



데이터베이스

SQL 기초





학습목표

- ➔ SQL의 기본 구문을 숙지하여 간단한 질의문을 작성할 수 있다.
- ➔ SQL 구문에 따라 테이블을 생성하고 제약조건을 설정할 수 있다.



학습내용

- ➔ SQL의 기본 구문
- ➔ 테이블의 생성과 제약조건 설정



SQL의 기본 구문



SQL의 역사와 특징

01 SEQUEL(Structured English Query Language)

■ SEQUEL의 역사와 특징

- 1974년, IBM San Jose Lab(현재 IBM Almaden 연구소)에서 개발

최초의 관계형 데이터베이스 관리 시스템 프로토타입인
SystemR을 위한 데이터베이스 언어로 개발





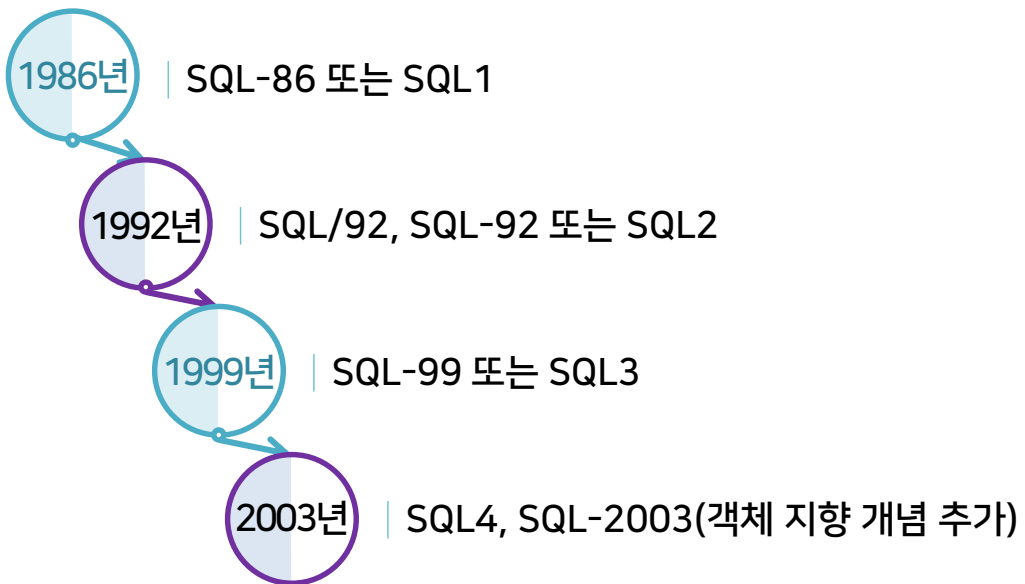
SQL의 기본 구문



SQL의 역사와 특징

02 SQL의 역사

- 1986년 ANSI에서 관계형 데이터베이스 표준 언어로 인증
- 이후 지속적인 개선 진행





SQL의 기본 구문



SQL의 역사와 특징

03 SQL의 특징

01 종합 데이터베이스 언어

- ▶ 데이터 정의(DDL), 조작(DML), 제어(DCL)
- ▶ 무엇(What)을 표시하며 어떻게(How)는 표시하지 않음
- ▶ 어떻게(How)는 DBMS가 알아서 처리

02 관계 대수식 대신 SQL을 사용

- ▶ 관계 대수식 연산자 기호는 **키보드로 표기하기 어려움**

03 관계 대수식(Relational Algebra)과 SQL과의 차이점

- ▶ 관계 대수식: Relation - 튜플의 집합(Set) - 중복을 허용하지 않음
- ▶ SQL: 튜플의 백(Bag) - 튜플들 간의 순서는 없으나, 중복 허용



SQL에서 중복을 허용하는 이유는?

- 집합을 유지하려면 SQL의 결과에서 항상 중복되는 내용을 제거해야 함
- 중복을 제거하는 것은 튜플을 정렬(Sorting)하는 것과 같은 문제
- 결과 생성 시 시간이 오래 걸림(성능 문제 발생)
- 특별한 언급이 없을 시 중복을 놔두고 Bag으로 보면 됨



SQL의 기본 구문



SQL문

01

DDL문: 데이터 정의문

테이블 생성: CREATE문

- 제약조건: NOT NULL, PRIMARY KEY, UNIQUE 등

```
CREATE TABLE 테이블명  
(속성명 속성타입 [제약조건],  
  속성명 속성타입,  
  ...  
)
```

테이블 삭제: DROP문

```
DROP TABLE 테이블명
```

» 예 | 학생(STUDENT) 테이블을 제거하시오.

```
DROP TABLE STUDENT
```



SQL의 기본 구문



SQL문

01 DDL문: 데이터 정의문

테이블 구조 변경: ALTER문

● 속성 추가

```
ALTER TABLE 테이블명(ADD 속성명 속성타입)
```

● 속성 제거

```
ALTER TABLE 테이블명(DROP 속성명 속성타입)
```

● 속성 타입 변경

```
ALTER TABLE 테이블명(ALTER 속성명 속성타입)
```



SQL의 기본 구문



SQL문

02 DDL문: 데이터 조작문

튜플 삽입: INSERT문

```
INSERT INTO 테이블명(속성명, 속성명, ... )  
VALUES(속성값, 속성값, ...)
```

튜플 변경: UPDATE문

```
UPDATE 테이블명  
SET 속성명 = 수식  
[WHERE 조건]
```

튜플 삭제: DELETE문

```
DELETE FROM 테이블명  
[WHERE 조건]
```

튜플 검색: SELECT문

```
SELECT 속성명, 속성명, ...  
FROM 테이블명  
[WHERE 조건]
```




SQL의 기본 구문



SQL문

03 DCL문: 데이터 제어문

| 트랜잭션 관련

- COMMIT, ROLLBACK

| 사용자 권한 제어 관련

- GRANT, REVOKE, DENY



SQL의 기본 구문



MS-SQL 설치



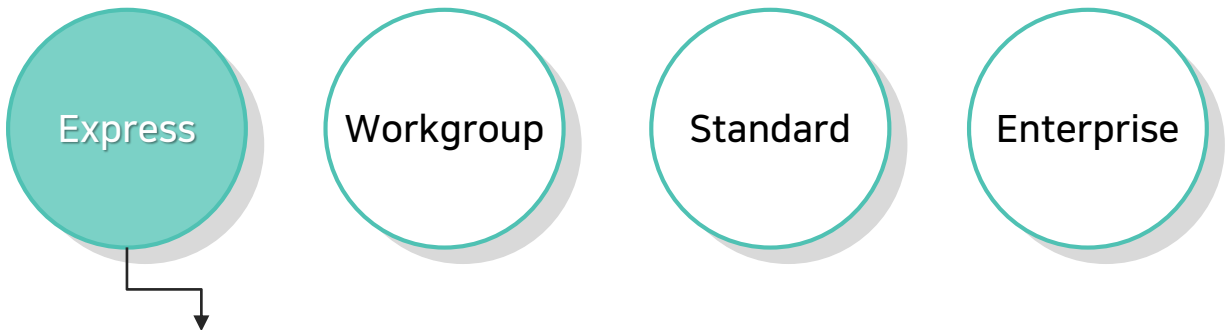
01 MS-SQL

| MS-SQL Server

- Microsoft에서 제공하는 데이터베이스 관리 시스템

| 4종류의 Edition

- Microsoft에서 제공하는 데이터베이스 관리 시스템



실습을 위하여 MS-SQL Server 2014 **Express version** 사용

※ T-SQL: MS-SQL Server에서 사용되는 SQL문



SQL 기초



SQL의 기본 구문



MS-SQL 설치



01 MS-SQL

MS-SQL 다운받기

● <https://msdn.microsoft.com/sqlserver2014expressresources.aspx>

다운로드 / 평가판 소프트웨어 / SQL Server 2014 Express 리소스

Microsoft SQL Server 2014 Express 평가 리소스

평가 가이드, 가상 랩, 온라인 교육 과정, 포럼 등의 기술 리소스에 액세스하세요.

기능 및 업무상 이점에 대한 자세한 내용은 [SQL Server 제품 정보 페이지](#)를 참조하세요.

지금 시작

[하드웨어 및 소프트웨어 요구 사항](#)

[SQL Server 2014용 개발자 참조](#)

[Microsoft Azure에 SQL Server 백업 도구 다운로드](#)

지금 사용해 보기

[DBA를 위한 SQL Server 2014의 새로운 기능: 가상 랩](#)

[SQL Server 2014 AlwaysOn 가용성 그룹을 통해 HA/DR 구현: 가상 랩](#)

온라인 학습

[APPLY 연산자로 T-SQL 향상: 교육 과정](#)

[데이터베이스 기본 사항: 교육 과정](#)

[무료 eBook](#)

소셜 미디어

[SQL Server 커뮤니티](#)

[SQL Server Express 포럼](#)

[Curah!를 사용하여 기술 콘텐츠를 관리하고 공유해 보세요!](#)

[Microsoft Tech 도우미 앱](#)

지금 시작하기

[SQL Server 2014 Express 다운로드](#)



SQL의 기본 구문



MS-SQL 설치



MS-SQL

Microsoft 계정이 없으면 새로 만들기

The screenshot shows the Microsoft login interface. At the top, there are icons for various services. The main heading is "로그인" (Login). Below it, the text says "Microsoft 계정을 사용합니다." (Use a Microsoft account.) and a link "제한 설명" (Terms of use). The login form has two input fields: the first contains the email "jkmin@koreatech.ac.kr", and the second contains masked characters ".....". Below the password field is a checkbox labeled "로그인 유지" (Keep me signed in). A blue "로그인" (Login) button is positioned below the form. A red box highlights the email and password fields, with a green callout bubble labeled "로그인" (Login) pointing to it. Below the login button, the text "계정이 없으신가요" (Don't have an account?) is followed by a link "계정을 만드세요!" (Create an account!). A red box highlights this link, with a green callout bubble labeled "계정 생성" (Account creation) pointing to it. At the bottom, there is a link "암호를 잊어버렸습니다." (Forgot your password?) and the text "일회용 코드를 사용하여 로그인" (Sign in with a one-time code). The Microsoft logo is at the very bottom.



MS-SQL 설치

01

MS-SQL

- 64bit 운영체제 사용시: '도구의 64비트와 SQL 서버 2014 익스프레스' 선택
- 웹 페이지 하단의 "동의함" 클릭 시, MS-SQL 설치 파일 다운로드 됨
- 설치 파일 명

32bit

SQLEXPRT_x86_KOR.exe

64bit

SQLEXPRT_x64_KOR.exe

* SQL 서버 2014 버전을 선택하십시오 인을 다운로드하고 싶습니다

- ☐ SQL 서버 2014 익스프레스하여 LocalDB 32 비트
- ☐ SQL 서버 2014 익스프레스 64 비트하여 LocalDB
- ☐ SQL 서버 익스프레스 2014 32 비트
- ☐ SQL 서버 2014 익스프레스 64 비트
- ☒ 도구와 SQL 서버 2014 익스프레스 32 비트
- ☐ 도구의 64 비트와 SQL 서버 2014 익스프레스
- ☐ SQL 서버 2014 관리 스튜디오 익스프레스 32 비트
- ☐ SQL 서버 2014 관리 스튜디오 익스프레스 64 비트
- ☐ 고급 서비스와 SQL 서버 2014 익스프레스 32 비트
- ☐ 고급 서비스 64 비트와 SQL 서버 2014 익스프레스

* SQL 서버의 언어 2014 익스프레스 W / 도구 32 비트 다운로드를 선택하세요

- ☐ 중국어 (간체)
- ☐ 중국어 (번체)
- ☐ 영어
- ☐ 프랑스어
- ☐ 독일어
- ☐ 이탈리아어
- ☐ 일본어
- ☒ 한국어
- ☐ 포르투갈어 (브라질)
- ☐ 러시아어
- ☐ 스페인어



MS-SQL 설치

02

MS-SQL 설치 방법

SQL Server 설치 센터

계획
설치
유지 관리
도구
리소스
옵션

클릭

새 SQL Server 독립 실행형 설치 또는 기존 설치에 기능 추가
SQL Server 2014를 비클러스터링 환경에 설치하거나 기존 SQL Server 2014 인스턴스에 기능을 추가하는 마법사를 시작합니다.

SQL Server 2005, SQL Server 2008, SQL Server 2008 R2 또는 SQL Server 2012에서 업그레이드
SQL Server 2005, SQL Server 2008, SQL Server 2008 R2 또는 SQL Server 2012를 SQL Server 2014로 업그레이드하는 마법사를 시작합니다.

Microsoft SQL Server

SQL Server 2014 설치 프로그램

사용 조건

SQL Server 2014를 설치하려면 Microsoft 소프트웨어 사용 조건에 동의해야 합니다.

사용 조건
전역 규칙
제품 업데이트
설치 파일 설치
설치 규칙
기능 선택
기능 규칙
기능 구성 규칙
설치 진행률
완료

동의함(A)

동의함 체크

MICROSOFT 소프트웨어 사용권 계약서
MICROSOFT SQL SERVER 2014 EXPRESS

본 사용권 계약은 Microsoft Corporation(또는 거주 지역에 따라 계열사 중 하나)과 귀하 간에 체결되는 계약입니다. 본 사용권 계약을 읽어 주시기 바랍니다. 본 사용권 계약은 위에 명시된 소프트웨어 및 이 소프트웨어가 포함된 미디어가 있는 경우 해당 미디어에 적용됩니다. 본 계약은 해당 품목에 별도의 약정이 있지 않는 한, 이 소프트웨어 및 온라인 서비스에 대한

복사(C) 인쇄(P)

자세한 내용은 [Microsoft SQL Server 2014 개인 정보 취급 방침을 참조하십시오.](#)

* Microsoft SQL Server 2014에는 CEIP 설정이 기본적으로 꺼져 있는 Visual Studio 구성 요소도 포함됩니다. Visual Studio가 설치된 경우 이 구성 요소는 CEIP 설정을 Visual Studio에 사용합니다.

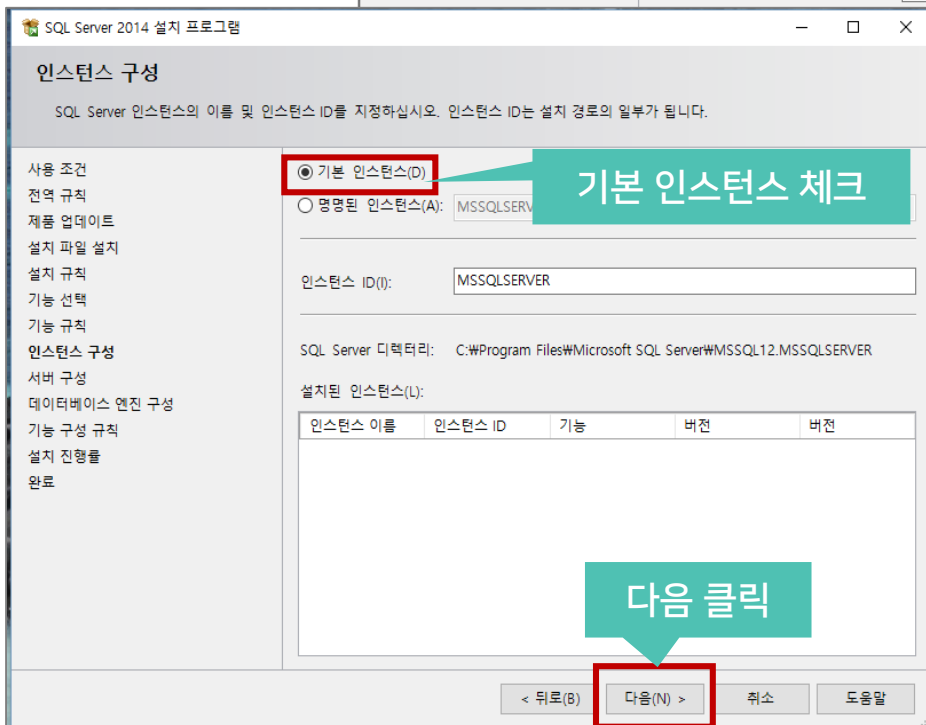
< 뒤로(B) 다음(N) > 취소

다음 클릭



MS-SQL 설치

02 MS-SQL 설치 방법





SQL 기초



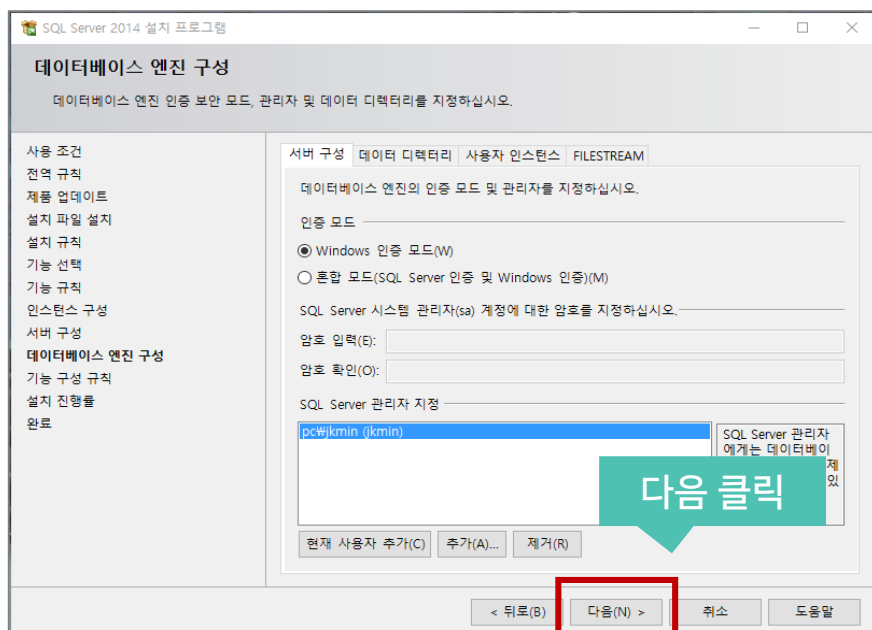
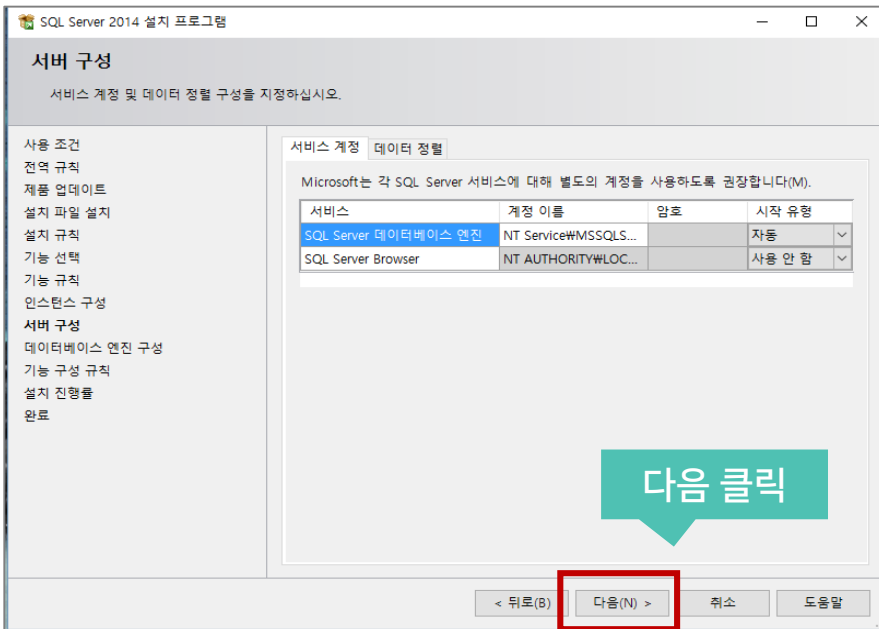
SQL의 기본 구문



MS-SQL 설치



02 MS-SQL 설치 방법

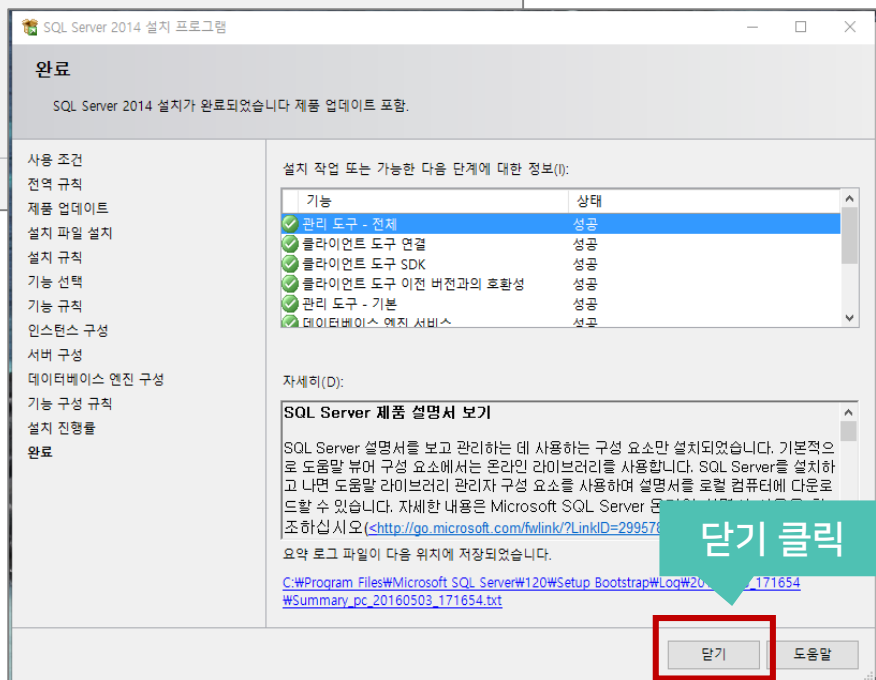
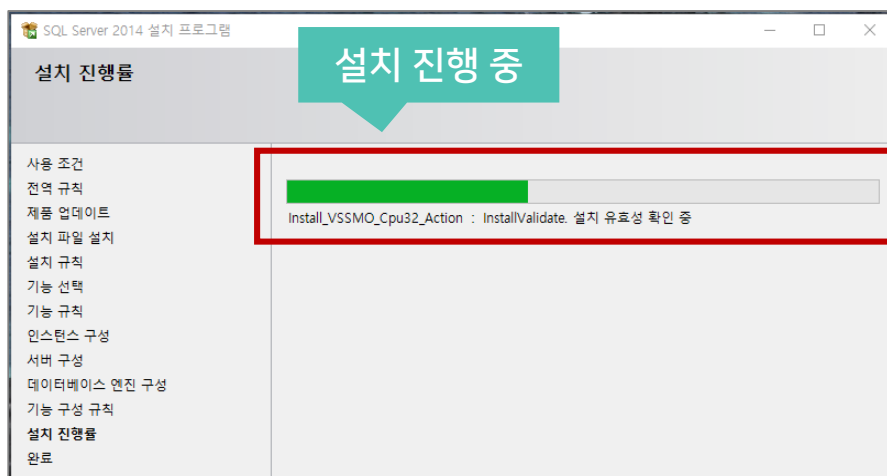




MS-SQL 설치



MS-SQL 설치 방법





SQL 기초



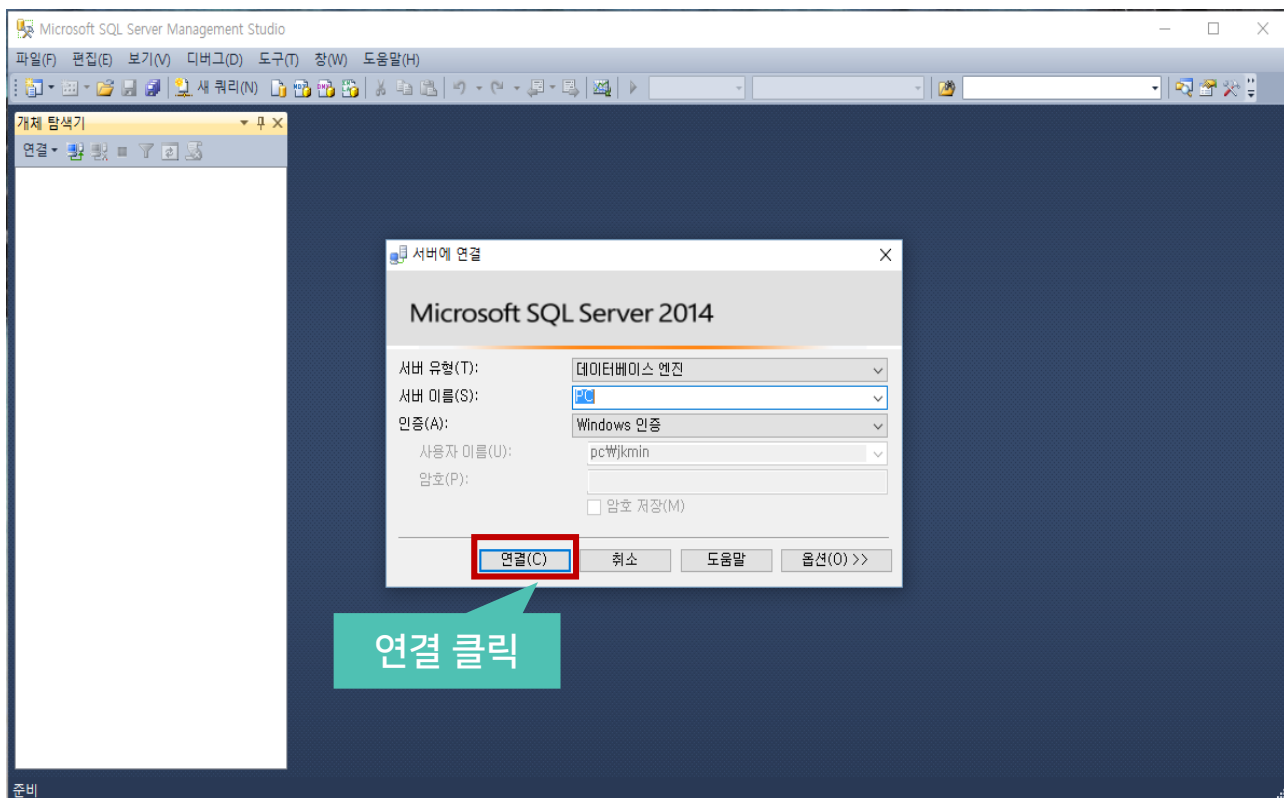
SQL의 기본 구문



MS-SQL 설치



03 MS-SQL 구동





SQL 기초



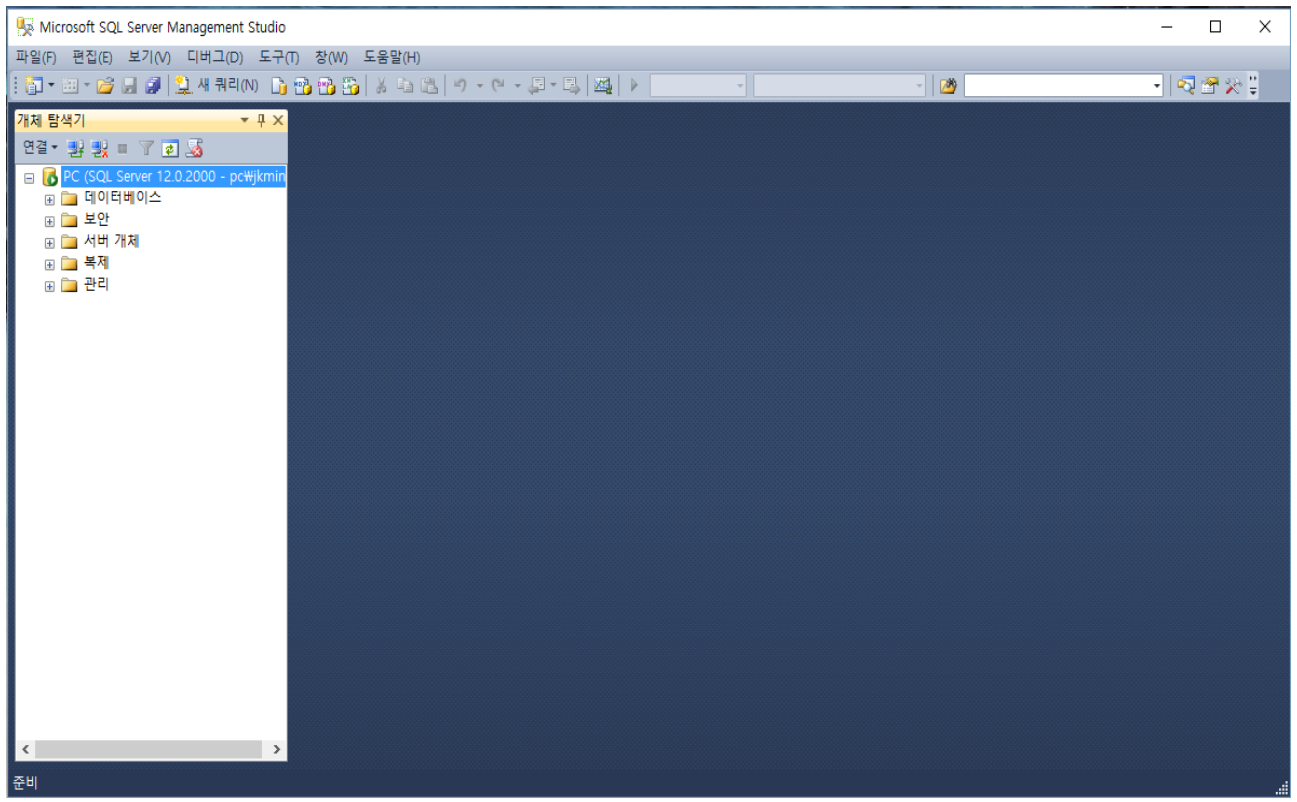
SQL의 기본 구문



MS-SQL 설치



03 MS-SQL 구동





SQL의 기본 구문



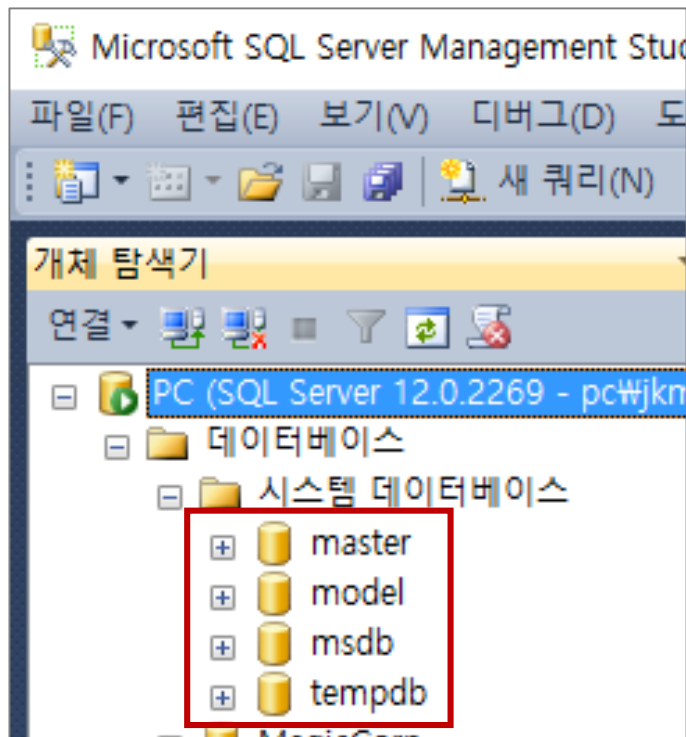
MS-SQL 설치

03

MS-SQL 구동

데이터베이스

- 시스템 데이터베이스(System Database)
 - ▶ master: 시스템 관리용
 - ▶ model: 새 데이터베이스 생성을 위한 위한 틀
 - ▶ msdb: SQL Sever agent를 위한 데이터베이스
 - ▶ tempdb: 질의 임시결과 유지를 위한 테이블
- 사용자 데이터베이스(User Database)
 - ▶ SSMS나 T-SQL(SQL of MS SQL)을 이용해서 생성





SQL의 기본 구문



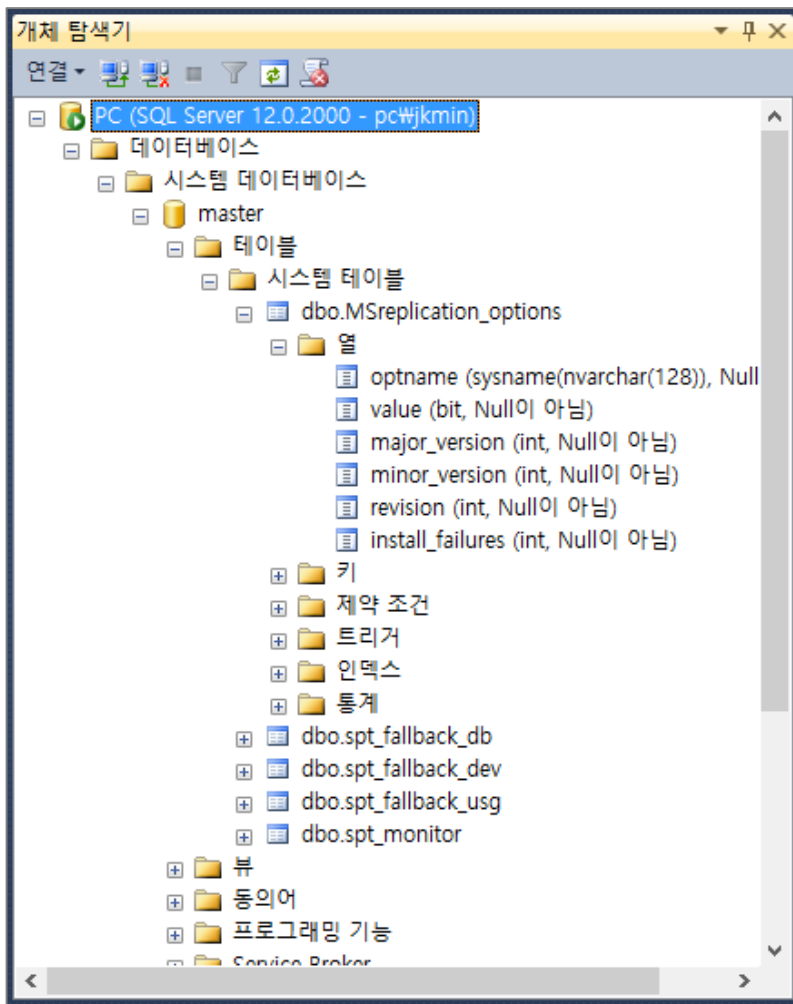
MS-SQL 설치

03

MS-SQL 구동

테이블 구조의 파악

- 데이터베이스 내에 있는 테이블의 구조는 **개체 탐색기**를 이용하여 손쉽게 파악 가능





테이블의 생성과 제약조건 설정



테이블 생성



01 데이터베이스=밥상

■ 테이블(그릇)을 올리기 전에 데이터베이스(밥상)부터 만들어야 함





SQL 기초



테이블의 생성과 제약조건 설정



테이블 생성

02 MS-SQL에서 데이터베이스 만들기

■ T-SQL을 이용하여 데이터베이스 만들기

USE master
CREATE DATABASE test01

DB 이름: test01
사용자 DB를 생성하려면, Master DB를 사용해야 함

준비



테이블의 생성과 제약조건 설정



테이블 생성

03 MS-SQL에서 제공하는 기본 속성 타입

숫자 타입

bit	1bit
tinyint	0~255(정수)
smallint	$-2^{15} \sim 2^{15}-1$ (정수)
int	$-2^{31} \sim 2^{31}-1$ (정수)
bigint	$-2^{63} \sim 2^{63}-1$ (정수)
decimal[(p[,s])], numeric[(p[,s])]	decimal(5,2) → 123.45
float	4 byte float(실수)
real	8 byte float(실수)
datetime	1755/1/1~9999/12/31(8byte)(날짜)
smalldatetime	1990/01/01~2079/6/6(4byte)(날짜)



테이블의 생성과 제약조건 설정



테이블 생성

03 MS-SQL에서 제공하는 기본 속성 타입

| 문자 타입

char[(n)]

고정길이 문자열

varchar[(n)]

0~255(정수)

nchar[(n)]

for unicode(2byte)→n→2n bytes
(유니코드를 위한 고정문자열)

nvarchar[(n)]

for unicode(유니코드를 위한 가변길이문자열)

| 문자 타입

- 자동 증가 속성 타입
- IDENTITY(10, 2)→초기값 10, 2씩 증가
 - ▶ 10, 12, 14, ...



테이블의 생성과 제약조건 설정



테이블 생성

04 CREATE TABLE 테이블명

```
( 속성명 속성타입 [제약조건],  
  속성명 속성타입,  
  ...  
)
```

test01 DB에 테이블을
만들 것임으로 use test01 기입

```
SQLQuery2.sql - ... (53))*  
use test01  
GO  
CREATE TABLE memberTEST(  
    id INT IDENTITY(1,1),  
    name VARCHAR(20)  
)
```



테이블의 생성과 제약조건 설정



테이블 생성

05

튜플

튜플 추가

→ 생략 가능

```
INSERT INTO 테이블명(속성명, 속성명, ... )  
VALUES(속성값, 속성값, ...)
```

- id 속성은 자동 증가 타입이므로 값을 지정할 수 없음

```
SQLQuery2.sql -...56Wadmin (Include Client  
INSERT memberTEST VALUES ('lee')  
INSERT memberTEST VALUES ('kim')
```



테이블의 생성과 제약조건 설정



테이블 생성

05 튜플

추가된 튜플의 검색

```
SELECT 속성명, 속성명, ...  
FROM 테이블명  
[WHERE 조건]
```

The screenshot shows a SQL query editor window titled "SQLQuery2.sql - ...56Wadmin (53))*". The query entered is "select * from memberTEST". The text "from memberTEST" is highlighted with a red box. A green callout bubble points to this text with the label "MemberTEST". Below the query editor, there are two tabs: "Results" and "Messages". The "Results" tab is active, displaying a table with two columns: "id" and "name". The table contains two rows of data: (1, lee) and (2, kim).

	id	name
1	1	lee
2	2	kim



테이블의 생성과 제약조건 설정



테이블 생성

06 테이블의 구조 변경(ALTER TABLE)

ADD column	<ul style="list-style-type: none">속성 추가
ALTER column	<ul style="list-style-type: none">속성 타입 변경
DROP column	<ul style="list-style-type: none">속성 제거

```
ALTER TABLE 테이블명{  
{ADD|ALTER|DROP} [COLUMN] 속성명 [타입]  
}
```



테이블의 생성과 제약조건 설정



테이블 생성

06 테이블의 구조 변경(ALTER TABLE)

주의사항

- ALTER column의 경우 속성값의 범위를 증가시키는 경우에는 문제가 없지만 **범위를 감소시킬 경우에는** 현재 테이블이 저장된 **속성값들에 따라서 허용이 안될 수도 있음**

예 | 이름 속성값으로 "김이름씨", "박이름씨", "이름이 이빠요" 가 저장된 경우

속성 타입은 varchar(3)로 변경하려고 하면 안됨

→ 모두 6byte 이상(한글 1자는 2byte)

→ varchar(50)으로는 변경 가능



테이블의 생성과 제약조건 설정



테이블 생성

06 테이블의 구조 변경(ALTER TABLE)

ADD column: 속성 추가

예 | datetime 형으로 pdate 속성을 memberTEST 테이블에 추가하자.

```
SQLQuery2.sql -...56Wadmin (F
ALTER TABLE memberTEST
ADD pdate datetime
```

ADD pdate datetime 추가

alter column: 속성 변경

예 | name 속성의 타입을 varchar(30)으로 변경하자.

```
ALTER TABLE memberTEST
ALTER column name varchar(30)
```

varchar(30)으로 변경



테이블의 생성과 제약조건 설정



테이블 생성

07 Drop table과 TRUNCATE table

테이블 지우기

```
DROP TABLE table_name
```

테이블의 모든 내용 지우기(단, 테이블은 남기기)

```
TRUNCATE TABLE table_name
```

CREATE, ADD, ALTER, DROP, TRUNCATE 등은 모두
데이터 정의어(DDL)이기 때문에
명령문이 수행되고 나면 회복이 불가능함



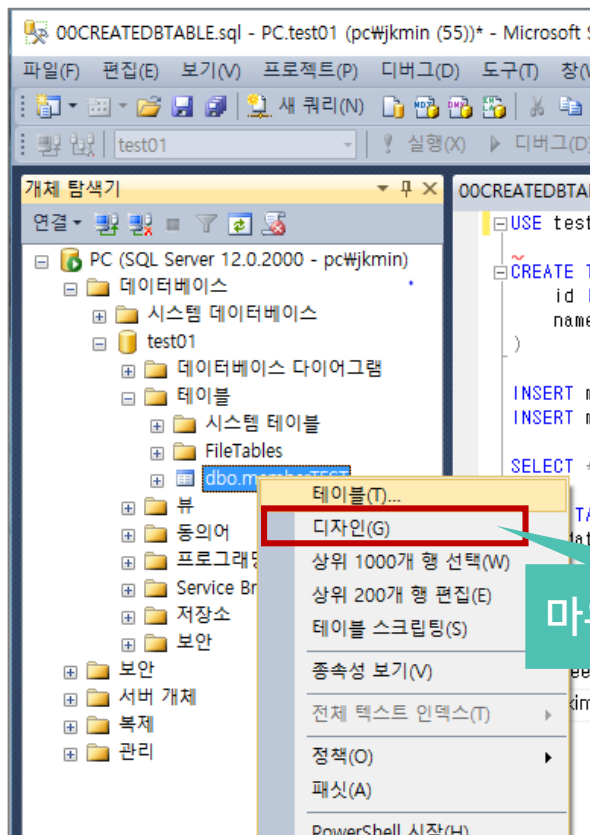
테이블의 생성과 제약조건 설정



테이블 생성

08

SSMS에서 테이블의 구조를 보는 방법



열 이름	데이터 형식	Null 허용
id	int	<input type="checkbox"/>
name	varchar(30)	<input checked="" type="checkbox"/>
pdate	datetime	<input checked="" type="checkbox"/>
		<input type="checkbox"/>



테이블의 생성과 제약조건 설정



제약조건 설정

01 무결성 제약조건

01 NOT NULL

- ▶ NULL 값 허용 불가

» 예 | 학생 테이블에서 학생의 이름은 NULL값일 수 없다.

02 UNIQUE

- ▶ 하나의 테이블 내에서 한번만 나옴(주로 대체키 설정 시 사용)

03 PRIMARY KEY: 기본키(의미: UNIQUE+NOT NULL)

04 FOREIGN KEY: 외래키

05 CHECK: 도메인 무결성



테이블의 생성과 제약조건 설정



제약조건 설정

02

테이블을 만들 때 속성에 제약조건 지정하기

```
CREATE TABLE 테이블명  
( 속성명 속성타입 [CONSTRAINT [제약조건명] 제약조건],  
  속성명 속성타입,  
  ...  
)
```



예 | name varchar(20) NOT NULL
id int CONSTRAINT PK_01 PRIMARY KEY

03

제약조건 설정 테이블 예제

```
CREATE TABLE customer(  
  id VARCHAR(20) CONSTRAINT PK_id PRIMARY KEY,  
  pwd VARCHAR(20) CONSTRAINT NN_pwd NOT NULL,  
  name VARCHAR(20) CONSTRAINT NN_name NOT NULL,  
  phone1 VARCHAR(3) NULL,  
  phone2 VARCHAR(8) NULL,  
  birthYear int NULL,  
  address VARCHAR(100) NULL  
)  
  
INSERT INTO customer  
VALUES('one', '1111', 'Kim', '010', '77727777', 1988, 'Daejeon')  
  
INSERT INTO customer  
VALUES('two', '2222', 'Lee', '010', '12324567', 1979, 'Seoul')
```



테이블의 생성과 제약조건 설정



제약조건 설정

04 제약조건 위반 튜플 삽입 예

```
INSERT INTO customer
VALUES (NULL, '2222', 'Lee', '010', '12324567', 1979, 'Seoul')
```

Messages

Msg 515, Level 16, State 2, Line 1
Cannot insert the value NULL into column 'id', table 'test01.dbo.customer'; column c
The statement has been terminated.

05 참조 무결성 제약조건

❑ 외래키 값은 다른 테이블의 기본키 값들 중에 하나이어야 함

➤ 예 | 사원 테이블의 부서번호는 부서 테이블의 기본키 값들 중 하나이어야 함

속성명 속성타입[CONSTRAINTS 제약조건명]
REFERENCE 참조테이블명(속성명)



테이블의 생성과 제약조건 설정



제약조건 설정

06

참조 무결성 설정 및 삽입 오류의 예

```
SQLQuery3.sql -...56Wadmin (52))*
CREATE TABLE orders (
    oseq      int IDENTITY(1,1) CONSTRAINT PK_oseq PRIMARY KEY,
    quantity  varchar(20) NULL,
    indate    datetime NULL,
    id        varchar(20) CONSTRAINT FK_id REFERENCES customer(id)
)

INSERT INTO orders(quantity, id) VALUES(5, 'ONE')
|
SELECT * FROM orders

INSERT INTO orders(quantity, id) VALUES(5, 'test')
```

Results Messages

(1 row(s) affected)

(1 row(s) affected)

Msg 547, Level 16, State 0, Line 14
The INSERT statement conflicted with the FOREIGN KEY constraint "FK_id". The
The statement has been terminated.

07

Check 제약조건

도메인 무결성 제약조건: 입력 값의 제한

예 | birthYear int CHECK(birthYear >= 1900)

Age int CHECK(Age between 1 and 150)



테이블의 생성과 제약조건 설정



제약조건 설정

08

테이블 수준 제약조건

- 속성 단위로 제약조건 설정은 표현에 있어서 제약이 따름
 - 복합키의 경우(여러 개의 속성이 합쳐져서 키가 됨)

```
CREATE TABLE 테이블명(  
    속성명 속성타입  
    ...  
    [CONSTRAINT 제약조건명] 제약조건(속성명)  
)
```

09

테이블 수준 제약조건 설정의 예

```
CREATE TABLE customer3(  
    name VARCHAR(20),  
    phone VARCHAR(11),  
    birthday DATETIME,  
    address VARCHAR(100)  
    CONSTRAINT customer3_COMBO_PK PRIMARY KEY(name, phone)  
)
```

ADMIN-D8B05BB56... - dbo.customer3			
Column Name		Data Type	Allow Nulls
	name	varchar(20)	<input type="checkbox"/>
	phone	varchar(11)	<input type="checkbox"/>
	birthday	datetime	<input checked="" type="checkbox"/>
	address	varchar(100)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>



테이블의 생성과 제약조건 설정



제약조건 설정

10

제약조건의 추가 및 제거

01

테이블을 생성한 후에 제약조건을 추가하거나 제거할 필요성이 있음

- ▶ 제약조건도 테이블의 구조 정보에 속함으로 **ALTER TABLE**을 이용
- ▶ 추가: ADD CONSTRAINT
- ▶ 제거: DROP CONSTRAINT

02

제약조건 이름을 지정해 두어야 추가나 제거가 쉬움

```
ALTER TABLE 테이블명  
ADD [CONSTRAINT 제약조건명] 제약조건(속성명)
```

```
ALTER TABLE 테이블명  
DROP CONSTRAINT 제약조건명
```



테이블의 생성과 제약조건 설정



제약조건 설정

10

제약조건의 추가 및 제거

제약조건 변경 예

```
CREATE TABLE orders5 (  
    oseq      int IDENTITY(1,1),  
    quantity  varchar(20) NULL,  
    indate    datetime NULL,  
    id        varchar(20),  
    pcode     varchar(20)  
)  
  
ALTER TABLE orders5  
ADD CONSTRAINT PK_oseq5 PRIMARY KEY(oseq)  
  
ALTER TABLE orders5  
ADD CONSTRAINT FK_id5  
FOREIGN KEY(id) REFERENCES customer(id)  
  
ALTER TABLE orders5  
DROP CONSTRAINT FK_id5
```


1 SQL의 기본 구문

- ✓ SQL의 역사: 최초의 관계형 데이터베이스 관리 시스템 프로토타입인 SystemR을 위한 데이터베이스 언어로 개발됨
- ✓ SQL의 특징: 관계 대수와 달리 중복을 허용하는 bag타입을 씀
- ✓ SQL문
 - DDL: CREATE, ALTER, DROP, TRUNCATE문
 - DML: INSERT, UPDATE, DELETE문
 - DCL: COMMIT, ROLLBACK, GRANT, REVOKE문

2 테이블의 생성과 제약조건 설정

- ✓ 테이블의 생성

```
CREATE TABLE 테이블명
( 속성명 속성타입 [제약조건], 속성명 속성타입, ...)
```

- ✓ 제약조건의 설정
 - NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, CHECK