**commands:(postgres User):**

create database ani;

create user ani with superuser login password 'ani';

create database vini;

create user vini with superuser login password 'vini';

create database lini;

create user lini with superuser login password 'lini';

**After this Open new cmd and open ani, vini, lini user. Same way to Open this users like postgres user:**

**COMMAND LIKE( psql -U username)**

**in ani::**

create extension dblink;

select dblink\_connect('host=127.0.0.2 user=vini password=vini dbname=vini');

create foreign data wrapper viniwrap validator postgresql\_fdw\_validator;

create server vinilink foreign data wrapper viniwrap options(hostaddr '127.0.0.2',dbname 'vini');

create user mapping for ani server vinilink options(user 'vini',password 'vini');

grant usage on foreign server vinilink to ani;

select dblink\_connect('host=127.0.0.2 user=lini password=lini dbname=lini');

create foreign data wrapper liniwrap validator postgresql\_fdw\_validator;

create server linilink foreign data wrapper liniwrap options(hostaddr '127.0.0.2',dbname 'lini');

create user mapping for ani server linilink options(user 'lini',password 'lini');

grant usage on foreign server linilink to ani;

**in vini::**

create extension dblink;

select dblink\_connect('host=127.0.0.2 user=ani password=ani dbname=ani');

create foreign data wrapper vtoaniwrap validator postgresql\_fdw\_validator;

create server vtoalink foreign data wrapper vtoaniwrap options(hostaddr '127.0.0.2',dbname 'ani');

create user mapping for vini server vtoalink options(user 'ani',password 'ani');

grant usage on foreign server vtoalink to vini;

**in lini :**

create extension dblink;

select dblink\_connect('host=127.0.0.2 user=ani password=ani dbname=ani');

create foreign data wrapper ltoaniwrap validator postgresql\_fdw\_validator;

create server ltoalink foreign data wrapper ltoaniwrap options(hostaddr '127.0.0.2',dbname 'ani');

create user mapping for lini server ltoalink options(user 'ani',password 'ani');

grant usage on foreign server ltoalink to lini;

**--create table and insert data in all users ani,vini,lini**

create table student

(

hscno smallint, name varchar(30),

dob date, aggr smallint , primary key(hscno)

);

CREATE TABLE COLLEGE

(

COLL\_CODE smallint PRIMARY KEY, NAME VARCHAR(50)

);

CREATE TABLE BRANCH

(

BCODE VARCHAR(3) PRIMARY KEY, BNAME VARCHAR(50)

);

create table coll\_branch

(

coll\_code smallint references college, bcode varchar(3) references branch,nos smallint,

primary key(coll\_code,bcode )

);

INSERT INTO STUDENT VALUES(12345,'VIJAY','13-OCT-85',200);

INSERT INTO STUDENT VALUES(12344,'SAM','12-OCT-85',198);

INSERT INTO STUDENT VALUES(12355,'SIM','13-OCT-85',197);

INSERT INTO STUDENT VALUES(12356,'JIM','12-MAR-86',198);

INSERT INTO COLLEGE VALUES('1','CEG');

INSERT INTO COLLEGE VALUES('2','ACTECH');

INSERT INTO COLLEGE VALUES('3','SAP');

INSERT INTO COLLEGE VALUES('4','MIT');

INSERT INTO BRANCH VALUES('CSE', 'COMPUTER SCIENCE AND ENGINEERING');

INSERT INTO BRANCH VALUES('IT','INFORMATION TECHNOLOGY');

INSERT INTO BRANCH VALUES('CH','CHEMICAL ENGINEERING');

INSERT INTO BRANCH VALUES('AR','ARCHITECTURE');

INSERT INTO COLL\_BRANCH VALUES(1,'CSE',120);

INSERT INTO COLL\_BRANCH VALUES(1,'IT',120);

INSERT INTO COLL\_BRANCH VALUES(4,'CSE',120);

INSERT INTO COLL\_BRANCH VALUES(4,'IT',120);

INSERT INTO COLL\_BRANCH VALUES(3,'AR',20);

INSERT INTO COLL\_BRANCH VALUES(2,'CH',30);

CREATE TABLE allotment (

rollno INT REFERENCES student(hscno) PRIMARY KEY,

coll\_code INT,

bcode VARCHAR(5),

FOREIGN KEY (coll\_code, bcode) REFERENCES coll\_branch(coll\_code, bcode));

**TRIGGERS**

**in ani :**

CREATE OR REPLACE FUNCTION allot()

RETURNS TRIGGER AS

$$

BEGIN

UPDATE COLL\_BRANCH SET NOS=NOS-1 WHERE COLL\_CODE=NEW.COLL\_CODE AND BCODE=NEW.BCODE;

PERFORM dblink\_exec('vinilink', 'UPDATE COLL\_BRANCH SET NOS=NOS-1 WHERE COLL\_CODE=''' || NEW.COLL\_CODE || ''' AND BCODE=''' || NEW.BCODE || '''');

PERFORM dblink\_exec('linilink', 'UPDATE COLL\_BRANCH SET NOS=NOS-1 WHERE COLL\_CODE=''' || NEW.COLL\_CODE || ''' AND BCODE=''' || NEW.BCODE || '''');

RETURN NEW;

END;

$$

LANGUAGE 'plpgsql';

CREATE TRIGGER AVAILABILITY

BEFORE INSERT ON ALLOTMENT

FOR EACH ROW

EXECUTE PROCEDURE allot();

**in vini:**

CREATE OR REPLACE FUNCTION insert\_ani()

RETURNS TRIGGER AS

$$

BEGIN

PERFORM dblink\_exec('vtoalink', 'INSERT INTO allotment(rollno,coll\_code, bcode) VALUES (''' || NEW.rollno || ''', ''' || NEW.coll\_code || ''', ''' || NEW.bcode || ''')');

RETURN NEW;

END;

$$

LANGUAGE 'plpgsql';

CREATE TRIGGER insert\_ani\_trigger

AFTER INSERT ON ALLOTMENT

FOR EACH ROW

EXECUTE PROCEDURE insert\_ani();

**in lini:**

CREATE OR REPLACE FUNCTION insert\_ani()

RETURNS TRIGGER AS

$$

BEGIN

PERFORM dblink\_exec('ltoalink', 'INSERT INTO allotment(rollno,coll\_code, bcode) VALUES (''' || NEW.rollno || ''', ''' || NEW.coll\_code || ''', ''' || NEW.bcode || ''')');

RETURN NEW;

END;

$$

LANGUAGE 'plpgsql';

CREATE TRIGGER insert\_ani\_trigger

AFTER INSERT ON ALLOTMENT

FOR EACH ROW

EXECUTE PROCEDURE insert\_ani();

**command:(In any Table)**

INSERT INTO ALLOTMENT VALUES (12345,1,'CSE');