데이터베이스」, 한국학중앙연구원 한국학대학원 수업 자료 (2020),



1. SQL이란?

2. SQL 실습 (데이터 정의)

SQL (Structured Query Language)

- ❖ 데이터베이스의 검색과 조작을 쉽고 빠르게 하기 위해 개발된 데이터베이스 언어
- ❖ 1970년대 초반 IBM사의 관계형 데이터베이스 질의어로 개발
 - 1973 SQUARE: Structured Queries as Relational Expressions
 - 1974 SEQUEL: Structured English as Query Language
 - 1976 SQL: Structured Query Language
- ❖ DBMS마다 상이한 형태의 질의어를 제공하는 데서 오는 사용자의 불편 해소 필요성 대두 ⇒ 데이터베이스 질의어 표준화 요구 발생
- ❖ 1986년 ANSI 와 ISO가 공동으로 SQL의 표준화 작업을 진행(SQL-86)
- ❖ 관계형 DB를 위한 SQL2(1992년), 객체지향형 DB를 위한 SQL3(1999), XML 데이터를 처리하기 위한 SQL4(2003), 멀티미디어 정보를 위한 SQL/MM 등 표준화 작업 진행

- 데이터가 저장되는 테이블(릴레이션)을 생성, 변경, 삭제하는 기능
- 테이블에 저장된 데이터에 빨리 접근하도록 인덱스(index)를 정의하는 기능
- 가상 테이블: 뷰(view)를 정의할 수 있는 기능

SQL Structured Query Language

- 테이블에 데이터를 삽입하거나, 삭제하고, 수정할 수 있는 기능
- 저장 데이터 중 조건에 맞는 것들을 검색하는 기능.

데이터 정의어 (DDL : Data Definition Language)

- 테이블 생성 기능(create table)
- 테이블 변경 기능(alter table)
- 테이블 삭제 기능(drop table)
- 뷰 생성 기능(create view)
- 뷰 삭제 기능(drop view)

데이터 조작어 (DML : Data Manipulation Language)

- 데이터 검색 기능(select)
- 데이터 삽입 기능(insert)
- 데이터 삭제 기능(delete)
- 데이터 수정 기능(update)

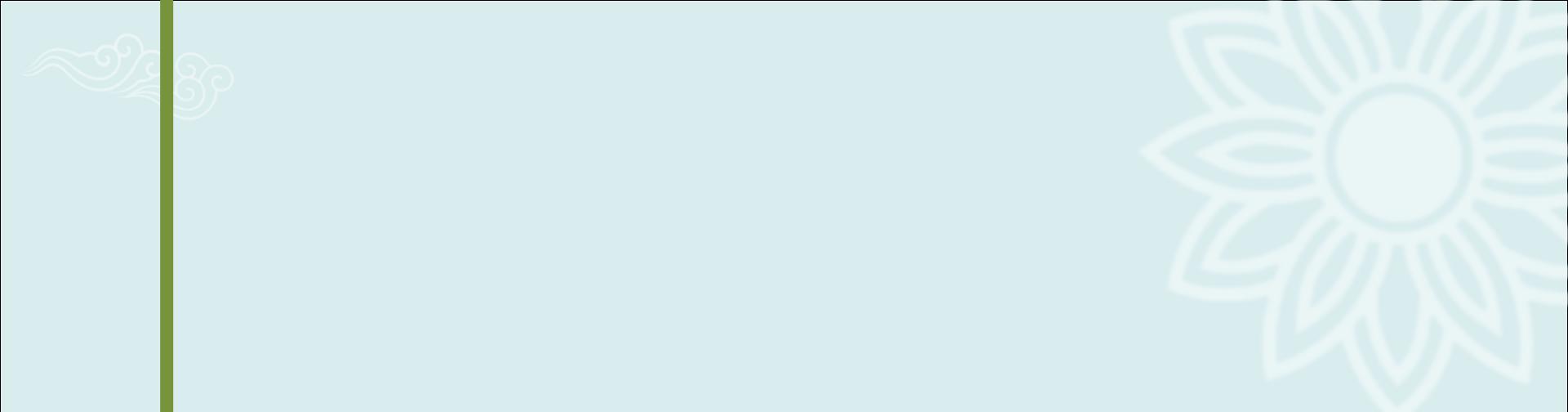
SQL 문의 기본 구조

select 표시 열 from 테이블 where 행 제약 조건 order by 기준 열

- ❖ select 이름, 생년, 몰년 from 인명 where 생년 > 1700 and 몰년 < 1900 order by 생년
- ☞ ‘인명’ 테이블에서 1700년 이후에 태어나고 1900년 이전에 죽은 사람을 찾아 그들의 ‘이름’, ‘생년’, ‘몰년’을 ‘생년’ 순서대로 보여라

MS SQL Server

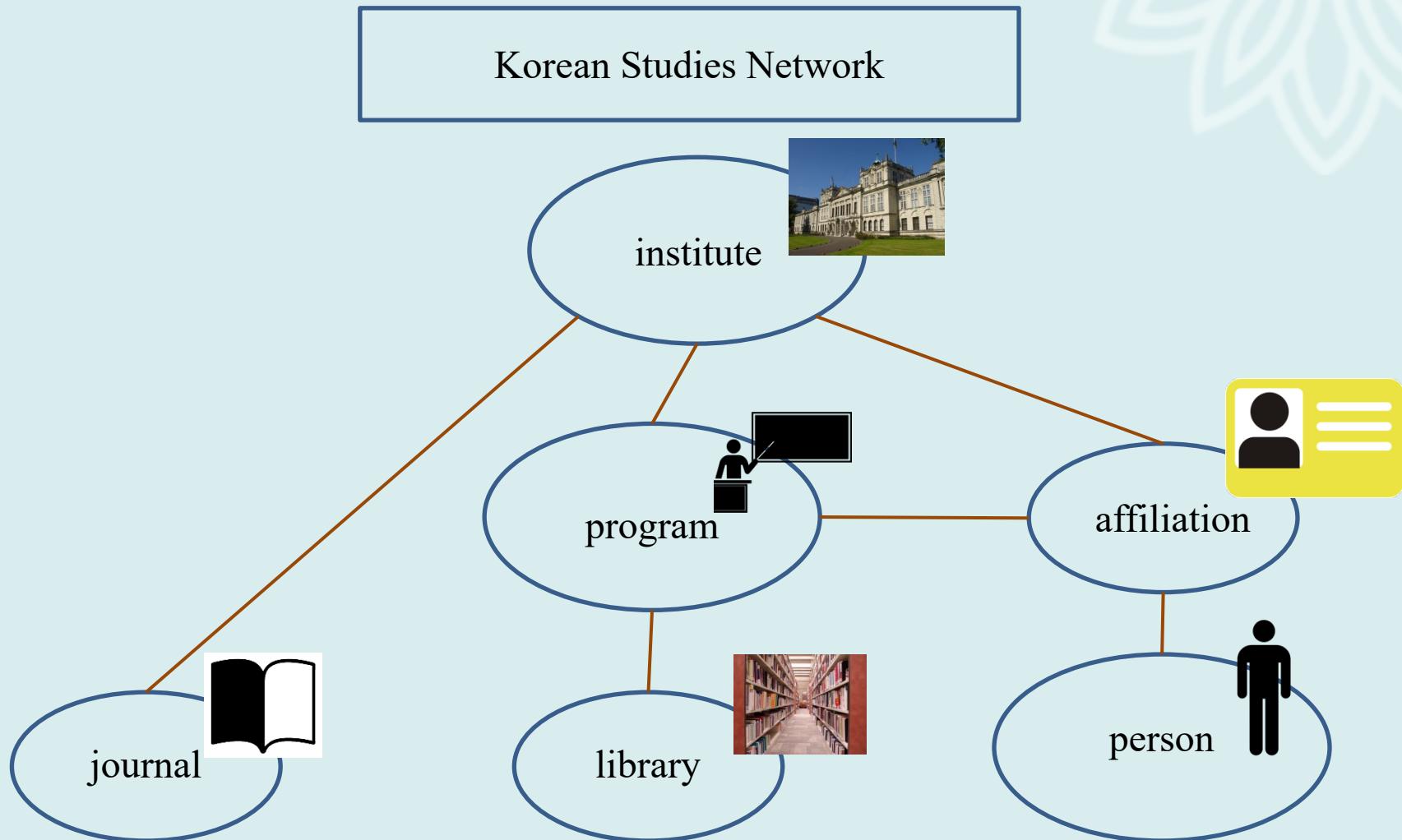
- ❖ 마이크로소프트사가 제공하는 관계형 데이터베이스 관리 시스템 ⇒ 소프트웨어 상품의 이름
- ❖ 1989년 - 기존의 DBMS 제품인 Sybase와 dBase의 기술을 도입하여 “Ashton-Tate/SQL 서버” 1.0 발표
- ❖ 1990년 - 윈도우 3.1 환경을 위한 “Ashton-Tate/SQL 서버” 1.1 발표
- ❖ 1994년 - 마이크로소프트사가 독자적으로 SQL 서버 개발
- ❖ 1997년 - SQL 서버 7.0
- ❖ 2000년 - SQL 서버 2000
- ❖ 2005년 - SQL 서버 2005. XML 데이터베이스 기능 지원
- ❖ 2008년 - SQL 서버 2008. 지리정보 데이터 처리 등 기능 강화
- ❖ 2012, 2014, 2016년 - SQL 서버 2012, 2014, 2016



1. SQL이란?

2. SQL 실습 (데이터 정의)

❖ 정규화



데이터 타입(Data Type)

- ❖ 문자형: 문자열을 정의하기 위한 데이터 타입
 - CHAR: 8000바이트 이하의 고정형 문자.
 - VARCHAR: 8000바이트 이하의 가변형 문자
 - TEXT: 8000바이트 이상의 문자열 데이터. (0 ~ 2 GB)
 - **NCHAR, NVARCHAR, NTEXT** : 유니코드 형식의 문자 데이터
- ❖ 정수형: 소수점이 없는 숫자형 데이터 타입
 - BIGINT, **INT**, SMALLINT, TINYINT, BIT
- ❖ 실수형: 소수점을 가진 숫자형 데이터 타입
 - NUMERIC, DECIMAL: 고정 소수점형
 - **FLOAT**, REAL: 부동 소수점형
- ❖ 날짜형: 날짜와 시간을 저장하기 위한 데이터 타입
 - DateTime: 날짜와 시간을 밀리세컨드 단위까지 저장.
 - SmallDateTime: 날짜와 시간을 분 단위까지 저장.
- ❖ 이진형: **BINARY**데이터를 저장하기 위한 데이터 타입.
 - BINARY, VARBINARY, IMAGE
- ❖ 화폐형: 금액을 저장하기 위한 데이터 타입.
 - MONEY, SMALLMONEY
- ❖ 특수형: 특수한 용도로 사용되는 데이터 타입.
 - TABLE, SQL_VARIANT, CURSOR, UNIQUEIDENTIFIER, TIMESTAMP
- ❖ XML: XML 문서를 위한 데이터 타입
 - **XML** (MS SQL 2005 이상)
- ❖ 계층형: 엔티티의 계층적 지위를 표현하기 위한 데이터 타입
 - **HIERARCHYID** (MS SQL 2008 이상)
 - /, /1/, /2/, /3/1/, /3/2/1/, /3/2/1.5/, /3/2/3/
- ❖ 공간형: 지리 공간 데이터를 처리하기 위한 데이터 타입
 - **GEOMETRY, GEOGRAPHY** (MS SQL 2008 이상)

❖ Create table: Institute

```
CREATE TABLE institute(
    institute_id      float            NOT NULL,
    institute_name   nvarchar(255)    NULL,
    type              nvarchar(255)    NULL,
    zone              float           NULL,
    state             float           NULL,
    state_name       nvarchar(255)    NULL,
    hasGis            float           NULL,
    latitude          float           NULL,
    longitude         float           NULL,
    primary key(institute_id)
)
```

테이블 생성(SQL)

- ❖ 기존 database의 내용을 copy

```
INSERT INTO institute( institute_id, institute_name, type, zone,
                      state, state_name, hasGis, latitude, longitude )
  SELECT institute_id, institute_name, type, zone, state,
         state_name, hasGis, latitude, longitude
    FROM common.dbo.institute
```

- ❖ 결과 조회

```
SELECT * from institute
```