Department of Computer Science
University of Cyprus

EPL342 –Databases

Lab 6
SQL-DDL Advanced in SQL Server 2008

Panayiotis Andreou

http://www.cs.ucy.ac.cy/courses/EPL342

Before We Begin

- Start the SQL Server Management Studio
  - Start →
    All Programs →
    Microsoft SQL Server →
    SQL Server Management Studio

  Server: APOLLO

  Authentication: SQL Server Authentication

  Username: <your username>

  Password: <your password>

COMPANY Database

- During Lab 6 we have created all the COMPANY DB objects (tables, primary keys, foreign keys) using commands of the SQL-DDL syntax.
- In Lab 7 we will learn about constraints in detail and create some constraints on the COMPANY DB

SQL Server 2008 Constraints

There are 5 types of constraints in SQL Server 2008

- **PRIMARY KEY Constraint** (covered partially in lab 6)
- **FOREIGN KEY Constraint** (covered partially in lab 6)
- **DEFAULT constraint**
- **UNIQUE Constraint**
- **CHECK constraint**

Depending on the constraint it can be created or added at different levels (column level, table level or statements)

PRIMARY KEY Constraint

When a PRIMARY KEY constraint is added to a column then the column must abide by the following rules:

- The column cannot allow for null values.
- There can be no duplicate values.

As soon as the PRIMARY KEY constraint is created the Database Engine automatically creates a UNIQUE INDEX to enforce the uniqueness requirement.

A PRIMARY KEY constraint cannot be deleted if the following exist:

- If it is referenced by a FOREIGN KEY constraint
- The table has a PRIMARY XML index applied on it.

PRIMARY KEY Constraint

- Column level creation

```
CREATE TABLE dbo.test (
    test_id int CONSTRAINT pk_test PRIMARY KEY,
    …)
```

- Table level creation

```
CREATE TABLE dbo.test (
    test_id int,
    …
    CONSTRAINT pk_test PRIMARY KEY (test_id) )
```

- With ALTER statement

```
ALTER TABLE dbo.test  ADD CONSTRAINT pk_test
PRIMARY KEY(test_id)
```

FOREIGN KEY Constraint

When a FOREIGN KEY constraint is added to a column then:

- The constraint enforces referential integrity by guaranteeing that changes cannot be made to data in the primary key table if those changes invalidate the link to data in the foreign key table.

FOREIGN KEY Constraint

When a FOREIGN KEY constraint is added to a column then:

- If WITH CHECK (default) is specified SQL Server examines the existing data in the columns to make sure that all values, except NULL, exist in the columns of the referenced PRIMARY KEY or UNIQUE constraint
- If WITH NOCHECK is specified then no examination is performed

FOREIGN KEY Constraint

- Column level creation

```
CREATE TABLE dbo.test (…,
    pid int CONSTRAINT fk_project FOREIGN KEY
    REFERENCES PROJECT (project_id),
    …)
```

- Table level creation

```
CREATE TABLE dbo.test (…,
    pid int,
    …
    CONSTRAINT fk_project FOREIGN KEY (pid)
    REFERENCES PROJECT (project_id)
    )
```

- With ALTER statement

```
ALTER TABLE dbo.test ADD CONSTRAINT fk_project
FOREIGN KEY(pid)REFERENCES PROJECT (project_id)
```

DEFAULT Constraint

The DEFAULT constraint defines the default value for a column. It is automatically applied when a value is not provided for a specific column

DEFAULT Constraint

- Column Level

```
CREATE TABLE dbo.test (…,
      testname nvarchar(50) CONSTRAINT df_testname
    DEFAULT 'Unknown',
    …)
```

- With ALTER statement

```
ALTER TABLE dbo.test ADD CONSTRAINT df_testname DEFAULT
'Unknown' FOR testname
```

UNIQUE Constraint

- You can use UNIQUE constraints to make sure that no duplicate values are entered in a column.
- Multiple UNIQUE constraints can be defined on a table
- Unlike PRIMARY KEY constraints, UNIQUE constraints allow for the value NULL. However, as with any value participating in a UNIQUE constraint, only one null value is allowed per column.

UNIQUE Constraint

- Column level creation

```
CREATE TABLE dbo.test (
    test_id2 int CONSTRAINT uk_test UNIQUE,
    …)
```

- Table level creation

```
CREATE TABLE dbo.test (
    test_id2 int,
    …
    CONSTRAINT uk_test UNIQUE (test_id2) )
```

- With ALTER statement

```
ALTER TABLE dbo.test  ADD CONSTRAINT uk_test
UNIQUE (test_id2)
```

CHECK Constraint

CHECK constraints enforce domain integrity by limiting the values that are accepted by a column. They are similar to FOREIGN KEY constraints in that they control the values that are put in a column.

You can apply multiple CHECK constraints to a single column. You can also apply a single CHECK constraint to multiple columns by creating it at the table level.

CHECK Constraint

- Column level creation

```
CREATE TABLE dbo.test (
 amount    int CONSTRAINT ch_amount CHECK
                  (amount>=5 AND amount<=10)
    …)
```

- Table level creation

```
CREATE TABLE dbo.test (
    test_id2 int,
    …
    CONSTRAINT ch_amount CHECK
                         (amount>=5 AND amount<=10)
    )
```

- With ALTER statement

```
ALTER TABLE dbo.test  ADD CONSTRAINT ch_amount CHECK
(amount>=5 AND amount<=10)
```

Complex CHECK Constraints

Requirement: Accept only the following values for the equipment column

```
ALTER TABLE office
ADD CONSTRAINT ch_check_equipment
CHECK (
   equipment IN
   ('Computer','Printer','Monitor')
)
```

Complex CHECK Constraints

Requirement: In table <payment>, adjust the <bonus> according to the <sales> of an employee:

- Sales<=1500 then bonus<100
- Sales>1500 and <2500 then bonus>=100 and <200
- Sales>=2500 then bonus>=300

```
ALTER TABLE payments ADD CONSTRAINT ch_fair_bonus
CHECK ( (sales<=1500 AND bonus<100)
        OR
        ((sales>1500 AND sales<2500) AND
                    (bonus>=100 and bonus<200))
        OR
        (sales>=2500 AND bonus>=300)
)
```

Enabling/Disabling a Constraint

- Enable the constraint

```
ALTER TABLE dbo.test
CHECK CONSTRAINT fk_project
```

- Disable the constraint

```
ALTER TABLE dbo.test
NOCHECK CONSTRAINT fk_project
```

Constraints of COMPANY DB

Create constraints on the COMPANY DB based on the following requirements

1. When entering a new employee, if a value for address is not supplied then automatically enter ‘N/A’.
2. When entering a new project, if a value is not supplied for location, then automatically enter ‘Nicosia’.
3. The department name must always be unique.

Constraints of COMPANY DB

4. The project name must also be unique.
5. To avoid accountant payment mistakes enforce that each employee is paid a salary between 1000 and 5000 euro
6. When a manager is assigned to a department, and the start date is not provided use the current date of the system

Constraints of COMPANY DB

7. A project can only located at a cypriot district (i.e., Nicosia, Larnaka, Lemesos, Paphos, Famagusta)
8. Each employee's first name, last name as well as a department's name must be a string whose length is greater than 0 (i.e., no Empty strings are allowed)

Other Information

- Constraint Naming
- View all constraints (or other objects) of a database

Constraint Naming

- You can always name the constraints yourself (e.g., pk_test, fk_project)
- **OR** you can let SQL Server name them for you

```
CREATE TABLE dbo.test (…,
    testname nvarchar(50) DEFAULT 'Unknown',
    --constraint is assigned a random name (e.g.,
    'DF__test__testname__46E78A0C')
    …)
```

View all constraints

```
USE COMPANY

SELECT      OBJECT_NAME(OBJECT_ID) AS NameofConstraint,
            SCHEMA_NAME(schema_id) AS SchemaName,
            OBJECT_NAME(parent_object_id) AS TableName,
            type_desc AS ConstraintType

FROM        sys.objects

WHERE       type_desc LIKE '%CONSTRAINT%‘

ORDER BY    type_desc
```