

DBMS LAB WEEK5

ATUL ANURAG
PES2UG19CS075
SECTION B

SQL DML- Insert (using select) Update and delete, Transactions commit rollback and savepoint.

1. 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.

```
Windows PowerShell
company=# \c
You are now connected to database "company" as user "atul".
company=# create table temp_table(
company(# lname          varchar(15)      NOT NULL,
company(# Pname          varchar(15)      NOT NULL,
company(# Hours          decimal(3,1)     NOT NULL
company(# );
CREATE TABLE
company=# insert into temp_table
company-# select E.Lname, P.Pname, W.Hours from Employee E, Project P, Works_On W
company-# where E.Ssn=W.Essn AND P.Pnumber=W.Pno;
INSERT 0 15
company=# select * from temp_table;
  lname |      pname      | hours
-----+-----+-----
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)

company=#
```

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

```
company=# update project
company-# set Plocation='Bellaire', Dnum=5
company-# where Pnumber=10;
UPDATE 1
company=# select * from project
company-# where pnumber=10;
  pname      | pnumber | plocation | dnum
-----+-----+-----+-----
Computerization | 10 | Bellaire | 5
(1 row)

company=#
```

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

```
Windows PowerShell
company=# update Employee
company=# set Salary= Salary + (0.1 * Salary)
company=# from department
company=# where Dno=Department.Dnumber AND Department.Dname='Research';
UPDATE 4
company=#
```

- Delete employee record whose lname ='Brown'

```
Windows PowerShell
company=# delete from Employee
company=# where Lname= 'Brown';
DELETE 0
company=#
```

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

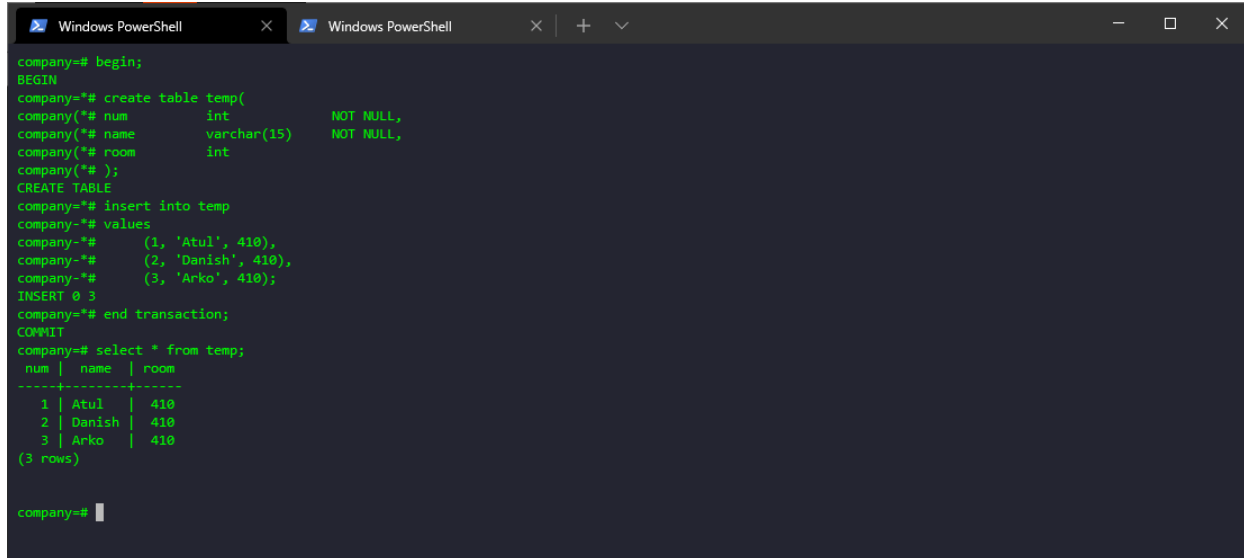
```
Windows PowerShell
company=# alter table employee
company=# drop constraint employee_super_ssn_fkey;
ALTER TABLE
company=# alter table department
company=# drop constraint department_mgr_ssn_fkey;
ALTER TABLE
company=# alter table works_on
company=# drop constraint works_on_essn_fkey;
ALTER TABLE
company=# delete from employee
company=# where ssn not in
company=# (select Dependent.Essn
company=# from Dependent);
DELETE 5
company=# select * from employee;
  fname | minit | lname | ssn | bdate | address | gender | salary | super_ssn | dno
-----|-----|-----|----|-----|-----|-----|-----|-----|----
Jennifer | S | Wallace | 987654321 | 1941-06-20 | 291 Berry, Bellaire,Tx | F | 43000.00 | 333445555 | 4
John | B | Smith | 123456789 | 1965-01-09 | 731 Fondren,Houston,TX | M | 33000.00 | 888665555 | 5
Franklin | T | Wong | 333445555 | 1955-12-08 | 638 voss,Houston,TX | M | 44000.00 | 888665555 | 5
(3 rows)

company=#
```

2. Transactions

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

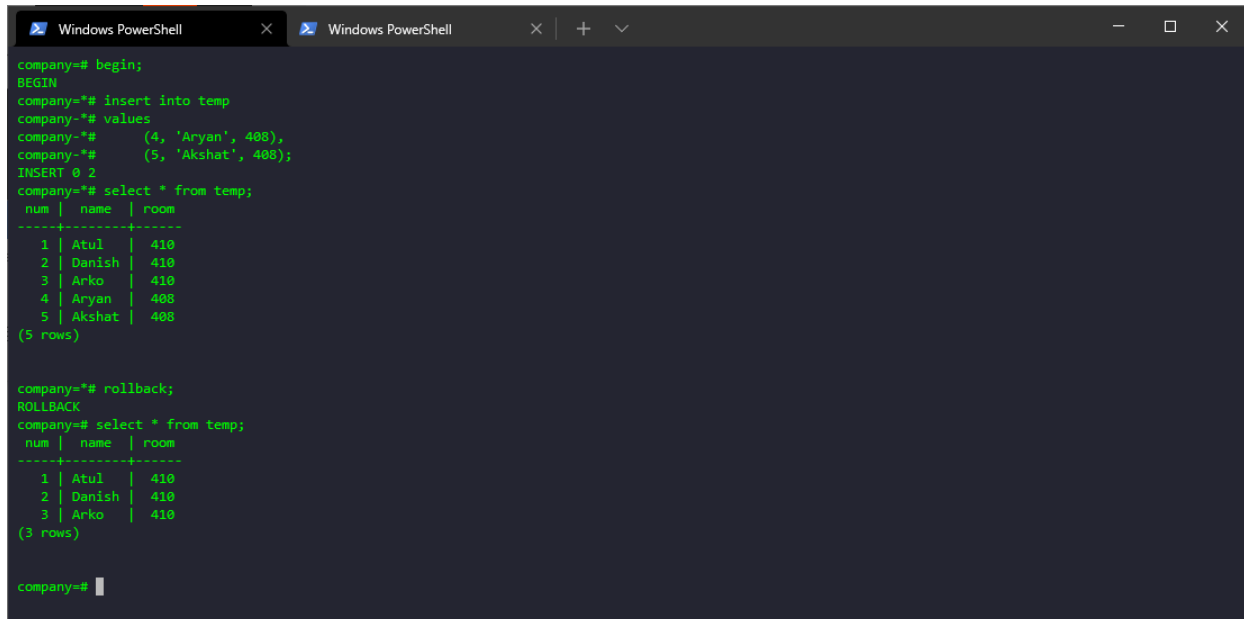
- Create a transaction consisting of a create table and multiple insert statements. After End transaction the changes should be committed and can be checked using select statement.



```
company=# begin;
BEGIN
company=# create table temp(
company=# num          int          NOT NULL,
company=# name         varchar(15)  NOT NULL,
company=# room         int
company=# );
CREATE TABLE
company=# insert into temp
company=# values
company=# (1, 'Atul', 410),
company=# (2, 'Danish', 410),
company=# (3, 'Arko', 410);
INSERT 0 3
company=# end transaction;
COMMIT
company=# select * from temp;
 num | name | room
-----+-----+-----
  1  | Atul | 410
  2  | Danish | 410
  3  | Arko | 410
(3 rows)

company=#
```

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

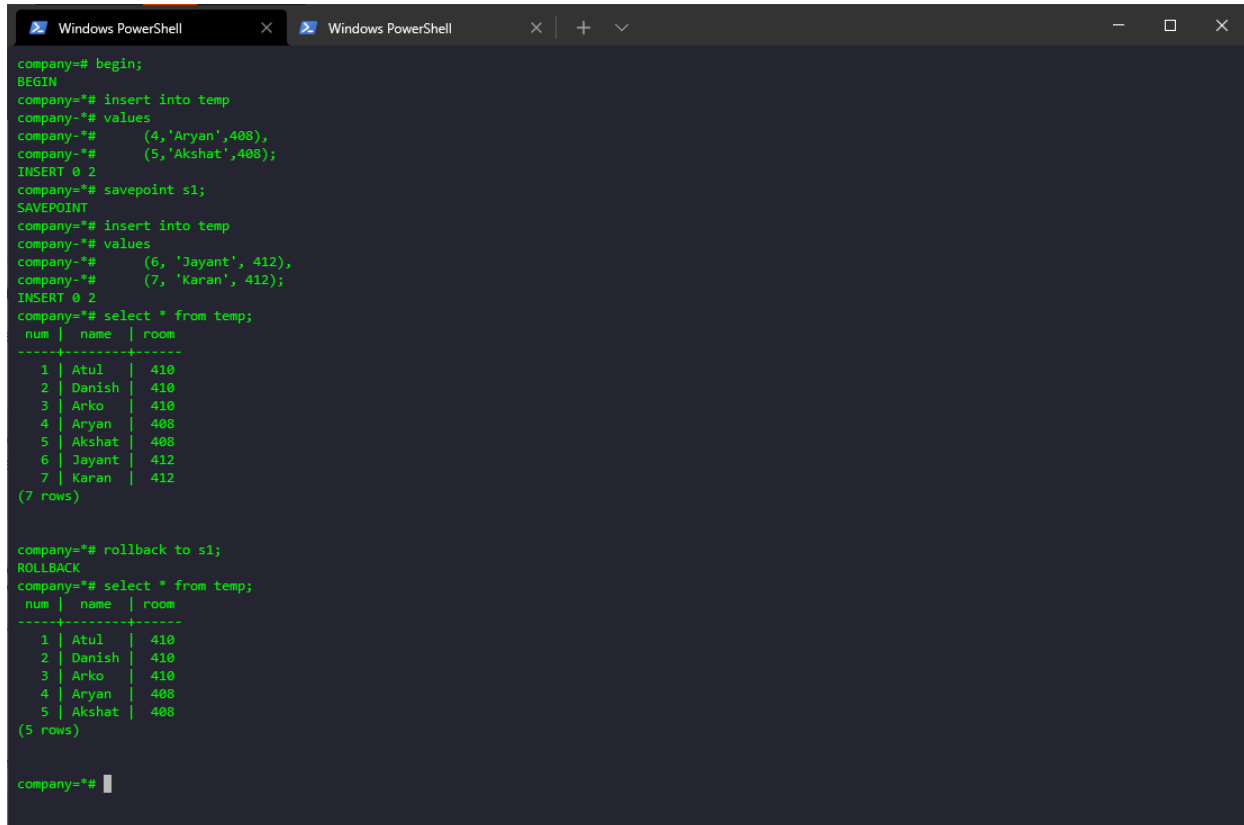


```
company=# begin;
BEGIN
company=# insert into temp
company=# values
company=# (4, 'Aryan', 408),
company=# (5, 'Akshat', 408);
INSERT 0 2
company=# select * from temp;
 num | name | room
-----+-----+-----
  1  | Atul | 410
  2  | Danish | 410
  3  | Arko | 410
  4  | Aryan | 408
  5  | Akshat | 408
(5 rows)

company=# rollback;
ROLLBACK
company=# select * from temp;
 num | name | room
-----+-----+-----
  1  | Atul | 410
  2  | Danish | 410
  3  | Arko | 410
(3 rows)

company=#
```

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



```
company=# begin;
BEGIN
company=# insert into temp
company=# values
company=#      (4,'Aryan',408),
company=#      (5,'Akshat',408);
INSERT 0 2
company=# savepoint s1;
SAVEPOINT
company=# insert into temp
company=# values
company=#      (6, 'Jayant', 412),
company=#      (7, 'Karan', 412);
INSERT 0 2
company=# select * from temp;
 num | name | room
-----+-----+-----
  1  | Atul | 410
  2  | Danish | 410
  3  | Arko | 410
  4  | Aryan | 408
  5  | Akshat | 408
  6  | Jayant | 412
  7  | Karan | 412
(7 rows)

company=# rollback to s1;
ROLLBACK
company=# select * from temp;
 num | name | room
-----+-----+-----
  1  | Atul | 410
  2  | Danish | 410
  3  | Arko | 410
  4  | Aryan | 408
  5  | Akshat | 408
(5 rows)

company=#
```