LAB 01

1.

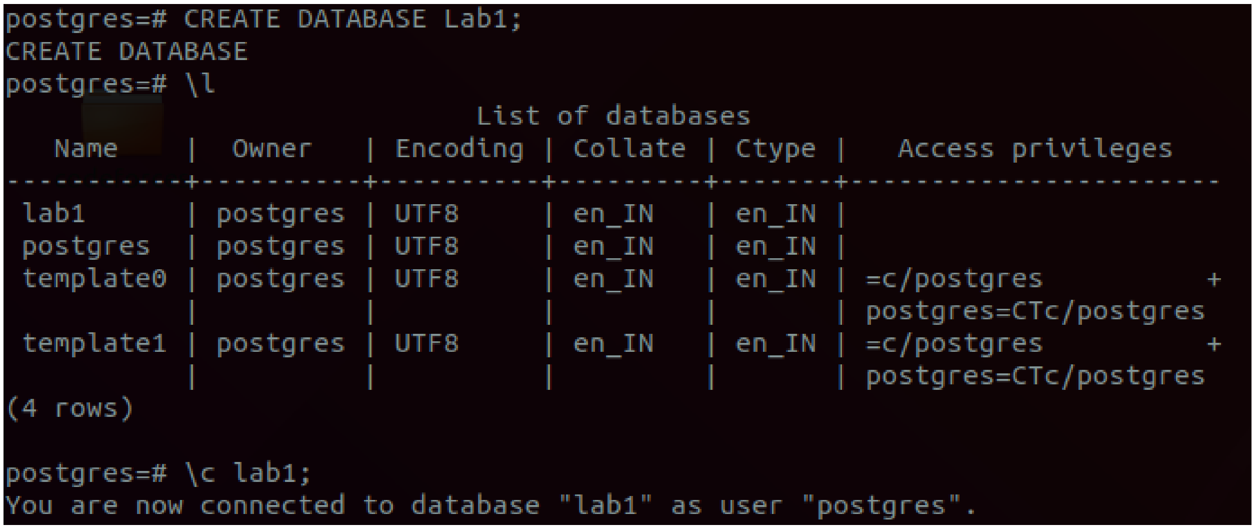
Student(RollNo, Name, CGPA,DeptID)

Department(DeptID, DeptName, NameofHOD)

Choose a unique column as a Primary Key , Foreign Key, and also other constraints as per your understanding. Create tables for the above schema[RollNo and DeptID are unique]

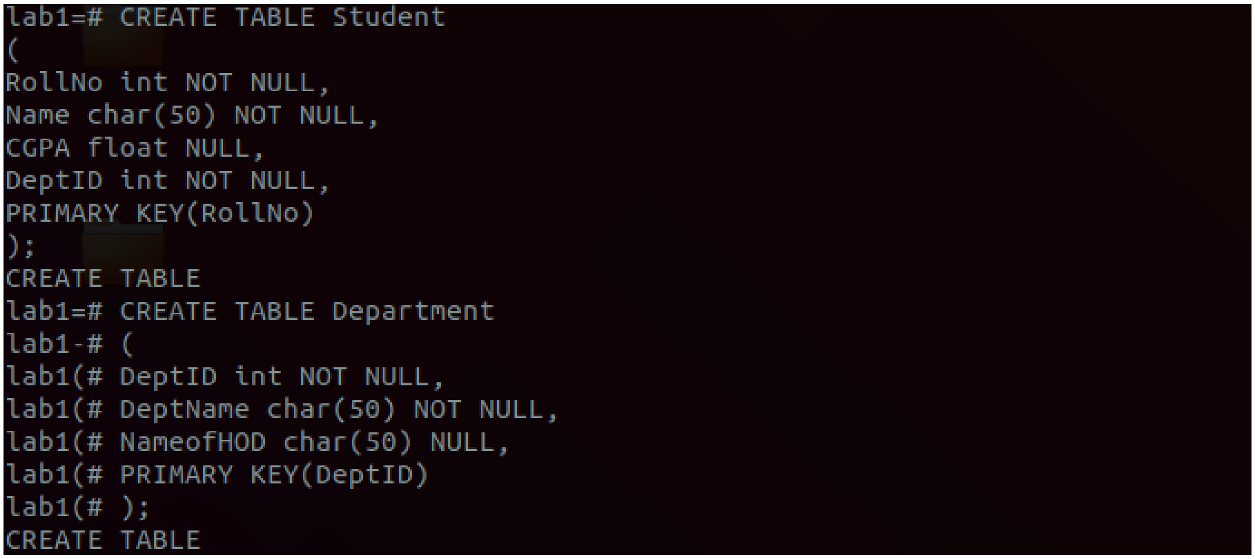
–

(creating database first, and connecting it)



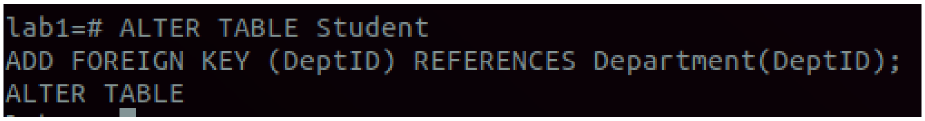
```
postgres=# CREATE DATABASE Lab1;
CREATE DATABASE
postgres=# \l
                             List of databases
   Name    |  Owner   | Encoding | Collate | Ctype |   Access privileges
-----------+----------+----------+---------+-------+-----------------------
 lab1      | postgres | UTF8     | en_IN   | en_IN |
 postgres  | postgres | UTF8     | en_IN   | en_IN |
 template0 | postgres | UTF8     | en_IN   | en_IN | =c/postgres          +
           |          |          |         |       | postgres=CTc/postgres
 template1 | postgres | UTF8     | en_IN   | en_IN | =c/postgres          +
           |          |          |         |       | postgres=CTc/postgres
(4 rows)

postgres=# \c lab1;
You are now connected to database "lab1" as user "postgres".
```

For the table Student, RollNo was set to be the PRIMARY KEY because that will uniquely identify the rows in the table. For the table Department, DeptID was set to be the PRIMARY KEY.



```
lab1=# CREATE TABLE Student
(
RollNo int NOT NULL,
Name char(50) NOT NULL,
CGPA float NULL,
DeptID int NOT NULL,
PRIMARY KEY(RollNo)
);
CREATE TABLE
lab1=# CREATE TABLE Department
lab1-# (
lab1(# DeptID int NOT NULL,
lab1(# DeptName char(50) NOT NULL,
lab1(# NameofHOD char(50) NULL,
lab1(# PRIMARY KEY(DeptID)
lab1(# );
CREATE TABLE
```

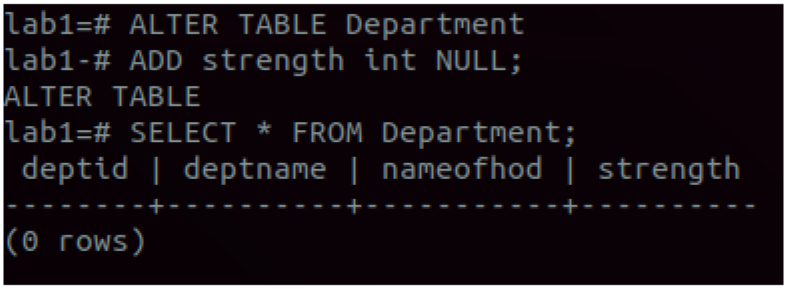
FOREIGN KEY of Student was set to the column DeptID, which refers to the column DeptID of the table Department. FOREIGN KEY was set later (not during CREATE TABLE) because of the reference (another way to do this can be to create the Department table before).



```
lab1=# ALTER TABLE Student
ADD FOREIGN KEY (DeptID) REFERENCES Department(DeptID);
ALTER TABLE
```

— —

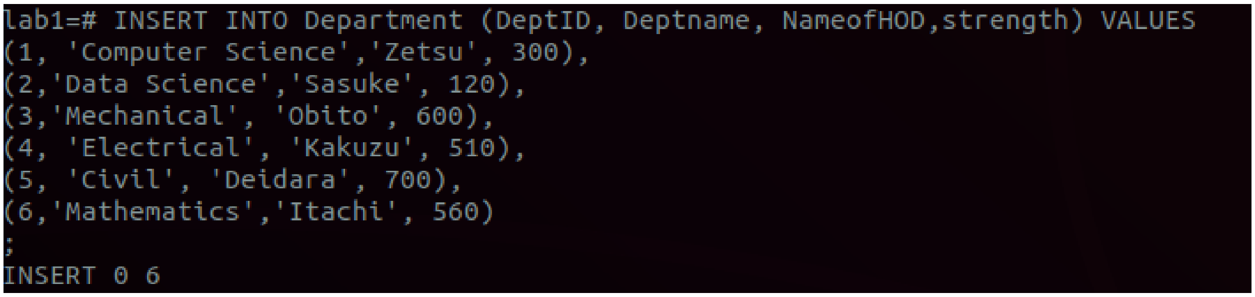
2. Change the Department table as,
Department(DeptID, Deptname, NameofHOD,strength)
–
Adding the new column using the ALTER TABLE command.



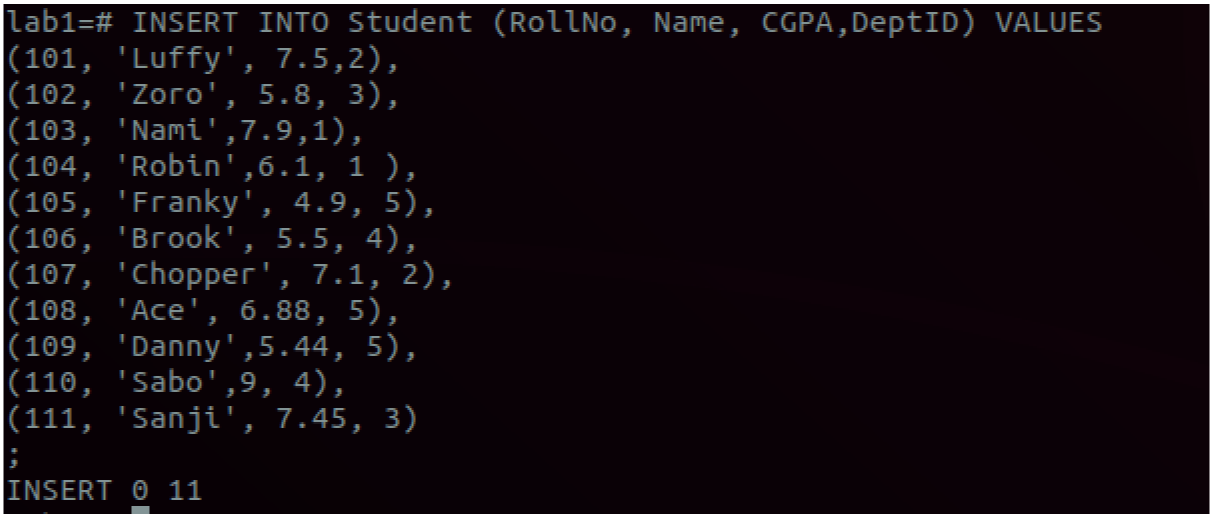
```
lab1=# ALTER TABLE Department
lab1-# ADD strength int NULL;
ALTER TABLE
lab1=# SELECT * FROM Department;
 deptid | deptname | nameofhod | strength
--------+----------+-----------+----------
(0 rows)
```

— — — —

3. Insert student details and department details as given below
Student(RollNo, Name, CGPA,DeptID)
(101, ‘Luffy’, 7.5,2),
(102, ‘Zoro’, 5.8, 3),
(103, ‘Nami’,7.9,1),
(104, ‘Robin’,6.1, 1),
(105, ‘Franky’, 4.9, 5),
(106, ‘Brook’, 5.5, 4),
(107, ‘Chopper’, 7.1, 2),
(108, ‘Ace’, 6.88, 5),
(109, ‘Danny’,5.44, 5),
(110, ‘Sabo’,9, 4),
(111, ‘Sanji’, 7.45, 3)
Department(DeptID, Deptname, NameofHOD,strength)
(1, ‘Computer Science’,’Zetsu’, 300),
(2,’Data Science’,’Sasuke’, 120),
(3,’Mechanical’, ‘Obito’, 600),
(4, ‘Electrical’, ‘Kakuzu’, 510),
(5, ‘Civil’, ‘Deidara’, 700),
(6,’Mathematics’,’Itachi’, 560)
–
Multiple inserts were done at the same time.



```
lab1=# INSERT INTO Department (DeptID, Deptname, NameofHOD,strength) VALUES
(1, 'Computer Science','Zetsu', 300),
(2,'Data Science','Sasuke', 120),
(3,'Mechanical', 'Obito', 600),
(4, 'Electrical', 'Kakuzu', 510),
(5, 'Civil', 'Deidara', 700),
(6,'Mathematics','Itachi', 560)
;
INSERT 0 6
```



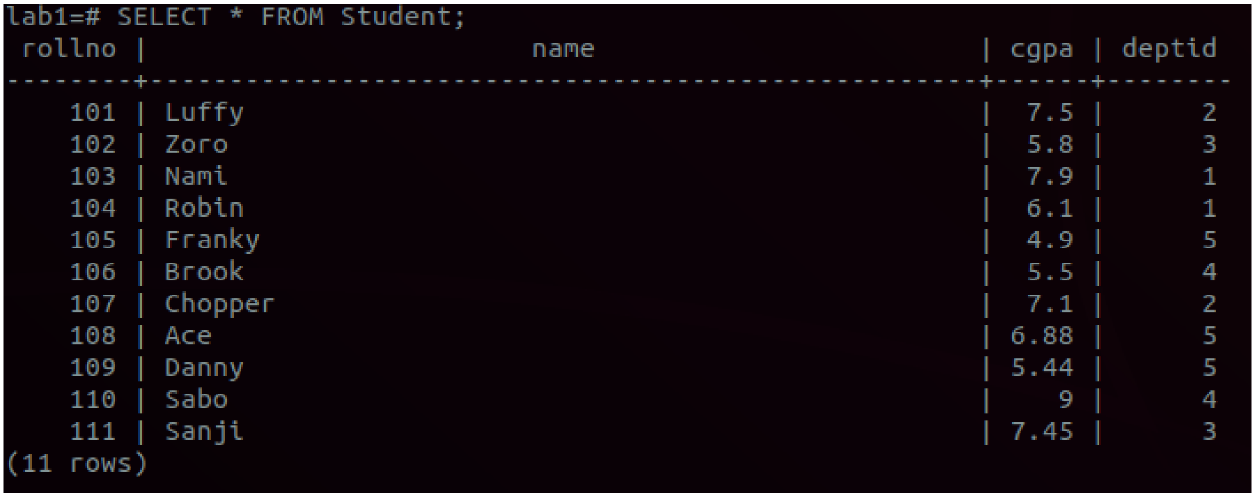
```
lab1=# INSERT INTO Student (RollNo, Name, CGPA,DeptID) VALUES
(101, 'Luffy', 7.5,2),
(102, 'Zoro', 5.8, 3),
(103, 'Nami',7.9,1),
(104, 'Robin',6.1, 1 ),
(105, 'Franky', 4.9, 5),
(106, 'Brook', 5.5, 4),
(107, 'Chopper', 7.1, 2),
(108, 'Ace', 6.88, 5),
(109, 'Danny',5.44, 5),
(110, 'Sabo',9, 4),
(111, 'Sanji', 7.45, 3)
;
INSERT 0 11
```

— — —

4. Display all Student details

–

* matches all coulumns



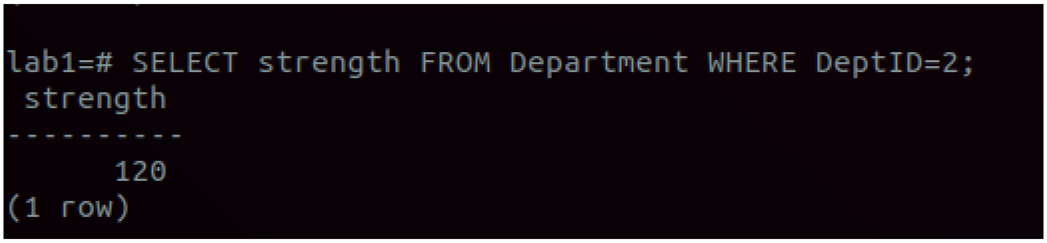
```
lab1=# SELECT * FROM Student;
 rollno |                         name                         | cgpa | deptid
--------+------------------------------------------------------+------+--------
    101 | Luffy                                                |  7.5 |      2
    102 | Zoro                                                 |  5.8 |      3
    103 | Nami                                                 |  7.9 |      1
    104 | Robin                                                |  6.1 |      1
    105 | Franky                                               |  4.9 |      5
    106 | Brook                                                |  5.5 |      4
    107 | Chopper                                              |  7.1 |      2
    108 | Ace                                                  | 6.88 |      5
    109 | Danny                                                | 5.44 |      5
    110 | Sabo                                                 |    9 |      4
    111 | Sanji                                                | 7.45 |      3
(11 rows)
```

— — —

5. Display the strength of the department when DeptID is given

–

DeptID was arbitrarily assumed to be 2 to proceed with the problem.



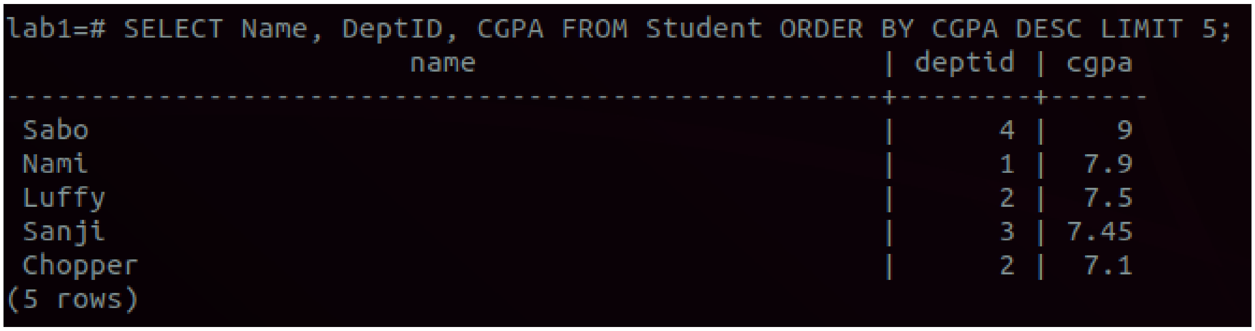
```
lab1=# SELECT strength FROM Department WHERE DeptID=2;
 strength
----------
      120
(1 row)
```

— — —

6. Display the Name, DeptID, CGPA of the students who secured the top 5 CGPAs in Descending order.

–

Selecting the required columns.. ORDER BY … DESC is the command to get descending sorted output. LIMIT 5 was given so that only top 5 candidates are displayed.



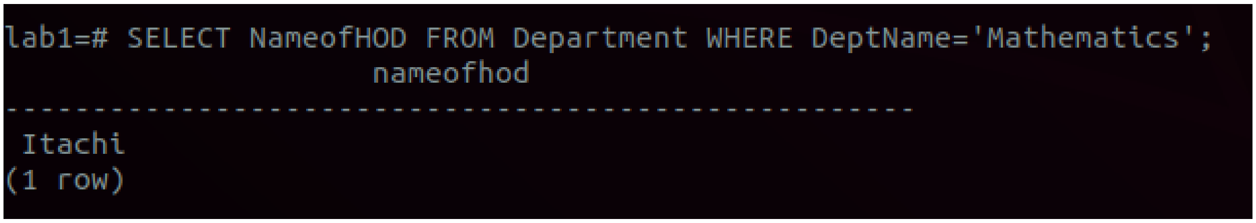
```
lab1=# SELECT Name, DeptID, CGPA FROM Student ORDER BY CGPA DESC LIMIT 5;
                        name                         | deptid | cgpa
-----------------------------------------------------+--------+------
 Sabo                                                |      4 |    9
 Nami                                                |      1 |  7.9
 Luffy                                               |      2 |  7.5
 Sanji                                               |      3 | 7.45
 Chopper                                             |      2 |  7.1
(5 rows)
```

— — —

7. Display the name of the HOD of the Mathematics department

–

Selecting NameofHOD. WHERE is the command used to give conditions to follow.



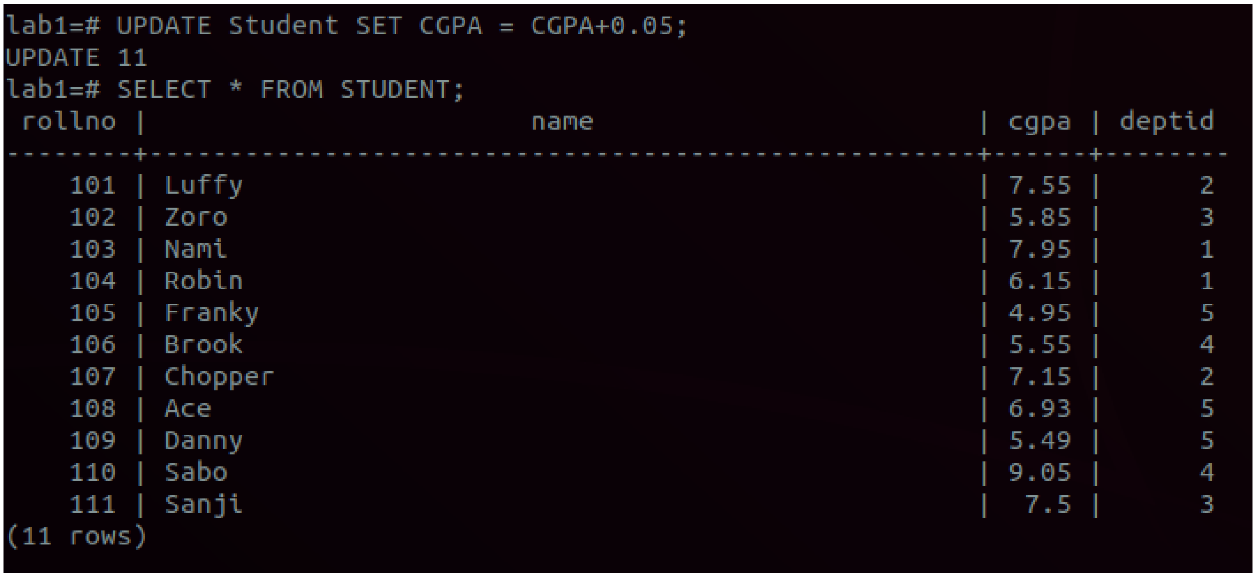
```
lab1=# SELECT NameofHOD FROM Department WHERE DeptName='Mathematics';
                      nameofhod
-----------------------------------------------------
 Itachi
(1 row)
```

— — —

8. Update the CGPA of all students by 0.05

–

We update the table Student. We set the new value of CGPA to be old value of CGPA+0.05. SET is the command used to do this.



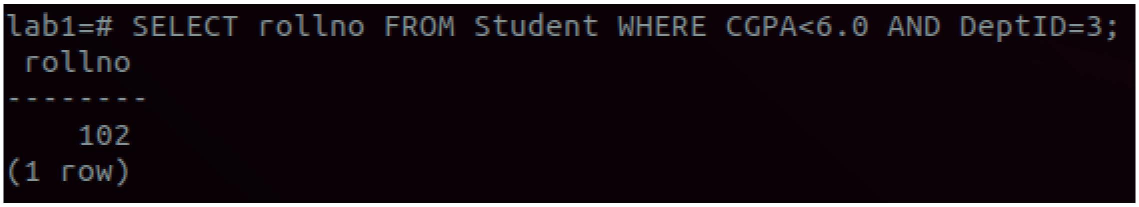
```
lab1=# UPDATE Student SET CGPA = CGPA+0.05;
UPDATE 11
lab1=# SELECT * FROM STUDENT;
 rollno |                         name                         | cgpa | deptid
--------+------------------------------------------------------+------+--------
    101 | Luffy                                                | 7.55 |      2
    102 | Zoro                                                 | 5.85 |      3
    103 | Nami                                                 | 7.95 |      1
    104 | Robin                                                | 6.15 |      1
    105 | Franky                                               | 4.95 |      5
    106 | Brook                                                | 5.55 |      4
    107 | Chopper                                              | 7.15 |      2
    108 | Ace                                                  | 6.93 |      5
    109 | Danny                                                | 5.49 |      5
    110 | Sabo                                                 | 9.05 |      4
    111 | Sanji                                                |  7.5 |      3
(11 rows)
```

— — —

9. Find the roll numbers of students who have CGPA < 6.0 and DeptID = ”3”

–

Basic command



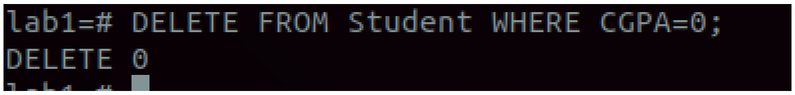
```
lab1=# SELECT rollno FROM Student WHERE CGPA<6.0 AND DeptID=3;
 rollno
--------
    102
(1 row)
```

— — —

10. Delete the details of students who have CGPA = 0

–

Basic command



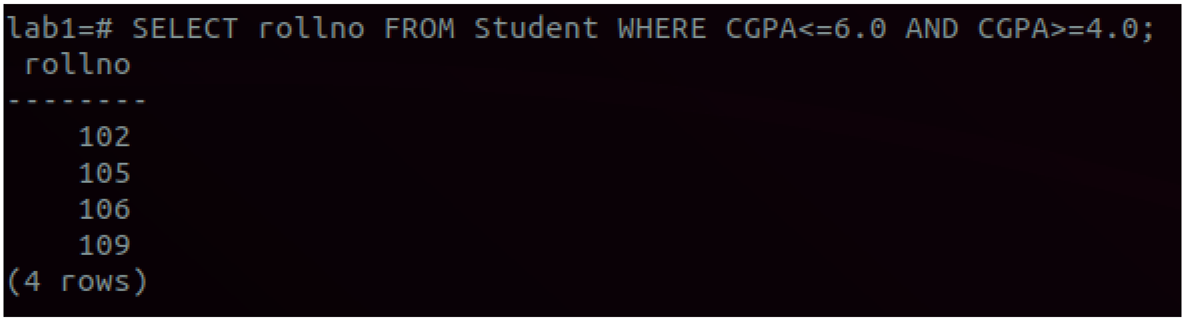
```
lab1=# DELETE FROM Student WHERE CGPA=0;
DELETE 0
```

— — —

11. Find the roll number of students who have CGPA between 4.0 to 6.0.

–

Basic command



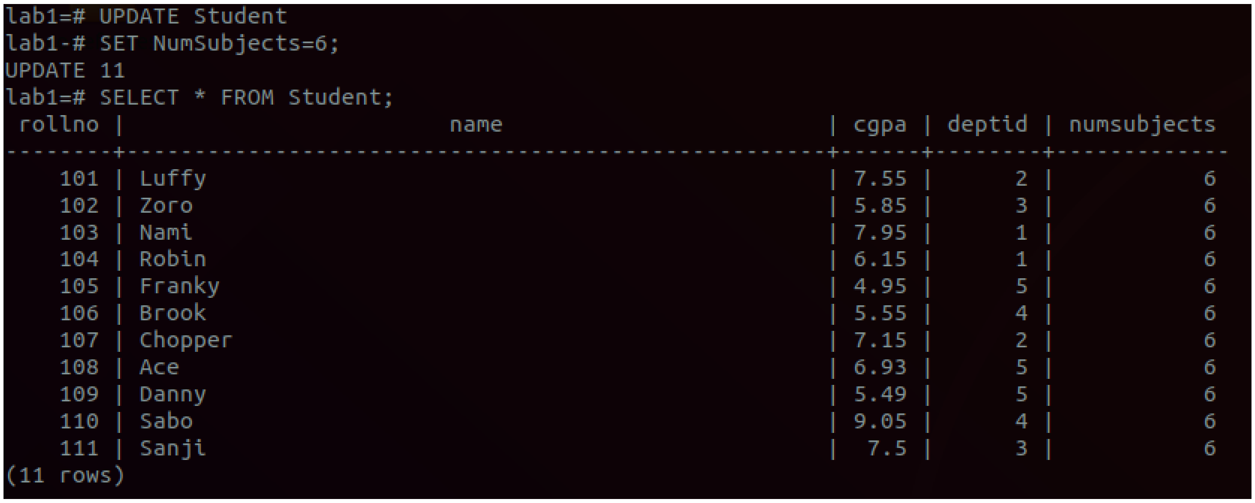
```
lab1=# SELECT rollno FROM Student WHERE CGPA<=6.0 AND CGPA>=4.0;
 rollno
--------
    102
    105
    106
    109
(4 rows)
```

— — —

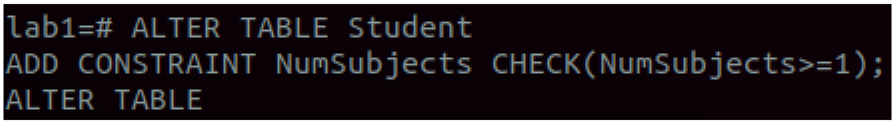
12. Add another column (NOT NULL) in the Student Table and insert
the data into this column. Use appropriate keywords to ensure that the column
should not contain any negative values and that all students must take a minimum of
one subject.

–

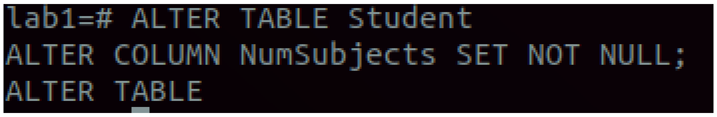
Added the constraint later, as it won’t cause an issue (because NOT NULL requires value to not be null, which before inserting into won’t be satisfied).



```
lab1=# UPDATE Student
lab1-# SET NumSubjects=6;
UPDATE 11
lab1=# SELECT * FROM Student;
 rollno |                         name                         | cgpa | deptid | numsubjects
--------+------------------------------------------------------+------+--------+-------------
    101 | Luffy                                                | 7.55 |      2 |           6
    102 | Zoro                                                 | 5.85 |      3 |           6
    103 | Nami                                                 | 7.95 |      1 |           6
    104 | Robin                                                | 6.15 |      1 |           6
    105 | Franky                                               | 4.95 |      5 |           6
    106 | Brook                                                | 5.55 |      4 |           6
    107 | Chopper                                              | 7.15 |      2 |           6
    108 | Ace                                                  | 6.93 |      5 |           6
    109 | Danny                                                | 5.49 |      5 |           6
    110 | Sabo                                                 | 9.05 |      4 |           6
    111 | Sanji                                                |  7.5 |      3 |           6
(11 rows)
```



```
lab1=# ALTER TABLE Student
ADD CONSTRAINT NumSubjects CHECK(NumSubjects>=1);
ALTER TABLE
```



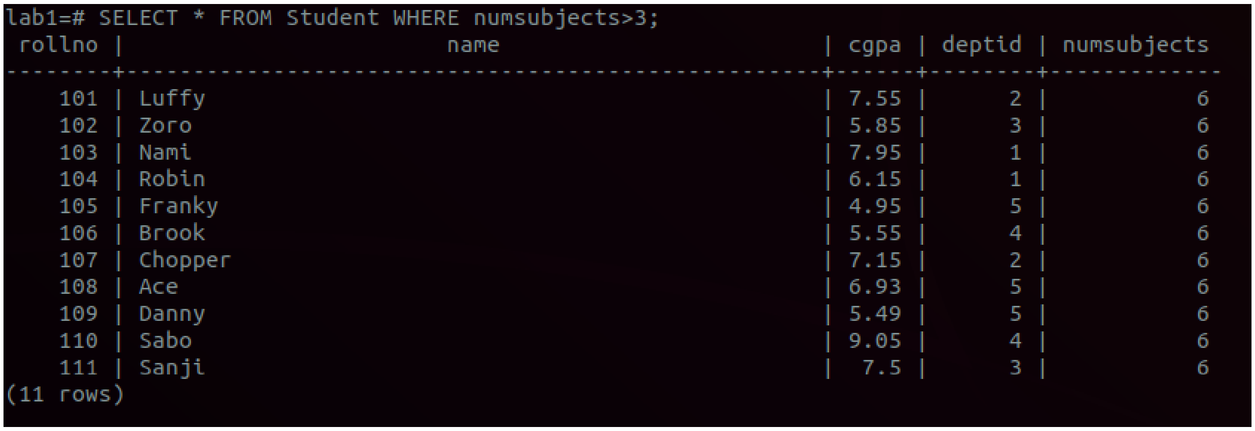
```
lab1=# ALTER TABLE Student
ALTER COLUMN NumSubjects SET NOT NULL;
ALTER TABLE
```

— — —

13. Display details of students who have taken more than 3 subjects.

–

Basic command



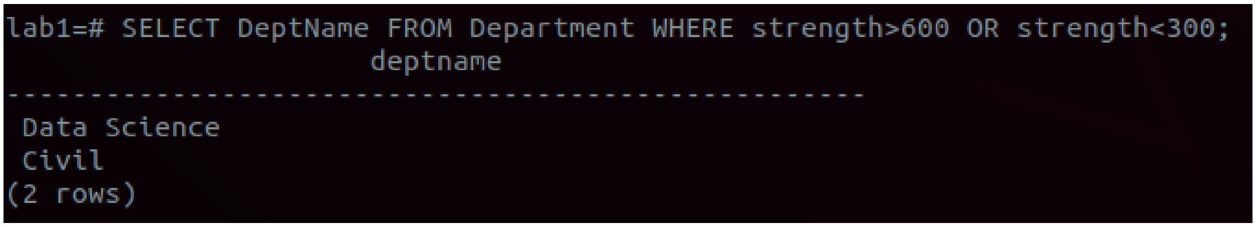
```
lab1=# SELECT * FROM Student WHERE numsubjects>3;
 rollno |                         name                       | cgpa | deptid | numsubjects
--------+----------------------------------------------------+------+--------+-------------
    101 | Luffy                                              | 7.55 |      2 |           6
    102 | Zoro                                               | 5.85 |      3 |           6
    103 | Nami                                               | 7.95 |      1 |           6
    104 | Robin                                              | 6.15 |      1 |           6
    105 | Franky                                             | 4.95 |      5 |           6
    106 | Brook                                              | 5.55 |      4 |           6
    107 | Chopper                                            | 7.15 |      2 |           6
    108 | Ace                                                | 6.93 |      5 |           6
    109 | Danny                                              | 5.49 |      5 |           6
    110 | Sabo                                               | 9.05 |      4 |           6
    111 | Sanji                                              |  7.5 |      3 |           6
(11 rows)
```

— — —

14. Find the DeptName where the total number of students is either less than 300 or greater than 600

–

Basic command



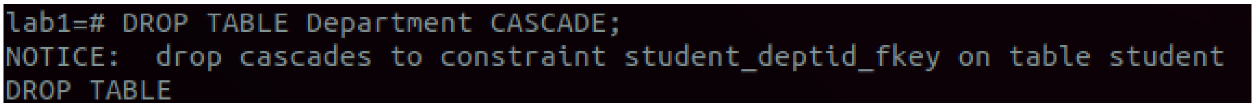
```
lab1=# SELECT DeptName FROM Department WHERE strength>600 OR strength<300;
                      deptname
----------------------------------------------------
 Data Science
 Civil
(2 rows)
```

— — —

15. Delete the table of Department

–

DROP TABLE … CASCADE was used because it removes the table along with the dependencies.



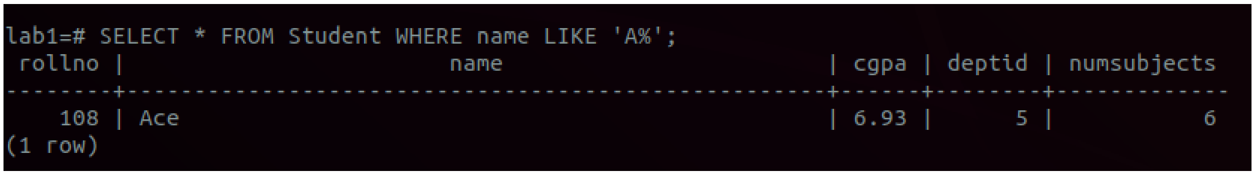
```
lab1=# DROP TABLE Department CASCADE;
NOTICE:  drop cascades to constraint student_deptid_fkey on table student
DROP TABLE
```

— — —

BONUS

1. Select the details of students whose names start with ‘A’

–

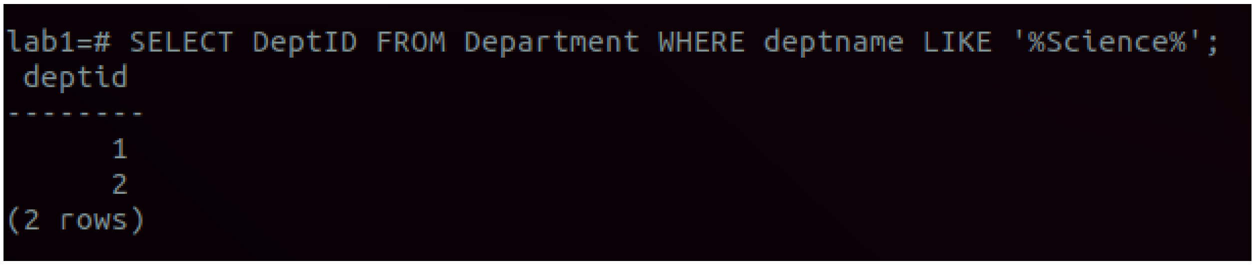


```
lab1=# SELECT * FROM Student WHERE name LIKE 'A%';
 rollno |                        name                        | cgpa | deptid | numsubjects
--------+----------------------------------------------------+------+--------+-------------
    108 | Ace                                                | 6.93 |      5 |           6
(1 row)
```

— — —

2. Select the DeptID where the department name ends with ‘science’

–

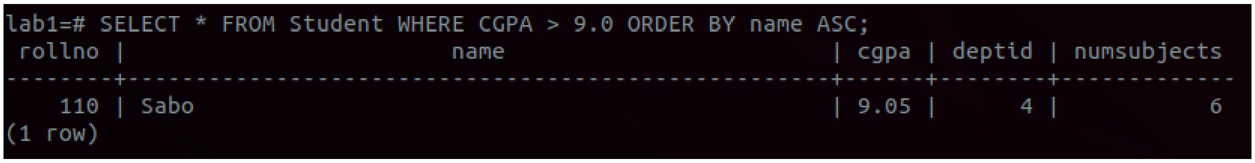


```
lab1=# SELECT DeptID FROM Department WHERE deptname LIKE '%Science%';
 deptid
--------
      1
      2
(2 rows)
```

— — —

3. Find students who have secured CGPA > 9.0 and display their details in the order of their names.

–



```
lab1=# SELECT * FROM Student WHERE CGPA > 9.0 ORDER BY name ASC;
 rollno |                        name                        | cgpa | deptid | numsubjects
--------+----------------------------------------------------+------+--------+-------------
    110 | Sabo                                               | 9.05 |      4 |           6
(1 row)
```

— — —