Lab Manual: Setting primary key and Foreign key:

1. At the time of table creation:

```
CREATE TABLE TableName(
   Column1 datatype(size),
   Column2 datatype(size),
   Column3 datatype(size),
   Primary key (Column1)
   );
```

-
  ```
  CREATE TABLE Persons (
      ID int NOT NULL,
      LastName varchar(255) NOT NULL,
      FirstName varchar(255),
      Age int,
      PRIMARY KEY (ID)
  );
  ```

2. After creating the table:

```
ALTER TABLE tableName ADD PRIMARY KEY (Column Name);
```

- `ALTER TABLE Persons add primary key(ID);`

3. For adding more than one primary key:

   I. We can normally add primary key by using CREATE/ALTER command like

-
  ```
  CREATE TABLE Persons (
      ID int NOT NULL,
      LastName varchar(255) NOT NULL,
      FirstName varchar(255),
      Age int,
      PRIMARY KEY (ID, LastName)
  );
  ```

   II. Or by using the following constraint:

-
  ```
  CREATE TABLE Persons (
      ID int NOT NULL,
      LastName varchar(255) NOT NULL,
      FirstName varchar(255),
      Age int,
     CONSTRAINT PK_Person PRIMARY KEY (ID,LastName)
  );
  ```

Note: In the example above there is only ONE PRIMARY KEY (PK_Person). However, the VALUE of the primary key is made up of TWO COLUMNS (ID + LastName).

4. Droping primary key:

```
Alter table tableName drop primary key;
```

- `Alter table persons drop primary key;`

5. Adding foreign key:

   I. While creating a table

-
  ```
  CREATE TABLE Orders (
      OrderID int NOT NULL,
      OrderNumber int NOT NULL,
      PersonID int,
      PRIMARY KEY (OrderID),
      FOREIGN KEY (PersonID) REFERENCES Persons(ID)
  );
  ```

But here without creating Foreign key constraints, we can not delete/drop the foreign key. So it’s good practice to use foreign key with constraints.

Or, by creating constraints:

-
  ```
  CREATE TABLE Orders (
      OrderIDint NOT NULL,
      OrderNumberint NOT NULL,
      PersonIDint,
      PRIMARY KEY (OrderID),
      CONSTRAINT FK_PersonOrder FOREIGN KEY (PersonID)
      REFERENCES Persons(ID)
  );
  ```

II. After creating

-
  ```
  ALTER TABLE Orders
  ADD FOREIGN KEY (PersonID) REFERENCES Persons(PersonID);
  ```

  Or, by creating constraints

-
  ```
  ALTER TABLE Orders
  ADD CONSTRAINT FK_PersonOrder
  FOREIGN KEY (PersonID) REFERENCES Persons(PersonID);
  ```

6. Dropping foreign keys:

   ```
   ALTER TABLE Orders
   DROP FOREIGN KEY FK_PersonOrder;
   ```

Referencing with DELETE and UPDATE rule:

- **(DELETE and UPDATE reference is done during the time of foreign key declaration and for proper referencing we need to set constraints while declaring the primary and foreign key constraint.)**

7. Delete and Update rule: There are four rules
   1. CASCADE: Delete or update the row from the parent table, and automatically delete or update the matching rows in the child table
   2. SET NULL: Delete or update the row from the parent table, and set the foreign key column or columns in the child table to NULL.
   3. RESTRICT: Rejects the delete or update operation for the parent table.
   4. NO ACTION: Equivalent to RESTRICT

Let’s say we have two tables:

-
  ```
  CREATE TABLE Persons (
      ID int NOT NULL,
      LastName varchar(255) NOT NULL,
      FirstName varchar(255),
      Age int,
      PRIMARY KEY (ID)
  );
  ```

-
  ```
  CREATE TABLE Orders (
      OrderIDint NOT NULL,
      OrderNumberint NOT NULL,
      PersonIDint,
      PRIMARY KEY (OrderID),
      FOREIGN KEY (PersonID) REFERENCES Persons(ID)
      //Here we have to write the delete or update rule
      ON DELETE *
      ON UPDATE *
  //Here the * can be replaced by any of the four rules
  );
  ```

8. After table creation:

   -
     ```
     alter table orders add constraint fk_pi foreign key(PersonId)
     references persons(ID) on delete set null on update cascade
     ```

```
//Code for the above 7 and 8

CREATE TABLE Persons (

    ID int NOT NULL,

    LastName varchar(255) NOT NULL,

    FirstName varchar(255),

    Age int,

    PRIMARY KEY (ID)

);


CREATE TABLE Orders (

    OrderID int NOT NULL,

    OrderNumber int NOT NULL,

    PersonID int,

    PRIMARY KEY (OrderID),

    FOREIGN KEY (PersonID) REFERENCES Persons(ID)

    ON DELETE cascade

    ON UPDATE cascade );
```