**Sheet No. – 01**

**Assignment No: 1**

1. **Creation of Tables:**

**(i). Creation of depts table:**

**Query:**

create table depts(

deptcode char(3) primary key,

deptname char(30) not null);

**Inserting values:**

insert into depts (deptcode, deptname)

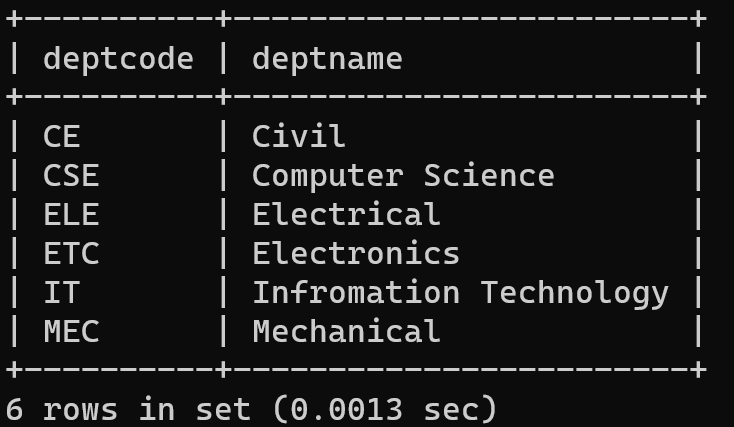
values

('CE', 'Civil'),

('CSE', 'Computer Science'),

('ELE', 'Electrical');

**depts table after insertion of values:**



**(ii). Creation of students table:**

**Query:**

create table students(

rollno numeric(8) primary key,

name char(30),

bdate date check(bdate < '1997-01-01'),

deptcode char(3) references depts(deptcode)

on delete cascade,

hostel numeric check(hostel<10),

parent\_inc numeric(8,1));

**Inserting values:**

insert into students (rollno, name, bdate, deptcode, hostel, parent\_inc)

values

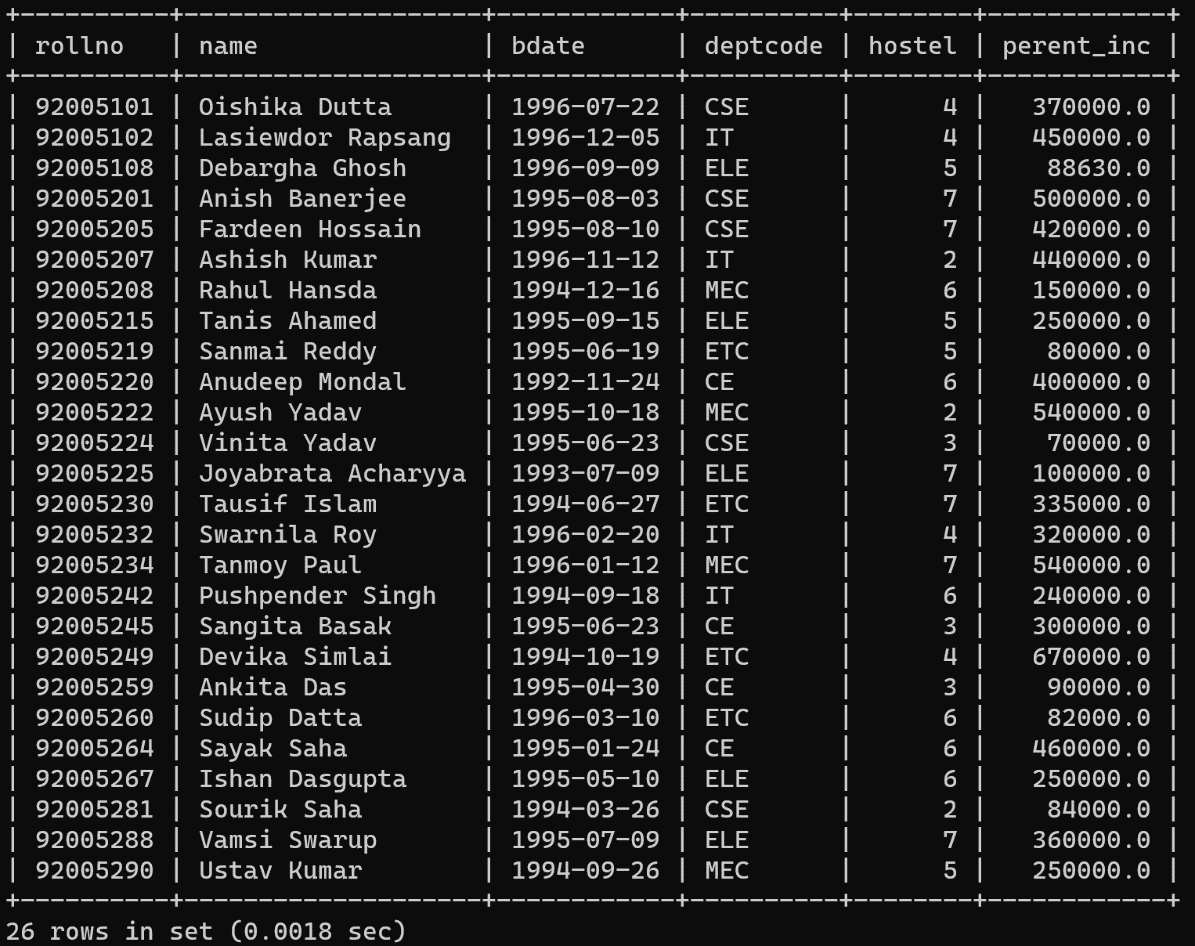
(92005101, 'Oishika Dutta', '1996-07-22', 'CSE', 4, 370000.0),

(92005102, 'Lasiewdor Rapsang', '1996-12-05', 'IT', 4, 450000.0),

(92005108, 'Debargha Ghosh', '1996-09-09', 'ELE', 5, 88630.0),

(92005201, 'Anish Banerjee', '1995-08-03', 'CSE', 7, 500000.0);

**students table after insertion of values:**



**(iii). Creation of faculty table:**

**Query:**

create table faculty(

fac\_code char(8) primary key,

fac\_name char(30) not null,

fac\_dept char(3) references depts(deptcode));

**Inserting values:**

insert into faculty (fac\_code, fac\_name, fac\_dept)

values

('AB', 'Aritra Banyopadhyay', 'ELE'),

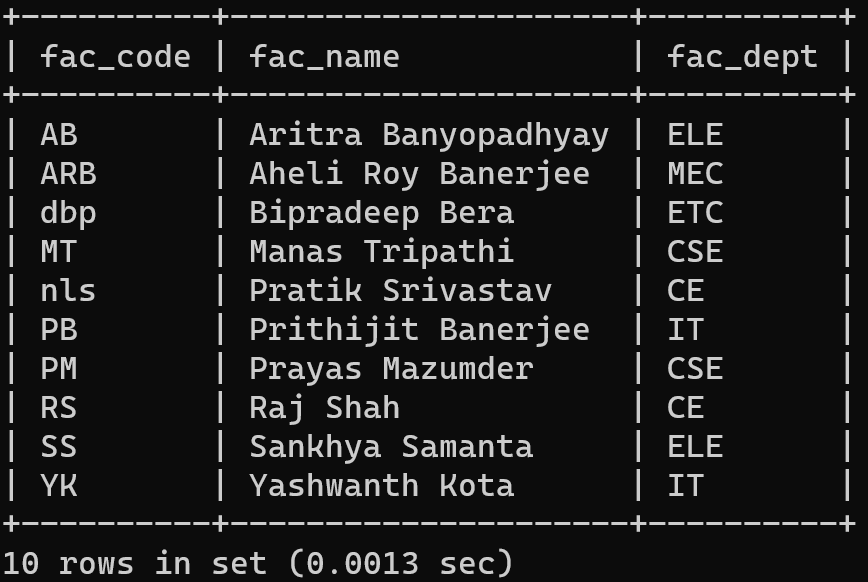
('ARB', 'Aheli Roy Banerjee', 'MEC'),

('dbp', 'Bipradeep Bera', 'ETC'),

('MT', 'Manas Tripathi', 'CSE'),

('nls', 'Pratik Srivastav', 'CE');

**faculty table after insertion of values:**

****

**(iv). Creation of crs\_offrd(course offered) table:**

**Query:**

create table crs\_offrd(

crs\_code char(5) primary key,

crs\_name char(35) not null,

crs\_credits numeric(2,1),

crs\_fac\_cd char(8) references faculty(fac\_code));

**Inserting values:**

insert into crs\_offrd (crs\_code, crs\_name, crs\_credits, crs\_fac\_cd)

values

('CE201', 'Water Resources', 6.0, 'RS'),

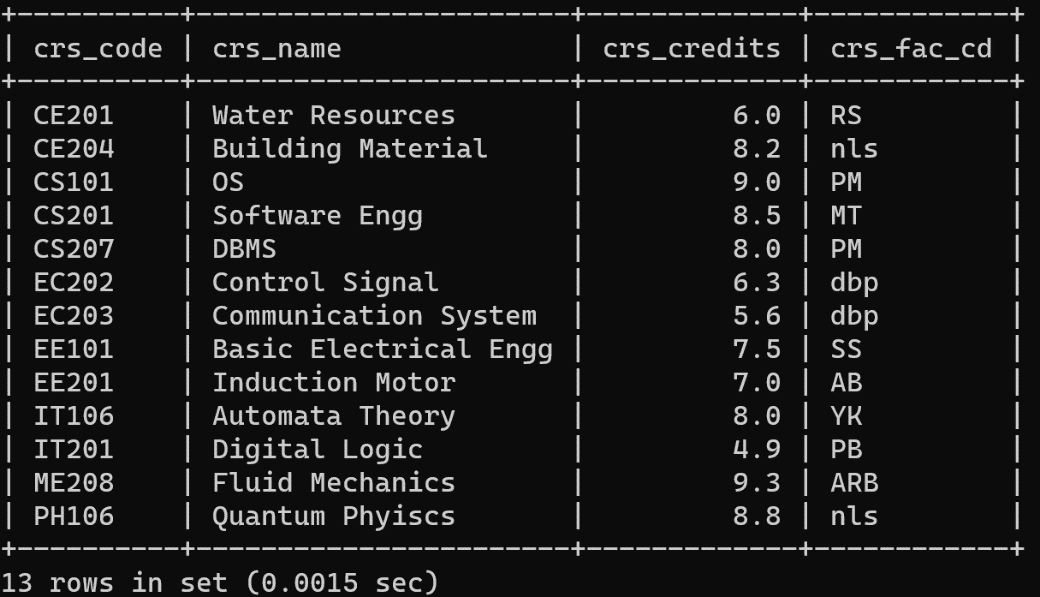
('CE204', 'Building Material', 8.2, 'nls'),

('CS101', 'OS', 9.0, 'PM'),

('CS201', 'Software Engg', 8.5, 'MT'),

('CS207', 'DBMS', 8.0, 'PM');

**crs\_offrd table after insertion of values:**

****

**(iv). Creation of crs\_regd(course registered) table:**

**Query:**

create table crs\_regd(

crs\_rollno numeric(8) references students(rollno),

crs\_cd char(5) references crs\_offrd(crs\_code),

marks numeric(5,2),

primary key(crs\_rollno,crs\_cd));

**Inserting values:**

insert into crs\_regd (crs\_rollno, crs\_cd, marks)

values

(92005101, 'CS101', 89.00),

(92005101, 'CS201', 59.00),

(92005101, 'CS207', 76.00),

(92005102, 'IT106', 68.00),

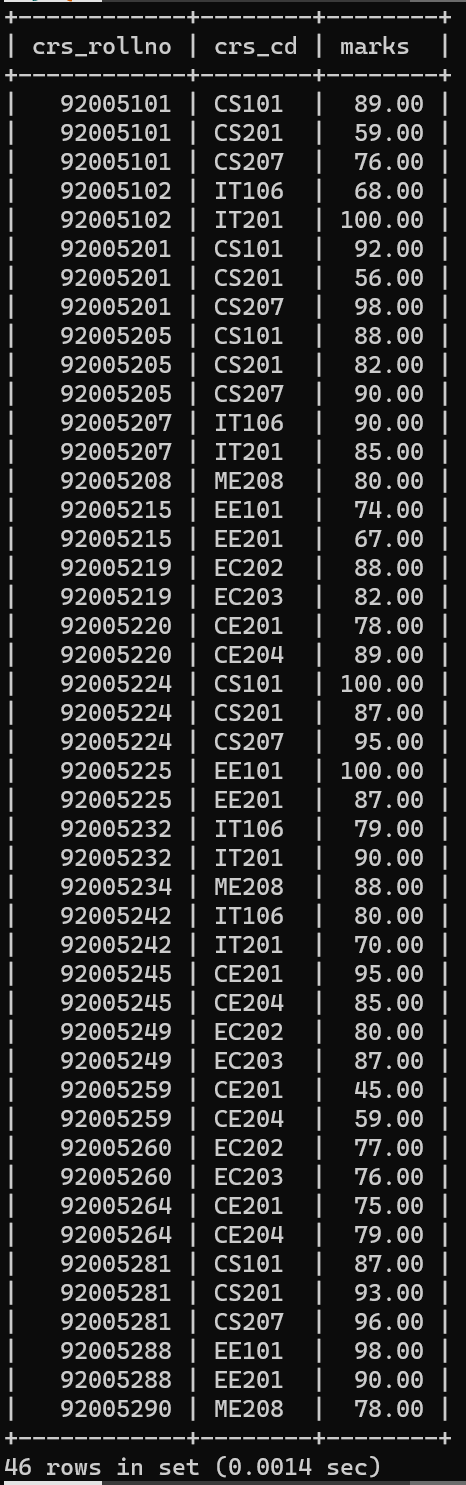
(92005102, 'IT201', 100.00),

(92005201, 'CS101', 92.00),

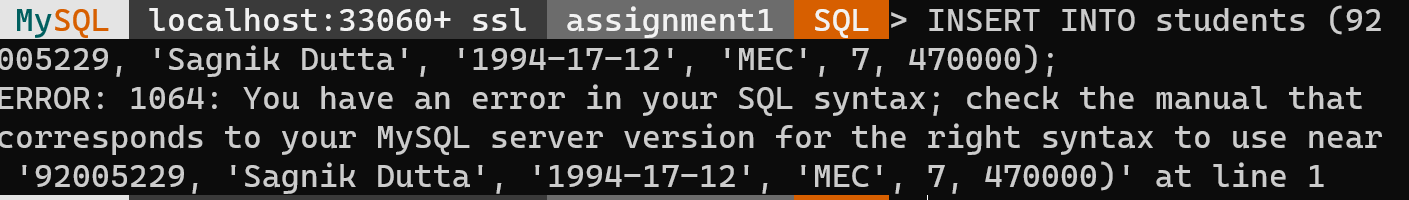
(92005201, 'CS201', 56.00),

(92005201, 'CS207', 98.00);

**crs\_regd table after insertion of values:**

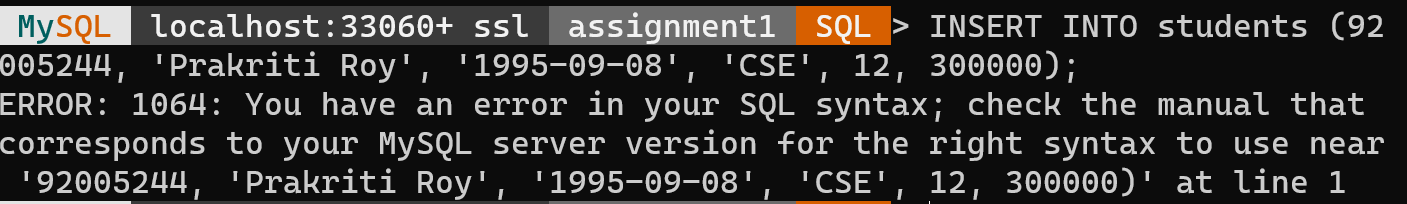
****

1. **Queries and their solutions:**
2. **Try inserting records which violate the constraints:**

****

**Explanation:**

Here it is showing error because, while creating the table ‘students’ , I added a check constraint that is bdate date check(bdate < '1997-01-01'). But here I added ‘1994-17-12’ where correct formatting of date is not used. Therefore check constraint is violated here.



**Explanation:**

Here it is showing error because, at the time of creation of table ‘students’, a check constraint was added that hostel numeric check(hostel<10). But I entered value 12. So check constraint is violated here.

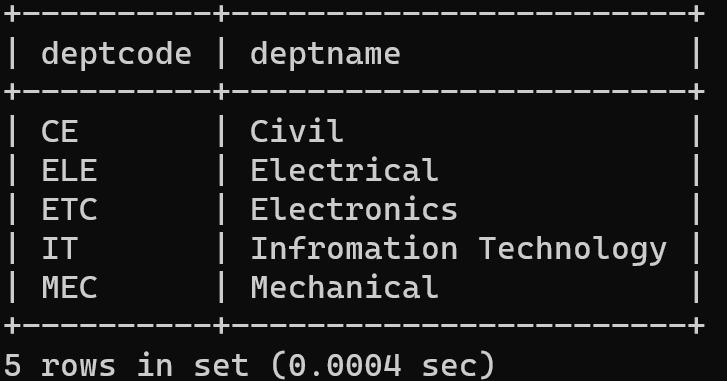
1. **Delete records from dept where deptcode='CSE'. (This deletes records from students whose deptcode='CSE'):**

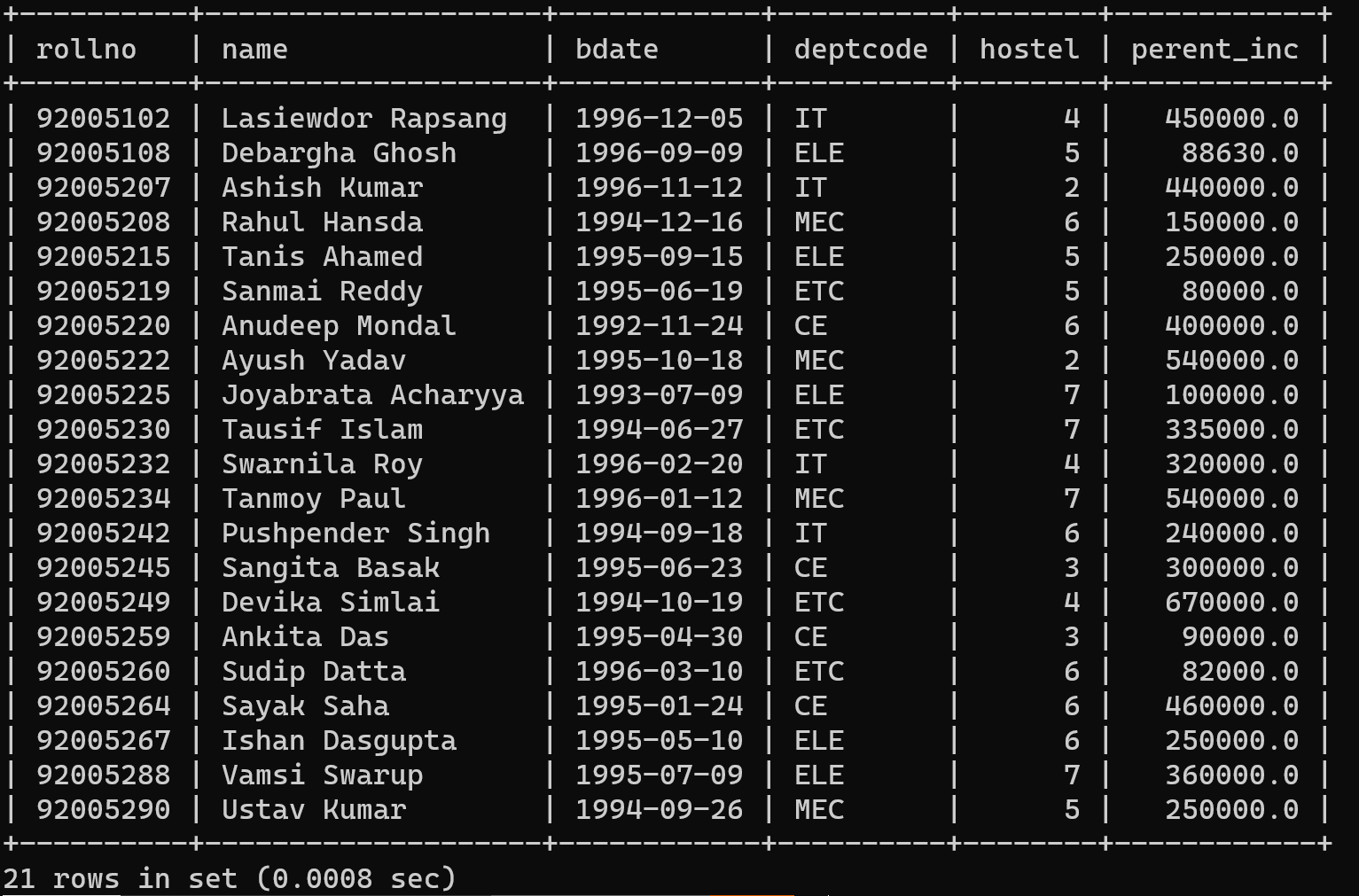
**Query:**

delete from dept

where deptcode=‘CSE’;

**Output:**

****

****

The result shows that the students who have deptocde as ‘CSE’ are removed from the students as well as from the depts table.

1. **Find out the courses offered by the faculty dbp and nls:**

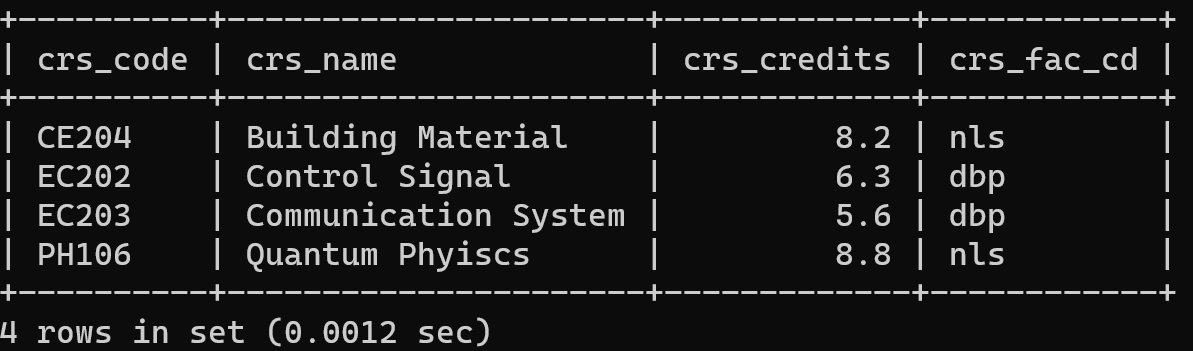
**Query:**

select \*

from crs\_offrd

where crs\_fac\_cd IN (‘dbp’, ‘nls’);

**Output:**



1. **Find out the courses with full details offered by dbp:**

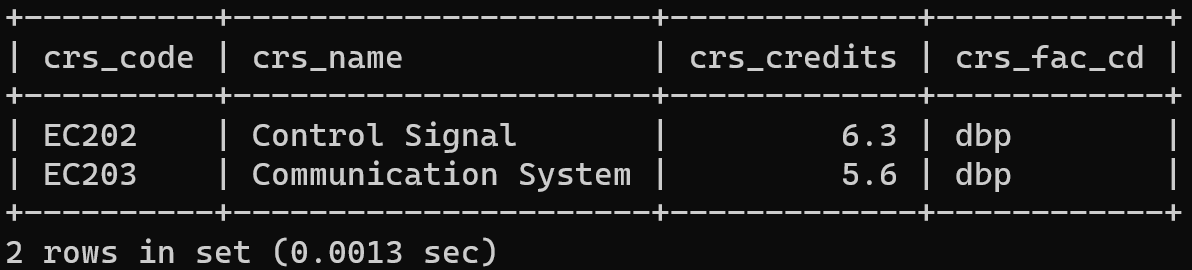
**Query:**

select \*

from crs\_offrd

where crs\_fac\_cd = 'dbp';

**Output:**

****

1. **Get the courses the credits of which lies between 4.0 and 6.0:**

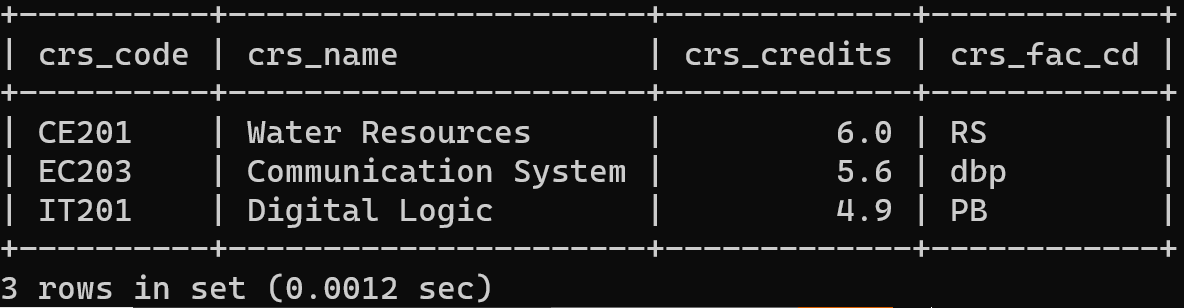
**Query:**

select \*

from crs\_offrd

where crs\_credits BETWEEN 4.0 AND 6.0;

**Output:**



1. **Get the courses the credits of which are > 6.5:**

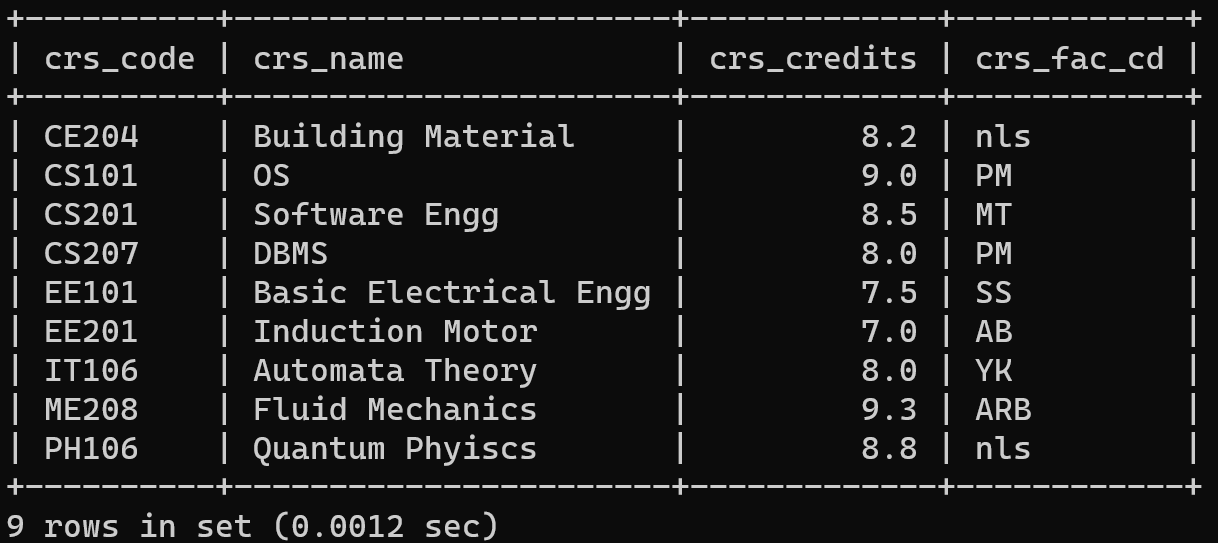
**Query:**

select \*

from crs\_offrd

where crs\_credits > 6.5;

**Output:**

****

**Assignment No : 2**

1. **Count the number of students in CSE dept:**

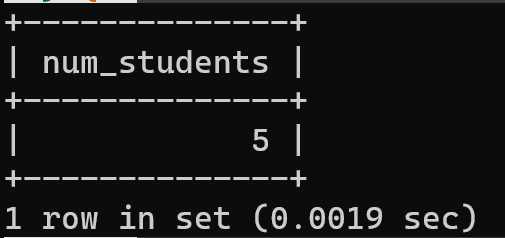
**Query:**

select COUNT(\*) AS num\_students

from students

where deptcode = 'CSE';

**Output:**

****

1. **Determine the minimum, maximum and average marks of each courses:**

**Query:**

select crs\_cd,

       MIN(marks) AS min\_marks,

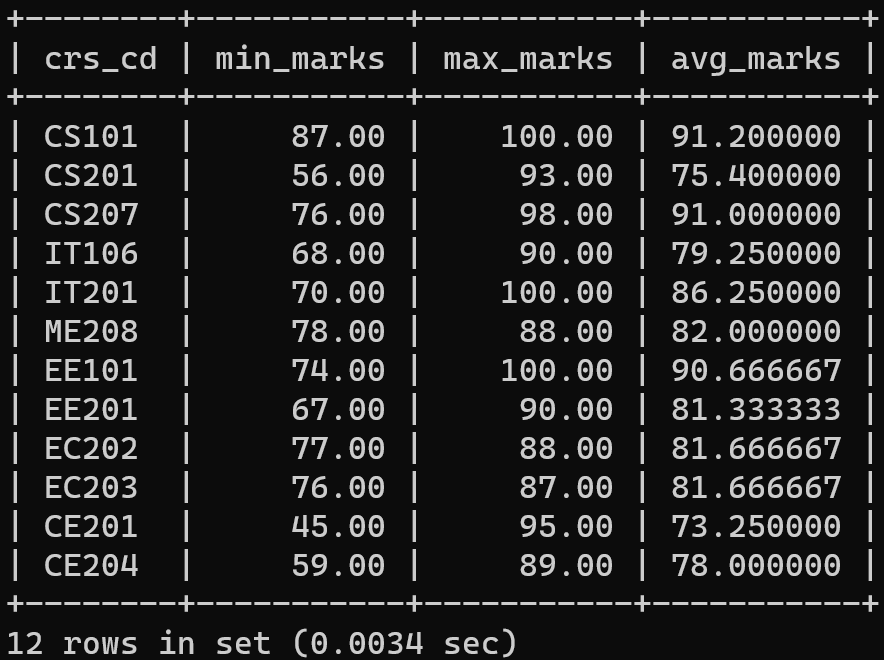
       MAX(marks) AS max\_marks,

       AVG(marks) AS avg\_marks

from crs\_regd

GROUP BY crs\_cd;

**Output:**

****

1. **Determine the total credits of the courses registered by a student:**

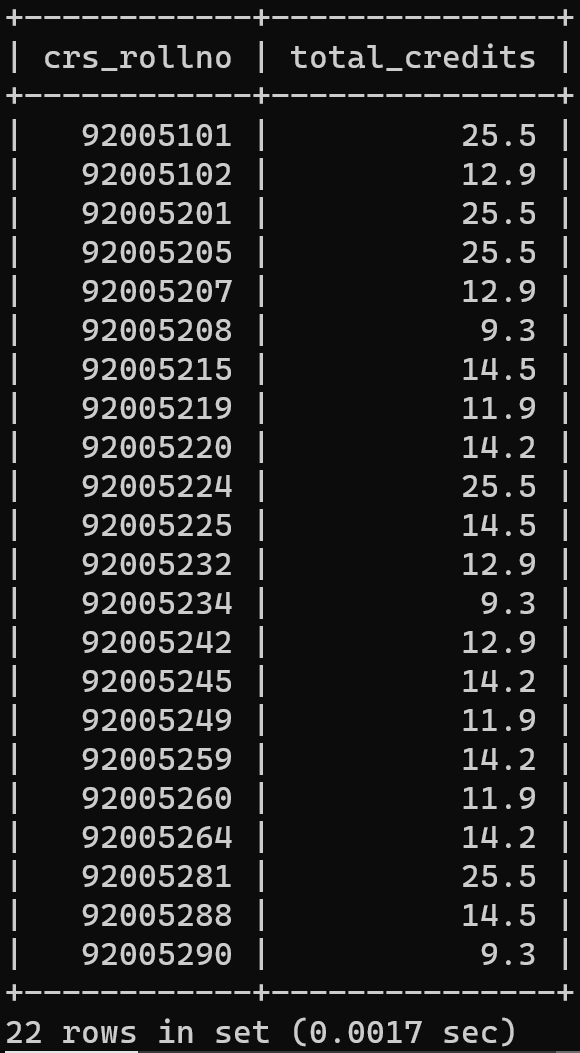
**Query:**

select cr.crs\_rollno, SUM(co.crs\_credits) AS total\_credits

from crs\_regd cr, crs\_offrd co where cr.crs\_cd = co.crs\_code

group by cr.crs\_rollno;

**Output:**

****

1. **Count the number of students in each hostel whose department is CSE:**

**Query:**

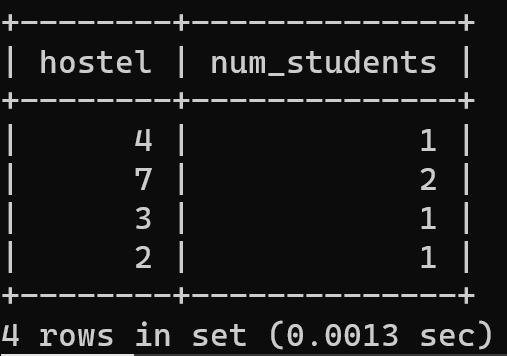
select s.hostel, COUNT(\*) AS num\_students

from students s

where s.deptcode = 'CSE'

group by s.hostel;

**Output:**

****

1. **Display the hostel, rollno, parent\_inc of the student who has the max(parent\_inc) in a hostel:**

**Query:**

Select hostel, rollno, perent\_inc

from students

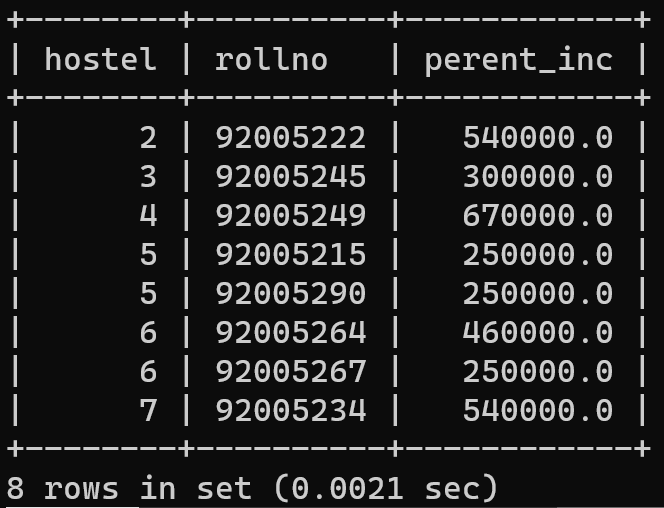
where perent\_inc in (select max(perent\_inc)

from students

group by hostel)

ORDER BY hostel;

**Output:**

****

1. **Display the name and parental income of each student greater than the parental income of some rollno 92005008:**

**Query:**

select name, perent\_inc

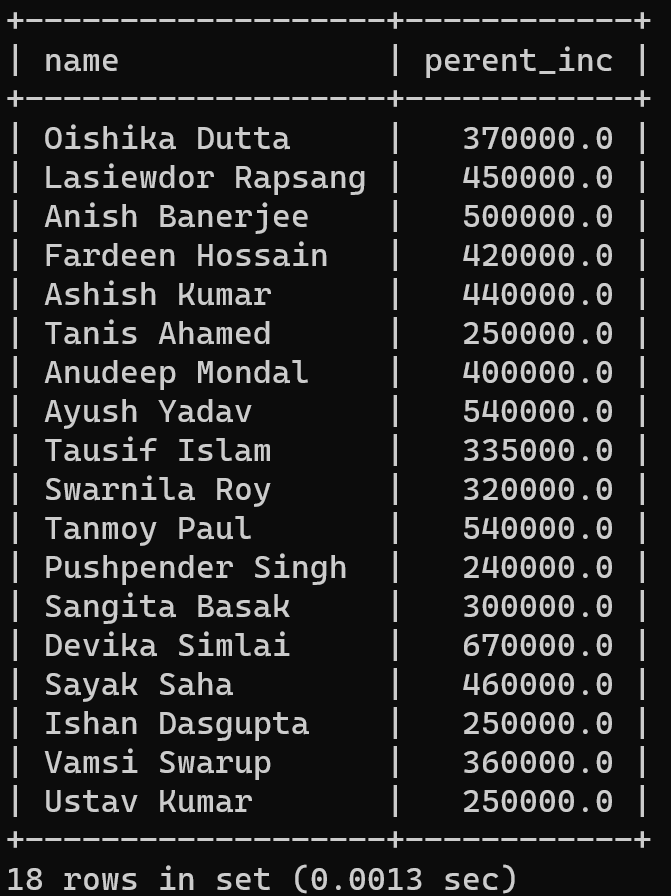
from students

where perent\_inc > (select perent\_inc

from students

where rollno = 92005208);

**Output:**

****

1. **Find out marks of students who have marks more than rollno 92005102 for course CH103 and PH106:**

**Query:**

select crs\_rollno, crs\_cd, marks

from crs\_regd

where(

(

crs\_cd = 'CE201'

AND

marks > (

select marks

from crs\_regd

where crs\_rollno = 92005102 AND crs\_cd = 'CE201'

)

)

OR

(

crs\_cd = 'PH106'

AND

marks > (

select marks

from crs\_regd

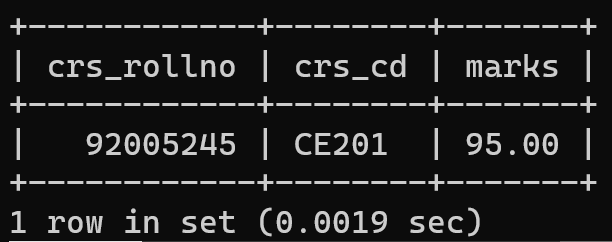
where crs\_rollno = 92005102 AND crs\_cd = 'PH106'

)

)

);

**Output:**

****

**Assignment No : 3**

1. **List students (rollno, name, deptcode) registered for course EE101:**

**Query:**

select rollno, name, deptcode

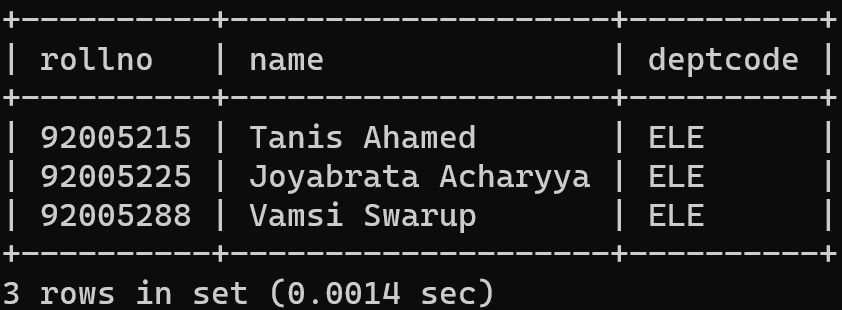
from students, crs\_regd

where

crs\_rollno = rollno

AND crs\_cd = 'EE101';

**Output:**

****

1. **List students (rollno, name) in ELE dept registered for course EE101:**

**Query:**

select rollno, name

from students, crs\_regd

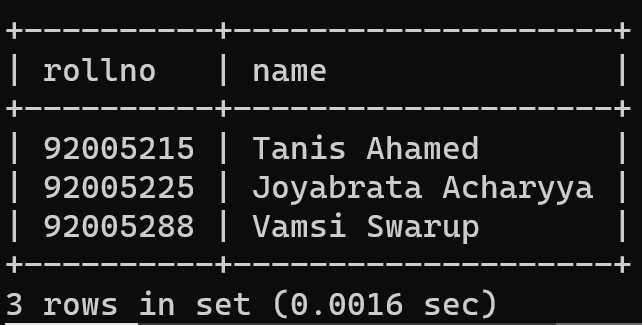
where

crs\_rollno = rollno

AND deptcode = 'ELE'

AND crs\_cd = 'EE101';

**Output:**

****

1. **List students (rollno, name) in ELE dept not registered for course EE101:**

**Query:**

select s.rollno, s.name

from students s, crs\_regd c

where c.crs\_rollno = s.rollno

AND s.deptcode = 'ELE'

EXCEPT

select s.rollno, s.name

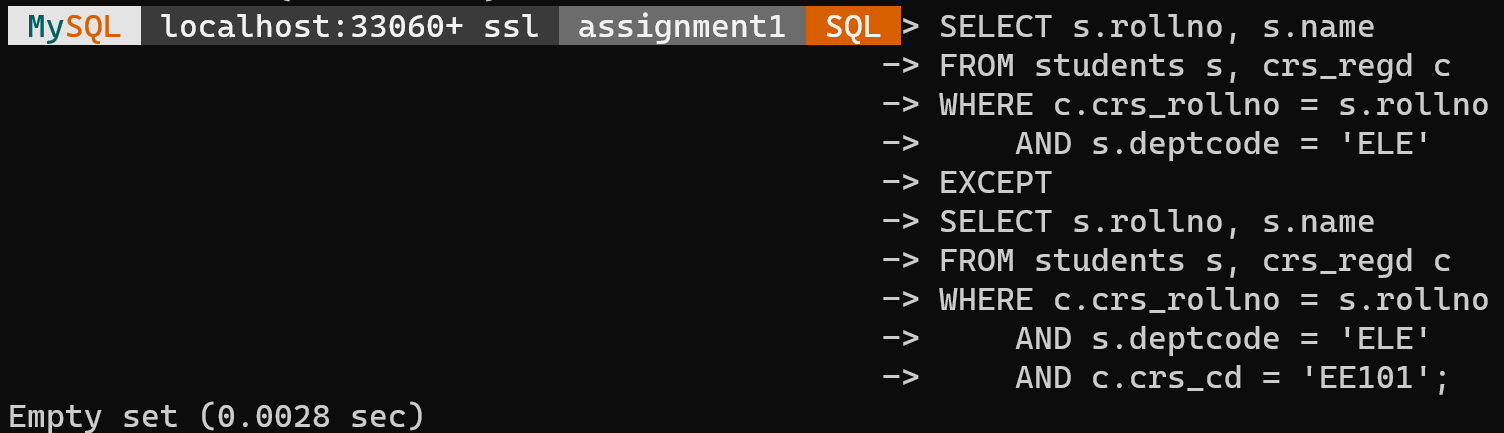
from students s, crs\_regd c

where c.crs\_rollno = s.rollno

AND s.deptcode = 'ELE'

AND c.crs\_cd = 'EE101';

**Output:**

****

This returns an empty set because there are no students who does not enrolled in any of the courses. ELE department students have enrolled in all the departmental courses.

1. **List the names of the students who have registered for both the courses 'DBMS' and 'OS':**

**Query:**

select name

from students, crs\_regd, crs\_offrd

where

crs\_rollno = rollno

AND crs\_cd = crs\_code

AND crs\_offrd.crs\_name = 'DBMS'

INTERSECT

select name

from students, crs\_regd, crs\_offrd

where

crs\_rollno = rollno

AND crs\_cd = crs\_code

AND crs\_offrd.crs\_name = 'OS';

**Output:**

****

1. **Find the names of the faculty members who have offered either 'Automata Theory' or 'Software Engg.':**

**Query:**

select fac\_name

from faculty, crs\_offrd

where

crs\_fac\_cd = fac\_code

AND crs\_name = ‘Automata Theory'

UNION

select fac\_name

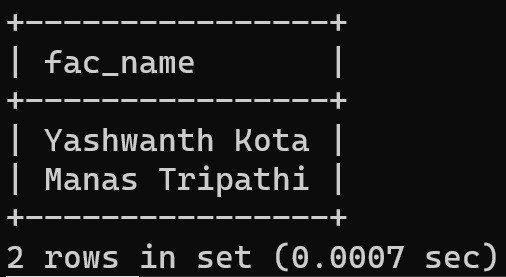
from faculty, crs\_offrd

where

crs\_fac\_cd = fac\_code

AND crs\_name = 'Software Engg';

**Output:**

****

1. **Find the names of the faculty members who have offered 'Automata Theory' but not offered 'Software Engg.':**

**Query:**

select fac\_name from faculty, crs\_offrd

where

fac\_code = crs\_fac\_cd

AND crs\_name = 'Automata Theory'

EXCEPT

select fac\_name

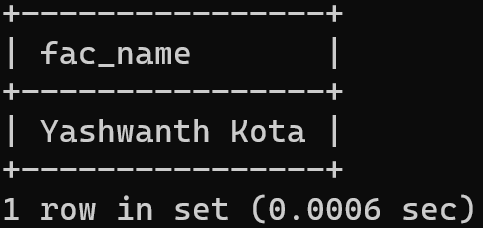
from faculty, crs\_offrd

where

fac\_code = crs\_fac\_cd

AND crs\_name = 'Software Engg';

**Output:**

****

1. **Find out the students in each hostel who are not registered for any course:**

**Query:**

(SELECT rollno, name, hostel

FROM students

WHERE rollno NOT IN (SELECT crs\_rollno FROM crs\_regd))

UNION

(SELECT rollno, name, hostel

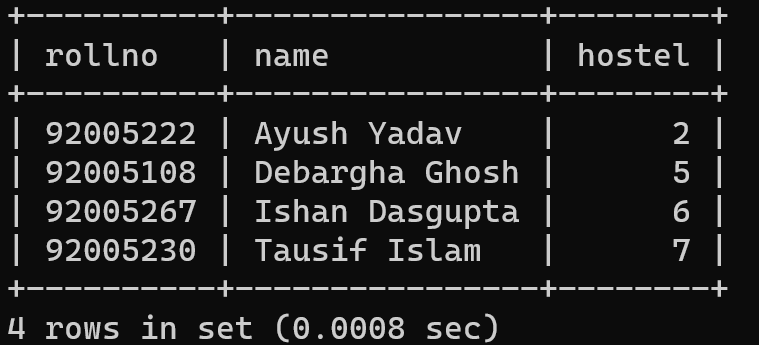
FROM students

WHERE rollno IN (SELECT crs\_rollno FROM crs\_regd)

AND rollno NOT IN (SELECT rollno FROM students WHERE rollno IN (SELECT crs\_rollno FROM crs\_regd)))

ORDER BY hostel;

**Output:**

****

1. **Select the students who are in ELE dept or who have registered for course CS101:**

**Query:**

select rollno, name, deptcode

from students

where deptcode = 'ELE'

UNION

select rollno, name, deptcode

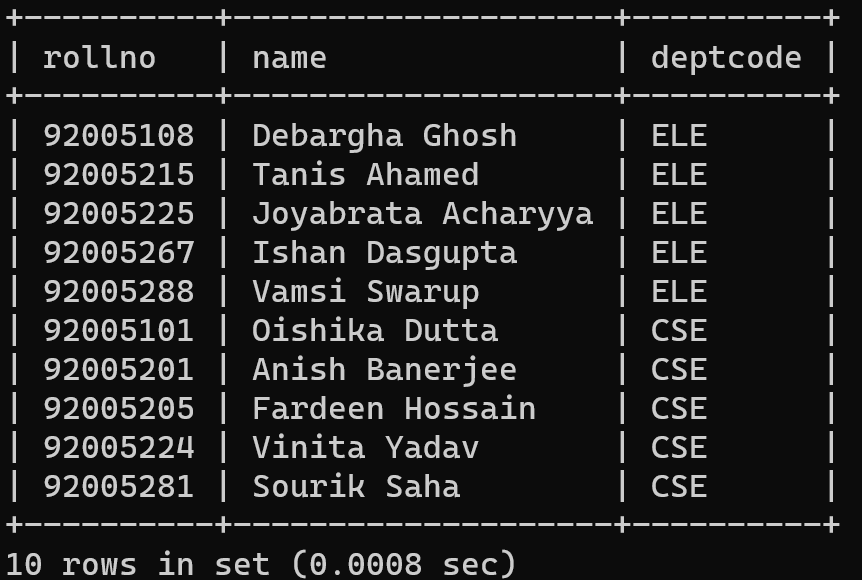
from students, crs\_regd

where

rollno = crs\_rollno

AND crs\_cd = 'CS101';

**Output:**

****

1. **Display the students who have registered to all the courses:**

**Query:**

select crs\_rollno, name, deptcode

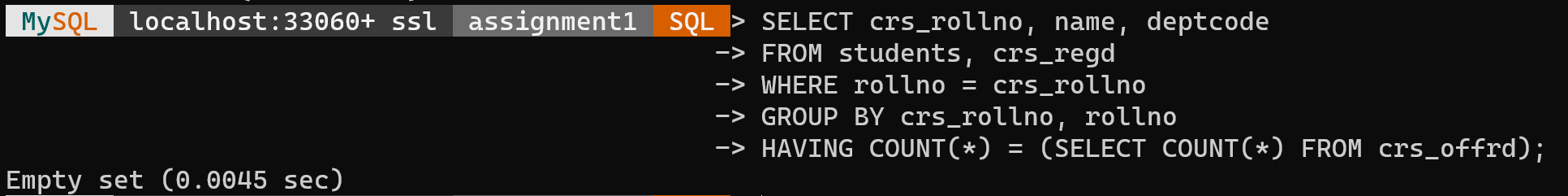
from students, crs\_regd

where rollno = crs\_rollno

group by crs\_rollno, rollno

HAVING COUNT(\*) = (SELECT COUNT(\*) from crs\_offrd);

**Output:**

****

This returns an empty set because there are no students who has registered to all the courses.

1. **Give Grace Marks 5 in subject ‘DBMS’ to the students who have scored less than 50 in that subject:**

**Query:**

UPDATE crs\_regd

SET marks = marks + 5

FROM crs\_offrd

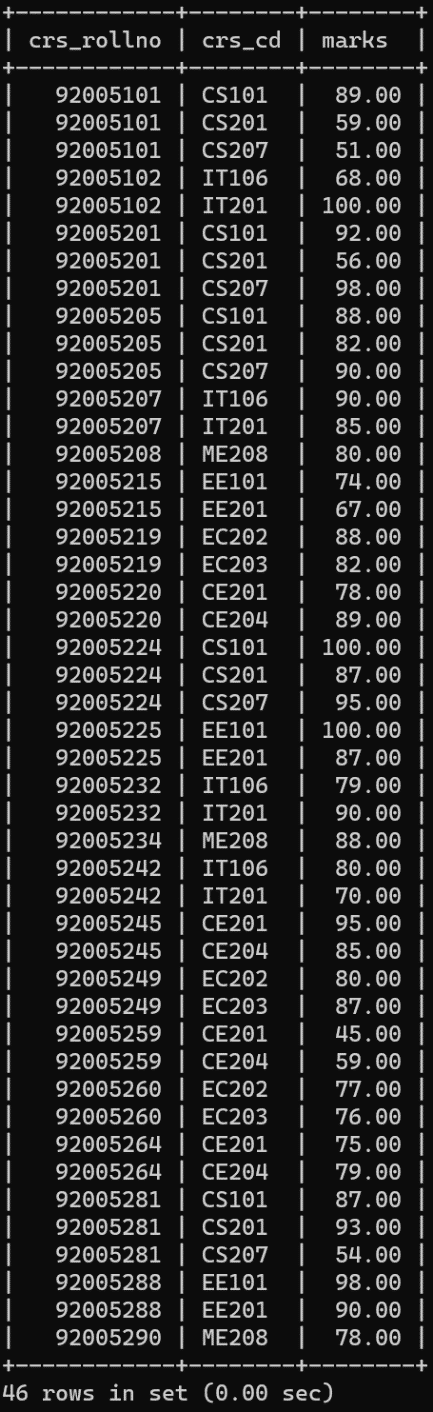
WHERE

crs\_code = crs\_cd

AND crs\_name = 'DBMS'

AND marks < 50;

**Output:**

****