



សាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ
មហាវិទ្យាល័យ វិទ្យាសាស្ត្រ
ជេហ៍តាម៉ង់៖ ព័ត៌មានវិទ្យា

Database Systems Design

គ្រូ៖ Kheam Hong



ក្រុមទី១៖ ទេស ចន្ទមករា

បូ ម៉េងឡុង

ពៅ ជានីត

ពេស សុក្រី

ជឿង ភីយ៉ា

SQL

Basic SQL

+SQL (Structured Query Language)

ជាភាសាម៉ាស៊ីនមួយប្រើសម្រាប់ការផ្ទុក និង process លើ Information នៅក្នុង Relational Database ។ យើងត្រូវការ IDE ឬ DBMS ដូចជា SSMS, Oracle, MySQL... ដើម្បី Compile ភាសានេះ។ យើងប្រើ SQL statements សម្រាប់ store, update, remove, search, retrieve information ពី Database ។

+ Relational database:

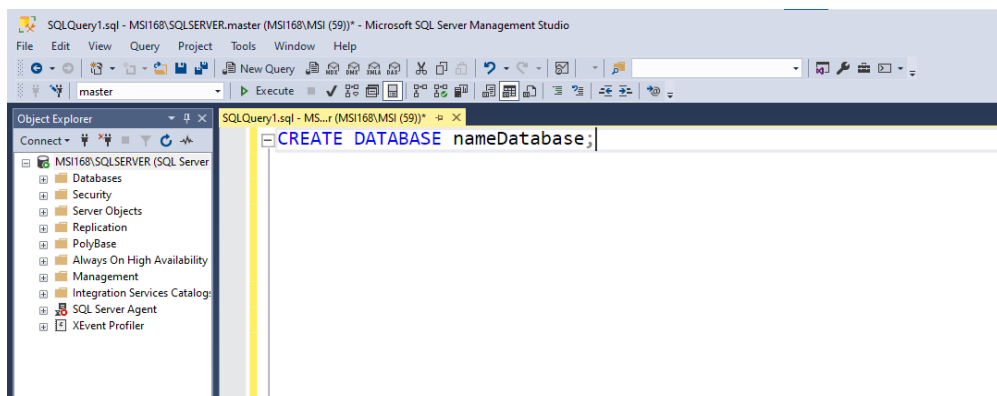
គឺជា data ដែលរក្សាទិន្នន័យនិងផ្តល់ទិន្នន័យដែលទាក់ទងគ្នាទៅវិញទៅមក។ ដូចជា table ដែលមាន primary key ដែលកំណត់អត្តសញ្ញាណជាក់លាក់ទៅលើ Record នីមួយៗនិង foreign key ដែលបង្កើតតំណភ្ជាប់រវាងតារាង។

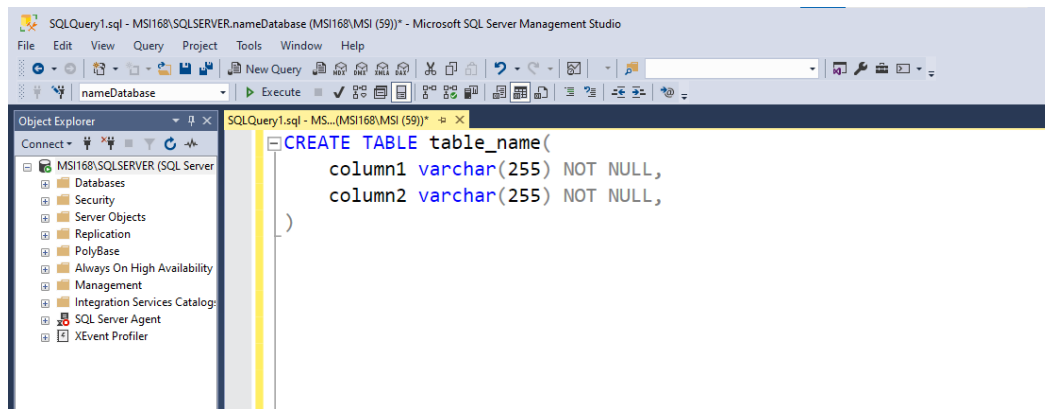
- ក្នុង **Relational database** ទិន្នន័យត្រូវបានរក្សាទុកក្នុង table ទៅតាម row និង column នីមួយៗ។

+ SQL Command:

- **DDL (Data definition language)**: មាននាទីសម្រាប់បង្កើត Table កែប្រែ Table និងលុបTable។
 - **CREATE**: ប្រើសម្រាប់បង្កើត Database, Table, View, Function, Trigger, Procedure,....

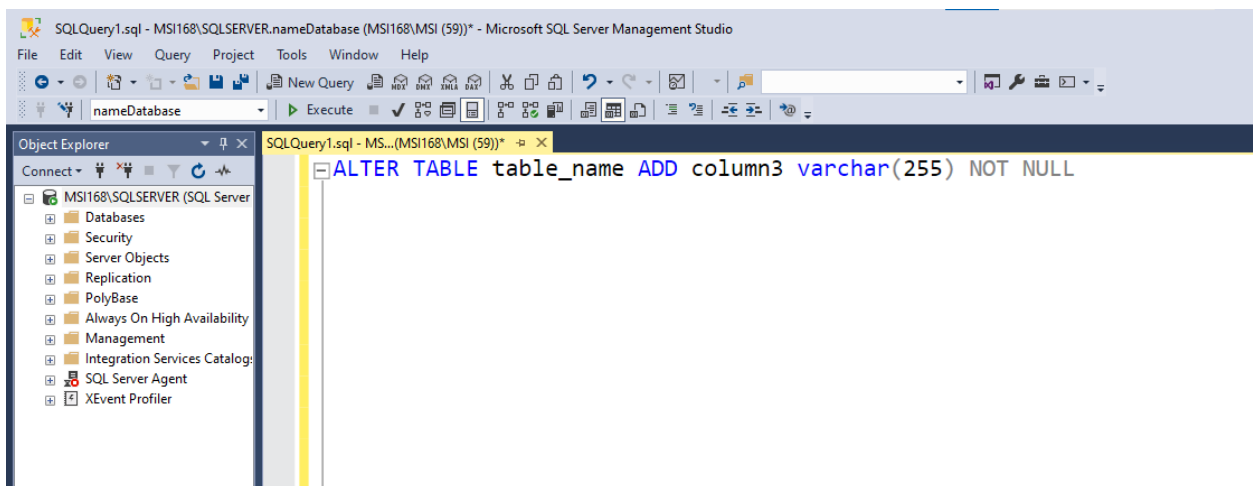
Example:





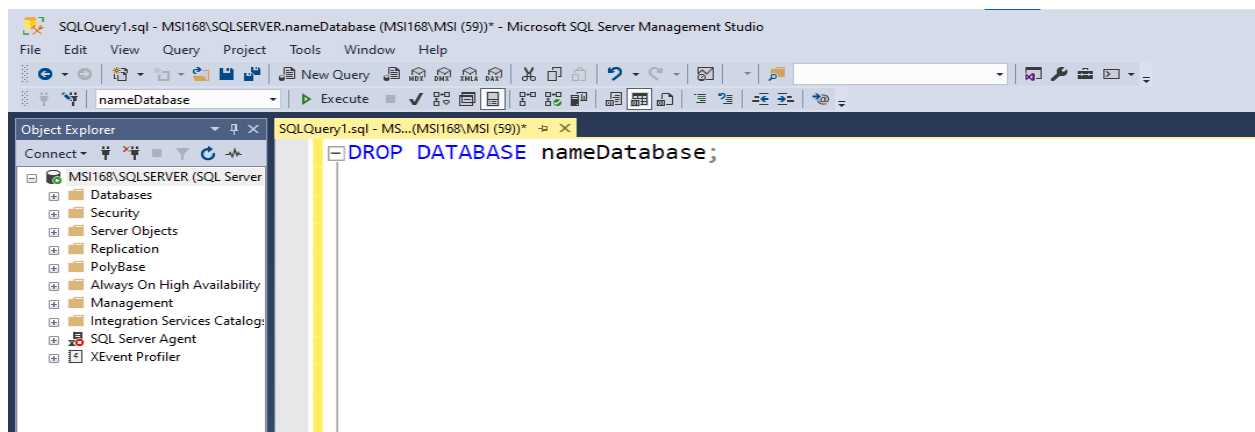
- ALTER: ប្រើសម្រាប់កែប្រែ Table, Database, Column,...

Example:



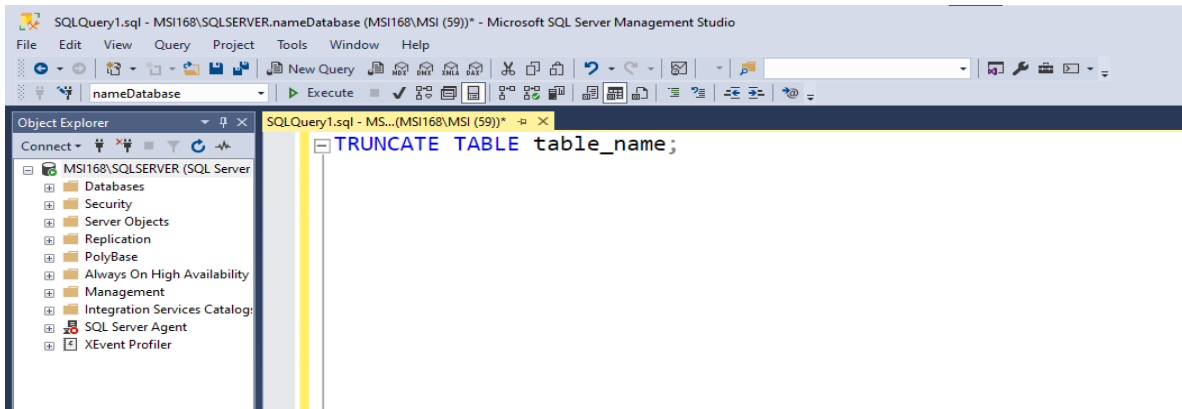
- DROP: ប្រើសម្រាប់លុប Table , Database , Column ,.....

Example:

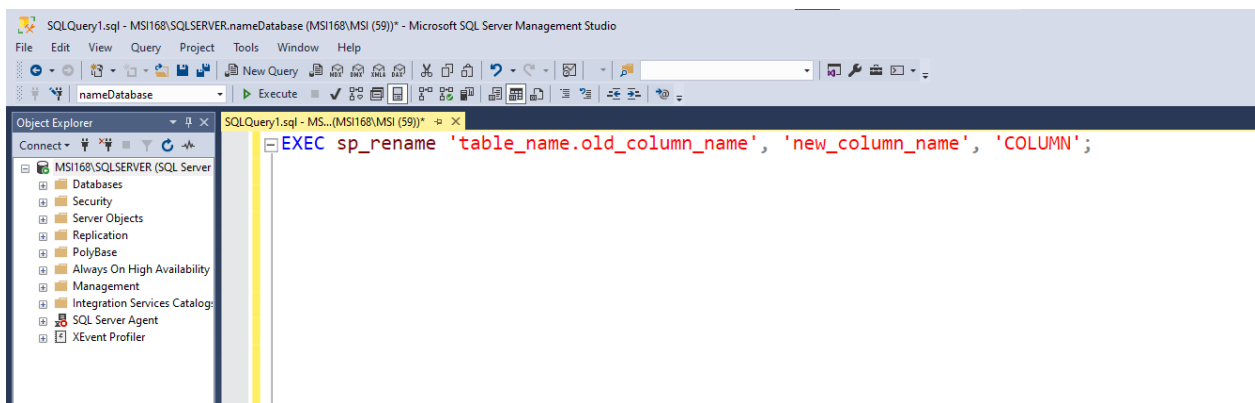


- TRUNCATE: ប្រើសម្រាប់ Reset data ក្នុង table និង Reset Identity column ហើយវាមិន អាចប្រើជាមួយ table ដែលមានតំណភ្ជាប់ Foreign key ទេ។

Example:

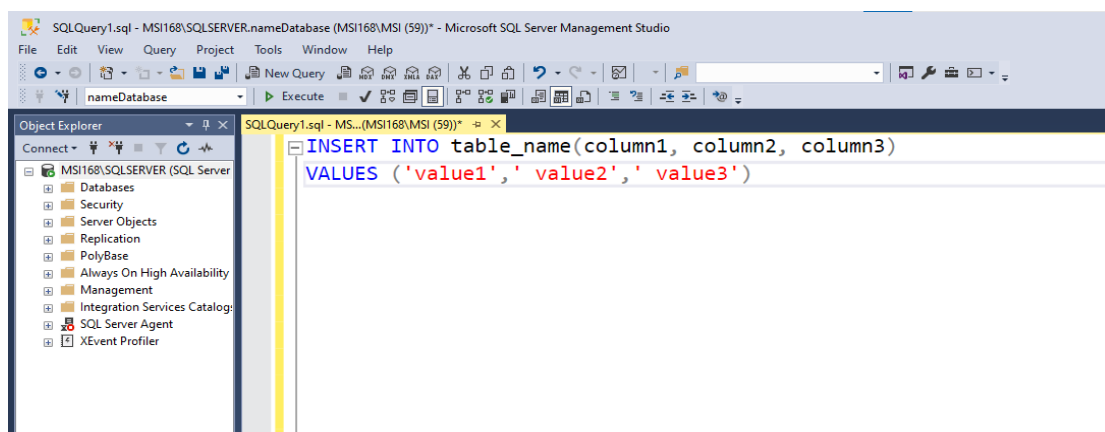


- RENAME: សម្រាប់ប្តូរឈ្មោះ table , database , column ,.....

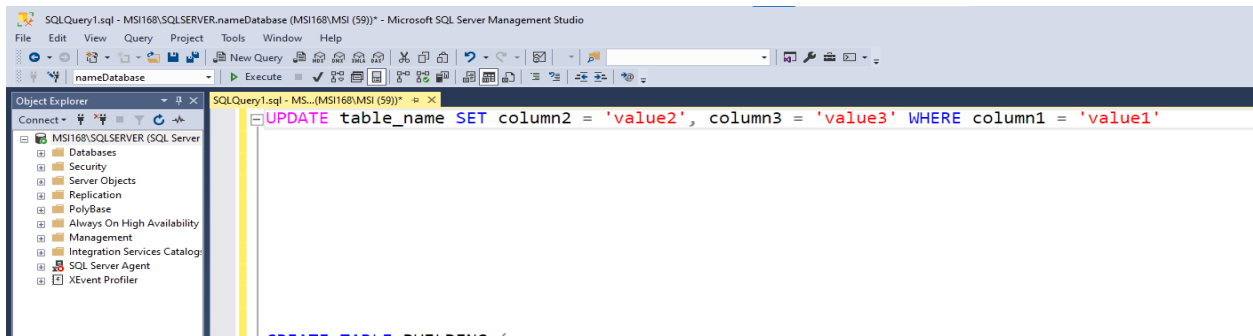


- DML (Data Manipulation language): មានតួនាទីសម្រាប់បញ្ចូល ធ្វើការកែប្រែ លុប និងទាញយកទិន្នន័យពីតារាងទិន្នន័យ។

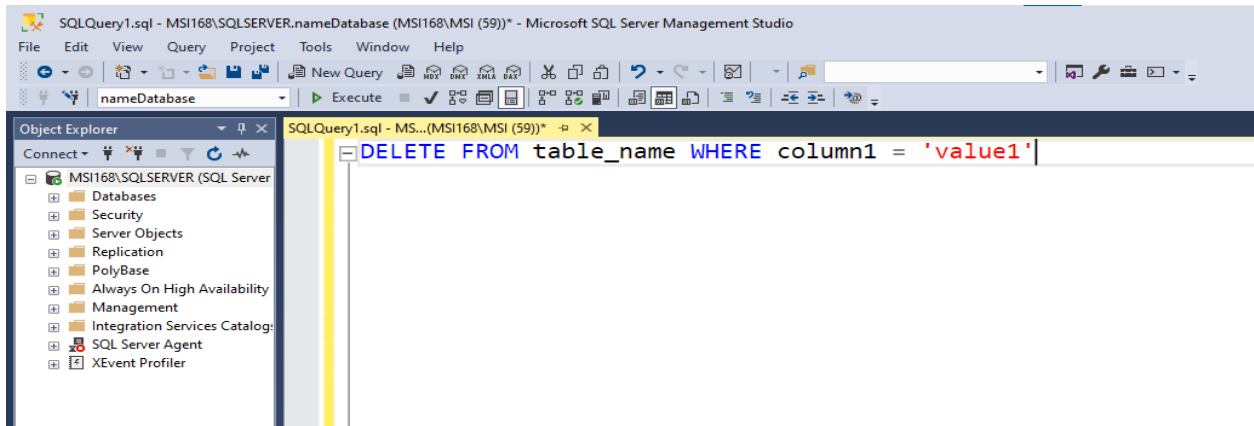
- INSERT: សម្រាប់បញ្ចូល data ទៅកាន់ table



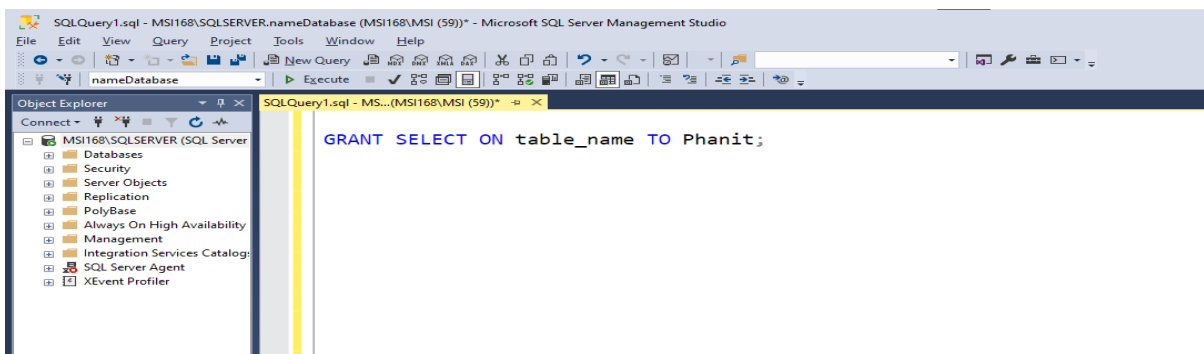
- UPDATE: សម្រាប់កែប្រែ data ក្នុង table



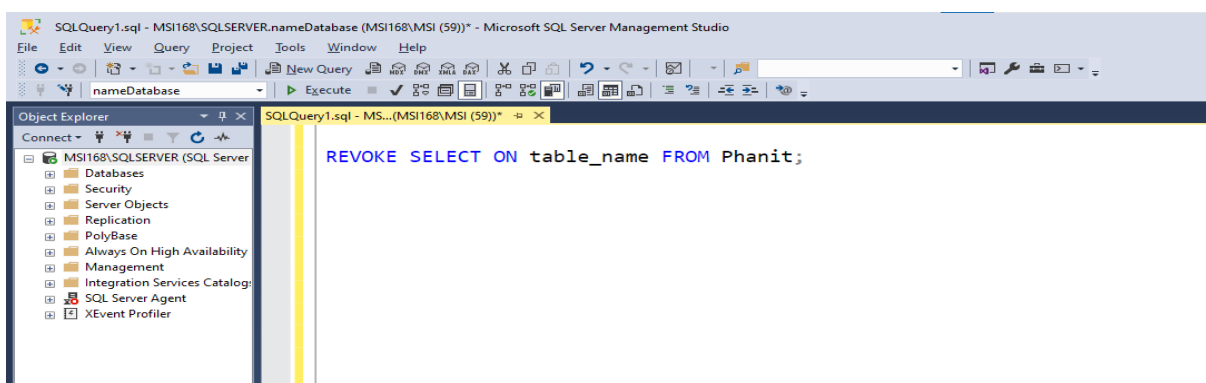
- DELETE: សម្រាប់លុបទិន្នន័យក្នុង table



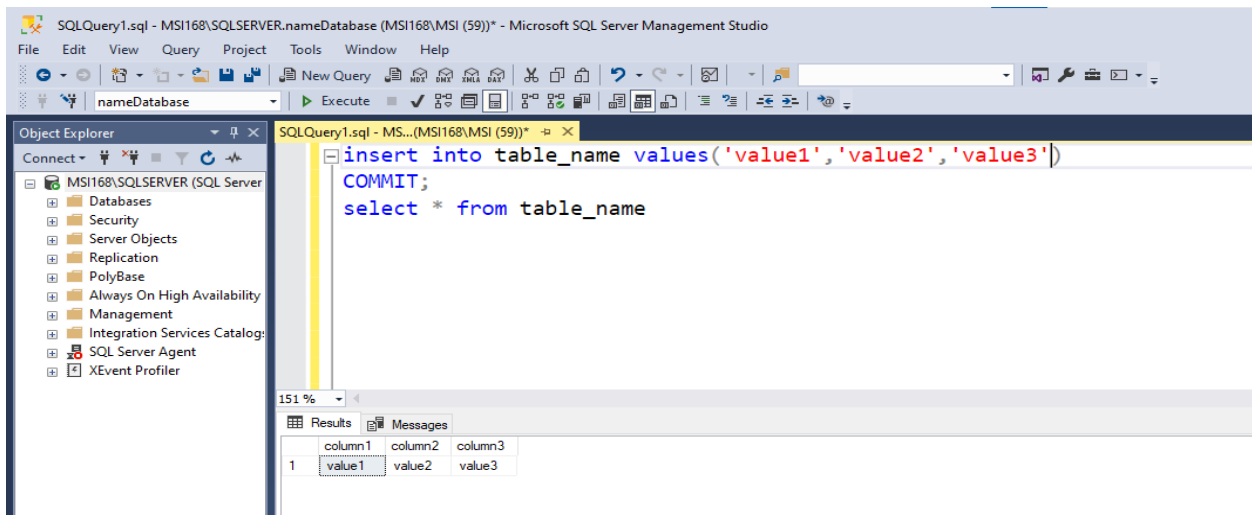
- MERGE: សម្រាប់ផ្គុំ table ពីឲ្យទៅជាមួយ table
- DCL (Data control language): មាននាទីសម្រាប់កំណត់ role ឲ្យ user និងដក role ដែលបានកំណត់ចេញពី user វិញ។
- GRANT: សម្រាប់កំណត់ role ឲ្យ user



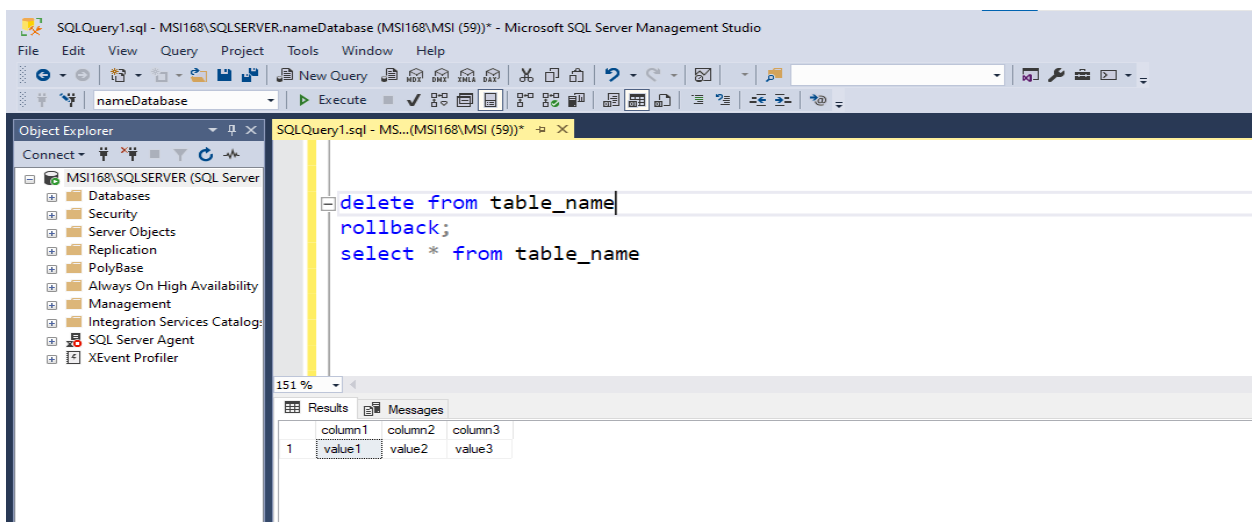
- REVOKE: សម្រាប់ដក role ដែលបានកំណត់ចេញពី user វិញ



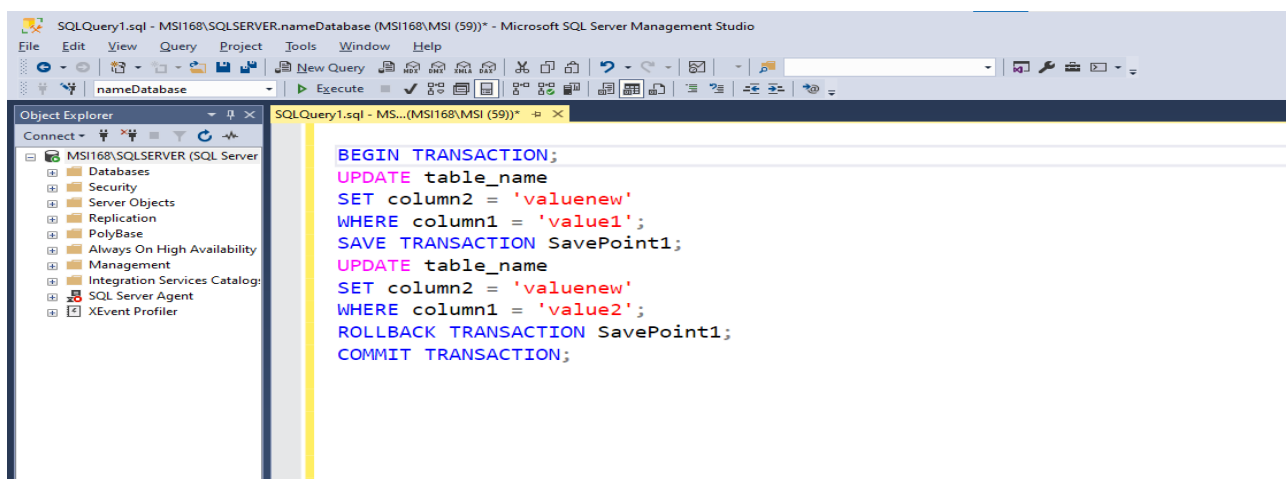
- TCL (Transaction control language): មាននាទីត្រួតពិនិត្យប្រែប្រួលទិន្នន័យ។
 - o COMMIT: ប្រើសម្រាប់រក្សាទុកទិន្នន័យក្នុងអំឡុងពេលប្រតិបត្តិការជៀសវាងdataត្រូវបានបាត់



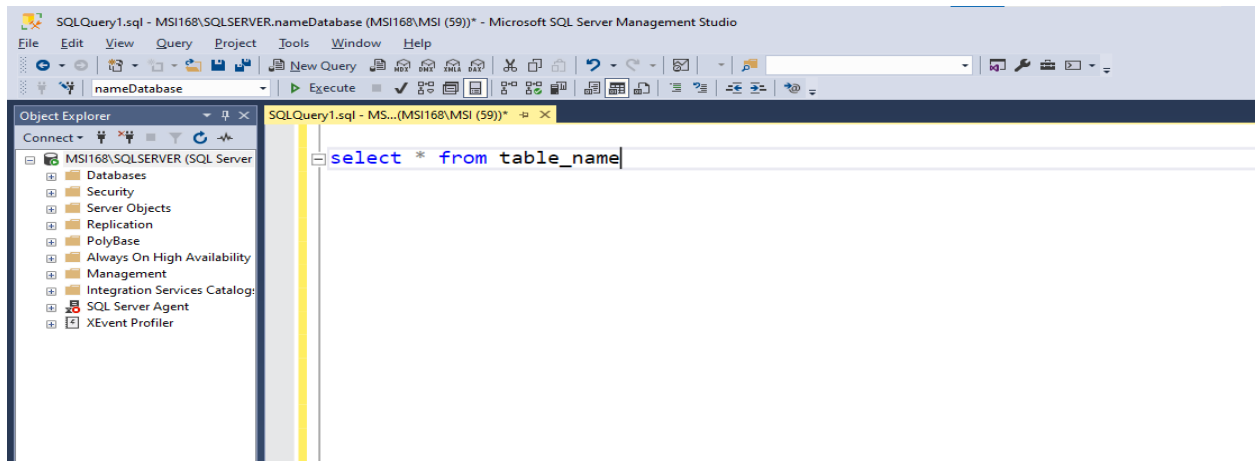
- o ROLLBACK: នៅពេលទិន្នន័យត្រូវបានបាត់យើងទាញមកវិញលុះត្រាតែយើងបានCommit



- o SAVE POINT : ប្រើសម្រាប់រក្សាទុកទិន្នន័យក្នុងអំឡុងពេលប្រតិបត្តិការប៉ុន្តែវាអនុញ្ញាតអោយយើងរំលងប្រតិបត្តិការដែលមិនចាំបាច់បាន



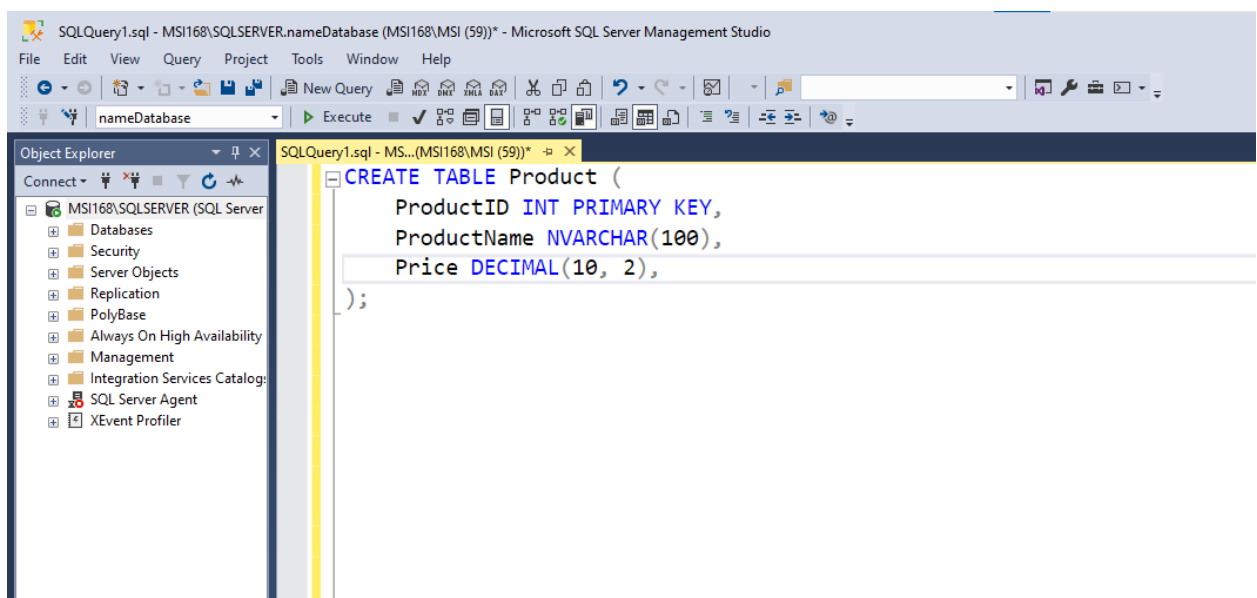
- **DQL (Data Query Language)**: មានសំខាន់ៗគឺ SELECT ។
វាអនុញ្ញាតឱ្យយើងស្វែងរកទិន្នន័យ សម្រាប់រៀបចំ, ការស្វែងរក, ការស្តិត,
និងការរាយការណ៍។
 - o **SELECT**: សម្រាប់ select data បញ្ចេញមក



+ Datatype:

- String: nvarchar, varchar, char, Text,...
- Integer: tinyint, smallint, int, bigint,...
- Date : date, time, datetime,....
- Float , decimal ,
- Boolean : Bit ,
- Identity (Auto increment)

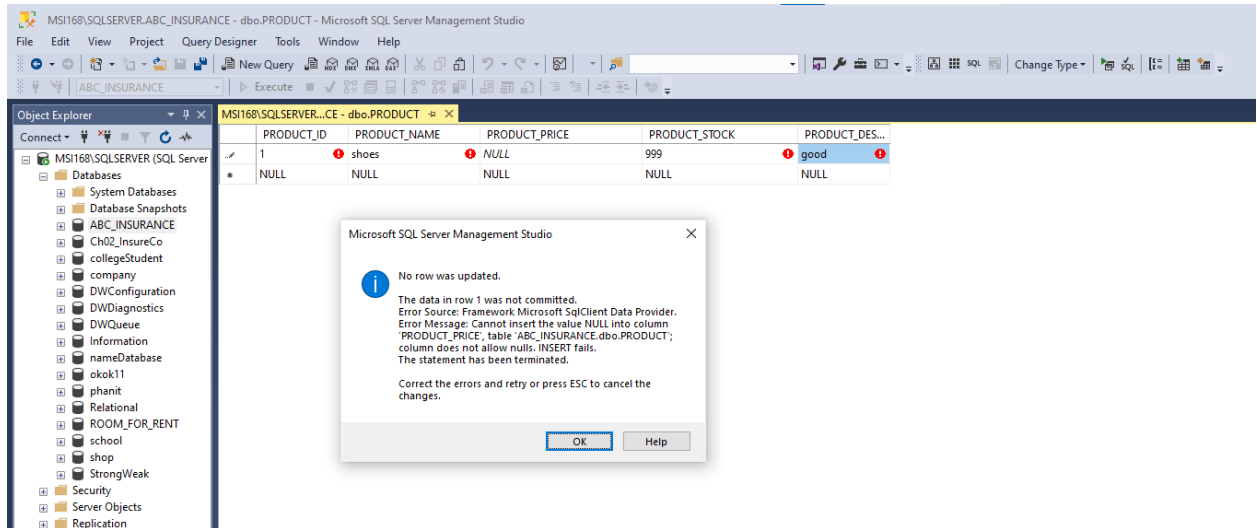
Example



+ Constraint

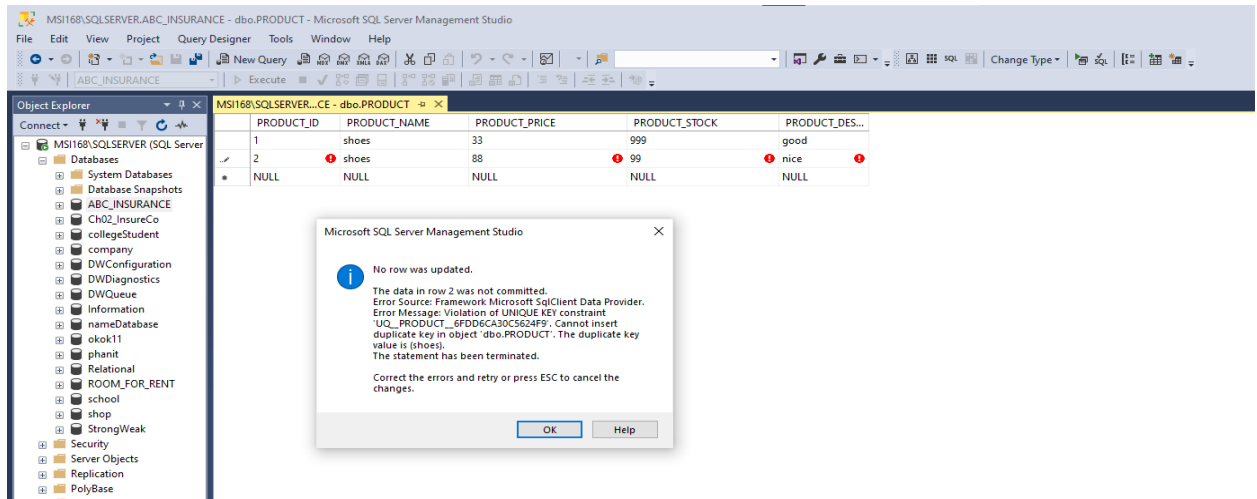
- Not Null: តម្លៃក្នុង table មិនអាច null ទេ

Example : ទៅលើ column PRODUCT_PRICE



- UNIQUE: តម្លៃក្នុង column ត្រូវតែខុសគ្នា

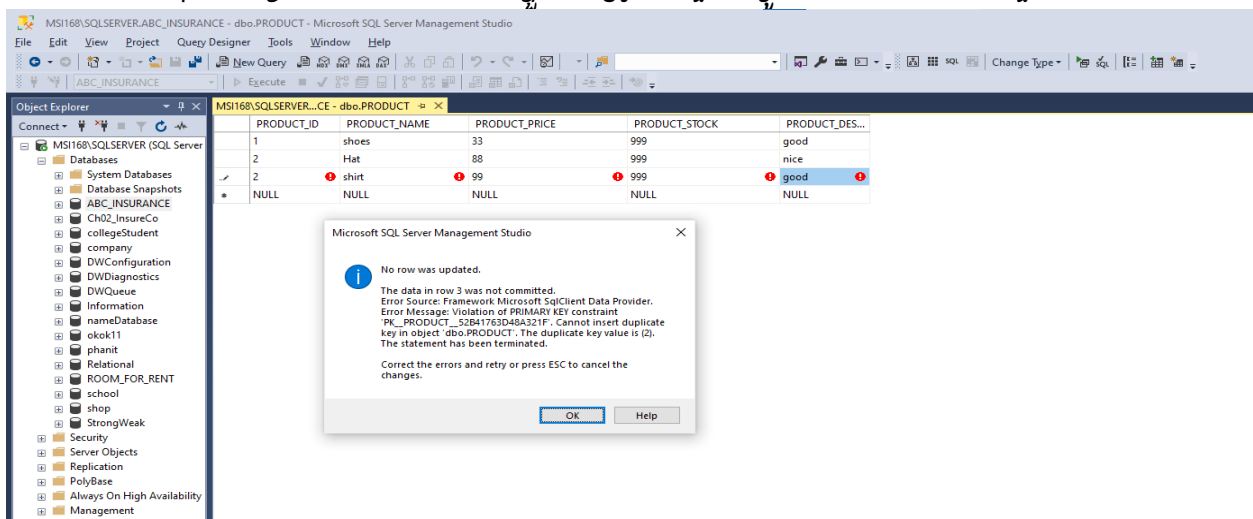
Example : Column PRODUCT_NAME មានឈ្មោះដូចគ្នា



- Primary Key: ត្រូវតែមានតម្លៃ និងអត្តសញ្ញាខុសគ្នា

Example : Column PRODUCT_ID មាន ID លេខ2 មាន 2 record ជា

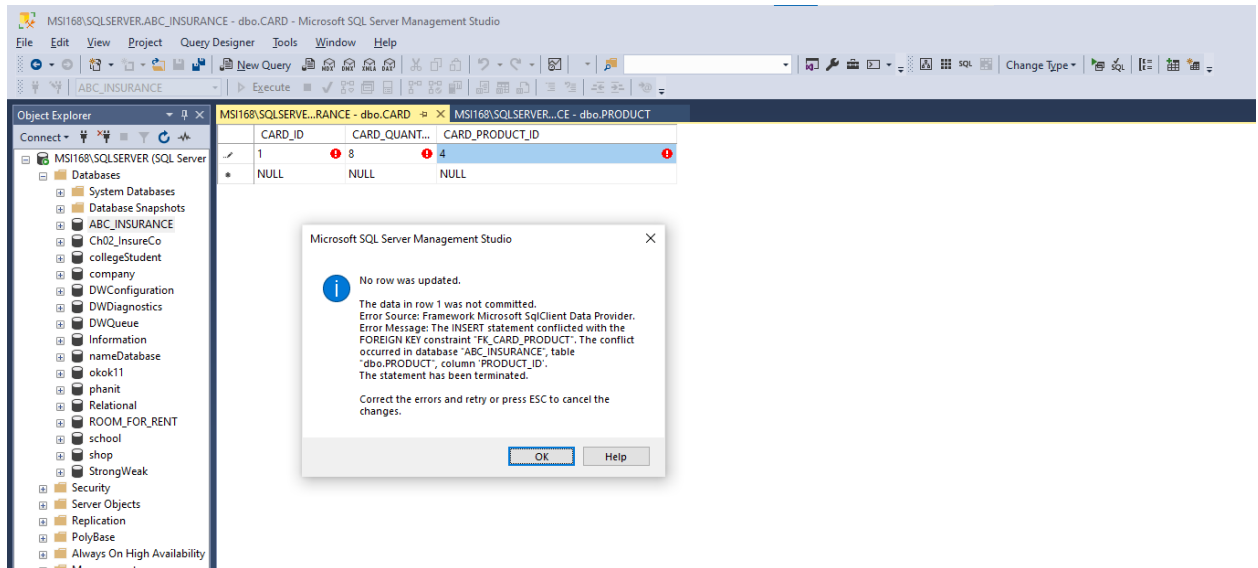
ហេតុដែលធ្វើអោយ Primakey ស្អាត (វាត្រូវបានរួមបញ្ចូល NOT NULL ជាមួយ UNIQUE)



- Foreign Key: តំណភ្ជាប់ table ពី primary key

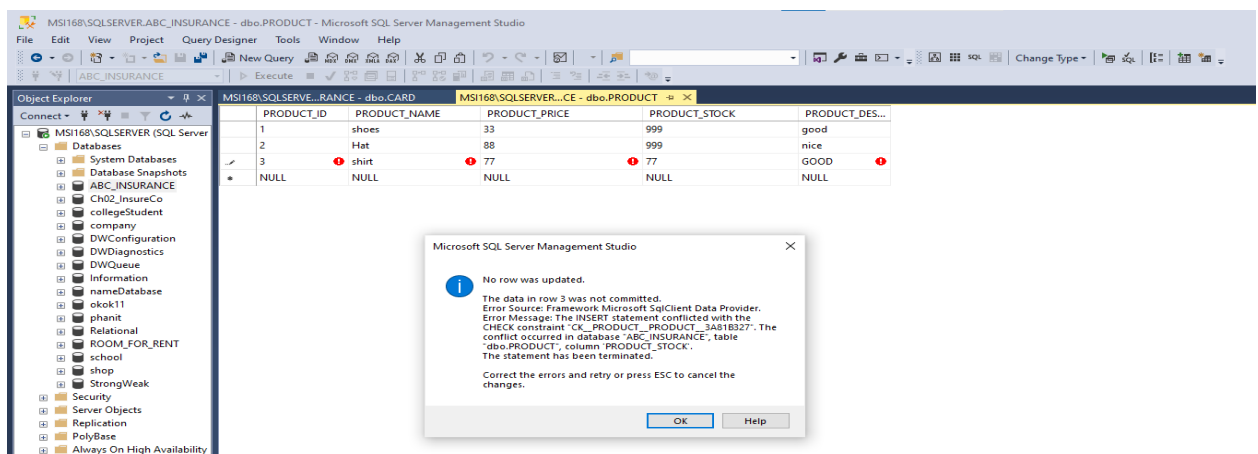
Example : Column CARD_PRODUCT_ID ដោយសារតែវាជា FOREIGN KEY

នោះតម្លៃរបស់វាត្រូវតែយកតាម PRIMARY KEY



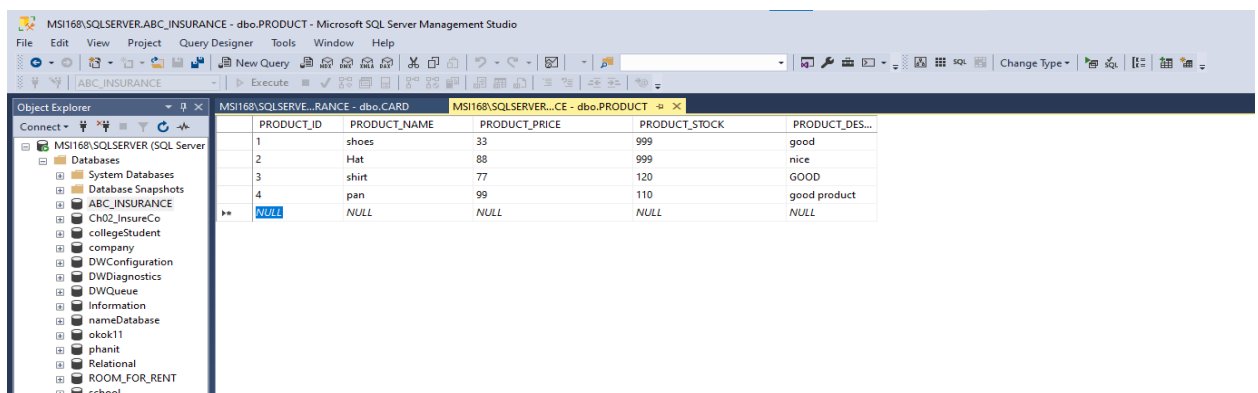
- Check: សម្រាប់ដាក់លក្ខខណ្ឌអោយ Column នីមួយៗ

Example : Column PRODUCT_STOCK ដោយសារតែចំនួនវា $77 < 100$



- default សម្រាប់ដាក់តម្លៃជាក់លាក់ ទៅអោយ Column នៅពេលដែលយើងមិនបានបញ្ចូលទិន្នន័យ ។

Example : Column PRODUCT_DESCRIPTION នៅពេលដែលមិនបានបញ្ចូលទិន្នន័យ វានឹងចាប់យក default 'good product' របស់យើង ។



+ Normalization

– 1NF:

- មិនឲ្យមាន repeating group
- ប្រាកដថា Cell នីមួយៗត្រូវតែមាន single value

Students

FirstName	LastName	Knowledge
Thomas	Mueller	Java, C++, PHP
Ursula	Meier	PHP, Java
Igor	Mueller	C++, Java

Startsituation

Result after Normalisation

Students

FirstName	LastName	Knowledge
Thomas	Mueller	C++
Thomas	Mueller	PHP
Thomas	Mueller	Java
Ursula	Meier	Java
Ursula	Meier	PHP
Igor	Mueller	Java
Igor	Mueller	C++

– 2NF:

- ប្រាកដថាវានៅក្នុង 1NF
- បង្កើត Table នីមួយៗទៅតាម partial dependencyនិងកំណត់ key ឲ្យវា
- ករណីដែលវាអស់ transitive dependency នោះវានឹងក្លាយជា 3NF (auto)

Student

<u>Stu_ID</u>	FirstName	LastName
1	Thomas	Mueller
2	Ursula	Meier
3	Igor	Mueller

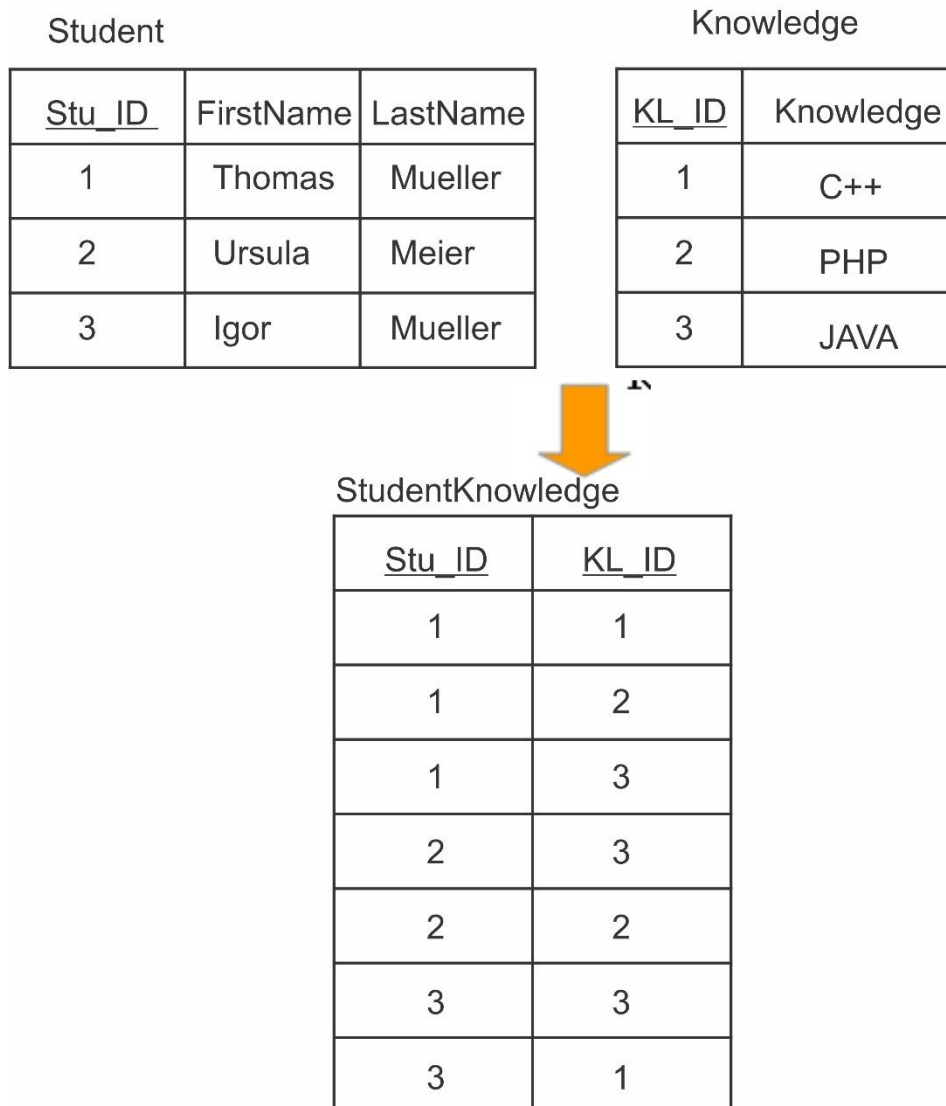
Knowledge

<u>KL_ID</u>	Knowledge
1	C++
2	PHP
3	JAVA

StudentKnowledge

<u>Stu_ID</u>	<u>KL_ID</u>
1	1
1	2
1	3
2	3
2	2
3	3
3	1

- 3NF:
 - ប្រាកដថាវានៅក្នុង 2NF
 - បង្កើត Table ថ្មីមួយទៅតាម transitive dependency ជាមួយ key



+ Operator

- Arithmetic operator (+ , - , * , % , /)

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL query:

```
select * , PRODUCT_PRICE -10 AS TOTAL_PRICE from PRODUCT
```

The Results pane displays the following data:

	PRODUCT_ID	PRODUCT_NAME	PRODUCT_PRICE	PRODUCT_STOCK	PRODUCT_DESCRIPTION	TOTAL_PRICE
1	1	shoes	33	999	good	23
2	2	Hat	88	999	nice	78
3	3	shirt	77	120	GOOD	67
4	4	pan	99	110	good product	89

- Comparative Operator (= , < , > , <= , >= !=)

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL statement:

```
SELECT * FROM PRODUCT WHERE PRODUCT_ID=1
```

The query has been executed, and the results are displayed in the Results pane. The results show a single row with the following data:

PRODUCT_ID	PRODUCT_NAME	PRODUCT_PRICE	PRODUCT_STOCK	PRODUCT_DESCRIPTION
1	shoes	33	999	good

- Union : សម្រាប់បង្ហាញទិន្នន័យ TABLE ពីរបញ្ចូលគ្នា ឬ ច្រើន និង លុបទិន្នន័យដែលដូចគ្នាចោល ។

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL statement:

```
SELECT PRODUCT_ID FROM PRODUCT
UNION
SELECT CARD_ID FROM CARD
```

The query has been executed, and the results are displayed in the Results pane. The results show a list of product IDs from the PRODUCT table and card IDs from the CARD table, with duplicate values removed:

PRODUCT_ID
1
2
3
4

- Union All: សម្រាប់បង្ហាញទិន្នន័យ TABLE ពីរបញ្ចូលគ្នា ឬ ច្រើន ។

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL statement:

```
SELECT PRODUCT_ID FROM PRODUCT
UNION ALL
SELECT CARD_ID FROM CARD
```

The query has been executed, and the results are displayed in the Results pane. The results show a list of product IDs from the PRODUCT table and card IDs from the CARD table, including duplicate values:

PRODUCT_ID
1
2
4
3
1
2
3

+ Importance SQL

- Distinct: កុំឱ្យចេញ record ដែលមានឈ្មោះដដែល

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL statement:

```
SELECT DISTINCT (PRODUCT_NAME) FROM PRODUCT
```

The Results pane displays the output of the query:

PRODUCT_NAME
1 Hat
2 pan
3 shirt
4 shoes

- Order by: សម្រាប់ sort តាម column នីមួយៗ

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL statement:

```
SELECT * FROM PRODUCT ORDER BY PRODUCT_PRICE
```

The Results pane displays the output of the query, sorted by PRODUCT_PRICE:

PRODUCT_ID	PRODUCT_NAME	PRODUCT_PRICE	PRODUCT_STOCK	PRODUCT_DESCRIPTION
1	shoes	33	999	good
2	shirt	77	120	GOOD
3	Hat	88	999	nice
4	pan	99	110	good product

- Limit: សម្រាប់កំណត់ record ទៅតាមការទាញ

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL statement:

```
SELECT TOP (2) * FROM PRODUCT
```

The Results pane displays the output of the query, limited to the top 2 records:

PRODUCT_ID	PRODUCT_NAME	PRODUCT_PRICE	PRODUCT_STOCK	PRODUCT_DESCRIPTION
1	shoes	33	999	good
2	Hat	88	999	nice

+ Inner Join

- ទាញទិន្នន័យពី table ច្រើនតាមរយៈការ join

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL query:

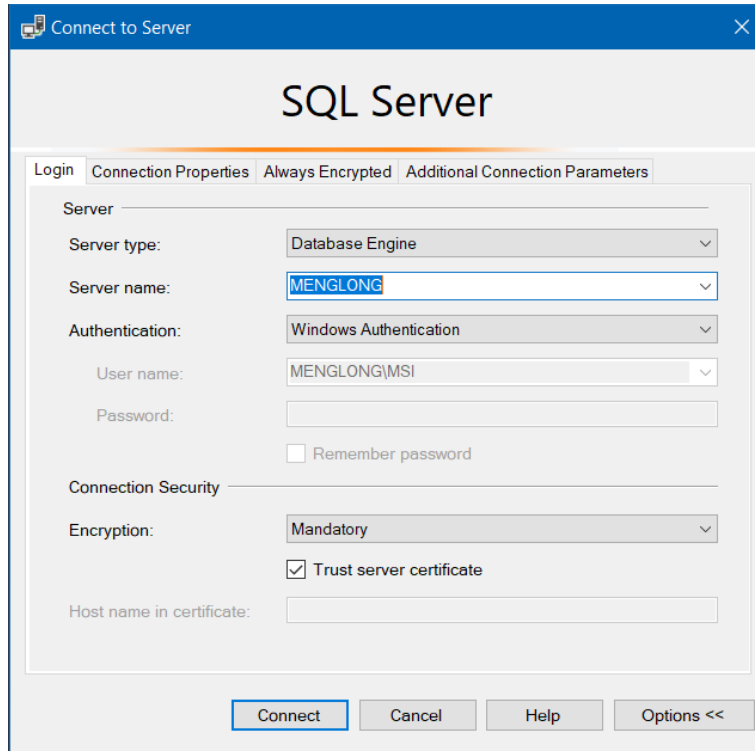
```
SELECT * FROM CARD CD INNER JOIN PRODUCT PD ON CD.CARD_PRODUCT_ID=PD.PRODUCT_ID
```

The query results are displayed in the Results pane, showing three rows of data. The columns are: CARD_ID, CARD_QUANTITY, CARD_PRODUCT_ID, PRODUCT_ID, PRODUCT_NAME, PRODUCT_PRICE, PRODUCT_STOCK, and PRODUCT_DESCRIPTION.

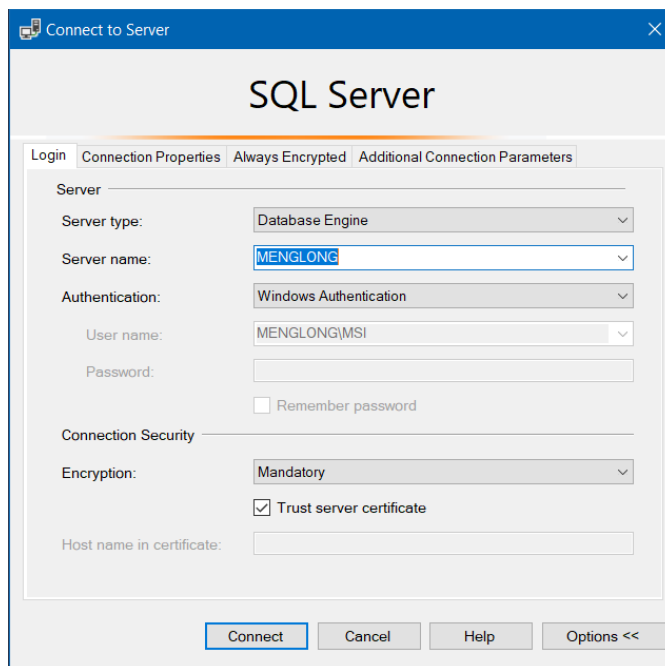
CARD_ID	CARD_QUANTITY	CARD_PRODUCT_ID	PRODUCT_ID	PRODUCT_NAME	PRODUCT_PRICE	PRODUCT_STOCK	PRODUCT_DESCRIPTION
1	8	1	1	shoes	33	999	good
2	7	1	1	shoes	33	999	good
3	6	2	2	Hat	88	999	nice

+ SQL Server Management Studio (SSMS)

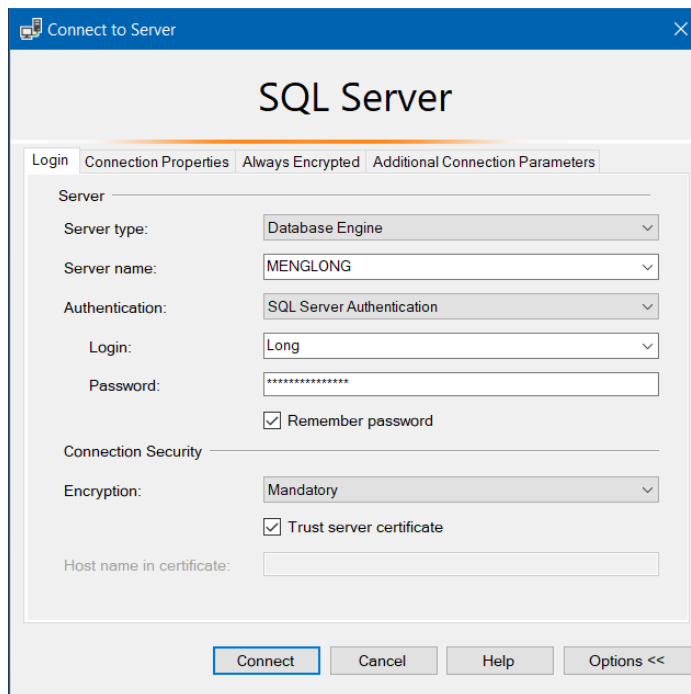
- របៀបចូលទៅប្រើប្រាស់ SQL Server Management Studio:
 - បើក IDE SSMS យើងនឹងឃើញផ្ទាំងដូចខាងក្រោម:



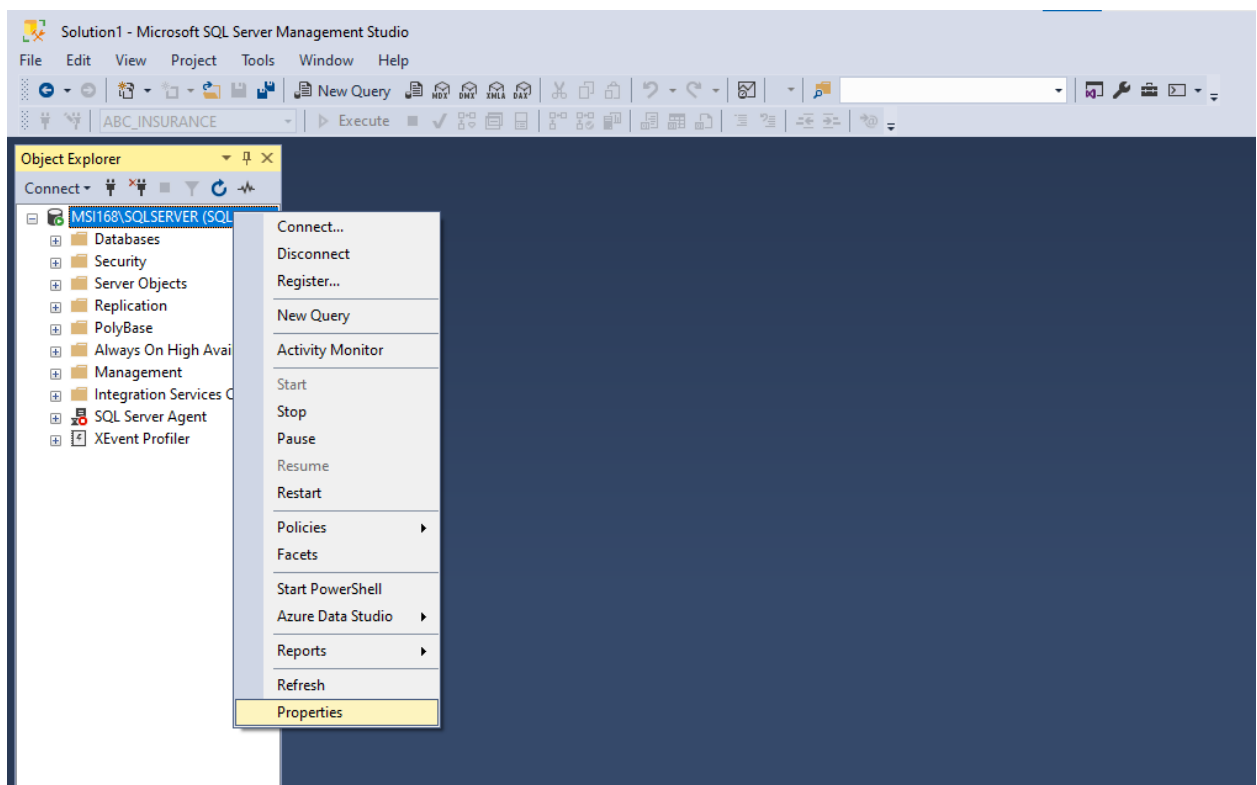
- ការ connect ចូល server មាន ២ របៀបគឺតាម
 - Window Authentication



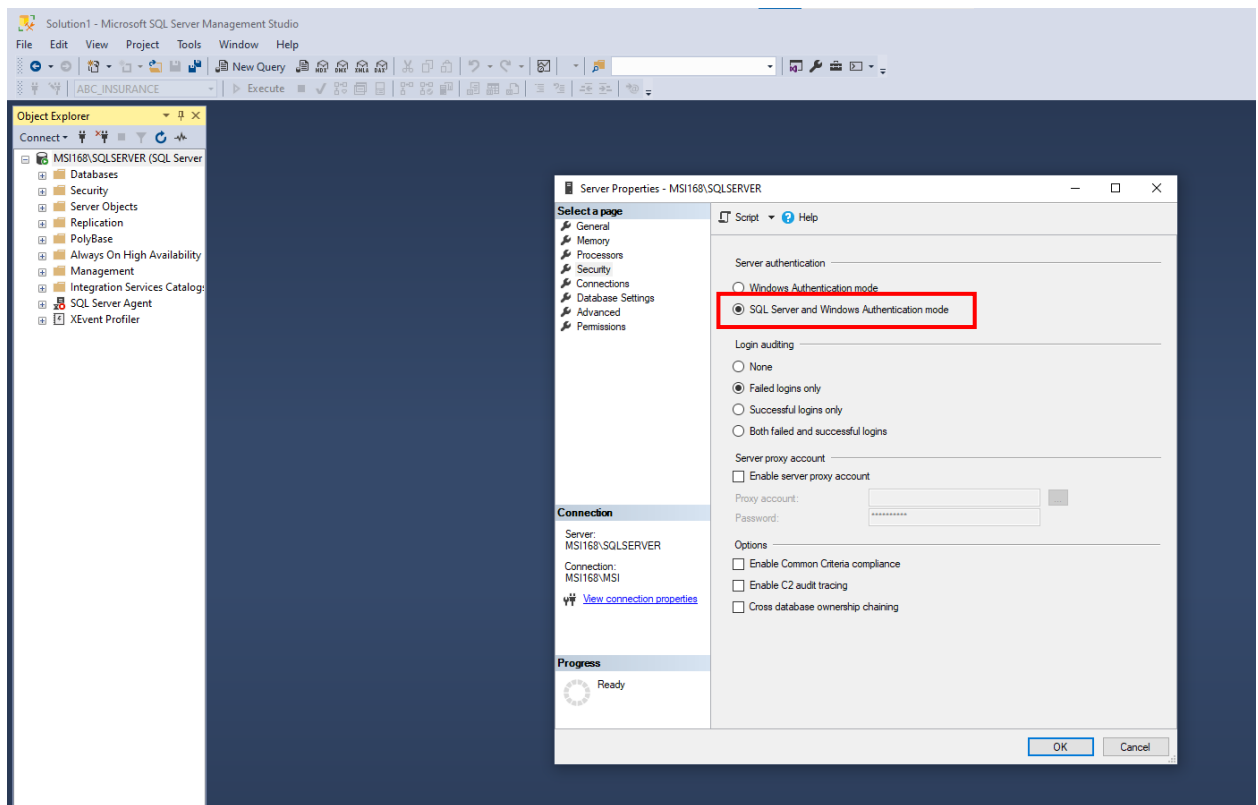
- SQL Server Authentication



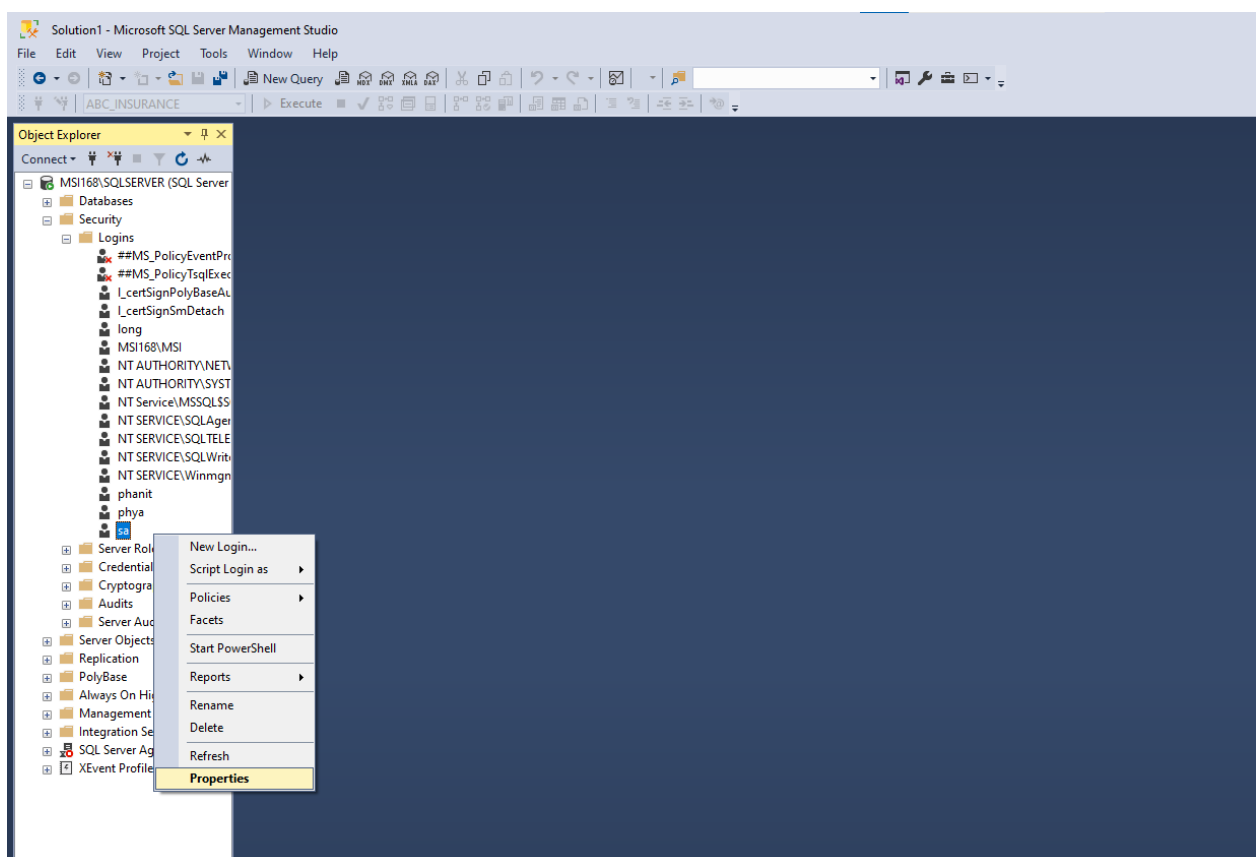
- ការ connect តាម Window Authentication មានភាពសាមញ្ញ តែវាភ្ជាប់នឹង Window ស្រាប់ តម្រូវឲ្យ Client machine និង SQL Server ប្រើក្នុង Window Domain ដូចគ្នា។
- ការ connect តាម SQL Server Authentication មានភាពស្មុគស្មាញដោយតម្រូវឲ្យយើងបង្កើត login and password ផ្តាច់ផ្សេងគ្នាសម្រាប់ SQL Server។ ដូចនេះ Client machine និង SQL Server អាចប្រើក្នុង Window Domain ផ្សេងគ្នាបាន។
- វិធីសម្រាប់បើក អោយ USER LOGIN ជាមួយ SQL Server Authentication



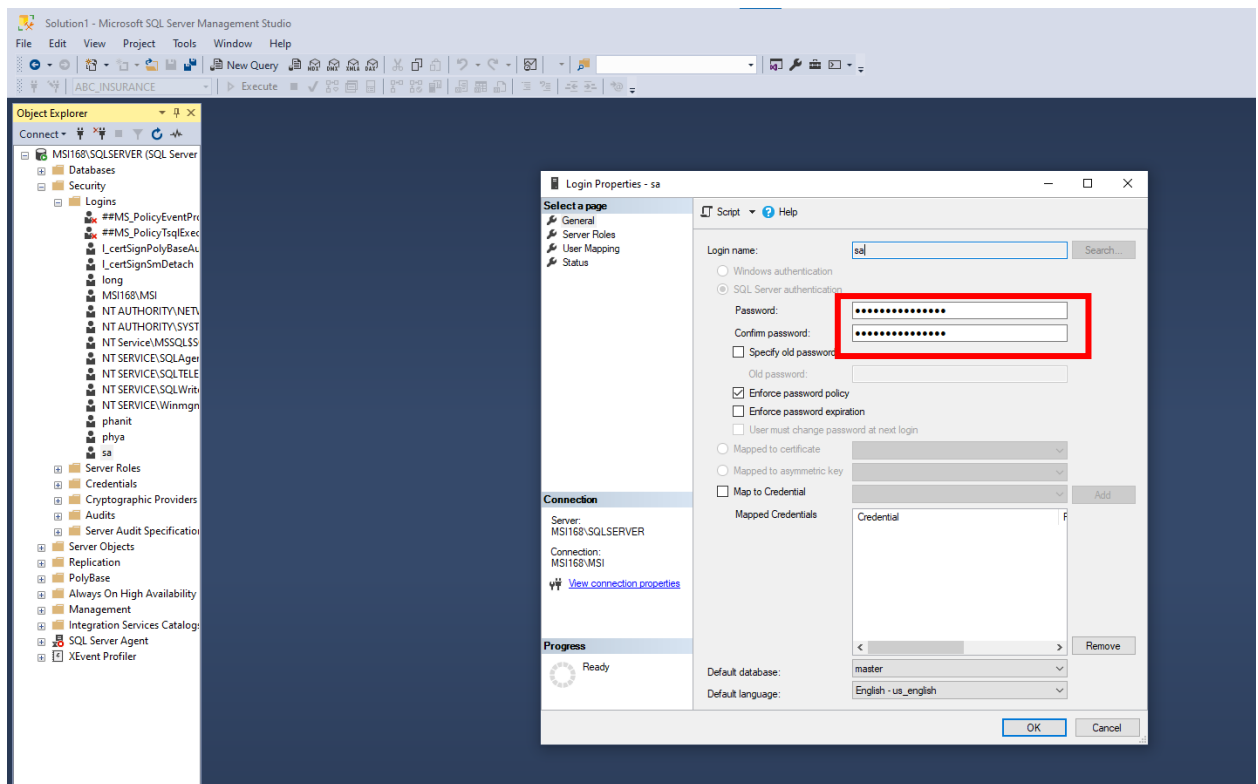
បើក SQL Server and Window Authentication mode



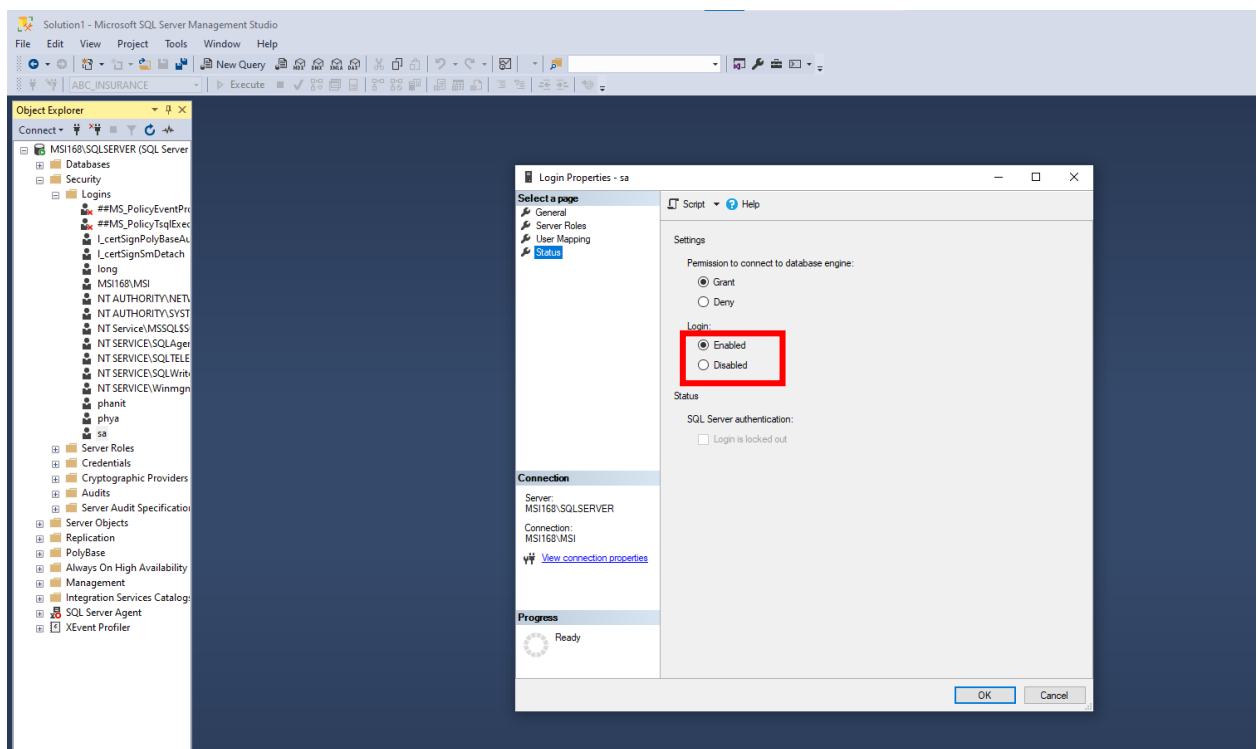
ចូលទៅកាន់ folder security > logins ហើយចុច right click ទៅលើ user sa យកពាក្យ Properties



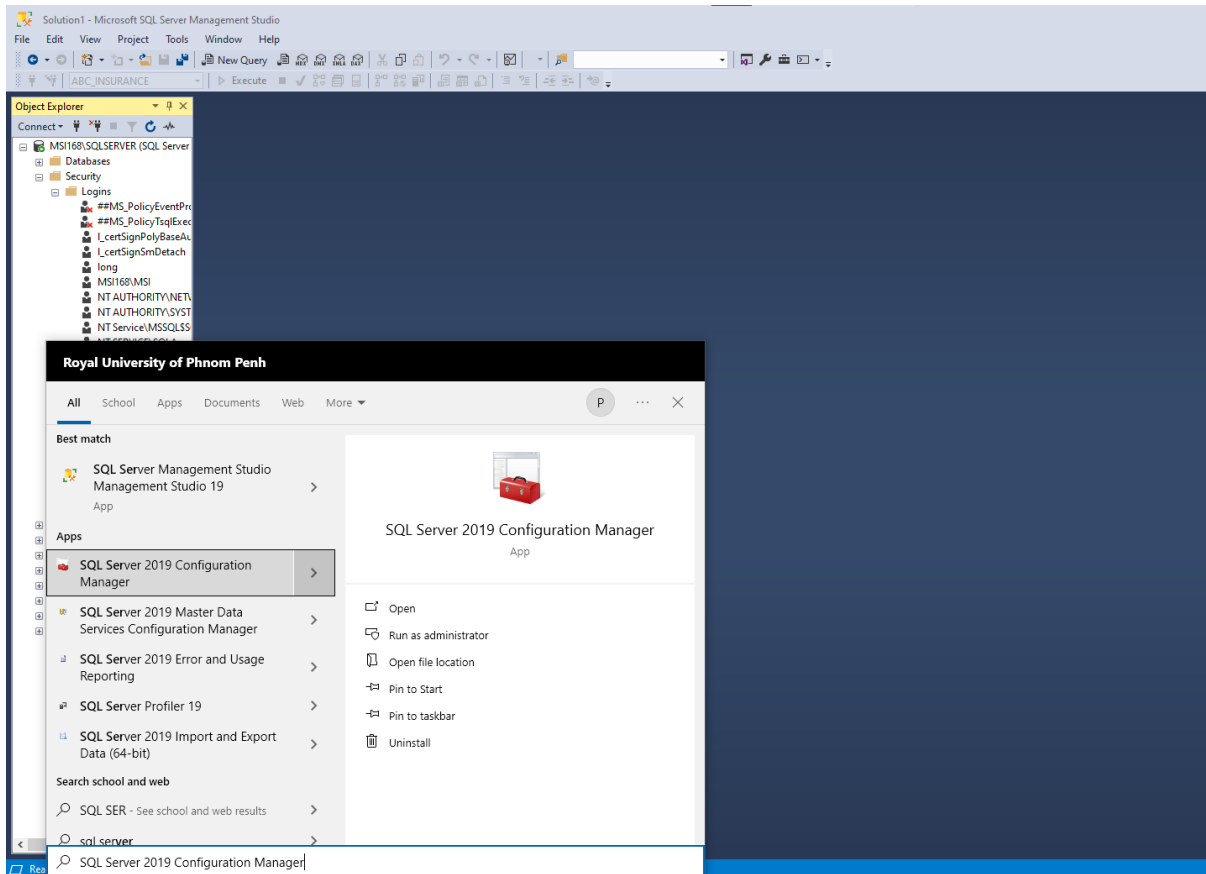
ប្តូរ Password ថ្មីសម្រាប់ login ជាមួយ user sa



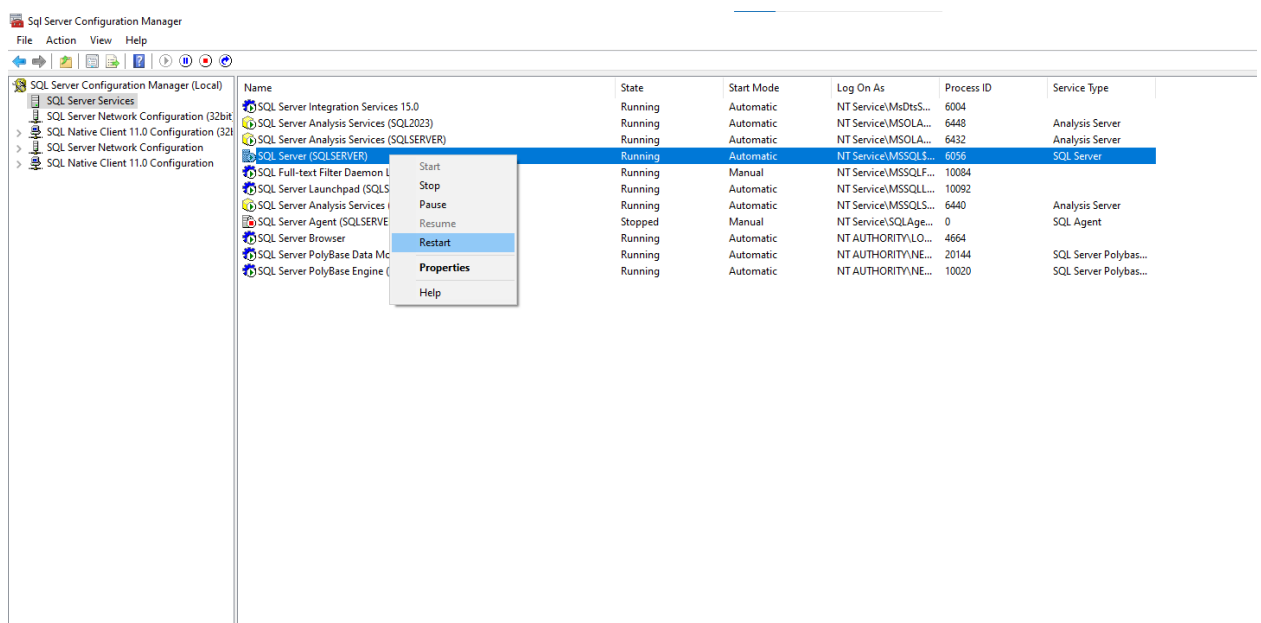
ចូលទៅកាន់ status ដើម្បីបើក Enabled login របស់ user sa



ប្រសិនបើការ Login មានបញ្ហា Error ណាមួយ សូមចូលទៅកាន់ Configuration របស់ SQL SERVER

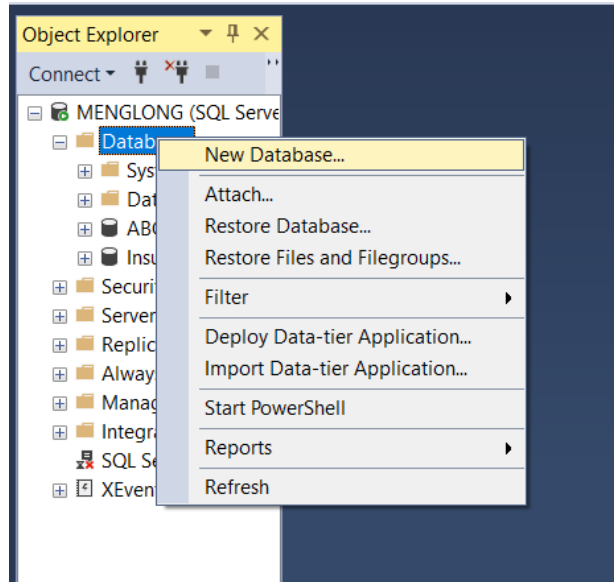


បន្ទាប់មកចុច Restart ទៅលើ server name នៃ SQL SERVER របស់យើង

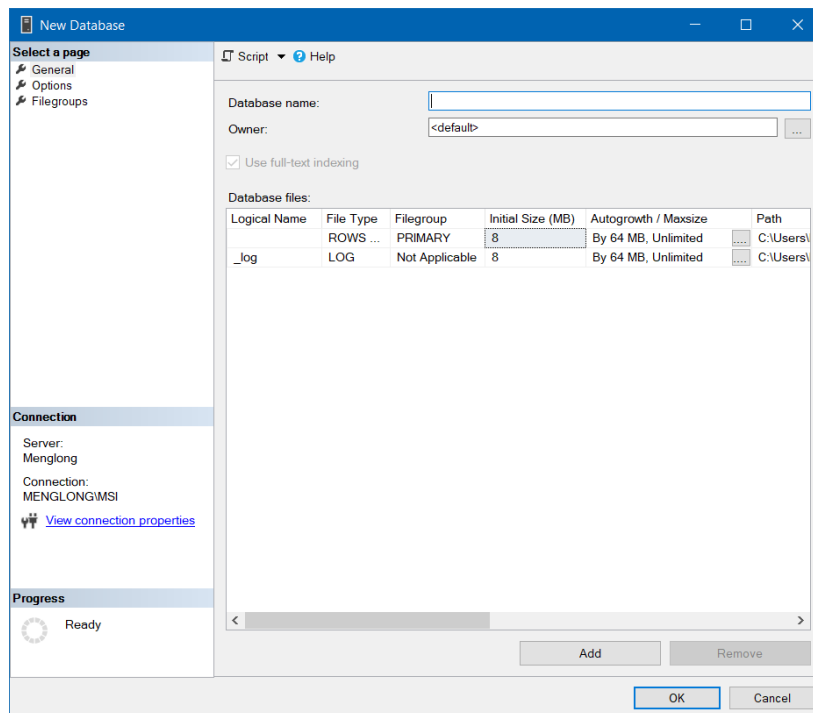


- របៀបបង្កើត DATABASE មាន ២ របៀប:

- របៀបទី១ បង្កើតដោយ tool
- ចុច Mouse ស្តាំទៅលើ folder database ហើយយើងយក new database យើងនិងឃើញ:



- បន្ទាប់ពីយក new database រួចយើងនិងឃើញផ្ទាំង



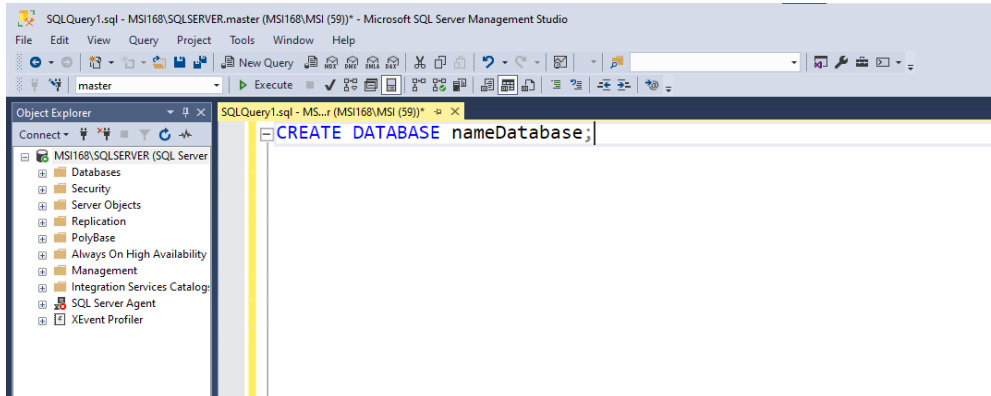
- Database name ជាកន្លែងដែលយើងដាក់ឈ្មោះ: files
- Owner: សម្រាប់ឲ្យ user មានសិទ្ធិចូលទៅកាន់database និងបាន
- Database files: មាន ២ file:
 - mdf extension
 - Logicalname: ដែលមាន file group ជា primary ហើយរាល់ទិន្នន័យទាំងអស់ត្រូវបានផ្ទុកក្នុង mdf extension និង autogrowth/maxsize, namepath។

- Idf extension

- Logicalname_log: ដែលមាន file group ជា Not applicable វា recode រាល់សកម្មភាពទាំងអស់ដែលបានប្រើប្រាស់និង autogrowth/maxsize, namepath។

- របៀបទី២ បង្កើតដោយ script

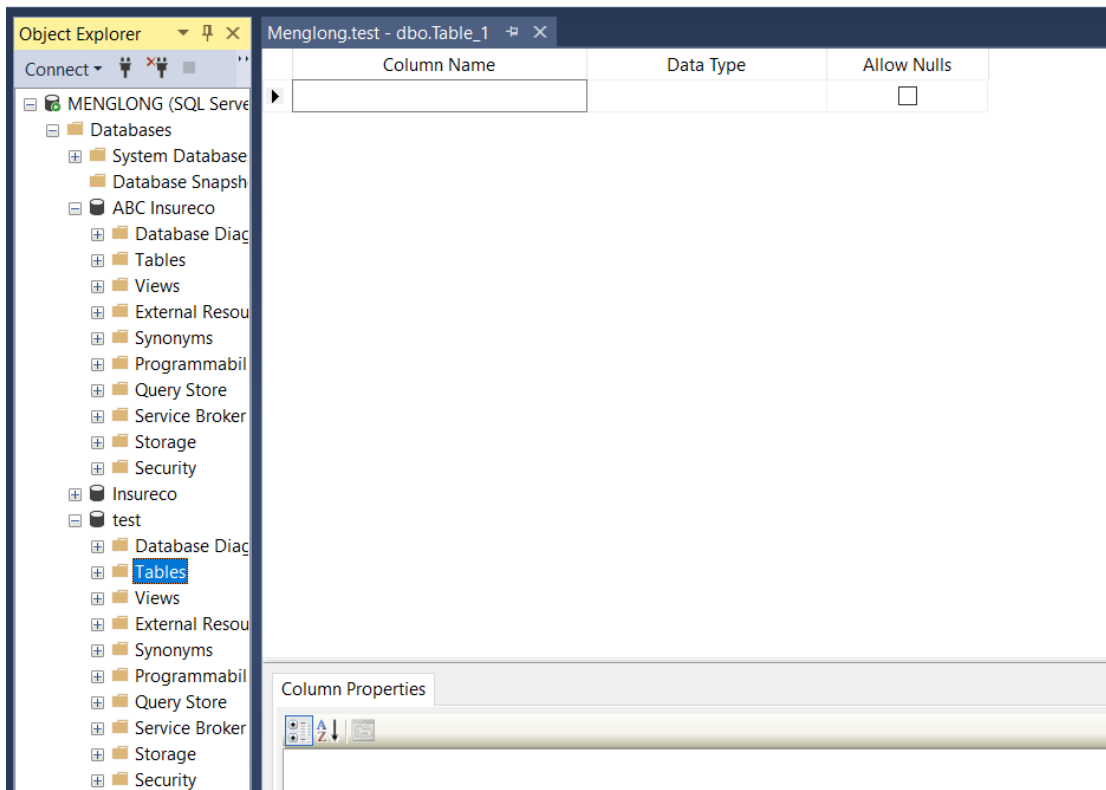
- គេត្រូវចុច New Query ឬ Ctrl + N បន្ទាប់មកសរសេរ Query
 - Create database namedatabae



- របៀបបង្កើត Table មាន២របៀប

- របៀបទី ១ ដោយ tool

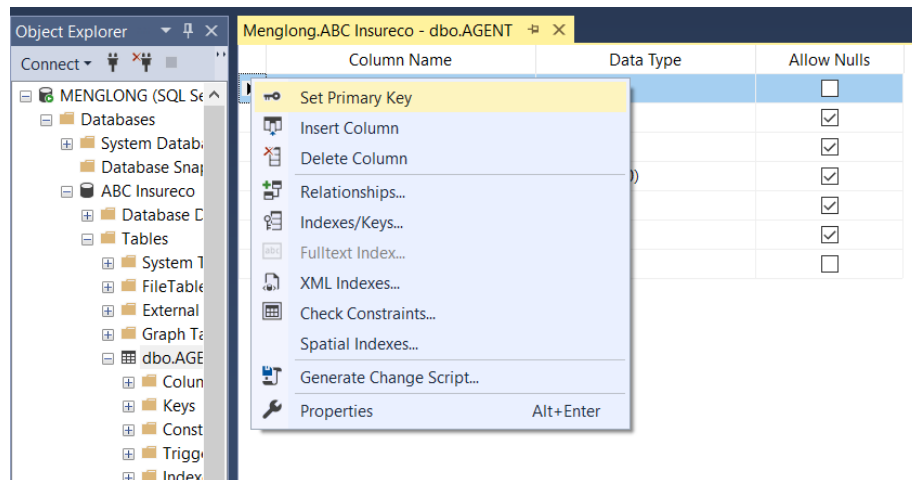
- ចុចMouse ស្តាំទៅលើfolder database ហើយយើងយក Namedatabase។ បន្ទាប់ពីដាក់ឈ្មោះ database name យើងចុចលើឈ្មោះ database ហើយយក folder table បន្ទាប់ចុចMouse ស្តាំយកពាក្យ New ហើយយកពាក្យ table យើងនិងឃើញ 3 column:



- Column name: ដាក់ឈ្មោះ attribute
- Data Type: សម្រាប់កំណត់ប្រភេទ datatype ទៅឲ្យ attribute
- Allow Null: បើសិនចុច tick មានន័យថាតម្លៃអាចទទេបាន ហើយបើដក tick មានន័យថាតម្លៃត្រូវតែមាន

- ការកំណត់ primary key ឲ្យ attribute:

- ចុច Mouse ស្តាំទៅលើសញ្ញាព្រួញនៅជិត column name ហើយយក set primary key



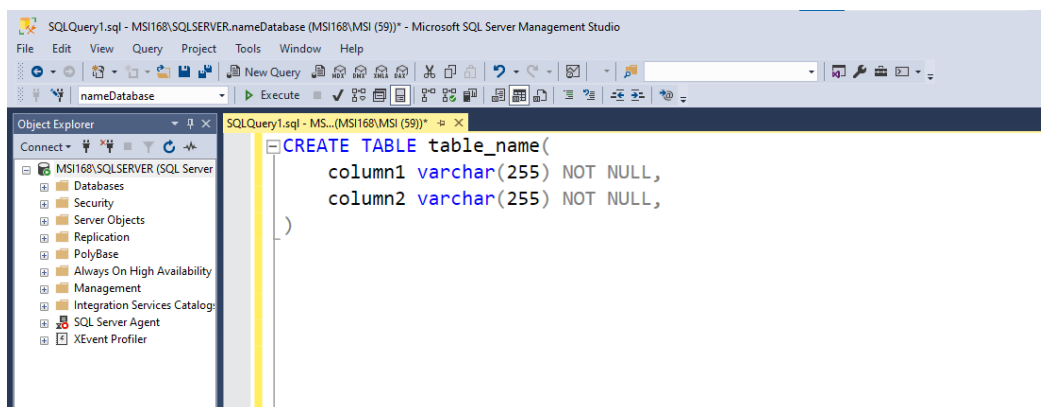
- របៀបទី២ បង្កើតដោយ script

- គេត្រូវចុច New Query ឬ Ctrl + N បន្ទាប់មកសរសេរ Query

- Syntax

- Create table **nametable**

(
 NameAttribute datatype constraint,
)



ឯកសារយោង

<https://www.educba.com/sql-ddl-commands/>

<https://aws.amazon.com/what-is/sql/>