



Database Systems

MS SQL SVR, SSMS & SQL Introduction



SQL language

- SQL language
 - Standard database language for relational database management
 - Considered one of the major reasons for the commercial success of relational databases.
 - Each statement in SQL ends with a semicolon.
- **SQL**
 - **Structured Query Language**
 - Statements for **data definitions, queries, and updates** (both DDL and DML)
 - **Core specification**
 - Plus specialized **extensions**

SQL language (cont'd.)

- **Data Definition Language (DDL)**

- Defines the schemas (databases).
- Create, drop, and alter of schema, table, integrity, view, and index.
- Used by only DBA to prevent ill-uses

- **Data Manipulation Language (DML)**

- Manipulates instances in table.
- Insert, update, delete or query instances from tables.
- Used by application programmers or interactive users.

- **Data Control Language (DCL)**

- Controls access privileges and backup/restore.
- Grant, and revoke
- Used by DBA.

SQL data definition

- Terminology:
 - **Table**, **row**, and **column** used for terms of relational model **relation**, **tuple**, and **attribute**, respectively.

The diagram shows a table titled "DESKTOP-86CGPLD...B - dbo.EMPLOYEE" with 11 columns: SSN, Super_SSN, Fname, Minit, Lname, Bdate, Address, Sex, Salary, and Dno. The table contains 4 rows of data. Annotations include a red box labeled "Relation" pointing to the table title, a red box labeled "Attributes" with arrows pointing to each column header, and a red box labeled "Tuples" with arrows pointing to each row of data.

	SSN	Super_SSN	Fname	Minit	Lname	Bdate	Address	Sex	Salary	Dno
	20170001	00000000	John	B	Smith	1965-01-09 00:00:00	731 Fondren, Houston, TX	M	30000	5
	00000000	00000000	Mack	H	Ruddy	1960-05-17 00:00:00	610 Fondren, Houston, TX	M	55000	1
	20170002	00000000	Mee	L	Kim	1995-05-22 00:00:00	381 Roxton, Houston, TX	F	15000	5
	20170003	00000000	Kane	P	Wood	1994-12-01 00:00:00	555, Roxton, Houston, TX	M	15000	5

Attribute data types

- We will use the **basic data types** of SQL2.
- **Basic data types**
 - **Numeric data types**
 - Integer numbers: INTEGER, INT, and SMALLINT
 - Floating-point (real) numbers: FLOAT or REAL, and DOUBLE PRECISION
 - **Character-string data types**
 - Fixed length: CHAR (*n*) , CHARACTER (*n*)
 - Varying length : VARCHAR (*n*) , CHAR VARYING (*n*) , CHARACTER VARYING (*n*)

Attribute data types (cont'd.)

– **Bit-string** data types

- Fixed length: `BIT (n)`
- Varying length: `BIT VARYING (n)`

– **Boolean** data type

- Values of `TRUE` or `FALSE` or `NULL`

– **DATE** data type

- Ten positions
- Components are `YEAR`, `MONTH`, and `DAY` in the form `YYYY-MM-DD`.

Attribute data types (cont'd.)

- **Additional data types**

- **Timestamp** data type (TIMESTAMP)

- Includes the DATE and TIME fields.
 - Plus a minimum of six positions for decimal fractions of seconds.
 - Optional WITH TIME ZONE qualifier

- **INTERVAL** data type

- Specifies a relative value that can be used to increment or decrement an absolute value of a date, time, or timestamp.

Attribute data types (cont'd.)

- In MS-SQL SERVER...

Data type	Length	Description
bigint	8	Integer from -2^{63} (-9 223 372 036 854 775 808) to $2^{63}-1$ (9 223 372 036 854 775 807).
int	4	Integer from -2^{31} (-2 147 483 648) to $2^{31}-1$ (2 147 483 647).
smallint	2	Integer from -2^{15} (-32 768) to $2^{15}-1$ (32 767).
tinyint	1	Integer from 0 to 255.
bit	1 bit	Integer 0 or 1.
decimal(precision, scale)	5-17	Numeric data type with fixed precision and scale (accuracy 1-38, 18 by default and scale 0-p, 0 by default).
numeric	5-17	Same as data type 'decimal'.
money	8	Financial data type from -2^{63} (-922 337 203 685 477.5808) to $2^{63}-1$ (922 337 203 685 477.5807) with the precision of one ten-thousandth unit.
smallmoney	4	Financial data type from -2^{31} (-214 748.3648) to $2^{31}-1$ (214 748.3647) with the precision of one ten-thousandth unit.
float(n)	4-8	Numeric data type with float precision, where n is the number of mantis bits (1-24, accuracy of 7 digits, size of 4 bytes and 25-53, accuracy of 15 digits and size of 8 bytes).
real	4	Numeric data type with float precision that is defined as a float(24).
datetime	8	Data type representing date and time from 1.1.1753 to 31.12.9999 with precision about 3ms. Values are rounded to .000, .003 and .007.
smalldatetime	4	Data type representing date and time from 1.1.1900 to 6.6.2079 with precision of 1min. Values up to 29.998 are rounded down and values from 29.999 are rounded down to the nearest minute.

Attribute data types (cont'd.)

- In MS-SQL SERVER ...

Data type	Length	Description
char	n	Char string of fixed length and max. length of 8000 chars.
varchar	n	Char string of variable length and max. length of 8000 chars.
text	n	Char string of variable length and max. length of $2^{31}-1$ (2 147 483 647) chars.
nchar	2*n	Unicode char string of fixed length and max. length of 4000 chars.
nvarchar	2*n	Unicode char string of variable length and max. length of 4000 chars.
ntext	2*n	Unicode char string of variable length and max. length of $2^{30}-1$ (1 073 741 823) chars.
binary	n+4	Binary data of fixed length and max. length of 8000 bytes.
varbinary	n+4	Binary data of variable length and max. length of 8000 bytes.
image	n	Binary data of variable length and max. length of $2^{31}-1$ (2 147 483 647) bytes.
cursor		For storing the reference to cursors in a variable or in a procedure (no for CREATE TABLE).
sql_variant		For storing value of another type (no text, ntext, image, timestamp, sql_variant) of max. length to 8016 bytes. ODBC doesn't fully support this data type.
table		For storing the query result for the later usage.
timestamp	8+4	Data type generates automatically binary numbers, unique in the database, used mostly to the rows identification. There can be only column of this data type in the table.
uniqueidentifier		Data type for storing GUID (new by means of the NEWID function or existing from the string in the form xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx, for example 6F9619FF-8B86-D011-B42D-00C04FC964FF).

Domain

- **Domain**

- Name used with the attribute specification
- Improves schema readability.
- Makes it easier **to change the data type for a domain** that is used by numerous attributes.
- Example:
 - `CREATE DOMAIN SSN_TYPE AS CHAR(9);`



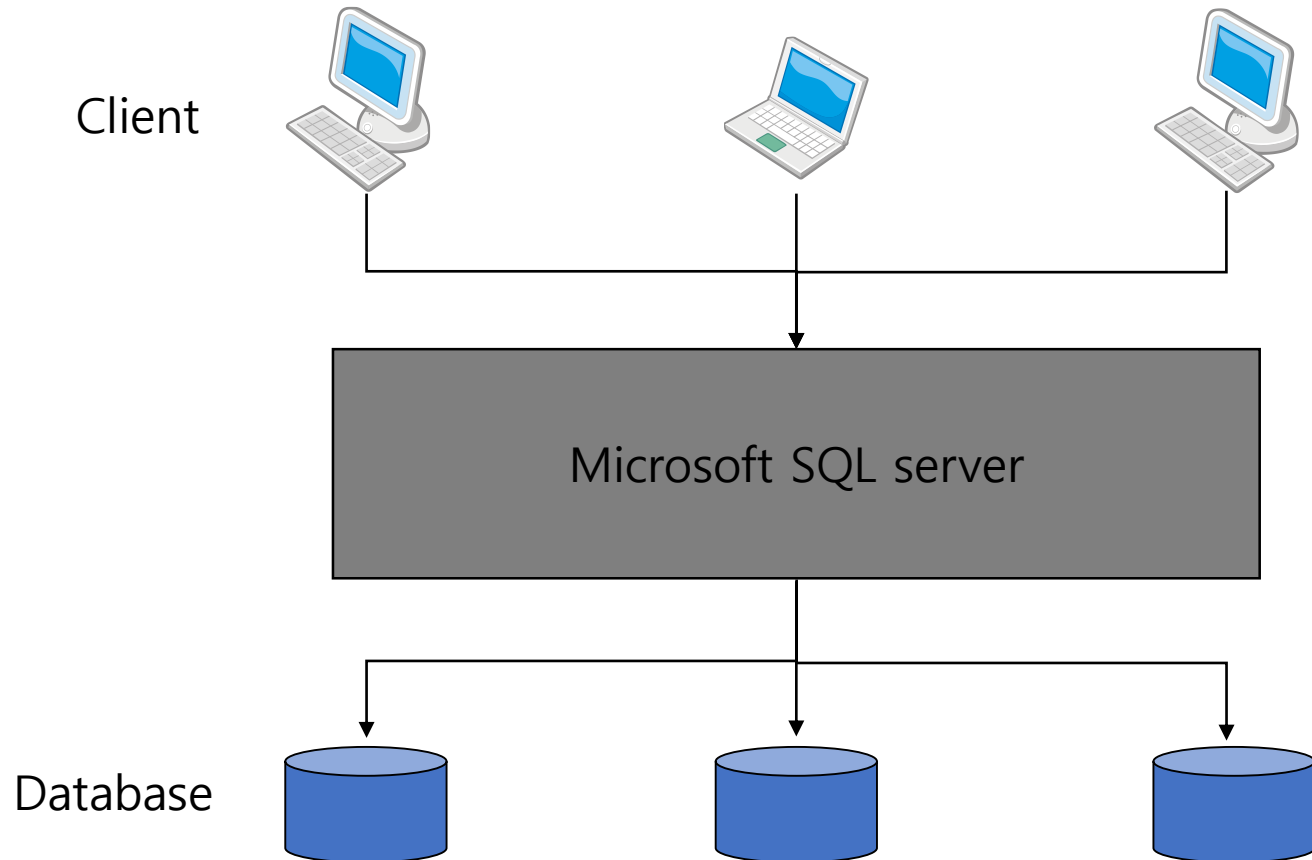
Relational database management system (RDBMS)

- Relational database management system
 - Microsoft SQL server
 - Oracle
 - MySQL
- Programs that can create, update, and administer relational databases
- Uses the structured query language (SQL) to access the database.

Microsoft SQL server

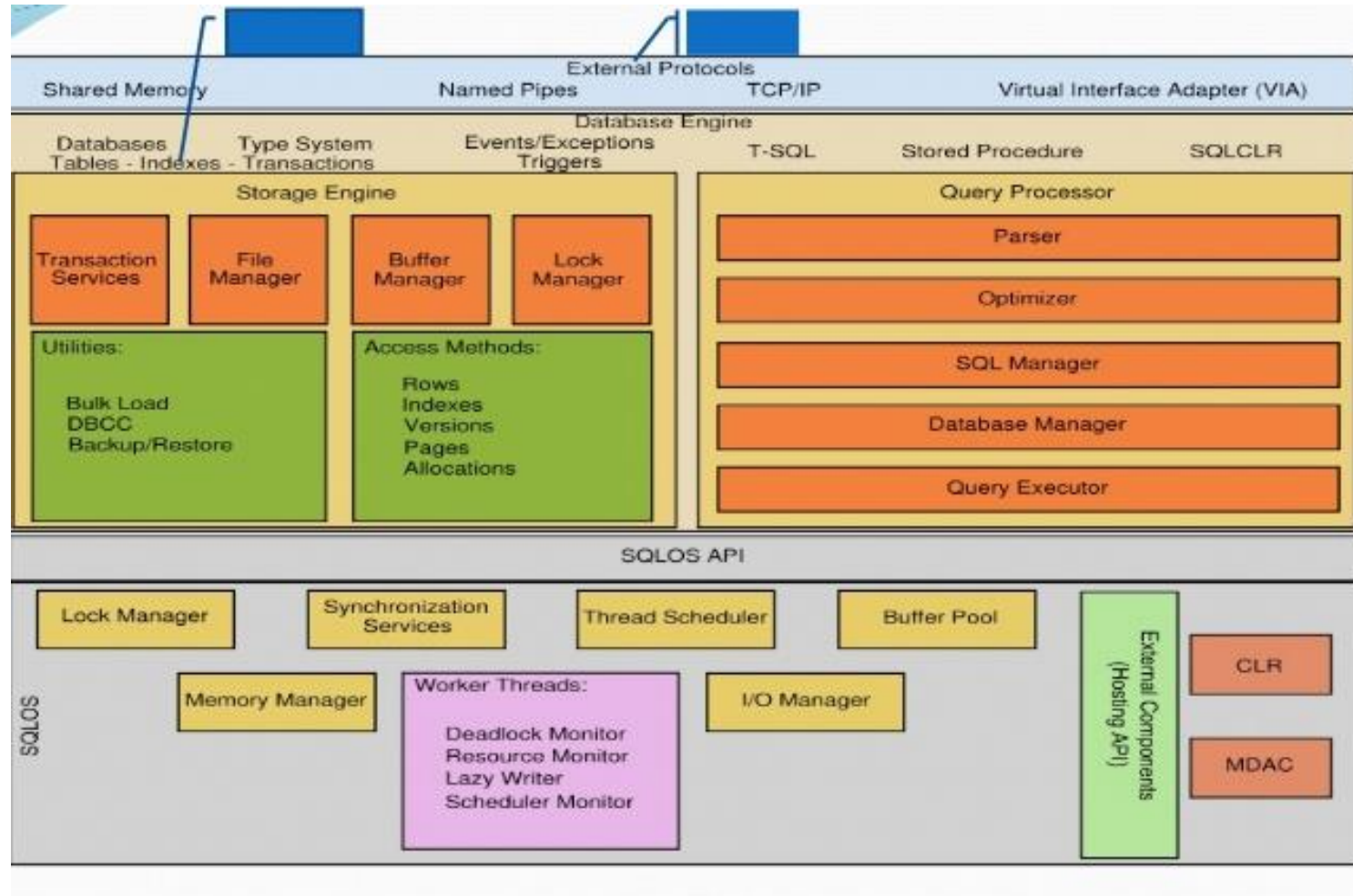
- A relational database management system developed by Microsoft
- As a database server, it is a software product with the primary functions of **defining**, **modifying**, and **retrieving data** as requested by other software applications which may run either on the same computer or on another computer across a network (including the Internet).

MS SQL server architecture



MS SQL server architecture (cont'd.)

- Internal structure of MS SQL sever (reference)



Install MS SQL server



개발자

SQL Server 2017 Developer는 비 프로덕션 환경에서 개발 및 테스트 데이터베이스로 사용하도록 라이선스가 제공되며 모든 기능을 갖춘 무료 버전입니다.

지금 다운로드하기 ↓



Express

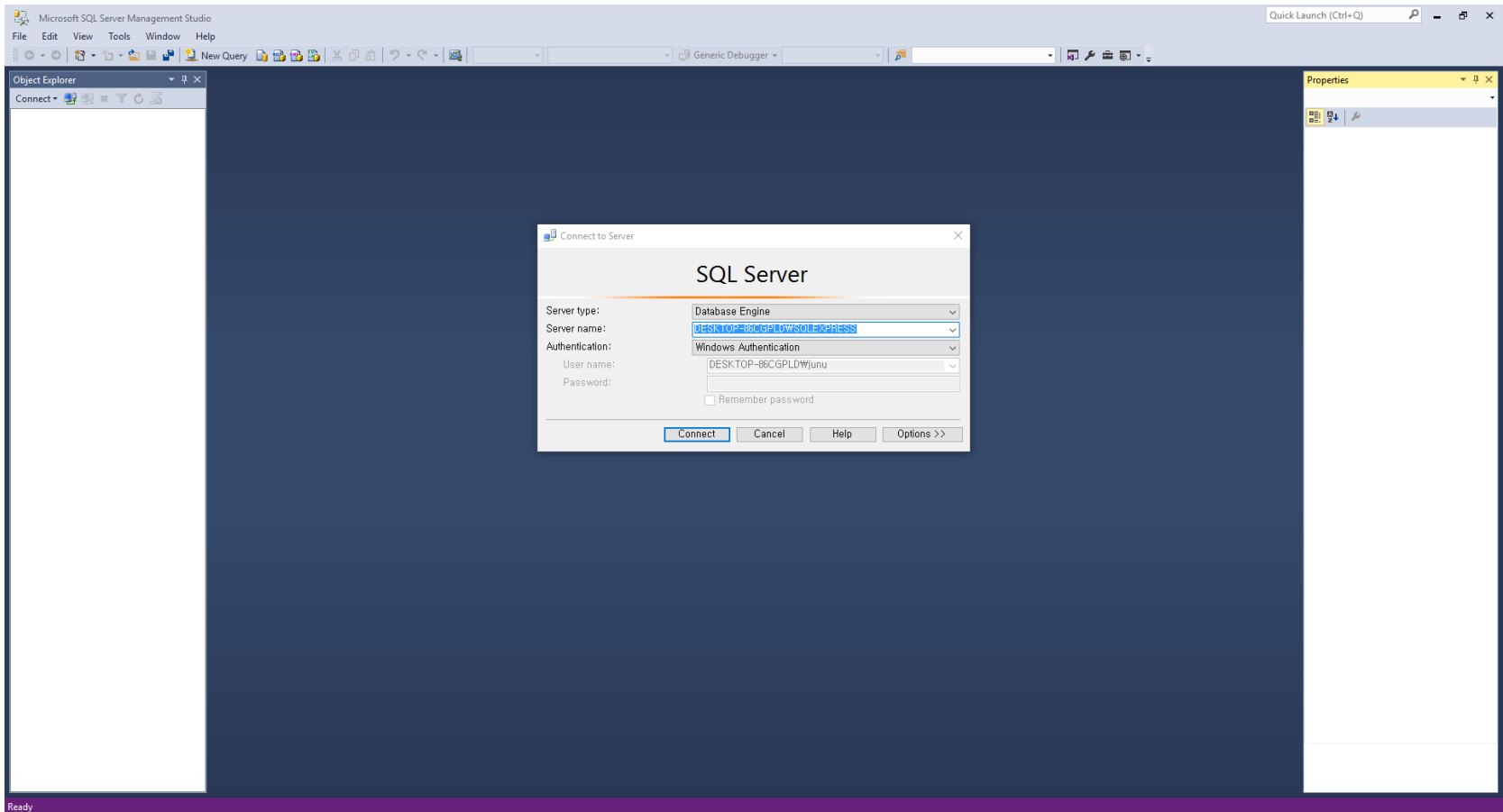
SQL Server 2017 Express는 데스크톱, 웹 및 소형 서버 애플리케이션의 개발 및 제작에 적합한 무료 SQL Server 버전입니다.

지금 다운로드하기 ↓

- Microsoft SQL Server 2016 Express 다운로드 또는
- Microsoft SQL Server 2017 Express 다운로드
- <https://www.microsoft.com/ko-kr/sql-server/sql-server-downloads>

Microsoft SQL server

- Program execution
 - [start button] → [program] → [Microsoft SQL Server]
→ [Microsoft SQL Server Management Studio]



Basic screen configuration

- It consists of one menu bar for simple selection of commands and four component windows.

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The top menu bar includes File, View, Tools, Window, and Help. The main workspace is divided into four panes:

- Object Explorer:** Located on the left, it shows the database structure for 'COMPANY_DB' (DESKTOP-86CGPLD\junu (53)). It includes folders for Databases, System Databases, Database Snapshots, Database Diagrams, Tables, Views, Synonyms, Programmability, Service Broker, Storage, Security, and Server Objects.
- Query editor:** The central pane shows a SQL query: `SELECT * FROM EMPLOYEE`.
- Results / messages page:** Located at the bottom, it displays the results of the query in a table format.
- Property page:** Located on the right, it shows the 'Properties' window for the current connection, including 'Aggregate status', 'Connection', and 'Connection Details'.

Red arrows point from the labels to the corresponding components in the screenshot.

	SSN	Super_SSN	Fname	Minit	Lname	Bdate	Address	Sex	Salary	Dno
1	00000000	00000000	Mack	H	Ruddy	1960-05-17 00:00:00,000	610 Fondren, Houston, TX	M	55000	1
2	20170001	00000000	John	B	Smith	1965-01-09 00:00:00,000	731 Fondren, Houston, TX	M	30000	5
3	20170002	00000000	Mee	L	Kim	1995-05-22 00:00:00,000	381 Roxton, Houston, TX	F	15000	5
4	20170003	00000000	Kane	P	Wood	1994-12-01 00:00:00,000	555, Roxton, Houston, TX	M	15000	5

Query executed successfully.

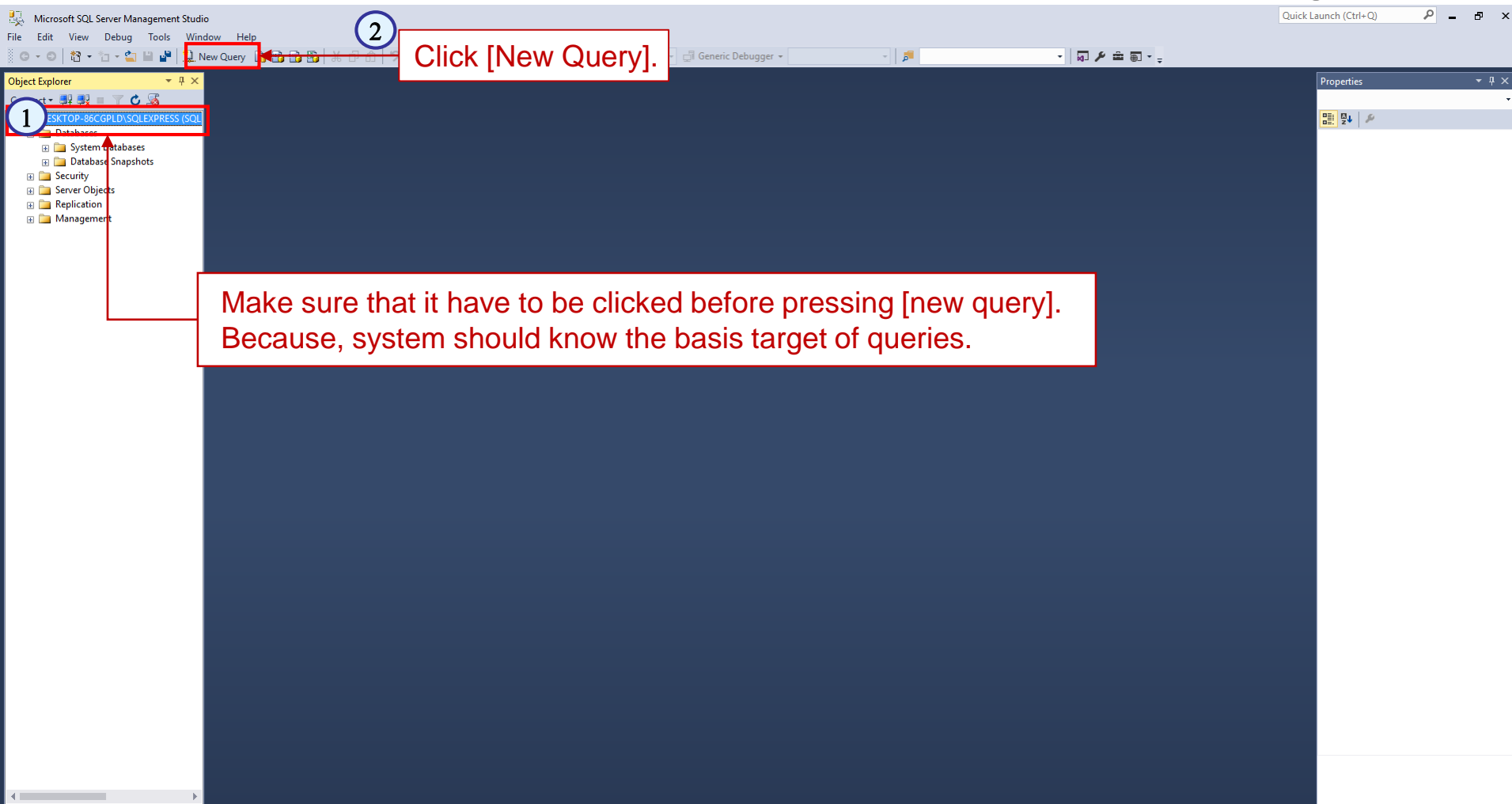
Basic screen configuration (cont'd.)

Division	Function
Menu bars	<ul style="list-style-type: none">➤ The buttons to help modeling works are gathered.➤ It can be located anywhere user want.
Query editor	<ul style="list-style-type: none">➤ By using the Database Engine Query Editor in SQL Server Management Studio you can write and edit queries as scripts.➤ You use scripts when you have to process Windows System commands and Transact-SQL statements in the same script.
Object Explorer	<ul style="list-style-type: none">➤ SQL Server Management Studio provides features for managing objects in instances of the Database Engine, Analysis Services, Integration Services, and Reporting Services.
Results / messages page	<ul style="list-style-type: none">➤ You can see results of query, messages.
Property page	<ul style="list-style-type: none">➤ Property page dialog boxes in Microsoft SQL Server Management Studio all use a common format displaying information with expanding and collapsing categories.➤ The fields shown depend on the particular property➤ Properties shown in gray are read-only.➤ Categorized and Alphabetic buttons are near the top of each property page.

References : <https://docs.microsoft.com/en-us/sql/ssms/use-sql-server-management-studio>

Query editor window

- Query editor window can be created by following steps



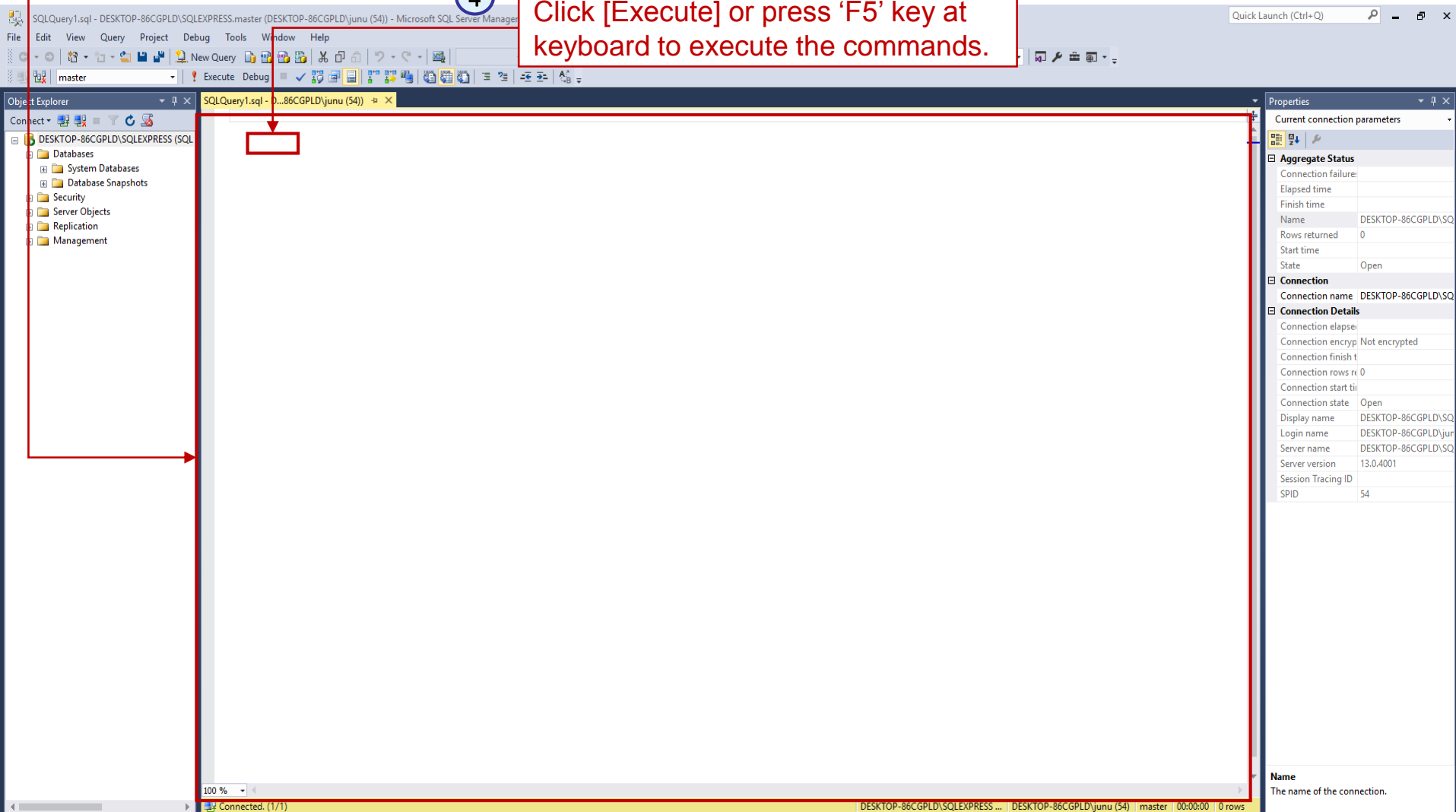
Query editor window (cont'd.)

3

Queries are noted at here.

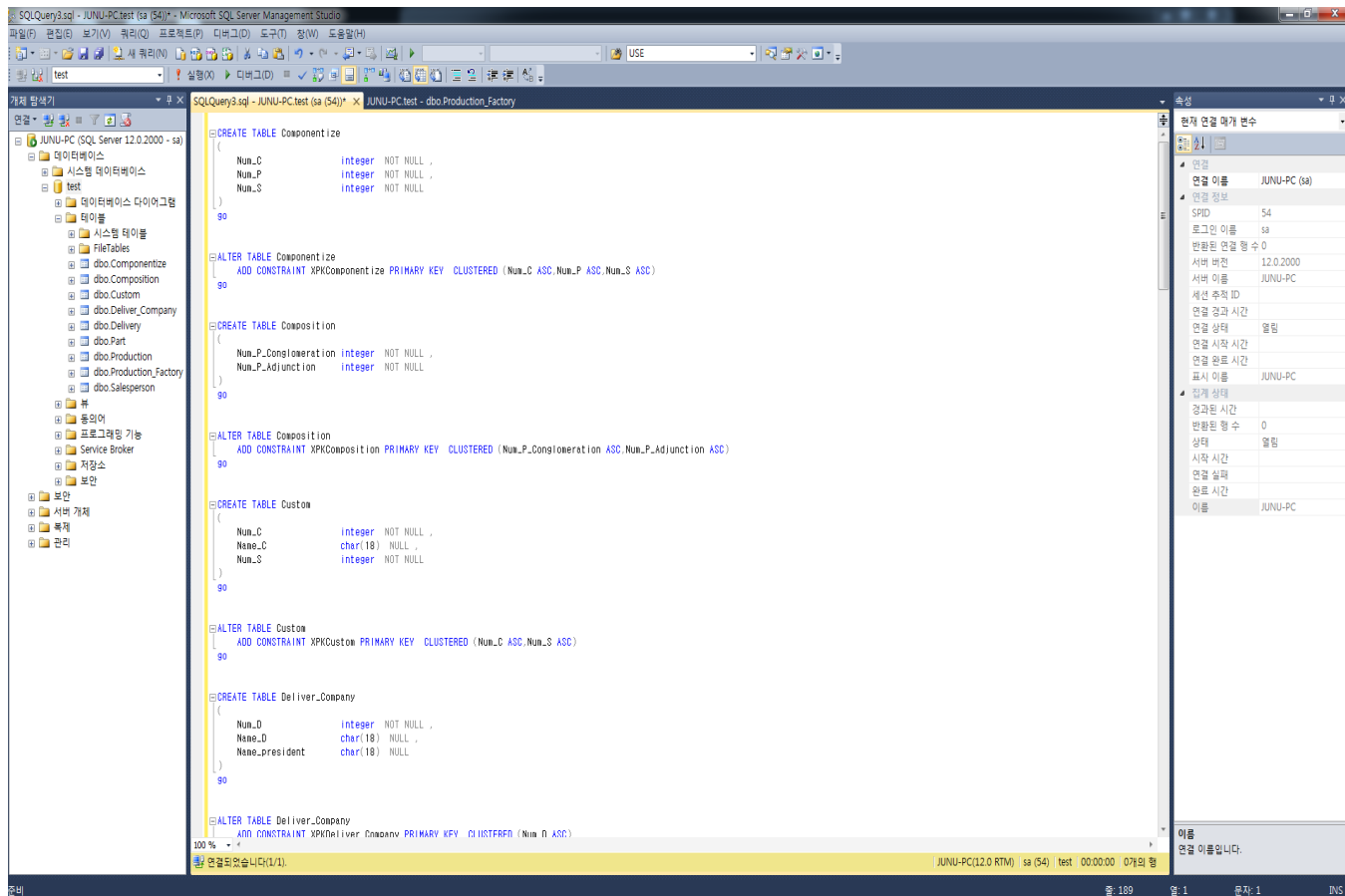
4

Click [Execute] or press 'F5' key at keyboard to execute the commands.



How to use SSMS (SQL server management studio) tool

- Using GUI

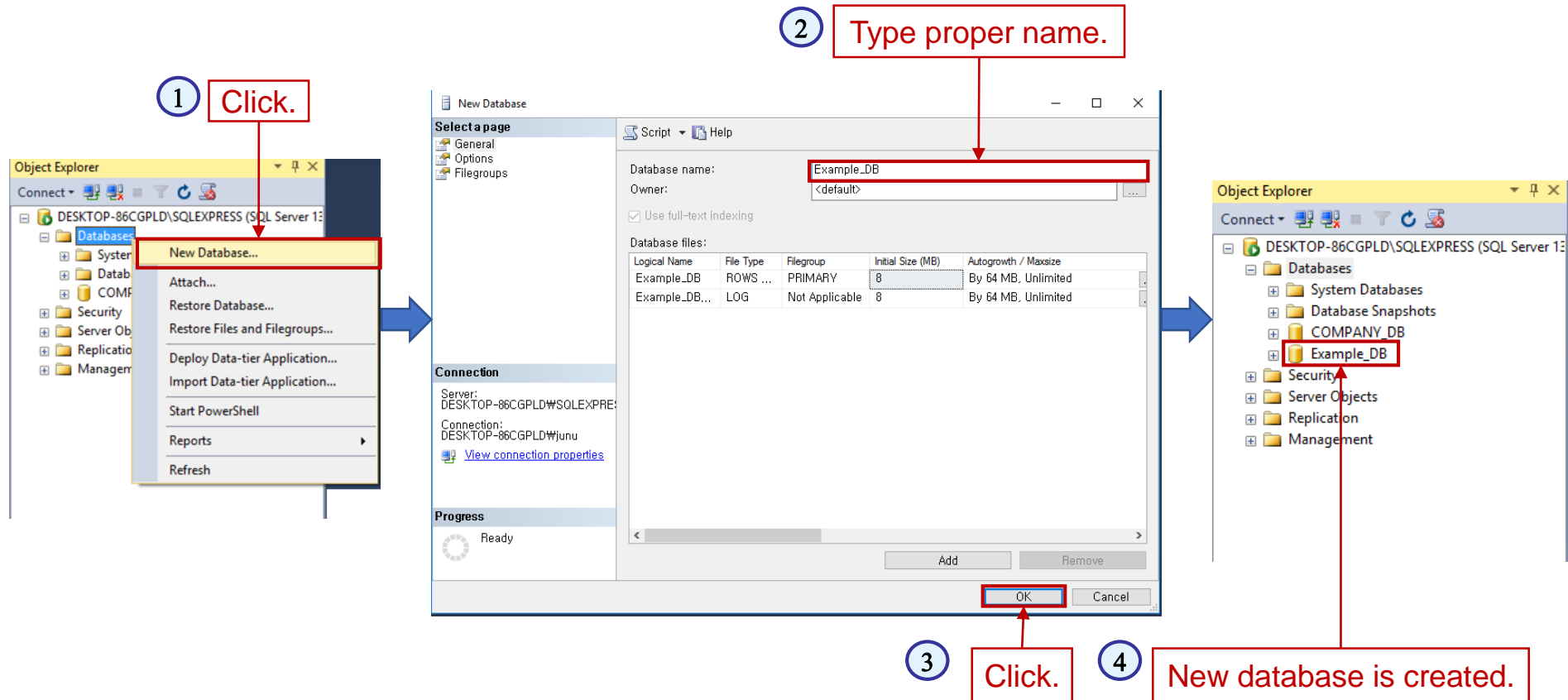


■ Management 기능과 쿼리분석기가 통합된 관리 툴

■ SSMS의 기능

- 서버관리, 데이터베이스 생성/변경,
DB백업 및 복구, 데이터 전송 및 변환

Create database



Create table

1 Click.

2 Set the table schema.
And push 'Ctrl'+ 's' at key board to save.

3 Type proper name.

4 Click.

5 New table is created.

Column Name	Data Type	Allow Nulls
A	int	<input type="checkbox"/>
B	varchar(50)	<input checked="" type="checkbox"/>
C	date	<input checked="" type="checkbox"/>

Column Name	Data Type	Allow Nulls
A	int	<input type="checkbox"/>
B	varchar(50)	<input checked="" type="checkbox"/>
C	date	<input checked="" type="checkbox"/>

Choose Name

Enter a name for the table:

Table_1

OK Cancel

DESKTOP-86CGPLD\SQLEXPRESS (SQL Server 13)

Databases

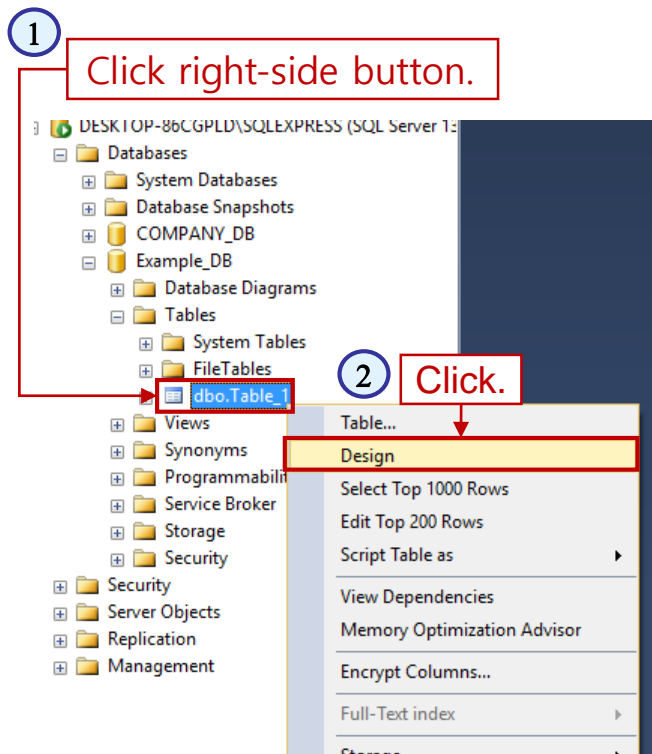
- System Databases
- Database Snapshots
- COMPANY_DB
- Example_DB
 - Database Diagrams
 - Tables
 - System Tables
 - FileTables
 - dbo.Table_1
 - Views
 - Synonyms
 - Programmability
 - Service Broker
 - Storage
 - Security
- Security

Modify table design

1 Click right-side button.

2 Click.

3 Modify table schema.



The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'Databases' folder expanded, showing 'System Databases', 'Database Snapshots', 'COMPANY_DB', 'Example_DB', 'Database Diagrams', 'Tables', 'System Tables', 'FileTables', and 'dbo.Table_1'. The 'dbo.Table_1' table is selected, and a context menu is open with the 'Design' option highlighted. A red box and arrow point to the 'Design' option, labeled '2 Click.' Another red box and arrow point to the right-side button in the top toolbar, labeled '1 Click right-side button.'

Column Name	Data Type	Allow Nulls
A	int	<input type="checkbox"/>
B	varchar(50)	<input checked="" type="checkbox"/>
C	date	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

The screenshot shows the 'Table Design' view for 'dbo.Table_1'. The table has three columns: 'A' (int), 'B' (varchar(50)), and 'C' (date). The 'Allow Nulls' column shows checkboxes for each column: 'A' is unchecked, 'B' is checked, and 'C' is checked. A red box and arrow point to the 'Table Design' tab, labeled '3 Modify table schema.'

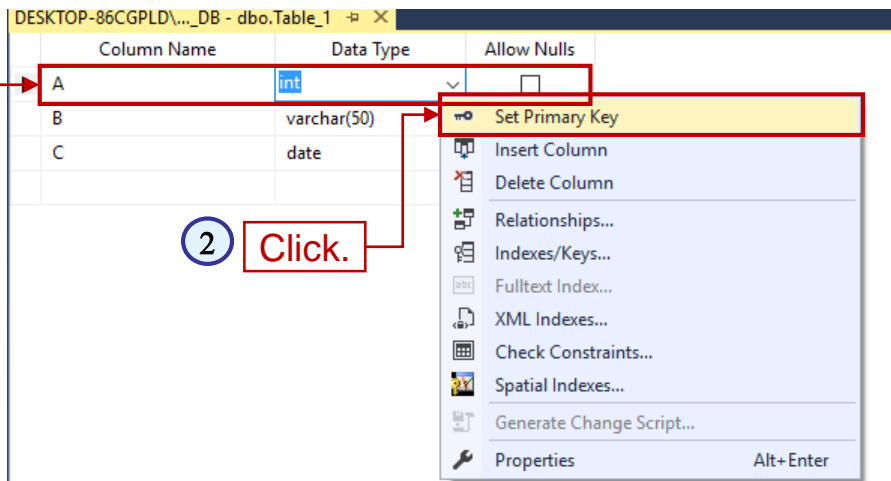
Set primary key

1

Click right-side button anywhere in row of attribute that you want to set primary key.

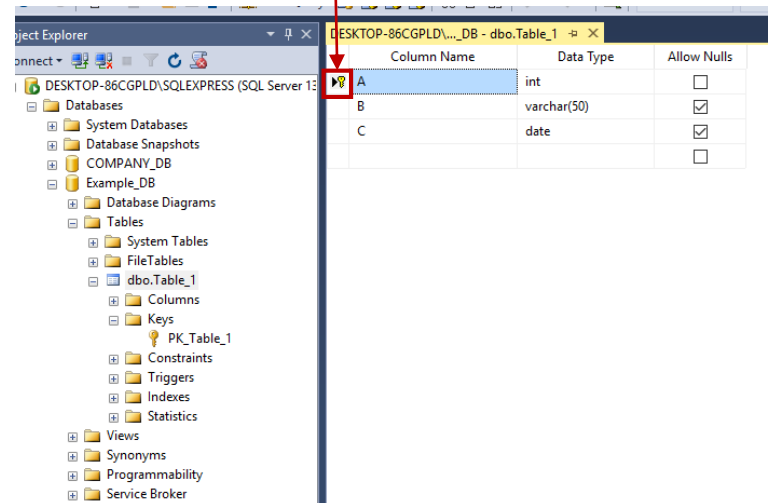
2

Click.



3

You can see key mark.



Set foreign key

- How to connect attribute 'A_' of 'Table_2' into attribute 'A' of 'Table_1':

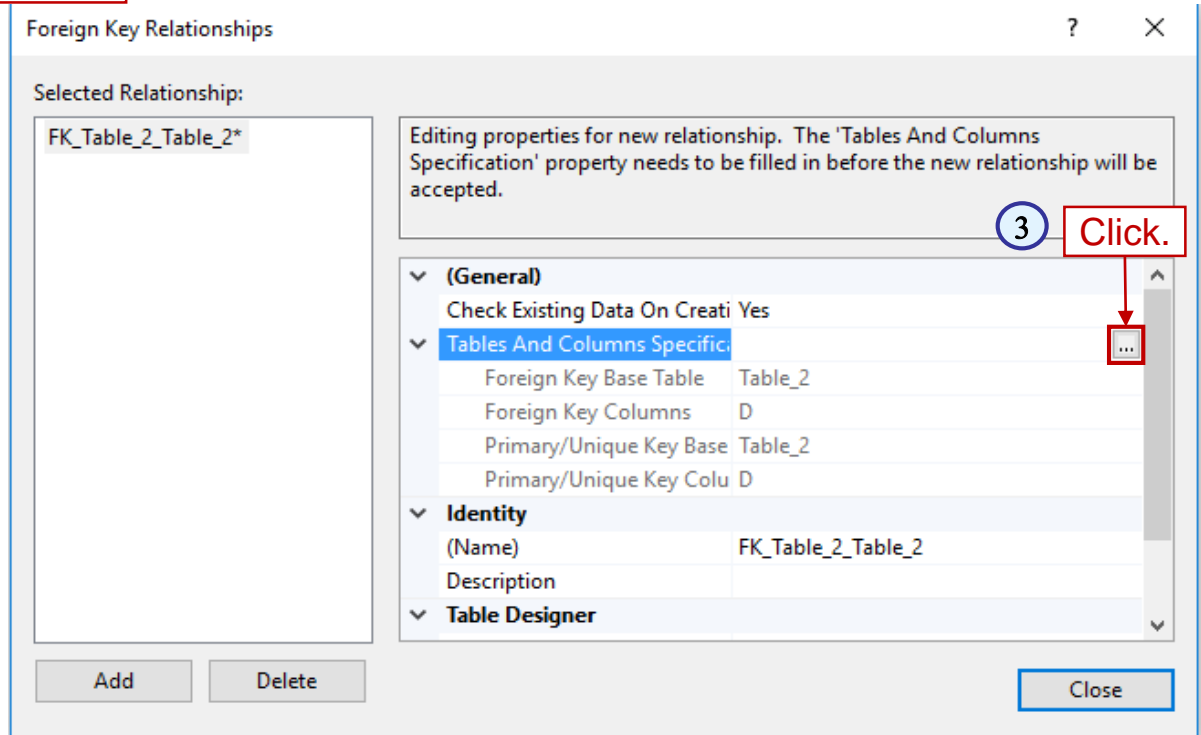
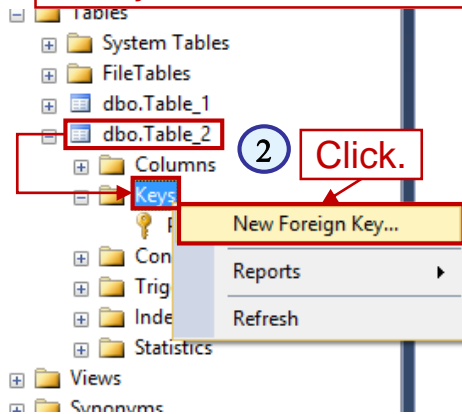
Column Name	Data Type	Allow Nulls
A	int	<input type="checkbox"/>
B	varchar(50)	<input checked="" type="checkbox"/>
C	date	<input checked="" type="checkbox"/>

Column Name	Data Type	Allow Nulls
D	int	<input type="checkbox"/>
E	nchar(10)	<input checked="" type="checkbox"/>
F	nchar(10)	<input checked="" type="checkbox"/>
A_	int	<input type="checkbox"/>

The attributes in FK have the same domain(s) as the primary key attributes PK.

Set foreign key (cont'd.)

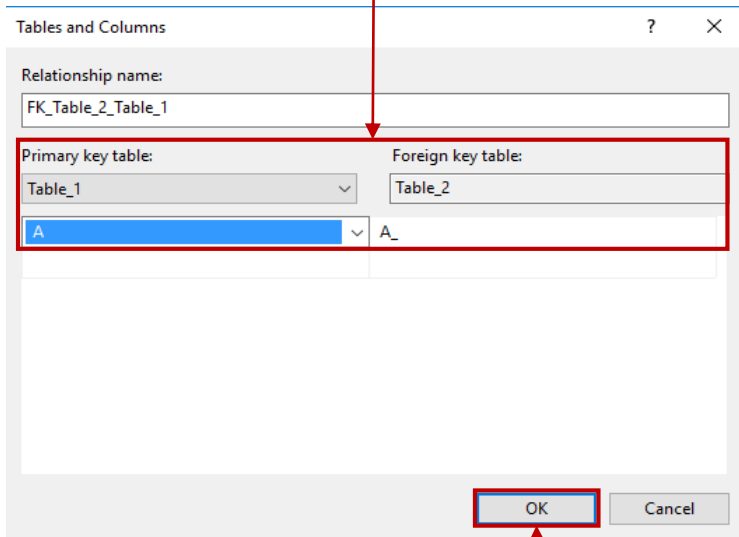
1 Click right-side button [Keys] of table that you want to set foreign key.



Set foreign key (cont'd.)

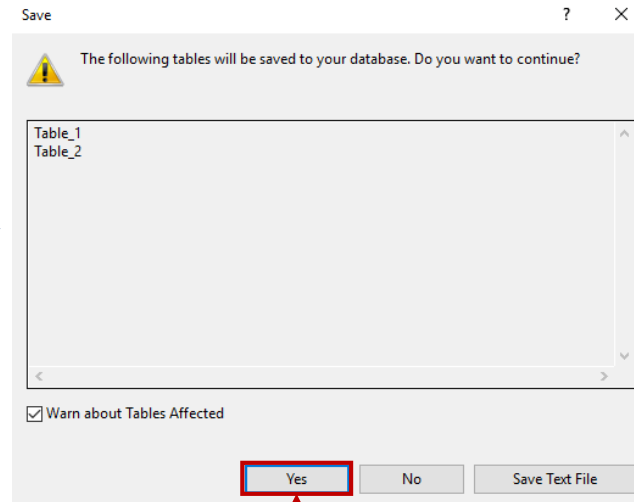
4

Set [Primary key table] and [Foreign key table].



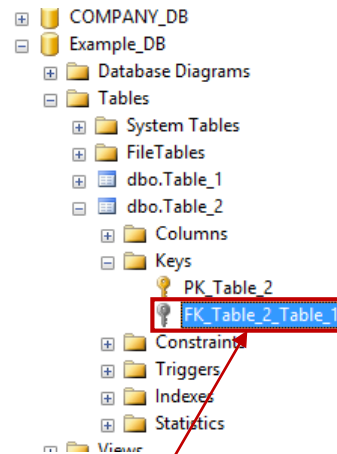
5

Click.



6

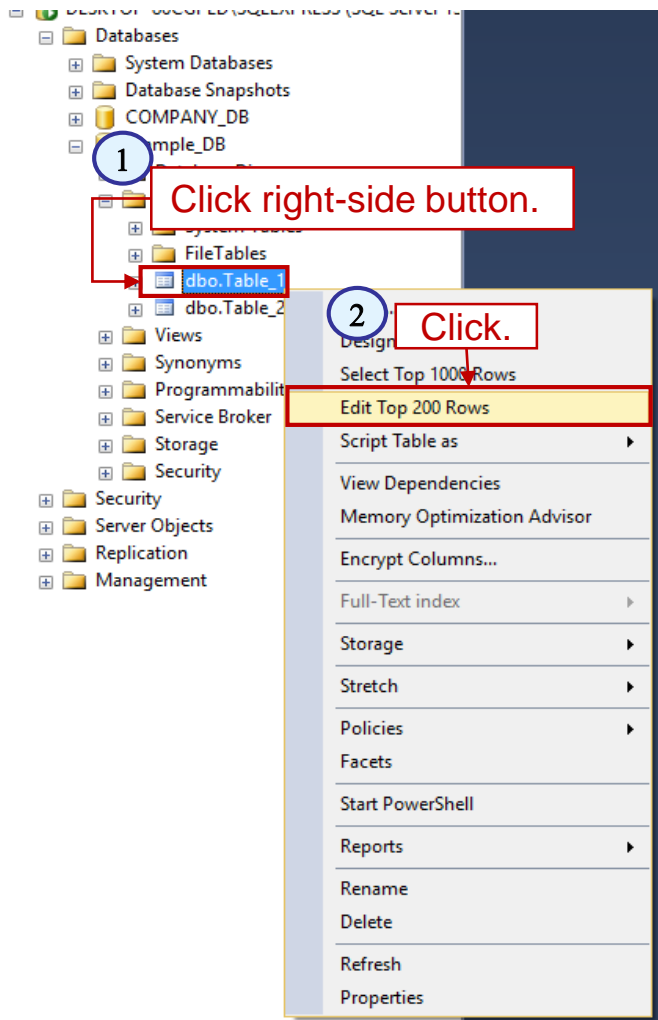
Click.



7

You can see the foreign key.

Insert instances



3 Type to insert instance.

DESKTOP-86CGPLD\..._DB - dbo.Table_1

	A	B	C
	1	AA	2017-03-01
▶*	NULL	NULL	NULL

The screenshot shows a table data view window titled 'DESKTOP-86CGPLD\..._DB - dbo.Table_1'. The table has four columns: an empty column, 'A', 'B', and 'C'. The first row contains the values '1', 'AA', and '2017-03-01'. The second row, which is highlighted with a red box, contains '▶*', 'NULL', 'NULL', and 'NULL|'. A red arrow points from the text 'Type to insert instance.' to the 'NULL|' cell in the second row.

Insert instances (cont'd.)

- When attribute 'A_' of 'Table_2' refers attribute 'A' of 'Table_1':

refers

	A	B	C
1	AA		2017-03-01
▶*	NULL	NULL	NULL

	D	E	F	A_
1		CC	BB	1
▶*	NULL	NULL	NULL	NULL

	D	E	F	A_
1		CC	BB	1
2		AC	QA	5
*	NULL	NULL	NULL	NULL

Microsoft SQL Server Management Studio

No row was updated.

Error occurs!

Insert statement conflicts with foreign key constraint rule!

Correct the errors and retry or press ESC to cancel the change(s).

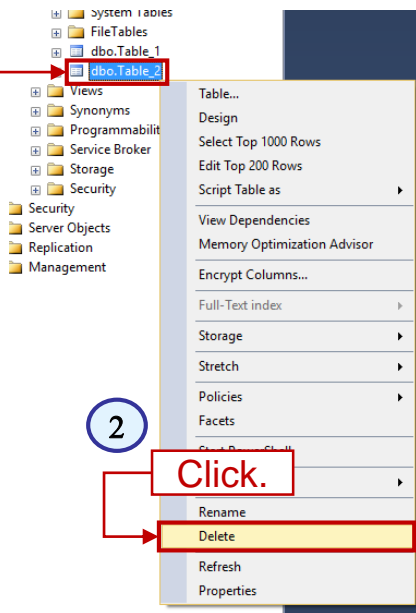
OK Help

Conflict referential integrity constraint.
Value of FK in a tuple t1 of the current state r1(R1) either occurs as a value of PK for some tuple t2 in the current state r2(R2) or is NULL.
(refer 25 slide in chapter 3.)

Delete table

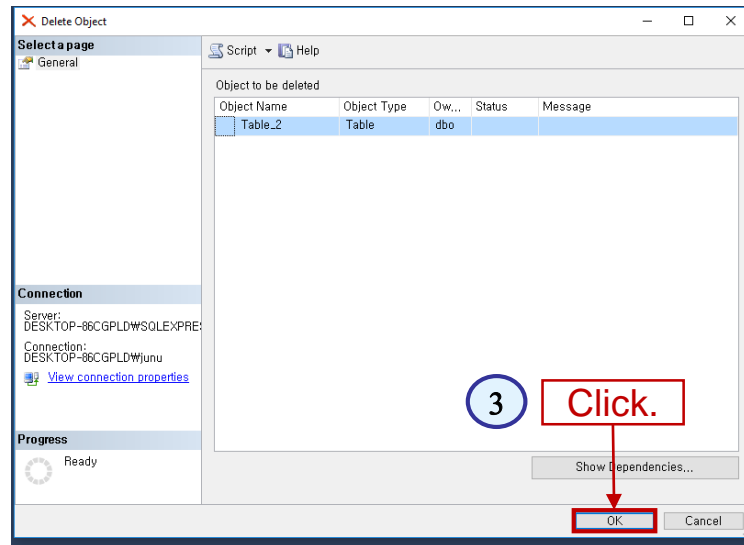
1

Click right-side button.



2

Click.

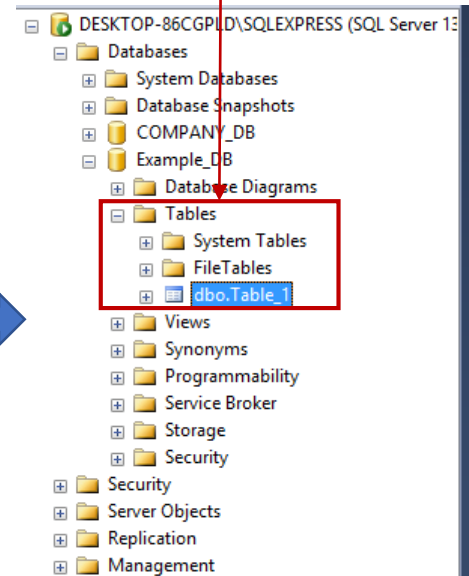


3

Click.

4

Check the table is deleted.



System view (catalogs)

1

Open [Views]→[System Views].

2

Click.

DESKTOP-86CGPLD\SQLEXPRESS (SQL Server 13.0.4001 - DESKTOP-86CGPLD\junu)

Databases

System Databases

Example_DB

Database Diagrams

Tables

Views

System Views

- INFORMATION_SCHEMA.CHECK_CONSTRAINTS
- INFORMATION_SCHEMA.COLUMN_DOMAIN_USAGE
- INFORMATION_SCHEMA.COLUMN_PRIVILEGES
- INFORMATION_SCHEMA.COLUMNS
- INFORMATION_SCHEMA.CONSTRAINT_COLUMN_USAGE
- INFORMATION_SCHEMA.CONSTRAINT_TABLE_USAGE
- INFORMATION_SCHEMA.DOMAIN_CONSTRAINTS
- INFORMATION_SCHEMA.DOMAINS
- INFORMATION_SCHEMA.KEY_COLUMN_USAGE
- INFORMATION_SCHEMA.PARAMETERS
- INFORMATION_SCHEMA.REFERENTIAL_CONSTRAINTS
- INFORMATION_SCHEMA.ROUTINE_COLUMNS
- INFORMATION_SCHEMA.ROUTINES
- INFORMATION_SCHEMA.SCHEMATA
- INFORMATION_SCHEMA.SEQUENCES
- INFORMATION_SCHEMA.TABLE_CONSTRAINTS
- INFORMATION_SCHEMA.TABLE_PRIVILEGES
- INFORMATION_SCHEMA.TABLES
- INFORMATION_SCHEMA.VIEW_COLUMN_USAGE
- INFORMATION_SCHEMA.VIEW_TABLE_USAGE

3

Can see all the tables in 'Example_DB'.

100 %

Results Messages

	TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	TABLE_TYPE
1	Example_DB	dbo	Table_1	BASE TABLE

How to use SSMS (SQL server management studio) tool

- Using DDL commands.

Create database

```
CREATE DATABASE database_name
```

```
[ ON
```

```
  [ < filespec > [ ,...n ] ]
```

```
  [ , < filegroup > [ ,...n ] ]
```

```
]
```

```
[ LOG ON { < filespec > [ ,...n ] } ]
```

```
[ COLLATE collation_name ]
```

```
[ FOR LOAD | FOR ATTACH ]
```

데이터베이스의 데이터 부분이 저장되는 데이터 파일 정의

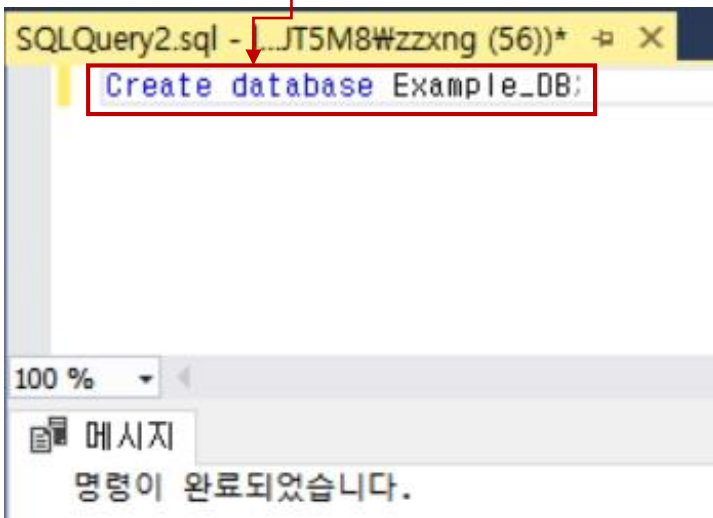
데이터베이스의 로그가 저장되는 로그파일을 정의

데이터 정렬 정의

이전 버전과의 호환성을 위해 지원되는 절

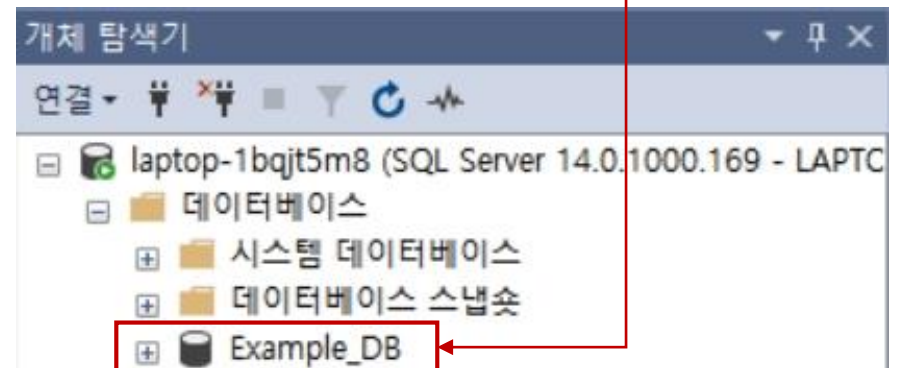
①

Type to create database



②

New database is created.



Create table

CREATE TABLE

```
[ database_name.[ owner ] . | owner. ] table_name
( { < column_definition >
  | column_name AS computed_column_expression
  | < table_constraint > ::= [ CONSTRAINT constraint_name ] {
    [ [ { PRIMARY KEY | UNIQUE } [ ,...n ]
  }
)
[ ON { filegroup | DEFAULT } ]
[ TEXTIMAGE_ON { filegroup | DEFAULT } ]
```

PRIMARY KEY, NOT NULL, UNIQUE, FOREIGN KEY 또는
CHECK 제약 조건 정의

constraint	Description
NULL NOT NULL	Null 허용 여부
PRIMARY KEY	지정한 열에 대해 고유한 인덱스 지정(1개)
UNIQUE	지정한 열에 대해 고유한 인덱스 지정(n개)
FOREIGN KEY...REFERENCES	참조 무결성 제공
ON DELETE {CASCADE NO ACTION}	부모 테이블에서 삭제시 해당 행 삭제
ON UPDATE {CASCADE NO ACTION}	부모 테이블에서 삭제시 해당 행 업데이트
CHECK	열에 입력할 수 있는 값 제한

① Type to create table

SQLQuery2.sql - I...JT5M8#zzxng (53) SQLQuery4.sql - I...JT5M8#zzxng (53)* X

```
Create Table Table_1
(
  A int NOT NULL,
  B varchar(52),
  C date,
  PRIMARY KEY(A));
```

```
Create Table Table_2
(
  D int NOT NULL,
  E varchar(52),
  F date,
  A_ int NOT NULL,
  PRIMARY KEY(D),
  FOREIGN KEY(A_) REFERENCES Table_1(A));
```

100 % 100 %

메시지 메시지

명령이 완료되었습니다. 명령이 완료되었습니다.

② New table is created.

Example_DB

- 데이터베이스 다이어그램
- 테이블
 - 시스템 테이블
 - FileTables
 - 외부 테이블
 - 그래프 테이블
 - dbo.Table_1
 - dbo.Table_2
- PK_Table_2_3BD019A96AB521B6
- FK_Table_2_A_398D8EEE

Modify table design

```
ALTER TABLE table
{ [ ALTER COLUMN column_name
  { new_data_type [ ( precision [ , scale ] ) ]
    [ COLLATE < collation_name > ]
    [ NULL | NOT NULL ]
    [ {ADD | DROP } ROWGUIDCOL }
  ]
  | ADD
    { [ < column_definition > ]
      | column_name AS computed_column_expression
      } [ ,...n ]
  | [ WITH CHECK | WITH NOCHECK ] ADD
    { < table_constraint > } [ ,...n ]
  | DROP
    { [ CONSTRAINT ] constraint_name
      | COLUMN column } [ ,...n ]
  | { CHECK | NOCHECK } CONSTRAINT
    { ALL | constraint_name [ ,...n ] }
  | { ENABLE | DISABLE } TRIGGER
    { ALL | trigger_name [ ,...n ] }
}
```

변경된 열의 새 데이터 형식

열에 null 값 허용 여부를 지정

ROWGUID 속성을 추가/삭제

하나 이상의 열 정의 / 테이블 제약 조건 추가 지정

테이블에서 열 / 제약 조건 제거

제약조건 설정 여부 지정

트리거 설정 여부 지정

Insert instances

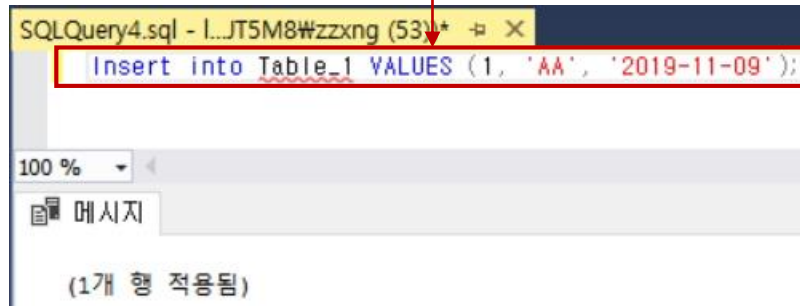
```
INSERT INTO [table_name] VALUES ('data1', 'data2', ... );
```

// 테이블 생성시의 Attribute 순서와 dataType에 맞게 입력

```
INSERT INTO [table_name] (attr1, attr2, ... ) VALUES ('data1', ....);
```

// 전체 Attribute가 아닌 특정 Attribute에만 입력 할 경우 사용

① Type to insert instances



② You can see inserted instances

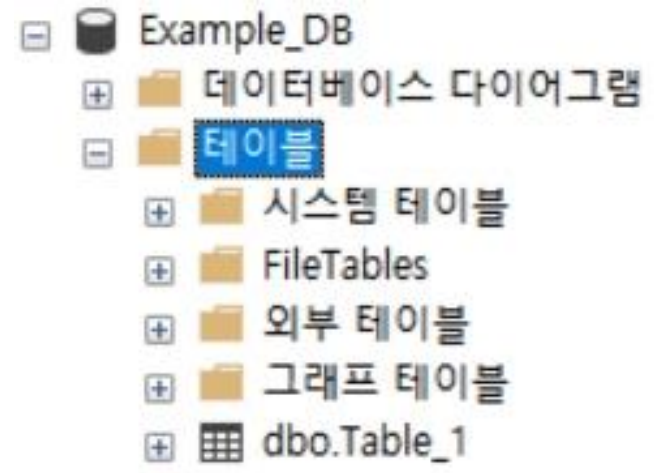
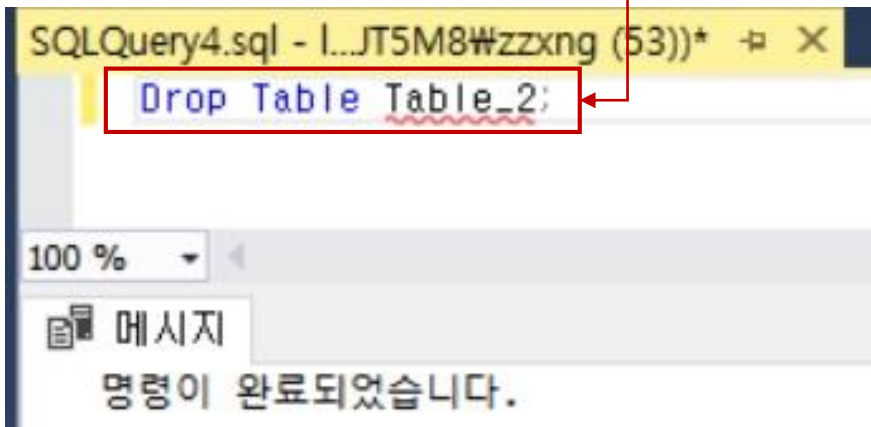
A screenshot of a table viewer showing the result of the insert operation. The table has columns A, B, and C. The first row contains the values 1, AA, and 2019-11-09. A red box highlights the first row. A blue arrow points from the SQL editor to this table viewer.

	A	B	C
1	1	AA	2019-11-09

Delete table

```
DROP TABLE table_name
```

- 1 Type to delete table





THANK YOU



인하대학교