

Creating and inserting values

[illegible]

SQL DML

- Create a temporary table that has the employee last name, project name, and hours per week for each employee working on a project. Insert the values into the table using insert into with select command.

```
company=# create table temporary_001(emp_last_name varchar(15),project_name varchar(15),hpw numeric(3,1));
CREATE TABLE
company=# \d temporary_001
      Table "public.temporary_001"
  Column      |      Type      | Collation | Nullable | Default
-----|-----|-----|-----|-----
emp_last_name | character varying(15) |           |          |
project_name  | character varying(15) |           |          |
hpw           | numeric(3,1)         |           |          |
```

```

company=# insert into temporary_001
select lname,pname,hours from employee,project,works_on where employee.ssn=works_on.essn and works_on.pno=project.pnumber;
INSERT 0 15
company=# select * from temporary_001 ;

```

emp_last_name	project_name	hpw
Smith	ProductX	32.5
Smith	ProductY	7.5
Narayan	ProductZ	40.0
English	ProductX	20.0
English	ProductY	20.0
Wong	ProductY	10.0
Wong	ProductZ	10.0
Wong	Computerization	10.0
Wong	Reorganization	10.0
Zelaya	Newbenefits	30.0
Zelaya	Computerization	10.0
Jabbar	Computerization	35.0
Jabbar	Newbenefits	5.0
Wallace	Newbenefits	20.0
Wallace	Reorganization	15.0

```

(15 rows)

```

- Update the location and controlling department number of project number 10 to 'Bellaire' and 5, respectively.

```

company=# update project
company=# set plocation='Bellaire'
company=# where pnumber='10';
UPDATE 1
company=# update project
company=# set dnum=5
company=# where pnumber='10';
UPDATE 1
company=# select * from project;

```

pname	pnumber	plocation	dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4
Computerization	10	Bellaire	5

```

(6 rows)

```

- Give all employees in the 'Research' department a 10% raise in salary.

```

company=## update test set salary=1.1*salary where dno in (select dnumber from department where dname='Research');
UPDATE 2

```

```
company==# select * from employee;
```

fname	minit	lname	ssn	bdate	address	gender	salary	super_ssn	dno
James	E	Borg	888665555	1937-11-10	450 Stone, Houston,TX	M	55000.00		1
John	B	Smith	123456789	1965-01-09	731 Fondren,Houston,TX	M	30000.00	888665555	5
Franklin	T	Wong	333445555	1955-12-08	638 voss,Houston,TX	M	40000.00	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring,Tx	F	25000.00	333445555	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire,Tx	F	43000.00	333445555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000.00	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice,Houston,TX	F	25000.00	333445555	5
Ahmed	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston,TX	M	25000.00	987654321	4

(8 rows)

- Delete employee record whose lname ='Brown'

```
company==# delete from employee where lname='Brown';
DELETE 0
company==#
```

- Delete all the records of the employee who doesn't have a dependent. (use sub query).

```
company=# alter table employee drop constraint employee_super_ssn_fkey;
ALTER TABLE
```

```
company=# alter table dependent drop constraint dependent_essn_fkey;
ALTER TABLE
```

```
company=# alter table department drop constraint department_mgr_ssn_fkey;
ALTER TABLE
company=# alter table works_on drop constraint works_on_essn_fkey;
ALTER TABLE
```

```
company=# delete from employee
where not exists (select essn from dependent where essn=employee.ssn)
;
DELETE 5
company=# select * from employee ;
```

fname	minit	lname	ssn	bdate	address	gender	salary	super_ssn	dno
John	B	Smith	123456789	1965-01-09	731 Fondren,Houston,TX	M	30000.00	888665555	5
Franklin	T	Wong	333445555	1955-12-08	638 voss,Houston,TX	M	40000.00	888665555	5
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire,Tx	F	43000.00	333445555	4

(3 rows)

2.

Transactions

Create a transaction using begin and end commands consisting of the following sql statements.

- Create a transaction consisting of a create table and multiple insert statements. After End transaction the changes should be committed and can be checked using the select statement.

```

company=# begin;
BEGIN
company=# create table narendiran_001(exam varchar(10),gpa numeric(3,1));
CREATE TABLE
company=# insert into narendiran_001 values('sem 1',7.55);
INSERT 0 1
company=# insert into narendiran_001 values('sem 2',8.5);
INSERT 0 1
company=# end transaction;
COMMIT
company=# select * from narendiran_001;
 exam | gpa
-----+-----
 sem 1 | 7.6
 sem 2 | 8.5
(2 rows)

```

- For the above transaction, introduce a roll back after inserting 2 records. The create and insert should not be reflected in the database.

```

company=# begin;
BEGIN
company=# create table narendiran_rollback_001(exam varchar(10),gpa numeric(3,1));
CREATE TABLE
company=# insert into narendiran_rollback_001 values('sem 3',8.00),('sem 4',8.79);
INSERT 0 2
company=# select * from narendiran_rollback_001;
 exam | gpa
-----+-----
 sem 3 | 8.0
 sem 4 | 8.8
(2 rows)

company=# rollback;
ROLLBACK
company=# select * from narendiran_r

company=# select * from narendiran_rollback_001;
ERROR:  relation "narendiran_rollback_001" does not exist
LINE 1: select * from narendiran_rollback_001;
                        ^

```

- For the first transaction, introduce a save point after inserting 2 records and insert 2 more records and rollback to savepoint . The database should reflect only the first 2 insertions.

```

company=# begin;
BEGIN
company=# create table narendiran_savepoint_001(exam varchar(10),gpa numeric(3,1));
CREATE TABLE
company=# insert into narendiran_savepoint_001 values('sem 1',7.55),('sem 2',8.5);
INSERT 0 2
company=# savepoint s1;
SAVEPOINT
company=# insert into narendiran_savepoint_001 values('sem 3',8.00),('sem 4',8.8);
INSERT 0 2
company=# select * from narendiran_savepoint_001 ;
 exam | gpa
-----+-----
 sem 1 | 7.6
 sem 2 | 8.5
 sem 3 | 8.0
 sem 4 | 8.8
(4 rows)

company=# rollback to s1;
ROLLBACK
company=# select * from narendiran_savepoint_001 ;
 exam | gpa
-----+-----
 sem 1 | 7.6
 sem 2 | 8.5
(2 rows)

```