




# DDL STATEMENTS

# Learning Goals

By the end of this lecture ✓ Categorize the main database objects  
trainees should be able to:

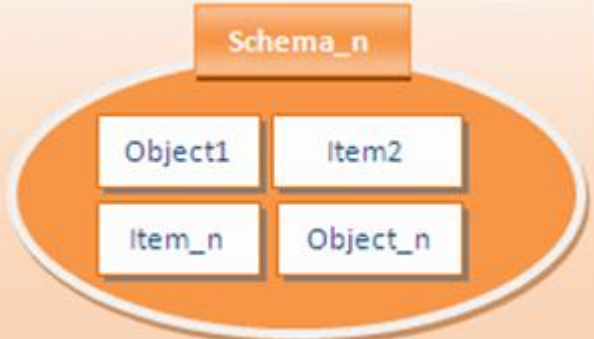


The diagram shows an 'HR Schema' containing two main categories: 'Tables' and 'Indexes'. The 'Tables' category is represented by a stack of three rectangles, with the top one showing a simple table structure with a horizontal line and a vertical line. The 'Indexes' category is represented by a hierarchical tree structure with a root node and several child nodes.

```
ScriptToCreateAn...SPG\dobso (51)
```

```
-- Specify column names and data types; a primary key is optional
CREATE TABLE [MSSQLTips].[Table1_from_SS] (
  [GUID_val] [uniqueidentifier] NOT NULL,
  [ID_val] [int] NOT NULL PRIMARY KEY,
  [Name] [varchar] (50) NULL,
```

## SQL CONSTRAINTS



The diagram shows a 'Schema\_n' container, represented by an orange oval, containing four objects: 'Object1', 'Item2', 'Item\_n', and 'Object\_n'. Each object is represented by a white rectangle with its name inside.

PRIMA  
FOREI  
UNIQUE  
NOT N  
CHECK  
DEFAULT

✓ Create a simple table

✓ Understand how constraints are created at the time of table creation

✓ Describe how schema objects work

✓ Understand and use commands as “create, alter, drop, and truncate table”

# Table of contents



## Introduction to DDL Statements



## Database Object



## Schema Object



## Table and Constraints

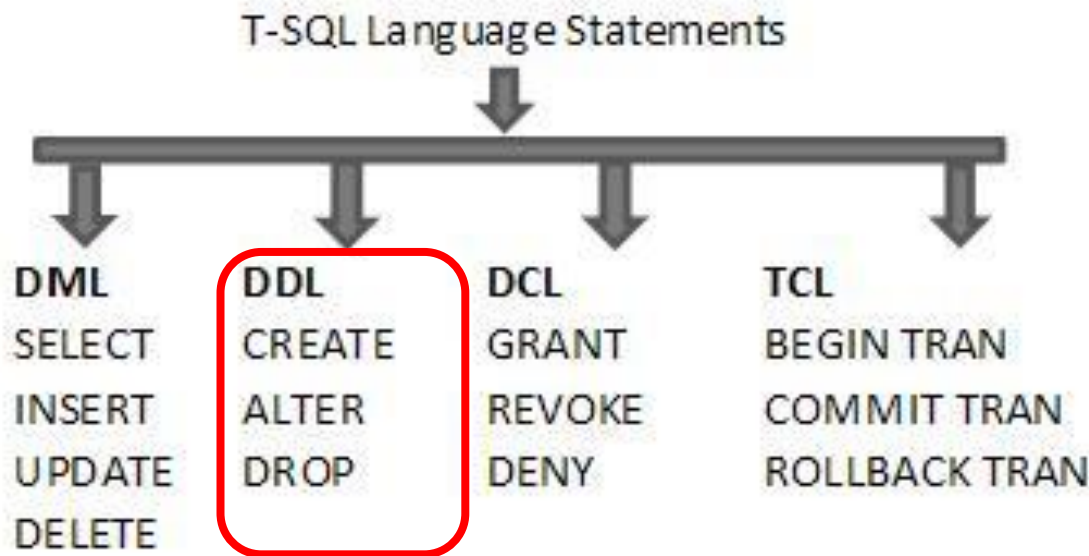


## Quiz

# Introduction to DDL Statements

# Introduction to DDL Statements

- ❁ **DDL** stands for **D**ata **D**efinition **L**anguage
- ❁ Define data structures in SQL Server as creating, altering, and dropping tables and establishing constraints...



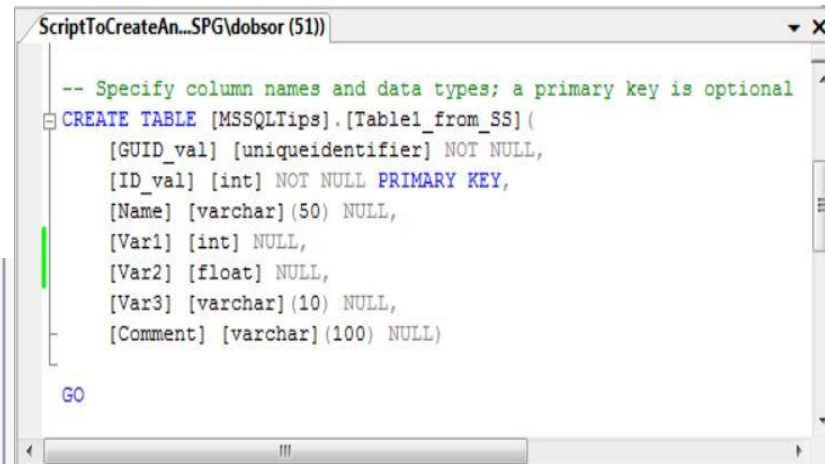
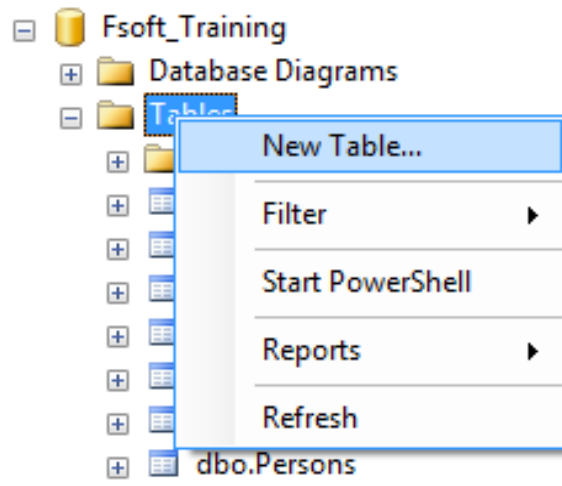
 **A SQL Server database has lot of objects like:**

- ✓ Database
- ✓ Schema
- ✓ Tables
- ✓ Views
- ✓ Stored Procedures
- ✓ Functions
- ✓ Rules
- ✓ Defaults
- ✓ Triggers

# Database Object

🌟 SQL Server supports both scripts editor and graphic tool in order to:

- ✓ Create a database
- ✓ Rename a database
- ✓ Drop a database





- ❁ **Scripts editor:**
  - ✓ Create a database
  - ✓ Rename a database
  - ✓ Drop a database

- ❁ **Graphic tool**
  - ✓ Create a database
  - ✓ Rename a database
  - ✓ Drop a database

- ❁ **Create database by using a template**

# Database Demo



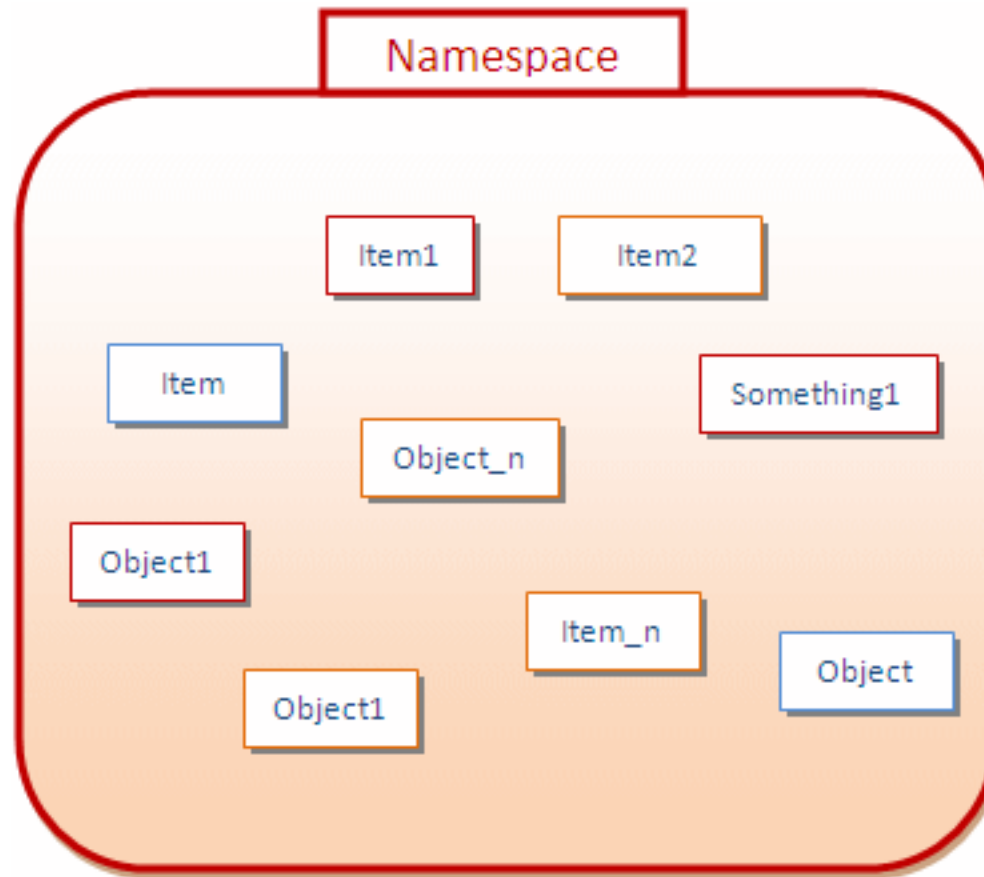
## Demo



# Schema Object

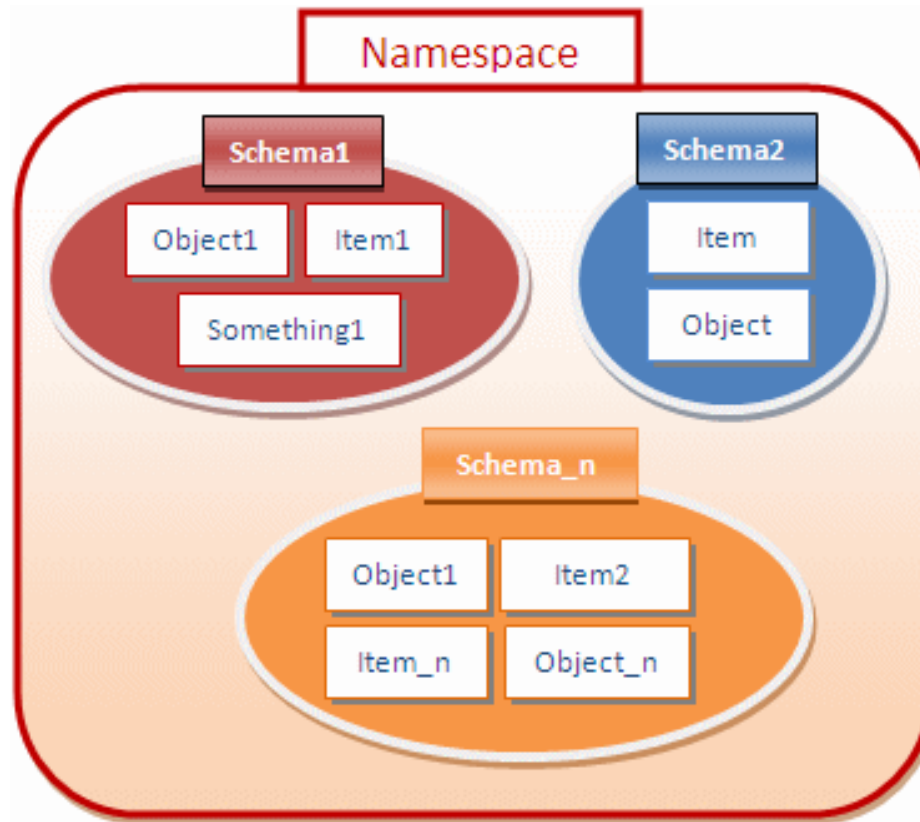
# Schema Object (1/3)

🌱 A namespace can have objects inside



# Schema Object (2/3)

- ❁ To further control and manage the objects inside of a namespace, you can put them in *sub-groups* called **schemas**.



## Schema default:

- ✓ **dbo** is default schema in every database
- ✓ Ex: SalesOrderDetail, HumanResources.Department
- ✓ **[linked-server].[DBName].[SchemaName].[Objectname]**

## Schema as:

- ✓ naming boundaries
- ✓ security boundaries

# Schema Demo



 Demo



# Table and Constraints



# Table

🌿 Table is a repository for data, with items of data grouped in one or more columns

- ✓ Data types
- ✓ Constraints
- ✓ Index

	EmployeeID	NationalIDNumber	ManagerID	Title	BirthDate	MaritalStatus	Gender	HireDate
1	1	14417807	16	Production Technician - WC60	1972-05-15 00:00:00.000	M	M	1996-07-31 00:00:00.000
2	2	253022876	6	Marketing Assistant	1977-06-03 00:00:00.000	S	M	1997-02-26 00:00:00.000
3	3	509647174	12	Engineering Manager	1964-12-13 00:00:00.000	M	M	1997-12-12 00:00:00.000
4	4	112457891	3	Senior Tool Designer	1965-01-23 00:00:00.000	S	M	1998-01-05 00:00:00.000
5	5	480168528	263	Tool Designer	1949-08-29 00:00:00.000	M	M	1998-01-11 00:00:00.000
6	6	24756624	109	Marketing Manager	1965-04-19 00:00:00.000	S	M	1998-01-20 00:00:00.000
7	7	309738752	21	Production Supervisor - WC60	1946-02-16 00:00:00.000	S	F	1998-01-26 00:00:00.000
8	8	690627818	185	Production Technician - WC10	1946-07-06 00:00:00.000	M	F	1998-02-06 00:00:00.000
9	9	695256908	3	Design Engineer	1942-10-29 00:00:00.000	M	F	1998-02-06 00:00:00.000

## Create table

## Alter table

- ✓ Add new column
- ✓ Change data type of existing column
- ✓ Delete a column
- ✓ Add or remove constraints

## Drop table

- ✓ Remove table structure and its data.



# Table demo

 Demo

# Table Constraints (1/4)

- 🌟 **Table Constraints:** Are used to limit the type of data that can go into a table.
- 🌟 We will focus on the following constraints:
  - ✓ NOT NULL
  - ✓ CHECK
  - ✓ UNIQUE
  - ✓ PRIMARY KEY
  - ✓ DEFAULT
  - ✓ FOREIGN KEY

# Table Constraints (2/4)

-  **NOT NULL:** Specifies that the column does not accept NULL values.
-  **CHECK:** Enforce domain integrity by limiting the values that can be put in a column.
  - ✓ **Syntax:**  
[CONSTRAINT *constraint\_name*]  
**CHECK** (*condition*)

# Table Constraint (3/4)

 **UNIQUE**: Enforce the uniqueness of the values in a set of columns

✓ **Syntax:**

CONSTRAINT unique\_name **UNIQUE** (col\_names)

 **PRIMARY KEY**: Specify primary key of table.

✓ **Syntax:**

[CONSTRAINT *PK\_Name*]

**PRIMARY KEY** [col\_names]


 **FOREIGN KEY:** Used to define relationships between tables in the database.

✓ **Syntax:**

[CONSTRAINT *FK\_Name*]

**FOREIGN KEY** [(*col\_names*)]

REFERENCES *reference\_table*(*col\_names*)

 **DEFAULT:** Defaults specify what values are used in a column if you do not specify a value for the column when you insert a row.

# SQL Constraints Scope

 SQL constraints can be applied at:

## ✓ **Table level**

- Are declared independently from the column definition
- declare table-level constraints at the end of the CREATE TABLE statement

## ✓ **Column level:**

- Are declared when define columns for the table.
- It is applied particularly to the column where it attached to

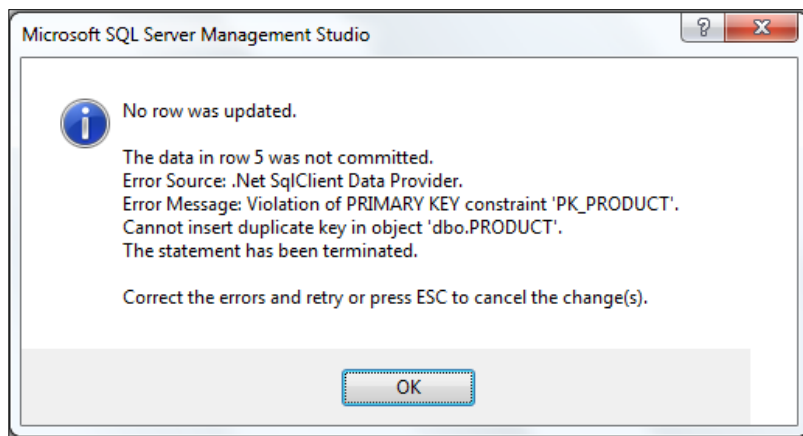


# Identity (1/1)

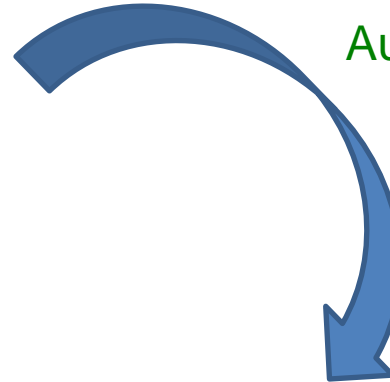
## Primary key

	PRODUCT_ID	PWIDTH	PLENGTH	PRICE
	1	40	50	2000.0000
	2	45	55	2000.0000
	3	40	60	3000.0000
	4	50	55	2500.0000
✎	4	❗ 45	❗ 50	❗ 2100 ❗
*	NULL	NULL	NULL	NULL

Error



Auto increment



	PRODUCT_ID	PWIDTH	PLENGTH	PRICE
	1	40	50	2000.0000
	2	45	55	2000.0000
	3	40	60	3000.0000
	4	50	55	2500.0000
	5	45	50	2100.0000
►*	NULL	NULL	NULL	NULL

# Identity (1/2)

- ❁ Identity has:
  - ✓ A seed
  - ✓ An increment
- ❁ Seed is the initial value
- ❁ Increment is the value by which we need to skip to fetch the next value
- ❁ **For example:**
  - ✓ Identity(1,2) will generate sequence numbers 1,3,5,7...

1
2
3
4
5
...

Identity(1,1)

1
4
7
10
13
...

Identity(1,3)

1
3
5
7
9
...

Identity(1,2)

# Truncate statement

- ❁ Removes all rows in a table.
- ❁ Table structure and its columns, constraints, indexes, ...remain.
- ❁ Resets the identity value.
- ❁ Releases the memory used.

# Table & Constraints Demo

 Demo

# Quiz!

*Now let's check how you understand the lecture!*

*There are 13 questions below.  
Click **NEXT** button to start!*

Now let's check how you  
understand the lecture!

# Quiz!

*There are 13 questions below.  
Click **NEXT** button to start!*

# Summary



## Introduction to DDL Statements

- SQL Server Database Objects



## Database Object

- Create, Rename, Drop a database: Graphic, Scripts, Template



## Schema Object

- What is schema in database? Schema default?



## Table and Constraints

- Create, Alter, Drop Table. NOT NULL, CHECK, UNIQUE, PRIMARY KEY, DEFAULT, FOREIGN KEY



## Quiz



# THANK YOU

You have completed "**Lecture \_04**" course.

Click EXIT button to exit course and discover the next Lecture "**Lecture\_05**".

EXIT