**ST. XAVIER’S COLLEGE**

**MAITIGHAR, KATHMANDU**

****

**Database Management System**

**Theory Assignment #8**

**Submitted by:**

Anu Kadel

013BSCIT007

**Submitted to:**

|  |  |
| --- | --- |
| **Er. Sanjay Kumar Yadav** |  |

Lecturer

Department of Computer Science

St. Xavier’s College, Maitighar

**[1]Data definition language**

A data definition language or data description language (**DDL**) is a syntax similar to a computer programming language for defining data structures, especially database schemas.

* 1. **Domain type in SQL**

The SQL standard supports a variety of built-in domain types:

* **char**(n) (or **character**(n)): fixed-length character string, with user-specified length.
* **varchar**(n) (or **character varying**): variable-length character string, with user-specified maximum length.
* **int** or **integer**: an integer (length is machine-dependent).
* **smallint**: a small integer (length is machine-dependent).
* **numeric**(*p, d*): a fixed-point number with user-specified precision, consists of *p* digits (plus a sign) and *d* of *p* digits are to the right of the decimal point. E.g., **numeric**(*3, 1*) allows 44.5 to be stored exactly but not 444.5.
* **real** or **double precision**: floating-point or double-precision floating-point numbers, with machine-dependent precision.
* **float**(n): floating-point, with user-specified precision of at least *n* digits.
* **date**: a calendar date, containing four digit year, month, and day of the month.
* **time**: the time of the day in hours, minutes, and seconds.
* SQL allows arithmetic and comparison operations on various numeric domains, including, **interval** and *cast* (*type coercion*) such as transforming between *smallint* and *int*.
* It considers strings with different length are compatible types as well.
* SQL allows **create domain** statement,

e.g.,

**Create domain** *person-name* **char**(20)

* 1. **Schema Definition in SQL**

An SQL relation is defined by:

**create table** *r* ( tex2html_wrap_inline1854

*integrity-* tex2html_wrap_inline1856 ,

..., *integrity-* tex2html_wrap_inline1856 )

where *r* is the relation name, tex2html_wrap_inline1730is the name of an attribute, and tex2html_wrap_inline1864is the domain of that attribute. The allowed integrity-constraints include

**primary key** tex2html_wrap_inline1866

and

**check(*P*)**

Example.

**create table** *branch* (

bname **char**(15) **not null**

bcity **char**(30)

assets **integer**

**primary key** (*bname*)

**check** (*assets >= 0*))

The values of primary key must be *not null* and *unique*. SQLconsider **not null** in primary key specification is redundant but SQL-89 requires to define it explicitly.

Check creates type checking functionality which could be quite useful. E.g.,

**create table** *student* (

*name* **char**(15) **not null**

*student-id* **char**(10) **not null**

*degree-level* **char**(15) **not null**

**check** (*degree-level* **in**

(``Bachelors'', ``Masters'', ``Doctorate'')))

Some checking (such as *foreign-key* constraints) could be costly, e.g.,

**check** (*bname* **in** (**select** *bname* **from** *branch*))

A newly loaded table is empty. The **insert** command can be used to load it, or use special bulk loader untilities.

To remove a relation from the database, we can use the **drop table** command:

**drop table** *r*

This is not the same as

**delete** *r*

which retains the relation, but deletes all tuples in it.

The **alter table** command can be used to add or drop attributes to an existing relation *r*:

**alter table** *r* **add** *A* *D*

where *A* is the attribute and *D* is the domain to be added.

**alter table** *r* **drop** *A*

where *A* is the attribute to be dropped.

**[2] Data Manipulation Language**

A **data manipulation language** (**DML**) is a family of syntax elements similar to a computer programming language used for selecting, inserting, deleting and updating data in a database.

**2.1 The Select Clause**

### The SELECT statement is used to select data from a database. The result is stored in a result table, called the result-set. SQL SELECT Syntax

SELECT *column\_name*,*column\_name*  
FROM *table\_name*;

and

SELECT \* FROM *table\_name*

# SQL SELECT DISTINCT Statement:

In a table, a column may contain many duplicate values; and sometimes you only want to list the different (distinct) values.

The DISTINCT keyword can be used to return only distinct (different) values

**SQL SELECT DISTINCT Syntax**

SELECT DISTINCT *column\_name*,*column\_name*  
FROM *table\_name*;

**2.2 The Where Clause**

The WHERE clause is used to extract only those records that fulfill a specified criterion.

**SQL WHERE Syntax**

SELECT *column\_name*,*column\_name*  
FROM *table\_name*  
WHERE *column\_name operator value*;

**2.3 The Form Clause**

The Rename Operation

Tuple Variable

**String Operations**

**ABC%** :

All strings that start with ABC.

Example ;ABCD and ABCDABCD would both satisft the condition.

**%XYZ:**

All string that end with XYZ.

Example: WXYZ and ZZXYZ would both satisfy the condition.

**%AN%**

All string that contain the pattern **AN** anywhere.

Example: LOS ANGLES and SAN FRANSCISCO.

Ordering the Display of Tuples

Duplicate Tuples