**EXPERIMENT 4**

**19L-1316 HAFIZ AHMADs**

# Introduction to SQL and DDL

**Objective**

* Familiarize students with the SQL DDL commands
* Enable students to use the basic DML commands (in order to check the working of above DDL commands)

**SQL (Structured Query Language)**

SQL commands can be broadly categorized under the following four types:

* **DDL**Data Definition Language (DDL) statements are used to define the database structure or schema. Some examples:

CREATE - to create objects in the database

ALTER - alters the structure of the database

DROP - delete objects from the database

TRUNCATE - remove all rows from a table

## Exercise1: Practice DDL commands using SQL server management studio

**Step1: Login to your database**

* Open SQL server management studio via Start->Program Files ->SQL server 2014 -> SQL server management studio
  + Enter your server name as “172.16.1.158” username as “sa” password as “12345678” depending on what you used in the last lab.
* Select your own database that you created in the last lab3

**Step2: Create a schema**

* Click on “New Query”
* In the window (Query Manager) that comes up run the following command

Use [your\_own\_DB\_created\_in\_lab3]

create schema lab4

**Step3: Create table command**

* Run the following statement in your own database

CREATE TABLE lab4.School(

schID [int] NOT NULL,

schName [varchar](50) NULL,

schdeanID [int] NULL,

)

CREATE TABLE lab4.Staff(

staffID [int] NOT NULL,

staffName [varchar](50) NULL,

staffRole [varchar](50)

)

* Refresh your database and verify the results from designer on left

**Step3: Drop table command**

* Run the following statement in your own database

drop table lab4.school

* Refresh your database and verify the results from designer on left

**Step4: Primary key on a new table**

* Recreate the table dropped in step3, this time with the primary key constraint in place by running the following command

CREATE TABLE lab4.Staff(

staffID [int] NOT NULL,

staffName [varchar](50) NULL,

staffRole [varchar](50),

constraint staffPK primary key (staffId)

)

* Refresh your database and verify the results from designer on the left by expanding Keys under Staff table

**Step5: Primary key on an existing table via ALTER table command**

* Using your own database run the following command

Alter TABLE lab4.School add constraint schPK primary key(schID)

* Refresh your database and verify the results from designer on the left by expanding Keys under School table

**Step6: Set Foreign Key on new table**

* Drop table school (you know how to do that now)
* drop table lab4.school
* drop table lab4.staff
* Recreate the table school by running the following statement

CREATE TABLE lab4.School(

schID [int] NOT NULL primary key,

schName [varchar](50) NULL,

schdeanID [int] NULL,

constraintdeanFK foreign key (schdeanID) references lab4.Staff(staffId)

)

**Step7: Drop Foreign Key constraint via Alter table command**

* Using your own database run the following command

Alter table lab4.school drop constraint deanFK

**Step8: Add Foreign Key constraint to an existing table via Alter table command**

* Using your own database run the following command

Alter table lab4.school add constraint deanFKforeign key (schdeanID) references lab4.Staff(staffId)

**Step9: Add Foreign key constraint to set behavior on delete/update**

* Drop the deanFK foreign key constraint added in step8 above

Alter table lab4.school drop constraint deanFK

* Recreate the foreign key constraint on school table to set behavior on update and delete by running the following command

Altertable lab4.school addconstraint deanFK foreign

key(schdeanID)references staff(staffID)

on delete set null on update cascade

**Step10: Check the behavior of the foreign key constraint set above by inserting a few rows in both the tables and then deleting and updating rows in Staff table**

* **Run the following insert commands to populate the two tables**

Insert into lab4.staff values (101, ‘Ahmed’, ‘Principal’)

Insert into lab4.staff values (102, ‘Ali’, ‘Head Teacher’)

Insert into lab4.staff values (103, ‘Salman’, ‘Teacher’)

Insert into Lab4.staff(staffName, staffId) values (‘Fatima’, 104)

Insert into lab4.school values (1, ‘DPS’, 101)

Insert into lab4.school values (2, ‘LGS’, 102)

Insert into lab4.school values (3, ‘BeaconHouse’, 103)

* **Select Data from the two tables by running the following commands**

Select \* from lab4.staff

Select \* from lab4.school

* **Update Data in Staff table by running the following command**

Update lab4.staff set staffId=106 where staffId=103

Check the behavior on update by selecting data from the tables

Select \* from lab4.staff

Select \* from lab4.school

* **Delete Data in Staff table by running the following command**

Delete from lab4.staff where staffId=102

Check the behavior on delete by selecting data from the tables

Select \* from lab4.staff

Select \* from lab4.school

**Step11:** Add column to an existing table

Alter table lab4.school add schAddress varchar(100)

**Step12: Remove column from an existing table**

Alter table lab4.school drop column schAddress

**Step13: TRUNCATE table**

TRUNCATE removes all rows from a table while structures still persists.

1. Insert 5 new rows in school table
2. Now execute truncate command as shown

truncate table lab4.school

1. select data from school table

**Exercise2: Please give the queries in the space provided after running it on SQL server**

director

|  |  |  |
| --- | --- | --- |
| did | name | bestRev |

movie

|  |  |  |
| --- | --- | --- |
| mid | mname | Directed\_by |

Given the relational model above:

1. Give the table creation statements with all constraints in place

CREATE TABLE director(

did [int] NOT NULL primary key,

name [varchar](50) NULL,

bestrev [varchar](50) NULL,

)

CREATE TABLE movie(

mid [int] NOT NULL,

mname [varchar](50) NULL,

directedby int

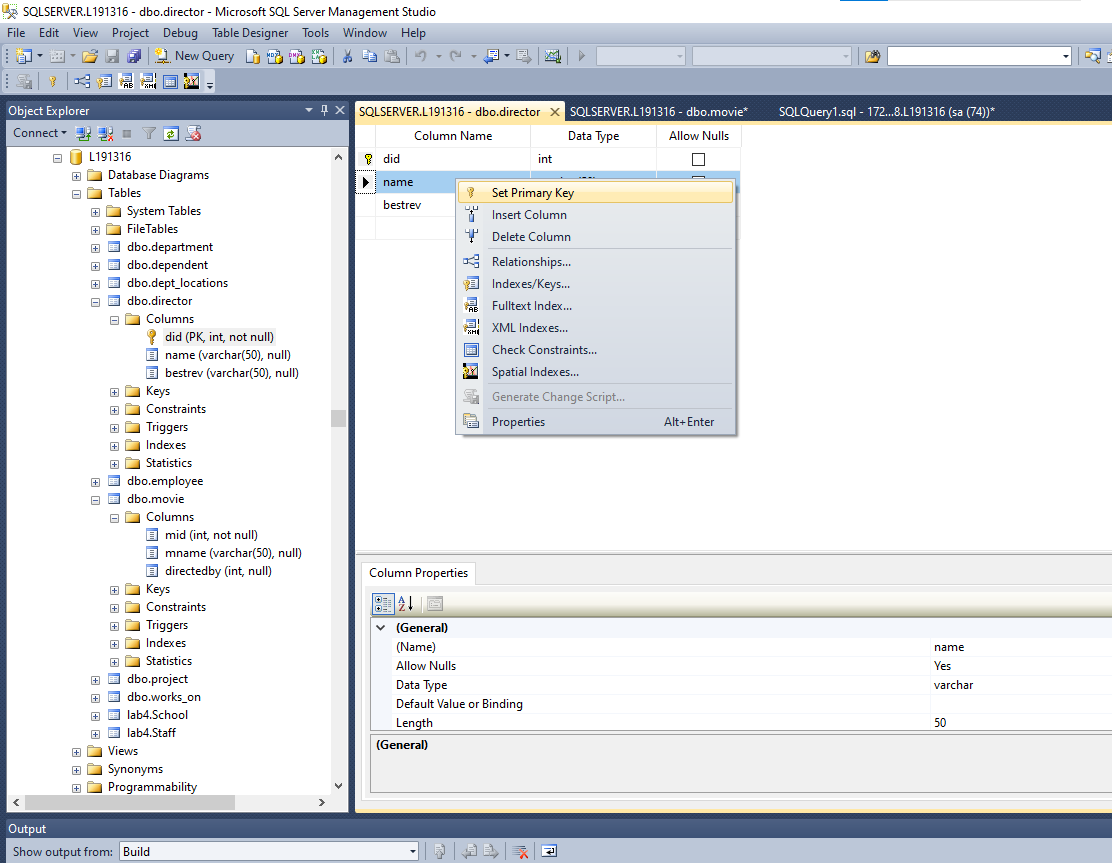
constraint fk foreign key references director(did)

)

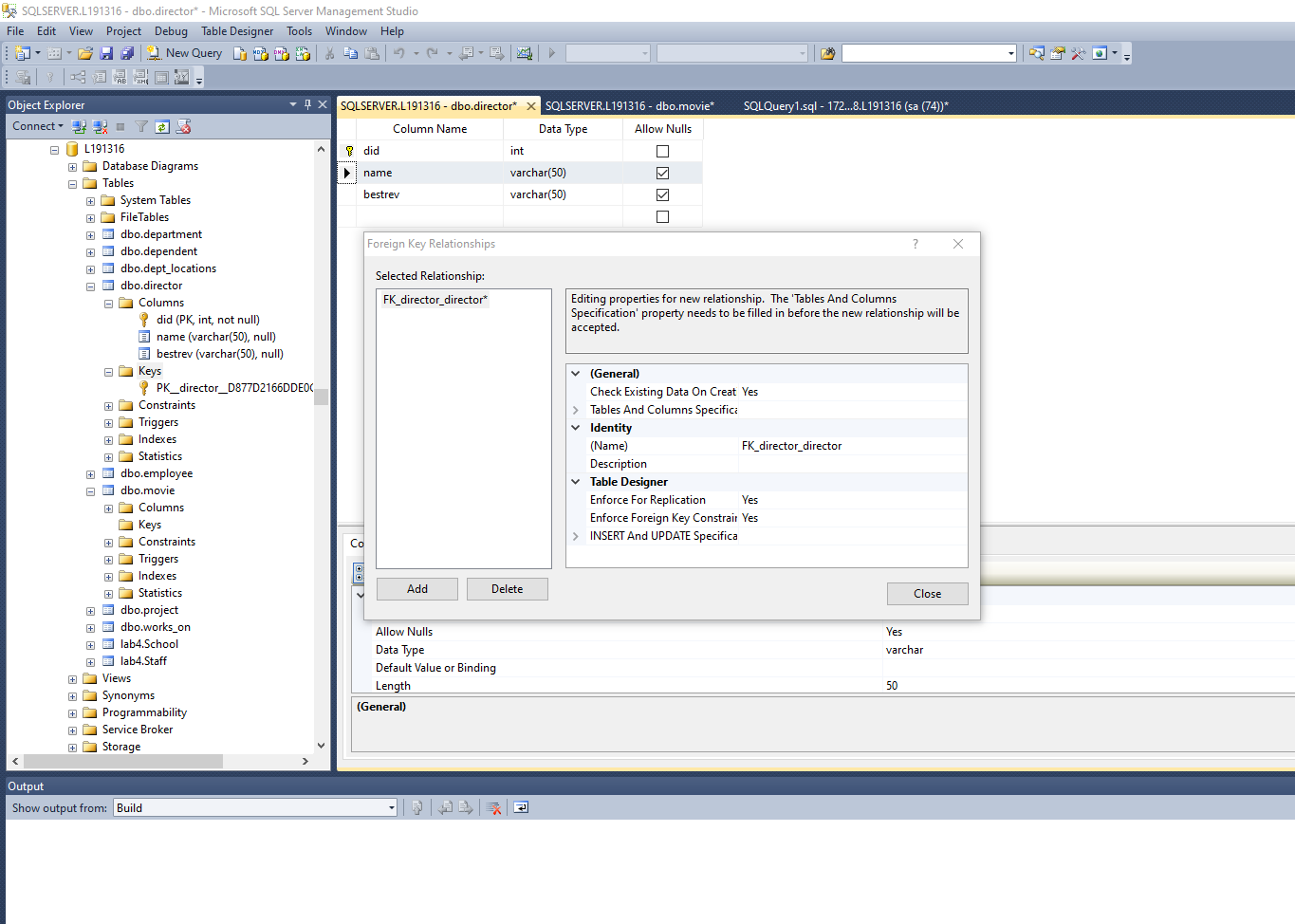
1. Give the DDL statements to drop the primary key and foreign key constraints in real time

alter table movie drop constraint fk

1. Explore how you can set primary key constraints on the director table using GUI provided by SQL Server, include screen shots



1. Explore how you can set foreign key constraints between tables above by using the GUI provided by SQL Server, include screen shots



1. Explore how Identity column works in SQL server, give the SQL DDL statement to add a column called Srno in movie table which would be an identity column

Identity column has unique values in and which is used to indentify tuples of values

1. Give the command needed to make the srno column added above a unique key for the table.

alter table director add sno int unique

1. List the differences between delete and truncate commands

Delete and Truncate look likes same commands as used for deletion both are different roles delete is used to delete selective rows whereas truncate is used to delete all the data that is present in the table in one time if there are triggers in place of using the truncate command will bypass them, whereas with the delete command the event will be triggered.