## **Objectives:**

The aim of this Lab is to practice the commands: insert, update and delete with referential integrity.

## **Lab Work:**

We create a table department:

CREATE TABLE DEPARTMENT

(Dname VARCHAR (15) NOT NULL,

Dnumber INT NOT NULL,

Mgr\_ssn CHAR (9) NOT NULL,

Mgr\_start\_date DATE,

PRIMARY KEY (Dnumber));

**Insert:**

**To insert rows into the department table:**

insert into department values ('Sales', 1, '123456789', '12-dec-2004');

insert into department values ('Sales', 2, '123456799', '11-dec-2004');

insert into department values ('Sales', 3, '123456999', '11-oct-2005');

insert into department values ('Sales', 4, '123456999', null);

**To view the contents of the table:**

select \* from department;

**Update:**

**Change the dnumber of the department 4 into 5.**

update department set dnumber = 5 where dnumber=4;

update department set mgr\_start\_date=null;

update department set mgr\_start\_date ='07-jan-1999', mgr\_ssn = 123456789;

update department set mgr\_start\_date='8-oct-1998' where dnumber =1 and dname ='Sales';

**Delete:**

delete from department where dnumber =1;

delete from department where dnumber =6;

update department set mgr\_start\_date=null where dnumber =2;

delete from department where mgr\_ssn='123456789' and mgr\_start\_date='07-jan-1999';

update department set mgr\_start\_date='12-feb-1978' where dnumber =2;

delete from department;

**Foreign keys and their constraints:**

we first create an employee table and make the dno of the employee table as foreign key to dnumber at department table as follows:

CREATE TABLE EMPLOYEE

(Fname VARCHAR (15) NOT NULL,

Minit CHAR,

Lname VARCHAR (15) NOT NULL,

Ssn CHAR (9) NOT NULL,

Bdate DATE,

Address VARCHAR (30),

Sex CHAR,

Salary DECIMAL (10,2),

Super\_ssn CHAR (9),

Dno INT NOT NULL,

PRIMARY KEY (Ssn));

\*Alter table employee add (constraint d\_number\_const foreign key(dno) references department(dnumber));

\*Now we add a tuple to the employee table. In this case we have to enter a valid department number otherwise referential integrity constraints will be violated.

The following insert is valid:

insert into employee values ('Omar', 'i', 'hassan', '12345', '09-oct-1997', null, 'M', 1234.45, null, 1);

This insert is not valid because there is no department with number 8:

insert into employee values ('hala', 'm', 'hassan', '1245', '09-oct-1997', null, 'M', 1234.45, null, 8);

The following insert cannot be executed either because of the not null constraint violation of the dno attribute:

insert into employee values ('hala', 'm', 'hassan', '1245', '09-oct-1997', null, 'M', 1234.45, null, null);

**In order to allow null values for the dno attribute of the employee relation:**

alter table employee modify (dno int null);

now the last insert operation will work.

**On delete set null, on delete cascade.**

We will drop the constraint d\_number\_const and add the foreign key again with the clause on delete set null (whenever we drop the referenced tuple we delete also all referencing tuple):

alter table employee drop constraint d\_number\_const;

Alter table employee add constraint d\_number\_const foreign key(dno) references department(dnumber) on delete set null;

Now if we delete a department:

delete from department where dnumber =1;

then all the employees who belong to this department will have dno =null.

In order to drop the constraint:

alter table employee drop constraint d\_number\_const;

## **Class Exercise:**

Based on the schema on the next page, perform the following:

1. Create the project table without Dnum and add the projects as per attachment.
2. Perform three modifications to this table where you suggest new values for existing entries in all three fields.
3. Delete one entry.
4. Create the department table with only two fields dname and dnumber.
5. Add the field dno into the project table. Add a foreign key constraint for dno to dnum of the department table (without the on delete cascade or on delete set null).
6. Fill in this field for all entries as per attachment.
7. Delete an entry from the project table. Will it be completed successfully?
8. Delete an entry from the department table. What is the message received and why?
9. Drop the constraint of the foreign key. Add the it again with on delete cascade.
10. Redo point E. check the results.
11. Drop the constraint of the foreign key. Add it again with on delete set null.
12. Redo the point E. check the results.



