SQL Course

DDL : Create Statement

Objectives

- To introduce why and where the SQL create statement is used.
- Describe the syntax of the **CREATE** statement and illustrate its use, using examples.

- Tables are created using the **CREATE TABLE** command.
- Each table consists of at least one column.
- Each column has a:
  - Name.
  - Type.
  - Size (depending on the type).
  - Optional integrity constraints.

A **CREATE TABLE** command creates a table in the sense that it makes the table known to the system; it gives the new table a name and defines its columns. A column is defined by:

- A name
- A data type
- A flag which indicates whether or not the column can accommodate **NULL** values

The table name must be system wide unique. In a system environment which supports many users and many applications this constraint may cause problems.
Some SQL systems solve this problem by allowing a user to choose names that are unique only among the set of tables that they created. The system wide unique name is then obtained by concatenating the name of the table with the name of the user who created the table and separating them with a dot. This is convention is known as **qualified table names**.
Generally the system knows which user is submitting a particular request to the system. Therefore the short name, without the qualifier, is often sufficient, since the system itself can add the qualifier when necessary.
Suppose we wish to specify a table representing an *Airport* with five attributes, namely : an *airport code* (maximum of three characters uniquely identifying the airport e.g. DUB, LHR), an *airport name*, *location*, *country* (each of up to 20 character in length), and *time difference* (number representing difference from GMT, with a default value of 0).

An appropriate table specification in SQL would be:

> **CREATE TABLE** *airport* (*id* CHAR(3) **NOT NULL,** *location* VARCHAR2(20) **NOT NULL,** *country* VARCHAR2(20) **NOT NULL,** *time_difference* NUMBER(2) **DEFAULT 0 NOT NULL, PRIMARY KEY**(*id*))**;**

The general syntax of the **CREATE TABLE** statement is:

> **CREATE TABLE** *tablename* (*column_name* DATA_TYPE [{NULL|NOT NULL}] [**DEFAULT** <*literal*>] [,*column_name* DATA_TYPE[{NULL|NOT NULL}] [**DEFAULT** <*literal*>]] [, ...] [**PRIMARY KEY** (*column_name* [, *column_name*] [, ...])] [**FOREIGN KEY**

(*column_name* [, *column_name*] [, ...])] **REFERENCES***table_name*] [**UNIQUE** (*column_name*[, *column_name*] [, ...])] [**CHECK** (search_condition)])**;**

The SQL2 standard allows several different parts of a column definition, which require the column to contain unique values, to specify that the column is a primary key or a foreign key, or to restrict the data values that a column can contain. The **DEFAULT** value is a value which will automatically be used by the database when a value is not specified by the **INSERT** statement into that particular column.

Note: More complicated examples of table creations which use all of the above features are explained in the **Table Constraints** module.