**SQL CREATE DATABASE Statement**

The CREATE DATABASE statement is used to create a new SQL database.

**Syntax**

CREATE DATABASE databasename;

**The SQL DROP DATABASE Statement**

The DROP DATABASE statement is used to drop an existing SQL database.

**Syntax**

DROP DATABASE databasename;

**The SQL CREATE TABLE Statement**

The CREATE TABLE statement is used to create a new table in a database.

**Syntax**

CREATE TABLE table\_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype  
   
);

**The SQL DROP TABLE Statement**

The DROP TABLE statement is used to drop an existing table in a database.

**Syntax**

DROP TABLE table\_name;

**SQL TRUNCATE TABLE**

The TRUNCATE TABLE statement is used to delete the data inside a table, but not the table itself.

**Syntax**

TRUNCATE TABLE table\_name;

**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.

**ALTER TABLE - ADD Column**

To add a column in a table, use the following syntax:

ALTER TABLE table\_name  
ADD column\_name datatype;

**ALTER TABLE - DROP COLUMN**

To delete a column in a table, use the following syntax (notice that some database systems don't allow deleting a column)

ALTER TABLE table\_name   
DROP COLUMN column\_name;

**SQL Constraints**

SQL constraints are used to specify rules for data in a table.

Constraints are used to limit the type of data that can go into a table.

Constraints can be specified when the table is created with the CREATE TABLE statement, or after the table is created with the ALTER TABLE statement.

* NOT NULL - Ensures that a column cannot have a NULL value
* UNIQUE - Ensures that all values in a column are different
* PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table
* FOREIGN KEY - Prevents actions that would destroy links between tables
* CHECK - Ensures that the values in a column satisfies a specific condition
* DEFAULT - Sets a default value for a column if no value is specified
* CREATE INDEX - Used to create and retrieve data from the database very quickly

CREATE TABLE table\_name (  
    column1 datatype *constraint*,  
    column2 datatype *constraint*,  
    column3 datatype *constraint*,  
     
);

**Example**

CREATE TABLE Persons (  
    ID int NOT NULL UNIQUE,  
    LastName varchar (255) NOT NULL,  
    FirstName varchar (255) NOT NULL,  
    Age int  
);

CREATE TABLE Persons (  
    ID int NOT NULL PRIMARY KEY,  
    LastName varchar (255) NOT NULL,  
    FirstName varchar(255),  
    Age int  
);

The MS SQL Server uses the IDENTITY keyword to perform an auto-increment feature.

CREATE TABLE Persons (  
    Personid int IDENTITY(1,1) PRIMARY KEY,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int  
);

the starting value for IDENTITY is 1, and it will increment by 1 for each new record.

**SQL**INSERT INTO**Statement**

INSERT INTO table\_name (column1, column2, column3)  
VALUES (value1, value2, value3);

**SQL**SELECT**Statement**

SELECT column1, column2 FROM table\_name;

SELECT \* FROM table\_name;

**SQL**SELECT DISTINCT**Statement**

The SELECT DISTINCT statement is used to return only distinct (different) values.

Inside a table, a column often contains many duplicate values; and sometimes you only want to list the different (distinct) values.

SELECT DISTINCT \* FROM table\_name;

**The SQL WHERE Clause**

The WHERE clause is used to filter records.

It is used to extract only those records that fulfill a specified condition.

*SELECT*\* FROMtable\_name *WHERE*condition*;*

**SQL**AND, OR and NOT**Operators**

The WHERE clause can be combined with AND, OR, and NOT operators.

The AND and OR operators are used to filter records based on more than one condition:

* The AND operator displays a record if all the conditions separated by AND are TRUE.
* The OR operator displays a record if any of the conditions separated by OR is TRUE.

The NOT operator displays a record if the condition(s) is NOT TRUE.

**AND Syntax**

SELECT column1, column2, ...  
FROM table\_name  
WHERE condition1 AND condition2 AND condition3 ...;

SELECT \* FROM tbl\_name  
WHERE Country='Pakistan' AND City='karachi';

**OR Syntax**

SELECT column1, column2, ...  
FROM table\_name  
WHERE condition1 OR condition2 OR condition3 ...;

SELECT \* FROM Customers  
WHERE Country='Pakistan' OR City='karachi';

**NOT Syntax**

SELECT column1, column2, ...  
FROM table\_name  
WHERE NOT condition;

SELECT \* FROM Customers  
WHERE NOT Country='Germany';

**SQL**ORDER BY**Keyword**

The ORDER BY keyword is used to sort the result-set in ascending or descending order.

The ORDER BY keyword sorts the records in ascending order by default. To sort the records in descending order, use the DESC keyword.

SELECT column1, column2 FROM table\_name  
ORDER BY column1, column2 ASC|DESC;

SELECT \* FROM tbl\_name ORDER BY Country DESC;

SELECT \* FROM tbl\_name ORDER BY columnname ASC, columnname DESC;