Suman Paudel (33)

Assignment I

Lab 1:

Prepare Lab Sheet of MYSQL Statements for following.

1. Create a database named "Yourname_Roll_COMPANY" e.g.: Atiz_02_Company and then create following tables within the database. Specify proper primary keys and the needed constraints while defining the tables. Use appropriate data types for the attributes.

   a. Employee (SSN, Ename, Gender, Bdate, Address, Salary, Ono, Years_of_experience); whereOno is a foreign key referencing to the Office table. Set default value of salary to 0.00. The Ename should not be null. Set SSN to auto increment. The Ename and address should be varchar, Gender should be char(1), Bdate should be date type, Salary should be decimal type with two digits after decimal.Years_of_experience should be integer. Use Check constraint for gender as CHECK (Gender IN ('M', 'F'))

   b. Office (Onumber, Oname, Country); where Oname should not be NULL. Country should be varchar.

   c. Project (Pnumber, Pname, Plocation, Onumber); where Onumber is a foreign key referencing Office table. Create a constraint name fk_pro for the foreign key. Pname should be unique and should not be null. Both Pname and Plocations should be of type varchar(40).

   d. Works_on(ESSN, Pno); where ESSN references Employee SSN and Pno references to Pnumber from Project . Set cascade on update and cascade on delete to both

   e. Dependents(Did, Dname, Dage, SSN); where SSN is Foreign key referencing the employee. Set NULL on delete and on update to the foreign key. Add constraint age_constraint using CHECK(Dage<16).

Create a database named "Yourname_Roll_COMPANY" e.g.: Atiz_02_Company and then create following tables within the database. Specify proper primary keys and the needed constraints while defining the tables. Use appropriate data types for the attributes.

SQL Script:

```
DROP DATABASE IF EXISTS suman_33_company;
CREATE DATABASE suman_33_company;
\c suman_33_company;
```

Output:

```
postgres=# DROP DATABASE IF EXISTS suman_33_company;
NOTICE:  database "suman_33_company" does not exist, skipping
DROP DATABASE
postgres=# CREATE DATABASE suman_33_company;
CREATE DATABASE                                    To switch the database that I just created.
postgres=# \c suman_33_company;
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, bits: 256, compression: off)
You are now connected to database "suman_33_company" as user "postgres".
suman_33_company=#
```

1a. Employee (<u>SSN</u>, Ename, Gender, Bdate, Address, Salary, Ono, Years_of_experience); where Ono is a foreign key referencing to the Office table. Set default value of salary to 0.00. The Ename should not be null. Set SSN to auto increment. The Ename and address should be varchar, Gender should be char(1), Bdate should be date type, Salary should be decimal type with two digits after decimal.Years_of_experience should be integer. Use Check constraint for gender as CHECK (Gender IN ('M', 'F'))

Note Error was thrown when trying to create employee table which has "Ono" column being referenced to Office table which is currently not in the database. So, first I had to create Office table then I created Employee table.

Error:

```
suman_33_company=# DROP TABLE IF EXISTS Employee;
NOTICE:  table "employee" does not exist, skipping
DROP TABLE
suman_33_company=# CREATE TABLE employee (
suman_33_company(#    SSN INT PRIMARY KEY,
suman_33_company(#    Ename VARCHAR(100) NOT NULL,
suman_33_company(#    Gender CHAR(1) CHECK (Gender IN ('M', 'F')),
suman_33_company(#    Bdate Date,
suman_33_company(#    Address VARCHAR(100),
suman_33_company(#    SALARY NUMERIC(10, 2) DEFAULT 0.00,
suman_33_company(#    Ono INT,
suman_33_company(#    Years_of_experience INT,
suman_33_company(#    FOREIGN KEY(Ono) REFERENCES Office(Onumber)
suman_33_company(# );
ERROR:  relation "office" does not exist        error because office table is being referenced
suman_33_company=#                              which doesn't exists.
```

SQL Script:

```sql
DROP TABLE IF EXISTS Employee;
CREATE TABLE employee (
    SSN INT PRIMARY KEY,
    Ename VARCHAR(100) NOT NULL,
    Gender CHAR(1) CHECK (Gender IN ('M', 'F')),
    Bdate Date,
    Address VARCHAR(100),
    SALARY NUMERIC(10, 2) DEFAULT 0.00,
    Ono INT,
    Years_of_experience INT,
    FOREIGN KEY(Ono) REFERENCES Office(Onumber)
);
```

Output:

```
suman_33_company=# DROP TABLE IF EXISTS Employee;
NOTICE:  table "employee" does not exist, skipping
DROP TABLE
suman_33_company=# CREATE TABLE employee (
suman_33_company(#    SSN INT PRIMARY KEY,
suman_33_company(#    Ename VARCHAR(100) NOT NULL,
suman_33_company(#    Gender CHAR(1) CHECK (Gender IN ('M', 'F')),
suman_33_company(#    Bdate Date,
suman_33_company(#    Address VARCHAR(100),
suman_33_company(#    SALARY NUMERIC(10, 2) DEFAULT 0.00,
suman_33_company(#    Ono INT,
suman_33_company(#    Years_of_experience INT,
suman_33_company(#    FOREIGN KEY(Ono) REFERENCES Office(Onumber)
suman_33_company(# );
CREATE TABLE
suman_33_company=# |
```

1b. Office (Onumber, Oname, Country); where Oname should not be NULL. Country should be varchar.

SQL Script:

```sql
DROP TABLE IF EXISTS Office;
    CREATE TABLE Office(
    Onumber INT PRIMARY KEY,
    Oname VARCHAR(100) NOT NULL,
    Country VARCHAR(50)
);
```

Output:

```
suman_33_company=# DROP TABLE IF EXISTS Office;
NOTICE:  table "office" does not exist, skipping
DROP TABLE
suman_33_company=# CREATE TABLE Office(
suman_33_company(#    Onumber INT PRIMARY KEY,
suman_33_company(#    Oname VARCHAR(100) NOT NULL,
suman_33_company(#    Country VARCHAR(50)
suman_33_company(# );
CREATE TABLE
suman_33_company=#
```

1c: Project (Pnumber, Pname, Plocation, Onumber); where Onumber is a foreign key referencing Office table. Create a constraint name fk_pro for the foreign key. Pname should be unique and should not be null. Both Pname and Plocations should be of type varchar(40).

SQL Script:

```sql
DROP TABLE IF EXISTS Project;
CREATE TABLE Project (
Pnumber INT PRIMARY KEY,
Pname VARCHAR(40) UNIQUE NOT NULL,
Plocation VARCHAR(40),
Onumber INT,
CONSTRAINT fk_pro FOREIGN KEY (Onumber) REFERENCES Office(Onumber)
);
```

Output:

```
suman_33_company=# DROP TABLE IF EXISTS Project;
NOTICE:  table "project" does not exist, skipping
DROP TABLE
suman_33_company=# CREATE TABLE Project (
suman_33_company(#    Pnumber INT PRIMARY KEY,
suman_33_company(#    Pname VARCHAR(40) UNIQUE NOT NULL,
suman_33_company(#    Plocation VARCHAR(40),
suman_33_company(#    Onumber INT,
suman_33_company(#    CONSTRAINT fk_pro FOREIGN KEY (Onumber) REFERENCES Office(Onumber)
suman_33_company(# );
CREATE TABLE
suman_33_company=#
```

1d: Works_on(<u>ESSN</u>, <u>Pno</u>); where ESSN references Employee SSN and Pno references to Pnumber from Project . Set cascade on update and cascade on delete to both.

SQL Script:

```sql
DROP TABLE IF EXISTS Works_on;
CREATE TABLE Works_on (
ESSN INT,
Pno INT,
PRIMARY KEY (ESSN, Pno),
FOREIGN KEY (ESSN) REFERENCES Employee(SSN) ON UPDATE CASCADE ON DELETE
CASCADE,
FOREIGN KEY (Pno) REFERENCES Project(Pnumber) ON UPDATE CASCADE ON DELETE
CASCADE
);
```

Output:

```
suman_33_company=# DROP TABLE IF EXISTS Works_on;
NOTICE:  table "works_on" does not exist, skipping
DROP TABLE
suman_33_company=# CREATE TABLE Works_on (
suman_33_company(#   ESSN INT,
suman_33_company(#   Pno INT,
suman_33_company(#   PRIMARY KEY (ESSN, Pno),
suman_33_company(#   FOREIGN KEY (ESSN) REFERENCES Employee(SSN) ON UPDATE CASCADE ON DELETE CASCADE,
suman_33_company(#   FOREIGN KEY (Pno) REFERENCES Project(Pnumber) ON UPDATE CASCADE ON DELETE CASCADE
suman_33_company(# );
CREATE TABLE
suman_33_company=#
```

1e: Dependents(<u>Did,</u> Dname, Dage, SSN); where SSN is Foreign key referencing the employee. Set NULL on delete and on update to the foreign key. Add constraint age_constraint using CHECK(Dage<16).

SQL Script:

```sql
DROP TABLE IF EXISTS Dependent;
CREATE TABLE Dependent (
Did INT PRIMARY KEY,
Dname VARCHAR(100),
Dage INT,
CONSTRAINT age_constraint CHECK (Dage < 16),
SSN INT,
FOREIGN KEY (SSN) REFERENCES Employee(SSN) ON UPDATE SET NULL ON DELETE SET
NULL
);
```

Output:

```
suman_33_company=# CREATE TABLE Dependent (
suman_33_company(#   Did INT PRIMARY KEY,
suman_33_company(#   Dname VARCHAR(100),
suman_33_company(#   Dage INT,
suman_33_company(#   CONSTRAINT age_constraint CHECK (Dage < 16),
suman_33_company(#   SSN INT,
suman_33_company(#     FOREIGN KEY (SSN) REFERENCES Employee(SSN) ON UPDATE SET NULL ON DELETE SET NULL
suman_33_company(# );
CREATE TABLE
suman_33_company=#
```

Outputs After Tables Creation.

```
suman_33_company=# \dt  ←                          lists all of tables in the database
              List of relations
  Schema  |    Name     | Type  |  Owner
----------+-------------+-------+-----------
  public  | dependent   | table | postgres
  public  | employee    | table | postgres
  public  | office      | table | postgres
  public  | project     | table | postgres
  public  | works_on    | table | postgres
(5 rows)


suman_33_company=# |
```

Q2: Alter table Dependent and add an attribute Drelation of type Char(50.

SQL Script:

```
ALTER TABLE dependent ADD COLUMN drelation CHAR(50);
```

Output:

```
suman_33_company=# alter table dependent add column Drelation char(50);
ALTER TABLE
suman_33_company=# SELECT column_name, data_type FROM information_schema.columns WHERE table_name = 'dependent';
 column_name |      data_type
-------------+--------------------
 did         | integer
 dname       | character varying
 dage        | integer
 ssn         | integer
 drelation   | character
(5 rows)

suman_33_company=# |
```

Q3: Alter table Dependent and modify the attribute Drelation of type Char(50) to Varchar(50)

SQL Script:

```
ALTER TABLE dependent ALTER COLUMN drelation TYPE varchar(50);
```

Output:

```
suman_33_company=# alter table dependent alter column drelation type varchar(50);
ALTER TABLE
suman_33_company=# SELECT column_name, data_type FROM information_schema.columns WHERE table_name = 'dependent';
 column_name |      data_type
-------------+--------------------
 did         | integer
 dname       | character varying
 dage        | integer
 ssn         | integer
 drelation   | character varying
(5 rows)

suman_33_company=# |
```

Q4: Insert at least five tuples into the tables. (Illustrate insertion of single tuple and multiple tuples both). During insertion insert following as well.

There should be one record in the Employee table having Ename "Your name" i. e. Deric and SSN "Your roll number" e.g. 6.

There should be one record in the Project table having Pname ="Your name_ProjMDS" and Pnumber = 2*Your Roll number.

One of the tuple in Office table should have office name "Yourname_Office_Roll" i.e. Deric_Office_06. Similarly one of the tuple in employee should have salary 30000.

In addition, there should be one tuple in office table having office name Yourname_Ncell_Roll.

In the dependents table insert the rows with Dname and Drelation having values from your family. For example, Deric has his elder brother and mother as his dependents. So the table will have records with values Dname=Denish and Drelation=Brother and Dname=Gayatri and Drelation=Mother. Take assumptions based on your family members while inserting the values.

For Office Table

Before Insertion:

```
suman_33_company=# select * from office;
 onumber | oname | country
---------+-------+---------
(0 rows)

suman_33_company=#
```

Single Insertion SQL Script:

```sql
INSERT INTO Office (Onumber, Oname, Country) VALUES (1, 'Suman_Office_33',
'Nepal');
```

Output:

```
suman_33_company=# INSERT INTO Office (Onumber, Oname, Country) VALUES (1, 'Suman_Office_33', 'Nepal');
INSERT 0 1
suman_33_company=#                              single insertion into table
```

Multiple Insertion SQL Script:

```sql
INSERT INTO Office (Onumber, Oname, Country)
VALUES
(2, 'Suman_Ncell_33', 'Nepal'),
(3, 'Prabhat Ale', 'USA'),
(4, 'Anish Thapaliya', 'India'),
(5, 'Gaurav Pandey', 'Japan');
```

Output:

```
suman_33_company=# INSERT INTO Office (Onumber, Oname, Country)
suman_33_company=# VALUES
suman_33_company=#    (2, 'Suman_Ncell_33', 'Nepal'),
suman_33_company=#    (3, 'Prabhat Ale', 'USA'),
suman_33_company=#    (4, 'Anish Thapaliya', 'India'),
suman_33_company=#    (5, 'Gaurav Pandey', 'Japan');
INSERT 0 4
suman_33_company=# |                          multi insertions
```

After Insertion into Office Table.

```
suman_33_company=# select * from office;
 onumber |       oname        | country
---------+--------------------+---------
       1 | Suman_Office_33    | Nepal
       2 | Suman_Ncell_33     | Nepal
       3 | Prabhat Ale        | USA
       4 | Anish Thapaliya    | India
       5 | Gaurav Pandey      | Japan
(5 rows)

suman_33_company=# |
```

For Employee Table

Before Insertion:

```
suman_33_company=# select * from employee;
 ssn | ename | gender | bdate | address | salary | ono | years_of_experience
-----+-------+--------+-------+---------+--------+-----+---------------------
(0 rows)

suman_33_company=# |
```

Single Insertion SQL Script:

```
-- for employee table
INSERT INTO employee (SSN, Ename, Gender, Bdate, Address, SALARY, Ono,
Years_of_experience)
VALUES (33, 'Suman Paudel', 'M', '1997-10-22', 'Kathmanu, Nepal', 30000.00,
1, 3);
```

Output:

```
suman_33_company=# INSERT INTO employee (SSN, Ename, Gender, Bdate, Address, SALARY, Ono, Years_of_experience)
suman_33_company=# VALUES (33, 'Suman Paudel', 'M', '1997-10-22', 'Kathmanu, Nepal', 30000.00, 1, 3);
INSERT 0 1
suman_33_company=# select * from employee;
 ssn |    ename     | gender |   bdate    |    address      |  salary  | ono | years_of_experience
-----+--------------+--------+------------+-----------------+----------+-----+---------------------
  33 | Suman Paudel | M      | 1997-10-22 | Kathmanu, Nepal | 30000.00 |   1 |                   3
(1 row)

suman_33_company=#                                                              single insertion
```

Multi Insertion SQL Script:

```
INSERT INTO employee (SSN, Ename, Gender, Bdate, Address, SALARY, Ono,
Years_of_experience)
VALUES
(1, 'Rekha Thapa', 'F', '1992-03-22', 'Kathmandu, Nepal', 55000.00, 2, 5),
(2, 'KP Oli', 'M', '1978-11-08', 'Bhaktapur, Nepal', 72000.00, 3, 12),
(3, 'Puspa Kamal Dahal Pracanda', 'M', '1990-09-01', 'Lalitpur, Nepal',
48000.00, 4, 3),
(4, 'Rabi Lamichane', 'M', '1983-04-30', 'Chitwan, Nepal', 60000.00, 4, 7);
```

Output:

```
suman_33_company=#  INSERT INTO employee (SSN, Ename, Gender, Bdate, Address, SALARY, Ono, Years_of_experience)
suman_33_company=# VALUES
suman_33_company=#    (1, 'Rekha Thapa', 'F', '1992-03-22', 'Kathmandu, Nepal', 55000.00, 2, 5),
suman_33_company=#    (2, 'KP Oli', 'M', '1978-11-08', 'Bhaktapur, Nepal', 72000.00, 3, 12),
suman_33_company=#    (3, 'Puspa Kamal Dahal Pracanda', 'M', '1990-09-01', 'Lalitpur, Nepal', 48000.00, 4, 3),
suman_33_company=#    (4, 'Rabi Lamichane', 'M', '1983-04-30', 'Chitwan, Nepal', 60000.00, 4, 7);
INSERT 0 4
suman_33_company=# select * from employee;
 ssn |            ename           | gender |   bdate    |    address       |  salary  | ono | years_of_experience
-----+----------------------------+--------+------------+------------------+----------+-----+---------------------
  33 | Suman Paudel               | M      | 1997-10-22 | Kathmanu, Nepal  | 30000.00 |   1 |                   3
   1 | Rekha Thapa                | F      | 1992-03-22 | Kathmandu, Nepal | 55000.00 |   2 |                   5
   2 | KP Oli                     | M      | 1978-11-08 | Bhaktapur, Nepal | 72000.00 |   3 |                  12
   3 | Puspa Kamal Dahal Pracanda | M      | 1990-09-01 | Lalitpur, Nepal  | 48000.00 |   4 |                   3
   4 | Rabi Lamichane             | M      | 1983-04-30 | Chitwan, Nepal   | 60000.00 |   4 |                   7
(5 rows)

suman_33_company=#
```

After Insertion into Employee Table.

```
suman_33_company=# select * from employee;
 ssn |            ename           | gender |   bdate    |    address       |  salary  | ono | years_of_experience
-----+----------------------------+--------+------------+------------------+----------+-----+---------------------
  33 | Suman Paudel               | M      | 1997-10-22 | Kathmanu, Nepal  | 30000.00 |   1 |                   3
   1 | Rekha Thapa                | F      | 1992-03-22 | Kathmandu, Nepal | 55000.00 |   2 |                   5
   2 | KP Oli                     | M      | 1978-11-08 | Bhaktapur, Nepal | 72000.00 |   3 |                  12
   3 | Puspa Kamal Dahal Pracanda | M      | 1990-09-01 | Lalitpur, Nepal  | 48000.00 |   4 |                   3
   4 | Rabi Lamichane             | M      | 1983-04-30 | Chitwan, Nepal   | 60000.00 |   4 |                   7
(5 rows)

suman_33_company=#
```

For Project Table

Before Insertion:

```
suman_33_company=# select * from project;
 pnumber | pname | plocation | onumber
---------+-------+-----------+---------
(0 rows)

suman_33_company=#
```

Single Insertion SQL Script:

```
INSERT INTO Project (Pnumber, Pname, Plocation, Onumber)
VALUES (66, 'Suman_ProjMDS', 'Kathmandu', 1);
```

Output:

```
suman_33_company=#  INSERT INTO Project (Pnumber, Pname, Plocation, Onumber)
suman_33_company-# VALUES (66, 'Suman_ProjMDS', 'Kathmandu', 1);
INSERT 0 1
suman_33_company=# select * from project;                    single insert
 pnumber |     pname     | plocation | onumber
---------+---------------+-----------+---------
      66 | Suman_ProjMDS | Kathmandu |       1
(1 row)
```

Multiple Insertion SQL Script:

```
INSERT INTO Project (Pnumber, Pname, Plocation, Onumber)
VALUES
(2, 'Project Kathmandu', 'Kathmandu', 2),
(3, 'Project Butwal', 'Butwal', 3),
(4, 'Project Chitwan', 'Chitwan', 4),
(5, 'Project Dhangadi', 'Dhangadi', 5);
```

Output:

```
suman_33_company=#  INSERT INTO Project (Pnumber, Pname, Plocation, Onumber)
suman_33_company-#  VALUES
suman_33_company-#  (2, 'Project Kathmandu', 'Kathmandu', 2),
suman_33_company-#  (3, 'Project Butwal', 'Butwal', 3),
suman_33_company-#  (4, 'Project Chitwan', 'Chitwan', 4),
suman_33_company-#  (5, 'Project Dhangadi', 'Dhangadi', 5);
INSERT 0 4
suman_33_company=# select * from project;
 pnumber |       pname       | plocation | onumber
---------+-------------------+-----------+---------
      66 | Suman_ProjMDS     | Kathmandu |       1
       2 | Project Kathmandu | Kathmandu |       2
       3 | Project Butwal    | Butwal    |       3
       4 | Project Chitwan   | Chitwan   |       4
       5 | Project Dhangadi  | Dhangadi  |       5    multi insert
(5 rows)

suman_33_company=#
```

After Insertion into Project Table.

```
suman_33_company=# select * from project;
 pnumber |        pname        | proj_location | onumber
---------+---------------------+---------------+---------
      66 | Suman_ProjMDS       | Kathmandu     |       1
       2 | Project Kathmandu   | Kathmandu     |       2
       3 | Project Butwal      | Butwal        |       3
       4 | Project Chitwan     | Chitwan       |       4
       5 | Project Dhangadi    | Dhangadi      |       5
(5 rows)

suman_33_company=#
```

For Works_on Table

Before Insertion:

```
suman_33_company=# select * from works_on;
 essn | pno
------+-----
(0 rows)

suman_33_company=#
```

Single Insertion SQL Script:

```sql
-- for employee table
INSERT INTO Works_on (ESSN, Pno)
VALUES (33, 66);
```

Output:

```
suman_33_company=# INSERT INTO Works_on (ESSN, Pno)
suman_33_company-# VALUES (33, 66);
INSERT 0 1
suman_33_company=# select * from works_on;
 essn | pno
------+-----
   33 |  66                              single insertion
(1 row)

suman_33_company=#
```

Multiple Insertion SQL Script:

```
INSERT INTO Works_on (ESSN, Pno)
VALUES
(1, 2),
(2, 3),
(3, 4),
(4, 5);
```

Output:

```
suman_33_company=#  INSERT INTO Works_on (ESSN, Pno)
suman_33_company-# VALUES
suman_33_company-#   (1, 2),
suman_33_company-#   (2, 3),
suman_33_company-#   (3, 4),
suman_33_company-#   (4, 5);
INSERT 0 4
suman_33_company=# select * from works_on;
 essn | pno
------+------
   33 |  66
    1 |   2          multiple insertion
    2 |   3
    3 |   4
    4 |   5
(5 rows)

suman_33_company=# |
```

After Insertion into Works_on Table.

```
suman_33_company=# select * from works_on;
 essn | pno
------+------
   33 |  66
    1 |   2
    2 |   3
    3 |   4
    4 |   5
(5 rows)

suman_33_company=# |
```

For Dependent Table

Before Insertion:

```
suman_33_company=# select * from dependent;
 did | dname | dage | ssn
-----+-------+------+-----
(0 rows)

suman_33_company=# |
```

Single Insertion SQL Script:

```sql
-- for employee table
INSERT INTO Dependent (Did, Dname, Dage, SSN, Drelation)
VALUES (1, 'Pushpa Panta Jaisi', 13, 33, 'Mother');
```

Output:

```
suman_33_company=# INSERT INTO Dependent (Did, Dname, Dage, SSN, Drelation)
VALUES (1, 'Pushpa Panta Jaisi', 13, 33, 'Mother');
INSERT 0 1
suman_33_company=# select * from dependent
suman_33_company-# ;
 did |       dname        | dage | ssn |               drelation
-----+--------------------+------+-----+--------------------------------------
   1 | Pushpa Panta Jaisi |   13 |  33 | Mother
(1 row)
                                                            single insertion

suman_33_company=# |
```

Multiple Insertion SQL Script:

```sql
INSERT INTO Dependent (Did, Dname, Dage, SSN, Drelation)
VALUES (2, 'Krishna Prasad Jaishi', 13, 33, 'Father'),
(3, 'Silu Poudel', 12, 33, 'Sister'),
(4, 'Sujita Sharma', 9, 33, 'Cousin Sister'),
(5, 'Prabin Adhikari', 14, 33, 'Cousin Brother');
```

Output:

```
suman_33_company=# INSERT INTO Dependent (Did, Dname, Dage, SSN, Drelation)
suman_33_company-# VALUES (2, 'Krishna Prasad Jaishi', 13, 33, 'Father'),
suman_33_company-#     (3, 'Silu Poudel', 12, 33, 'Sister'),
suman_33_company-#     (4, 'Sujita Sharma', 9, 33, 'Cousin Sister'),
suman_33_company-#     (5, 'Prabin Adhikari', 14, 33, 'Cousin Brother');
INSERT 0 4
suman_33_company=# select * from dependent;
 did |         dname         | dage | ssn |            drelation
-----+-----------------------+------+-----+-----------------------------
   1 | Pushpa Panta Jaisi    |   13 |  33 | Mother
   2 | Krishna Prasad Jaishi |   13 |  33 | Father
   3 | Silu Poudel           |   12 |  33 | Sister
   4 | Sujita Sharma         |    9 |  33 | Cousin Sister
   5 | Prabin Adhikari       |   14 |  33 | Cousin Brother
(5 rows)                                              multi insertion

suman_33_company=# |
```

After Insertion into Dependent Table.

```
suman_33_company=# select * from dependent;
 did |          dname          | dage | ssn |               drelation
-----+-------------------------+------+-----+--------------------------------------
   1 | Pushpa Panta Jaisi      |   13 |  33 | Mother
   2 | Krishna Prasad Jaishi   |   13 |  33 | Father
   3 | Silu Poudel             |   12 |  33 | Sister
   4 | Sujita Sharma           |    9 |  33 | Cousin Sister
   5 | Prabin Adhikari         |   14 |  33 | Cousin Brother
(5 rows)

suman_33_company=#
```

5. Update the name of office having office name "Yourname_Ncell_Roll" to "Yourname_Ntc_Roll".

SQL Script:

```
UPDATE office SET oname = 'Suman_NTC_33' WHERE oname = 'Suman_Ncell_33';
```

Output:

```
suman_33_company=# select * from office;
 onumber |      oname       | country
---------+------------------+---------
       1 | Suman_Office_33  | Nepal
       2 | Suman_Ncell_33   | Nepal
       3 | Prabhat Ale      | USA
       4 | Anish Thapaliya  | India
       5 | Gaurav Pandey    | Japan
(5 rows)

suman_33_company=# update office set oname = 'Suman_NTC_33' where oname = 'Suman_Ncell_33';
UPDATE 1
suman_33_company=# select * from office;
 onumber |      oname       | country
---------+------------------+---------
       1 | Suman_Office_33  | Nepal
       3 | Prabhat Ale      | USA
       4 | Anish Thapaliya  | India
       5 | Gaurav Pandey    | Japan
       2 | Suman_NTC_33     | Nepal
(5 rows)

suman_33_company=#
```

6. Delete those employees whose SSN is 1.

SQL Script:

```
DELETE FROM employee WHERE ssn = 1;
```

Output

7. Alter table Project to rename the attribute in Plcoation to Proj_location

SQL Script:

```
ALTER TABLE project RENAME plocation to Proj_location;
```

Output:

8. Select tuples from all of the tables individually.

Employee Table

SQL Script:

```
SELECT * FROM employee;
```

Output:

```
suman_33_company=# select * from employee;
 ssn |            ename             | gender |   bdate    |    address       |  salary  | ono | years_of_experience
-----+-----------------------------+--------+------------+------------------+----------+-----+---------------------
  33 | Suman Paudel                | M      | 1997-10-22 | Kathmanu, Nepal  | 30000.00 |   1 |                   3
   2 | KP Oli                      | M      | 1978-11-08 | Bhaktapur, Nepal | 72000.00 |   3 |                  12
   3 | Puspa Kamal Dahal Pracanda  | M      | 1990-09-01 | Lalitpur, Nepal  | 48000.00 |   4 |                   3
   4 | Rabi Lamichane              | M      | 1983-04-30 | Chitwan, Nepal   | 60000.00 |   4 |                   7
(4 rows)

suman_33_company=#
```

Office Table

SQL Script:

```
SELECT * FROM office;
```

Output:

```
suman_33_company=# select * from office;
 onumber |      oname      | country
---------+-----------------+---------
       1 | Suman_Office_33 | Nepal
       2 | Suman_Ncell_33  | Nepal
       3 | Prabhat Ale     | USA
       4 | Anish Thapaliya | India
       5 | Gaurav Pandey   | Japan
(5 rows)

suman_33_company=#
```

Project Table

SQL Script:

```sql
SELECT * FROM project;
```

Output:

```
suman_33_company=# select * from project;
 pnumber |        pname        | proj_location | onumber
---------+---------------------+---------------+---------
      66 | Suman_ProjMDS       | Kathmandu     |       1
       2 | Project Kathmandu   | Kathmandu     |       2
       3 | Project Butwal      | Butwal        |       3
       4 | Project Chitwan     | Chitwan       |       4
       5 | Project Dhangadi    | Dhangadi      |       5
(5 rows)

suman_33_company=#
```

Works_on Table

SQL Script:

```sql
SELECT * FROM workson;
```

Output:

```
suman_33_company=# select * from works_on;
 essn | pno
------+-----
   33 |  66
    1 |   2
    2 |   3
    3 |   4
    4 |   5
(5 rows)

suman_33_company=#
```

Dependents Table

SQL Script:

```
SELECT * FROM dependents;
```

Output:

```
suman_33_company=# select * from dependent;
 did |         dname         | dage | ssn |        drelation
-----+-----------------------+------+-----+------------------
   1 | Pushpa Panta Jaisi    |   13 |  33 | Mother
   2 | Krishna Prasad Jaishi |   13 |  33 | Father
   3 | Silu Poudel           |   12 |  33 | Sister
   4 | Sujita Sharma         |    9 |  33 | Cousin Sister
   5 | Prabin Adhikari       |   14 |  33 | Cousin Brother
(5 rows)

suman_33_company=#
```

All Tables at once.

```
suman_33_company=#
suman_33_company=# select * from office;
 onumber |     oname        | country
---------+------------------+---------
       1 | Suman_Office_33  | Nepal
       3 | Prabhat Ale      | USA
       4 | Anish Thapaliya  | India
       5 | Gaurav Pandey    | Japan
       2 | Suman_NTC_33     | Nepal
(5 rows)

suman_33_company=# select * from employee;
 ssn |          ename          | gender |   bdate    |    address       |  salary  | ono | years_of_experience
-----+-------------------------+--------+------------+------------------+----------+-----+--------------------
  33 | Suman Paudel            | M      | 1997-10-22 | Kathmanu, Nepal  | 30000.00 |   1 |                   3
   2 | KP Oli                  | M      | 1978-11-08 | Bhaktapur, Nepal | 72000.00 |   3 |                  12
   3 | Puspa Kamal Dahal Pracanda | M   | 1990-09-01 | Lalitpur, Nepal  | 48000.00 |   4 |                   3
   4 | Rabi Lamichane          | M      | 1983-04-30 | Chitwan, Nepal   | 60000.00 |   4 |                   7
(4 rows)

suman_33_company=# select * from project;
 pnumber |      pname       | proj_location | onumber
---------+------------------+---------------+--------
      66 | Suman_ProjMDS    | Kathmandu     |       1
       2 | Project Kathmandu| Kathmandu     |       2
       3 | Project Butwal   | Butwal        |       3
       4 | Project Chitwan  | Chitwan       |       4
       5 | Project Dhangadi | Dhangadi      |       5
(5 rows)

suman_33_company=# select * from works_on;
 essn | pno
------+-----
   33 |  66
    2 |   3
    3 |   4
    4 |   5
(4 rows)

suman_33_company=# select * from dependent;
 did |         dname         | dage | ssn |        drelation
-----+-----------------------+------+-----+------------------
   1 | Pushpa Panta Jaisi    |   13 |  33 | Mother
   2 | Krishna Prasad Jaishi |   13 |  33 | Father
   3 | Silu Poudel           |   12 |  33 | Sister
   4 | Sujita Sharma         |    9 |  33 | Cousin Sister
   5 | Prabin Adhikari       |   14 |  33 | Cousin Brother
(5 rows)

suman_33_company=#
```

9. Drop the table Works_on. Make sure to export your database before you drop it so that you can recover.

Exported the database before dropping tables and database.

```
→  suman   pg_dump -U postgres suman_33_company -h localhost > "backup.sql"
Password:
→  suman  |
```

SQL Script

```
DROP TABLE works_on;
```

Output:

```
suman_33_company=# select * from works_on;
 essn | pno
------+------
   33 |   66
    2 |    3
    3 |    4
    4 |    5
(4 rows)

suman_33_company=# drop table works_on;
DROP TABLE
suman_33_company=# select * from works_on;
ERROR:  relation "works_on" does not exist
LINE 1: select * from works_on;
                      ^
suman_33_company=# |
```

10. Drop the constraint age_constraint from dependent table

SQL Script:

```
ALTER TABLE dependent DROP CONSTRAINT age_constraint;
```

Output:

```
suman_33_company=# \d dependent
                    Table "public.dependent"
   Column    |          Type          | Collation | Nullable | Default
-------------+------------------------+-----------+----------+---------
 did         | integer                |           | not null |
 dname       | character varying(100) |           |          |
 dage        | integer                |           |          |
 ssn         | integer                |           |          |
 drelation   | character varying(50)  |           |          |
Indexes:
    "dependent_pkey" PRIMARY KEY, btree (did)
Check constraints:
    "age_constraint" CHECK (dage < 16)
Foreign-key constraints:
    "dependent_ssn_fkey" FOREIGN KEY (ssn) REFERENCES employee(ssn) ON UPDATE SET NULL ON DELETE SET NULL

suman_33_company=# alter table dependent drop constraint age_constraint;
ALTER TABLE
suman_33_company=# \d dependent
                    Table "public.dependent"
   Column    |          Type          | Collation | Nullable | Default
-------------+------------------------+-----------+----------+---------
 did         | integer                |           | not null |
 dname       | character varying(100) |           |          |
 dage        | integer                |           |          |
 ssn         | integer                |           |          |
 drelation   | character varying(50)  |           |          |
Indexes:
    "dependent_pkey" PRIMARY KEY, btree (did)
Foreign-key constraints:
    "dependent_ssn_fkey" FOREIGN KEY (ssn) REFERENCES employee(ssn) ON UPDATE SET NULL ON DELETE SET NULL

suman_33_company=#
```

11. Drop the database COMPANY. Make sure to export your database before you drop it so that you can recover.

SQL Script:

```
DROP DATABASE suman_33_company;
```

Output:

```
suman_33_company=# \c postgres
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, bits: 256, compression: off)
You are now connected to database "postgres" as user "postgres".
postgres=# drop database suman_33_company;
DROP DATABASE
postgres=# \c suman_33_company;
FATAL:  database "suman_33_company" does not exist
Previous connection kept
postgres=#
```