**Schema in MSSQL**

**Schema** is a container or namespace that helps to organize database objects such as tables, views, procedures, and functions.

* **By Default dbo Schema:** is Generated by MSSQL when we Create Database.
* You Can Create Multiple Schemas in Database
* Droping Schema will Drop All Tables which References to Schema
* **Location : Database / selectDatabase / secuirity / schema**

1. ***Create:***

---- Schema ---

create schema SchemaNameYouWantToGive;

---- Table ---

create table customerSch.customer(

id int primary key identity,

name varchar(25)

)

select \* from customerSch.customer

--\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Verify Schema \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

sELECT \* FROM sys.schemas

SELECT \* FROM sys.schemas WHERE name = 'customerSch';

1. ***Rename (Transfer):***

***Rename not Work -***🡪 ***Transfer 1st schema Object to 2nd schema***

---\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Rename Schema \_\_\_\_\_\_\_\_\_

-- Rename not Works in Schem

--EXEC sp\_rename 'customerSch', 'newCustomerSch';

---\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 1. Transfer Schema to Schema \_\_\_\_\_\_\_\_\_

create schema newSchema

--\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2. Transfer \_\_\_\_\_\_\_\_\_\_\_\_\_

alter schema newSchema

transfer customerSch.customer

--\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 3. Check New Schema Object (tables) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

select \* from newSchema.customer

--\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ testing \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

-- Old Schema

select \* from customerSch.customer

-- New Schema

select \* from newSchema.customer

--\_\_\_\_\_\_\_\_\_\_\_\_\_\_ See All objects of Schema \_\_\_\_\_\_\_\_\_\_\_\_\_

SELECT \*

FROM sys.objects

WHERE SCHEMA\_NAME(schema\_id) ='newSchema';

1. ***Drop:***

* ***If Schema Contans Object (table) Drop will not work.***
* ***Transfer all objects to newSchema***
* ***Then Drop Schema will work***

--\_\_\_ 1 Step \_\_\_\_\_\_\_\_\_\_\_\_\_

create schema myNewSchema

--\_\_\_ 2 Step \_\_\_\_\_\_\_\_\_\_\_\_\_

alter schema myNewSchema transfer newSchema.customer;

--\_\_\_ 3. Drop \_\_\_\_\_\_\_\_\_\_

drop schema newSchema

--\_\_\_\_\_ See All objects of Schema \_\_\_\_

SELECT \*

FROM sys.objects

WHERE SCHEMA\_NAME(schema\_id) = 'myNewSchema';

1. ***Transfer All Object ():***

***Let we have 20 Table in Schema. Then how we will transfer to 2nd schema ??***

--\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Transfer All Schema Object to NewSchema \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

-- We Can not Direclty Change Alll Thing

--- But We Can Get All The Structure of Schema ---> Then Copy past and -------> Exicute This

SELECT 'ALTER SCHEMA xyz TRANSFER [' + SysSchemas.Name + '].[' + DbObjects.Name + '];'

FROM sys.Objects DbObjects

INNER JOIN sys.Schemas SysSchemas ON DbObjects.schema\_id = SysSchemas.schema\_id

WHERE SysSchemas.Name = 'myNewSchema'

AND (DbObjects.Type IN ('U', 'P', 'V'))

-- Copy and Past

--Where type 'U' denotes user tables, 'V' denotes views and 'P' denotes stored procedures.

ALTER SCHEMA xyz TRANSFER [myNewSchema].[customer];

-- schema ---

SELECT \*

FROM sys.objects

WHERE SCHEMA\_NAME(schema\_id) = 'xyz';

***\_\_\_\_\_\_\_\_\_ permissions of User \_\_\_\_\_\_\_\_\_***

***End me parhan ga SQL ka***

--\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Permission \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

SELECT \*

FROM sys.database\_permissions

WHERE class = 3 -- 3 represents schema-level permissions

AND major\_id = SCHEMA\_ID('customerSch');

SELECT HAS\_PERMS\_BY\_NAME(OBJECT\_NAME(major\_id), 'SCHEMA', 'SELECT') AS has\_select\_permission,

HAS\_PERMS\_BY\_NAME(OBJECT\_NAME(major\_id), 'SCHEMA', 'UPDATE') AS has\_update\_permission,

HAS\_PERMS\_BY\_NAME(OBJECT\_NAME(major\_id), 'SCHEMA', 'DELETE') AS has\_delete\_permission,

HAS\_PERMS\_BY\_NAME(OBJECT\_NAME(major\_id), 'SCHEMA', 'INSERT') AS has\_insert\_permission,

HAS\_PERMS\_BY\_NAME(OBJECT\_NAME(major\_id), 'SCHEMA', 'REFERENCES') AS has\_references\_permission,

HAS\_PERMS\_BY\_NAME(OBJECT\_NAME(major\_id), 'SCHEMA', 'EXECUTE') AS has\_execute\_permission

FROM sys.database\_permissions

WHERE class = 3 AND major\_id = SCHEMA\_ID('customerSch');

SELECT \*

FROM fn\_my\_permissions(NULL, 'SCHEMA');

---\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Grand Permission to user\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

GRANT SELECT ON SCHEMA::customerSch TO "SAQIB\m4357";

-- This Grand only works for those users have Restraction ....