1. ***The SQL CREATE DATABASE Statement***

* The CREATE DATABASE statement is used to create a new SQL database.
* Syntax :
* **CREATE DATABASE databasename;**
* Example :
* **CREATE DATABASE testDB;**
* Note : Make sure you have admin privilege before creating any database. Once a database is created, you can check it in the list of databases with the following SQL command: SHOW DATABASES;

1. ***The SQL DROP DATABASE Statement***

* The DROP DATABASE statement is used to drop an existing SQL database.
* Syntax :
* **DROP DATABASE databasename;**
* Example :
* **DROP DATABASE testDB;**
* Notes :
* Be careful before dropping a database. Deleting a database will result in loss of complete information stored in the database!
* Make sure you have admin privilege before dropping any database. Once a database is dropped, you can check it in the list of databases with the following SQL command: SHOW DATABASES;

1. ***The SQL BACKUP DATABASE Statement***

* The BACKUP DATABASE statement is used in SQL Server to create a full back up of an existing SQL database.
* Syntax-1 :
* **BACKUP DATABASE databasename  
  TO DISK = 'filepath';**
* Syntax-2 (With DIFFERENTIAL Statement) :
* **BACKUP DATABASE databasename  
  TO DISK = 'filepath'  
  WITH DIFFERENTIAL;**
* Note : A differential back up only backs up the parts of the database that have changed since the last full database backup.
* Example-1 :
* **BACKUP DATABASE testDB  
  TO DISK = 'D:\backups\testDB.bak';**
* Note : Always back up the database to a different drive than the actual database. Then, if you get a disk crash, you will not lose your backup file along with the database.
* Example-2 (With DIFFERENTIAL Statement) :
* **BACKUP DATABASE testDB  
  TO DISK = 'D:\backups\testDB.bak'  
  WITH DIFFERENTIAL;**
* Note : A differential back up reduces the back up time (since only the changes are backed up).

1. ***The SQL CREATE TABLE Statement***

* The CREATE TABLE statement is used to create a new table in a database.
* Syntax-1 :
* **CREATE TABLE table\_name (  
      column1 datatype,  
      column2 datatype,  
      column3 datatype,  
     ....  
  );**
* Syntax-2 (Create Table Using Another Table) :
* **CREATE TABLE new\_table\_name AS  
      SELECT column1, column2,...  
      FROM existing\_table\_name  
      WHERE ....;**
* Example-1 :
* **CREATE TABLE Persons (  
      PersonID int,  
      LastName varchar(255),  
      FirstName varchar(255),  
      Address varchar(255),  
      City varchar(255)  
  );**
* Example-2 (Create Table Using Another Table) :
* **CREATE TABLE TestTable AS  
  SELECT customername, contactname  
  FROM customers;**

1. ***The SQL DROP TABLE Statement***

* The DROP TABLE statement is used to drop an existing table in a database.
* Syntax :
* **DROP TABLE table\_name;**
* Note : Be careful before dropping a table. Deleting a table will result in loss of complete information stored in the table!

1. ***The SQL TRUNCATE TABLE Statement***

* The TRUNCATE TABLE statement is used to delete the data inside a table, but not the table itself.
* Syntax :
* **TRUNCATE TABLE table\_name;**

1. ***The SQL ALTER TABLE Statement***

* The ALTER TABLE statement is used to **add**, **delete**, or **modify columns** in an existing table. The ALTER TABLE statement is also used to **add** and **drop various constraints** on an existing table.
* Syntax ADD COLUMN :
* **ALTER TABLE table\_name  
  ADD column\_name datatype;**
* Syntax DROP COLUMN :
* **ALTER TABLE table\_name  
  DROP COLUMN column\_name;**
* Syntax ALTER/MODIFY COLUMN :
* SQL Server / MS Access :
  + **ALTER TABLE table\_name  
    ALTER COLUMN column\_name datatype;**
* My SQL / Oracle (Prior version 10G) :
  + **ALTER TABLE table\_name  
    MODIFY COLUMN column\_name datatype;**
* Oracle 10G and later :
  + **ALTER TABLE table\_name  
    MODIFY column\_name datatype;**