**4.2 Data Definition language.**

**4.2.1. Domain Type in SQL**

**Character Domain Types in SQL**

• CHAR(n) :Fixed length character string, with user-specified length n. •

VARCHAR(n): Variable length character strings, with user-specified maximum length n.

• Null: values are allowed in all the domain types.

Declaring an attribute to be NOT NULL prohibits null values for that attribute.

**Number Domain Types in SQL**

• INT (also: INTEGER):Integer (a finite subset of the integers that is machine-dependent).

• SMALLINT: Small integer (a machine-dependent subset of the integer domain type)

. • DECIMAL(p,d): Fixed point number, with user-specified precision of p digits, with n digits to the right of decimal point.

• FLOAT(n): Floating point number, with user-specified precision of at least n digits.

• REAL (also: DOUBLE PRECISION) Floating point and double-precision floating point numbers, with machine-dependent precision. COMP5138 "Relational Database Management Systems"

**Date Domain Types of Oracle SQL**

• DATE : dates, containing a (4 digit) year, month and date – E.g. DATE ‘2001-7-27’ (ANSI Syntax) – E.g. TO\_DATE (‘01-JUL-27’,’YY-MON-DD’) (Oracle Syntax)

• TIMESTAMP: date plus time of day – E.g. TIMESTAMP ‘2001-7-27 09:00:30.75’

**4.2.2. Scheme 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_inline1730 is the name of an attribute, and tex2html_wrap_inline1864 is 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*. SQL-92 consider **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

**4.3. Data Manipulation Language.**

**4.3.1 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*;

**4.3.2 The Where Clause**

**4.3.3 The Form Clause**

**4.3.4 The Rename Operation**

**4.3.5 Tuple Variable**

**4.3.6 String Operations**

**4.3.7 Ordering The Display Of Tuples.**

**4.3.8 Duplicate Tuples**