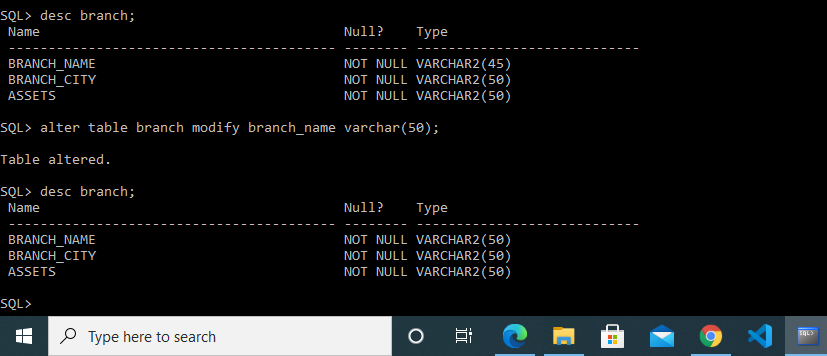
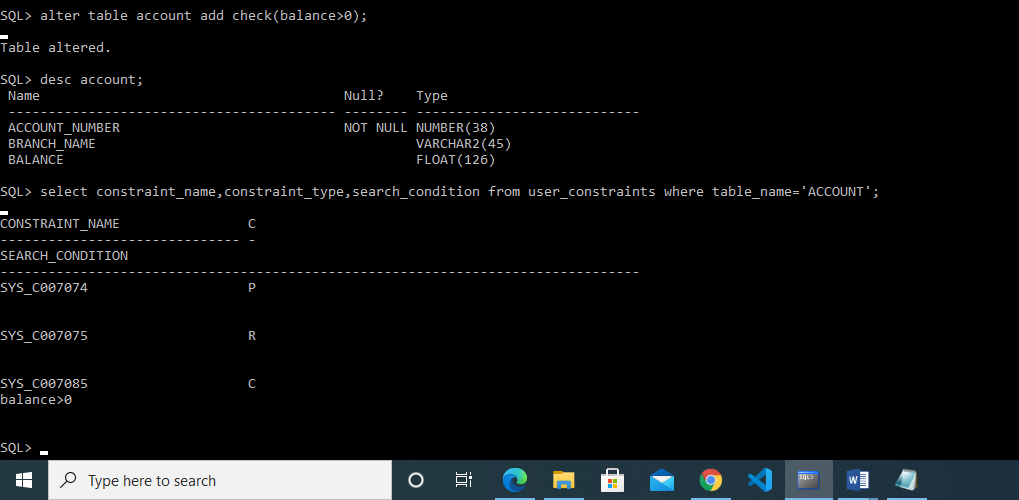
1. Increase the size of branch\_name field.

alter table branch modify branch\_name varchar(50);



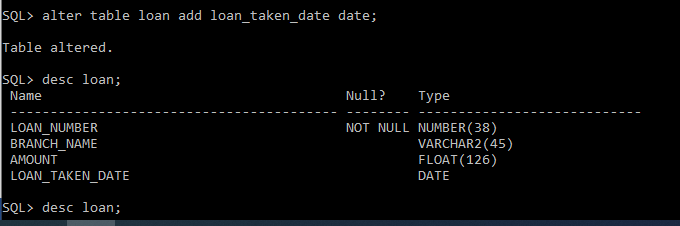
2. Add constraint to check whether balance is more than zero.

alter table account add check(balance>0);



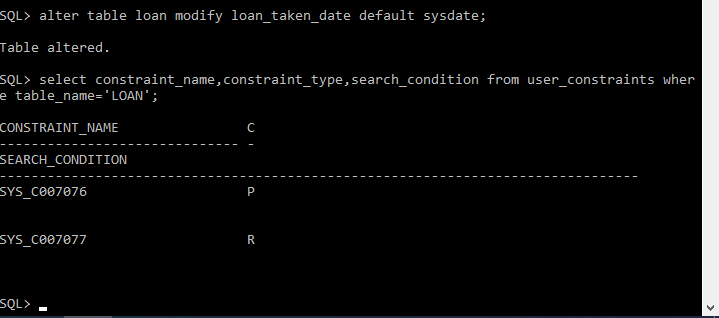
3. Add a column loan\_taken\_date in loan table.

alter table loan add loan\_taken\_date date;



4. Add default constraint on loan\_taken\_date to insert current date

alter table loan modify loan\_taken\_date default sysdate;



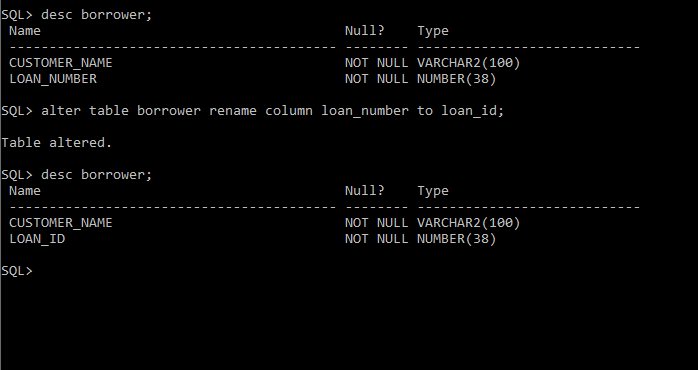
5. Drop foreign key constraint from loan\_number field of borrower table

alter table borrower drop foreign key loan\_number;

ERROR

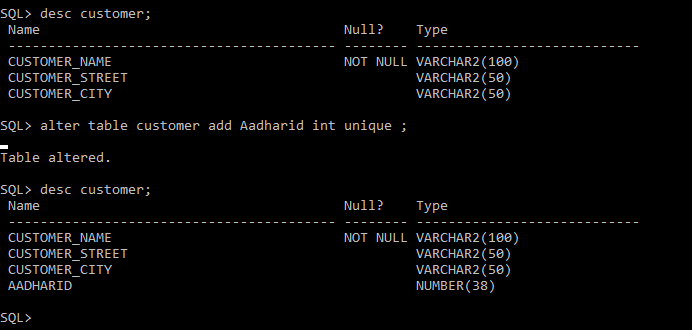
6. Rename column loan\_number of borrower table as “loan\_id”

alter table borrower rename column loan\_number to loan\_id;



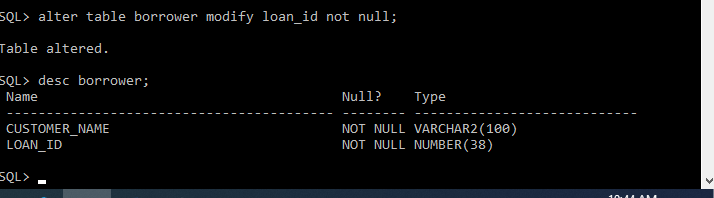
7. Add a field “Aadharid” with unique constraint in customer table

alter table customer add Aadharid int unique ;



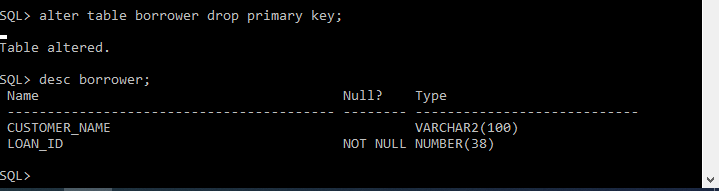
8. Add “Not Null” constraint on loan\_number field of borrower table

alter table borrower modify loan\_id not null;



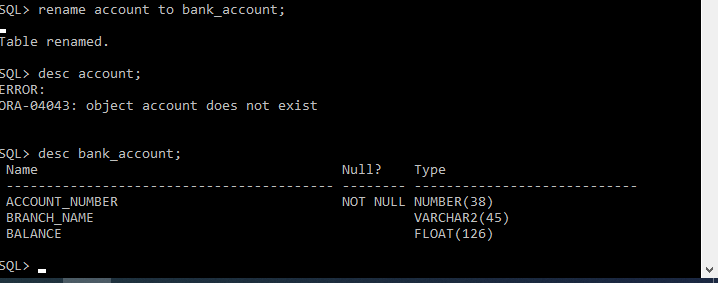
9. Drop primary key constraint from borrower table.

alter table borrower drop primary key;



10. Rename table account to bank\_account.

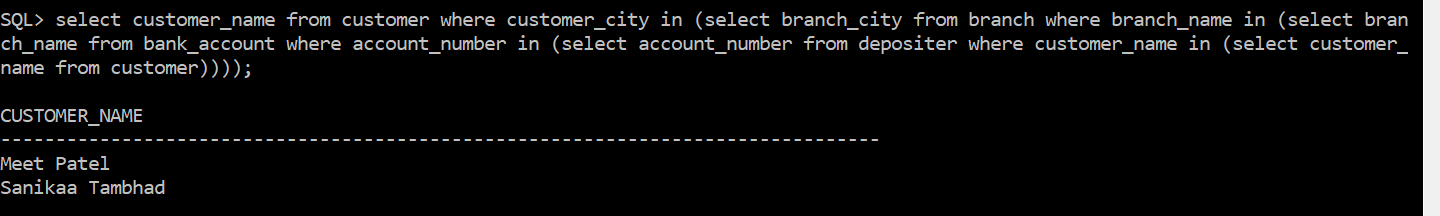
rename account to bank\_account;



Part 2

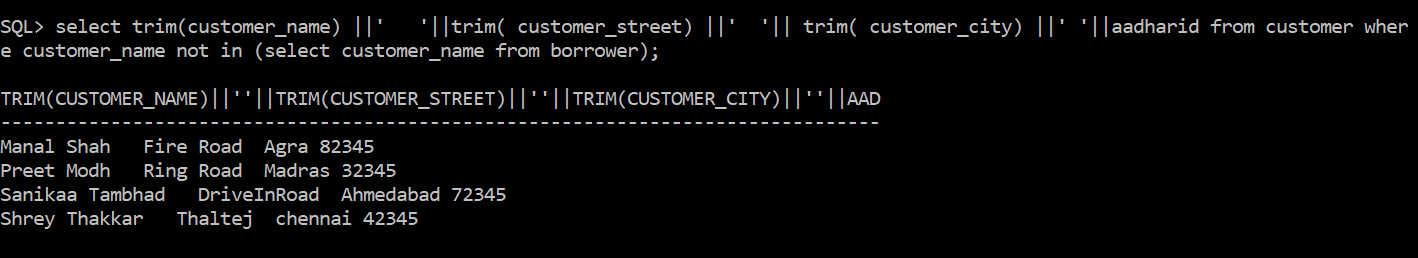
1. Display names of the customers who have account in the city where they live.

**select customer\_name from customer where customer\_city in (select branch\_city from branch where branch\_name in (select branch\_name from bank\_account where account\_number in (select account\_number from depositer where customer\_name in (select customer\_name from customer))));**



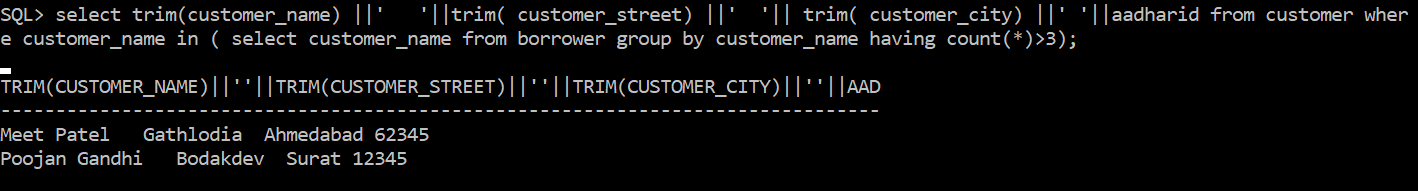
1. Display details of customers who are not borrower.

**select trim(customer\_name) ||' '||trim( customer\_street) ||' '|| trim( customer\_city) ||' '||aadharid from customer where customer\_name not in (select customer\_name from borrower);**



3. Display customer details who have taken loan more than 3 times

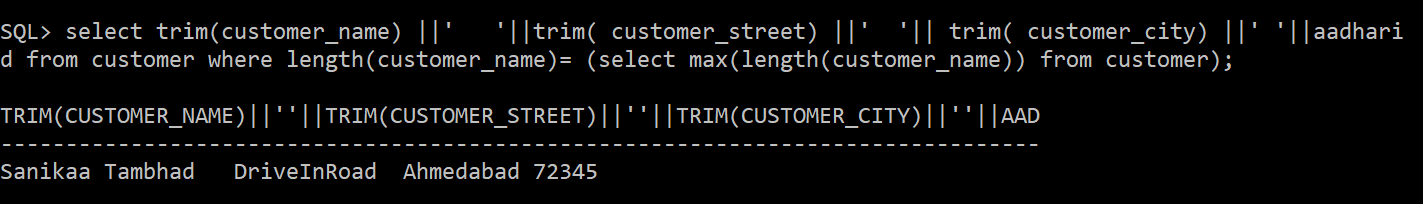
**select trim(customer\_name) ||' '||trim( customer\_street) ||' '|| trim( customer\_city) ||' '||aadharid from customer where customer\_name in ( select customer\_name from borrower group by customer\_name having count(\*)>3);**



4. Display pairs of customers who live in the same city.

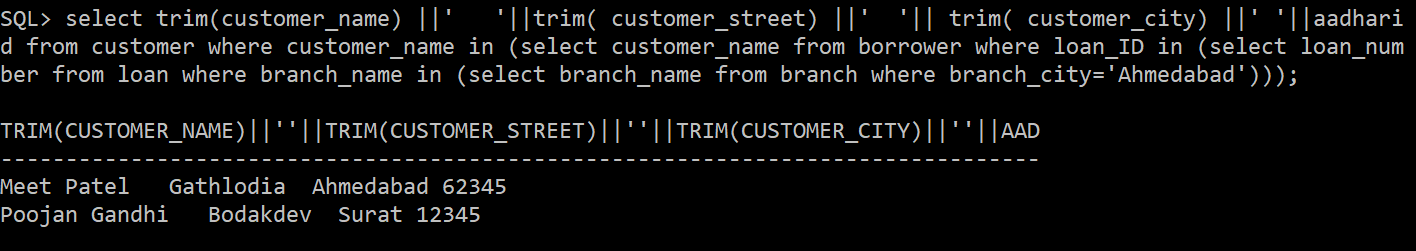
5. Display details of the customers whose name is the longest.

**select trim(customer\_name) ||' '||trim( customer\_street) ||' '|| trim( customer\_city) ||' '||aadharid from customer where length(customer\_name)= (select max(length(customer\_name)) from customer);**



6. Display details of borrowers who have taken loan from 'Ahmedabad' branch

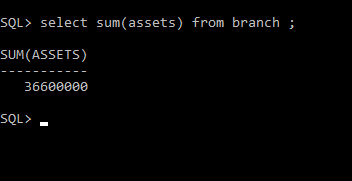
select trim(customer\_name) ||' '||trim( customer\_street) ||' '|| trim( customer\_city) ||' '||aadharid from customer **where customer\_name in (select customer\_name from borrower where loan\_ID in (select loan\_number from loan where branch\_name in (select branch\_name from branch where branch\_city='Ahmedabad')));**



7. Display details of borrowers who have taken **total** loan of more than Rs. 50,00,000.

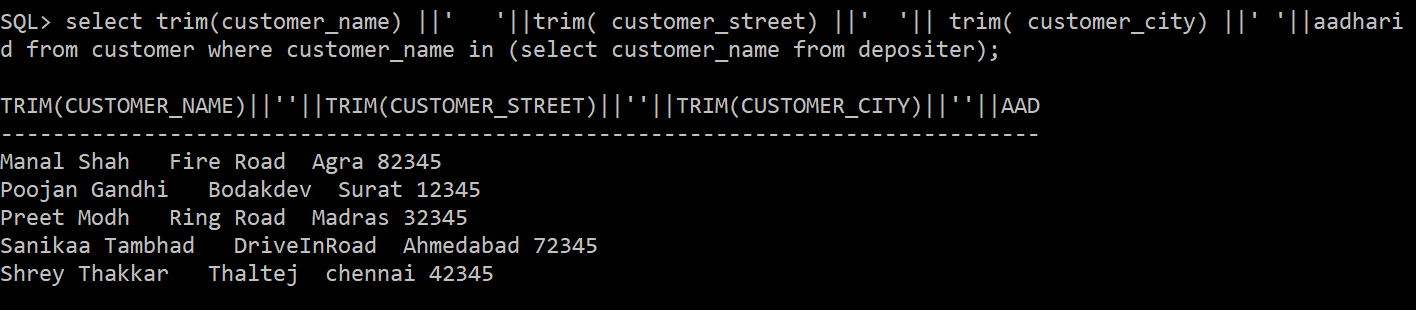
8. Display total assets of all the branches.

select sum(assets) from branch ;



9. Display details of customers who are depositors

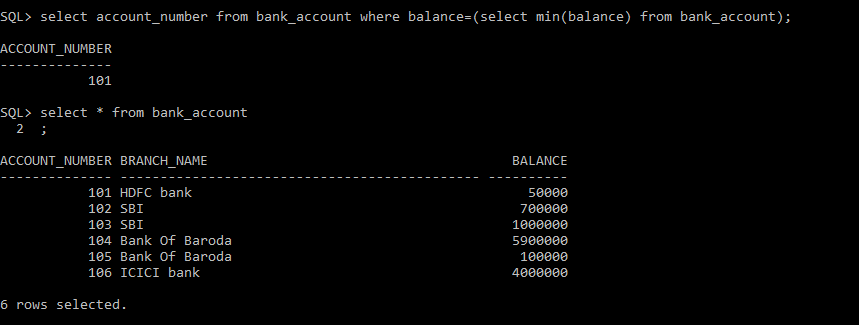
select trim(customer\_name) ||' '||trim( customer\_street) ||' '|| trim( customer\_city) ||' '||aadharid from customer where customer\_name in (select customer\_name from depositer);



10. Display left outer join of customer and depositor table

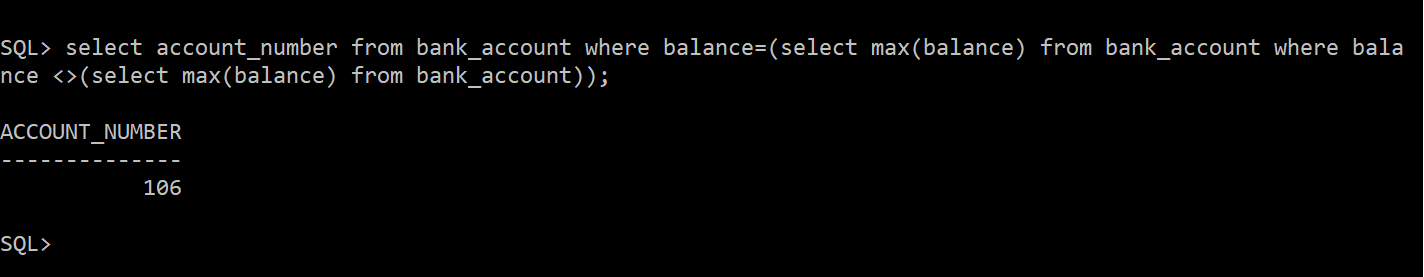
12. Display account number in which balance is minimum.

**select account\_number from bank\_account where balance=(select min(balance) from bank\_account);**



13. Display account number in which balance is second highest

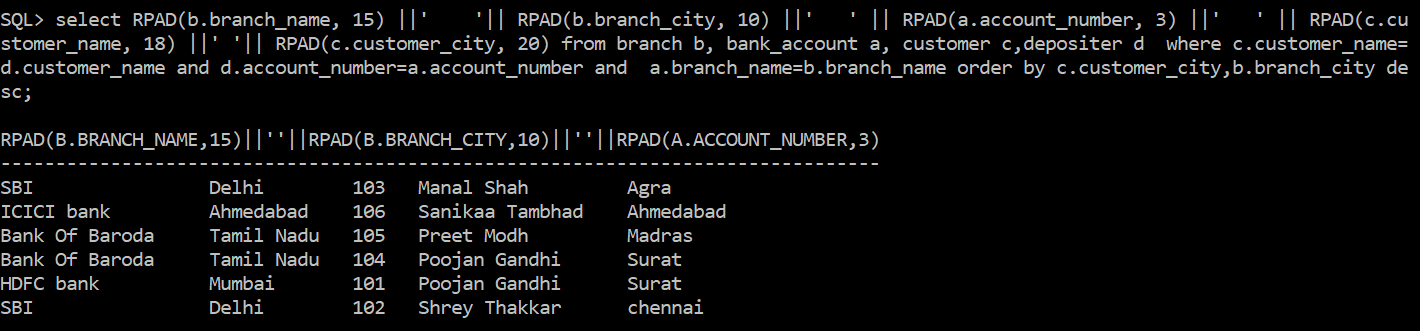
**select account\_number from bank\_account where balance=(select max(balance) from bank\_account where balance <>(select max(balance) from bank\_account));**



14. Display branch name, branch city, account number, customer name and customer city in ascending order of customer city and descending order of branch city

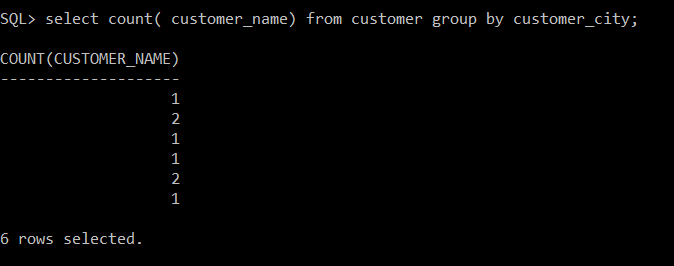
**select b.branch\_name||' '||b.branch\_city||' ' ||a.account\_number||' ' ||c.customer\_name||' '||c.customer\_city from branch b, bank\_account a, customer c,depositer d where c.customer\_name=d.customer\_name and d.account\_number=a.account\_number and a.branch\_name=b.branch\_name order by c.customer\_city,b.branch\_city desc;**

**OR**

**select RPAD(b.branch\_name, 15) ||' '|| RPAD(b.branch\_city, 10) ||' ' || RPAD(a.account\_number, 3) ||' ' || RPAD(c.customer\_name, 18) ||' '|| RPAD(c.customer\_city, 20) from branch b, bank\_account a, customer c,depositer d where c.customer\_name=d.customer\_name and d.account\_number=a.account\_number and a.branch\_name=b.branch\_name order by c.customer\_city,b.branch\_city desc;**

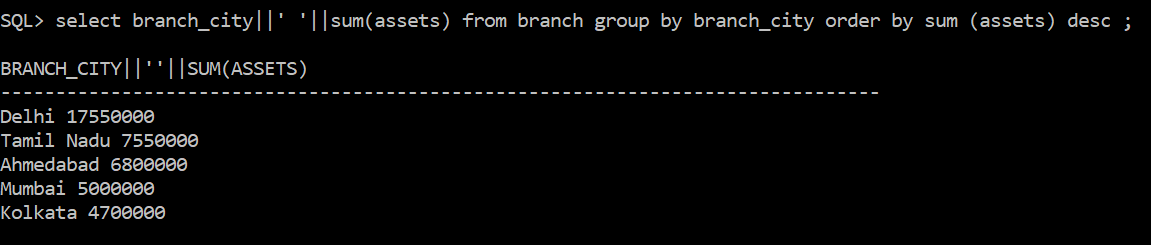
15. Display total no. of customers in each city

**select count( customer\_name) from customer group by customer\_city;**



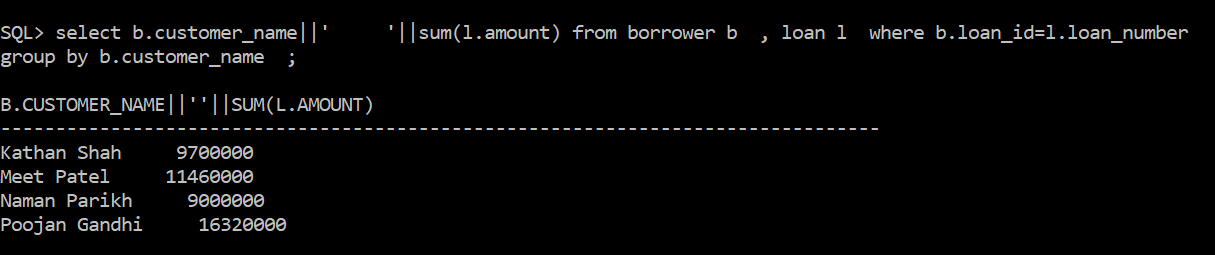
16. Display city wise total assets in descending order of total assets.

**select branch\_city||' '||sum(assets) from branch group by branch\_city order by sum (assets) desc ;**



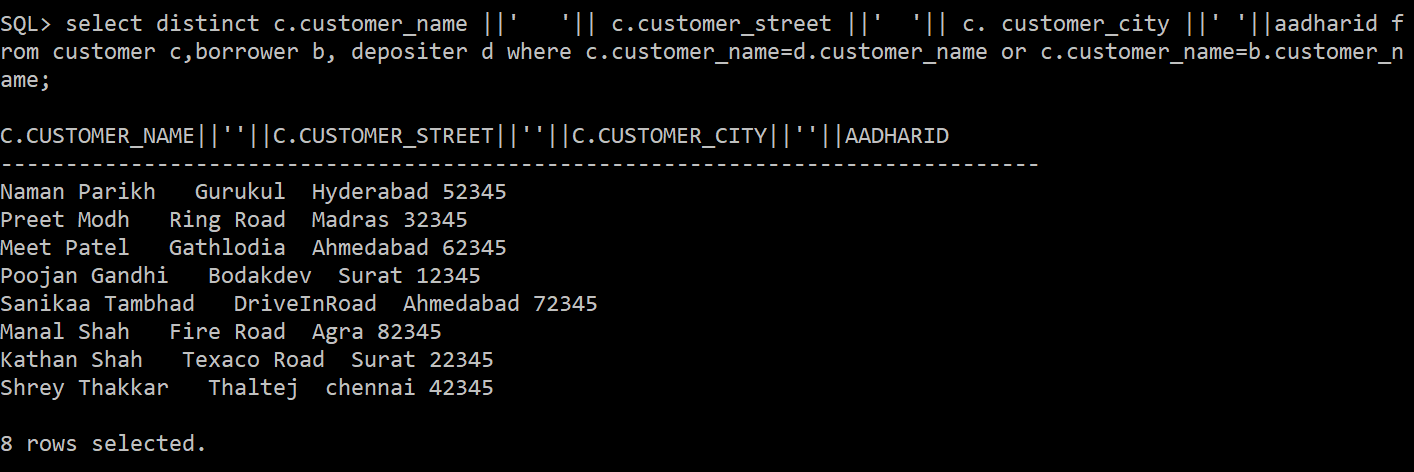
17. Display borrower names with total loan amount taken.

**select b.customer\_name||' '||sum(l.amount) from borrower b , loan l where b.loan\_id=l.loan\_number group by b.customer\_name ;**



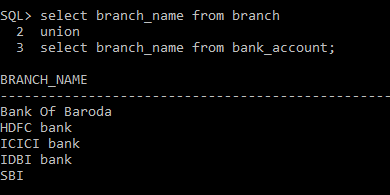
18. Display details of the customers who are depositors as well as borrower.

**select distinct c.customer\_name ||' '|| c.customer\_street ||' '|| c. customer\_city ||' '||aadharid from customer c,borrower b, depositer d where c.customer\_name=d.customer\_name or c.customer\_name=b.customer\_name;**



20. Display union of branch and account. (Use “union” operator)

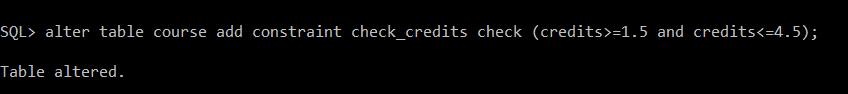
**select branch\_name from branch union select branch\_name from bank\_account;**



2ND TABLE

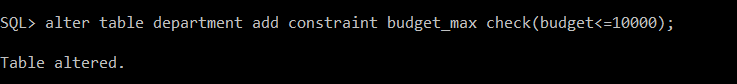
1. Add constraints:
   * + 1. Credits in course table should be minimum 1.5 and maximum 4.5

**alter table course add constraint check\_credits check (credits>=1.5 and credits<=4.5);**



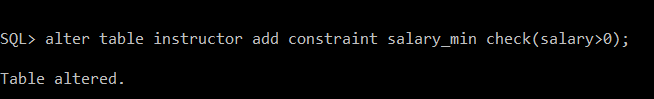
* + - 1. Budget in department table can’t exceed Rs. 10,000.

**alter table department add constraint budget\_max check(budget<=10000);**



* + - 1. Salary in instructor table should be more than 0.

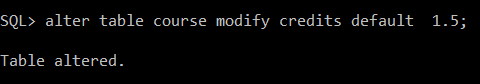
**alter table instructor add constraint salary\_min check(salary>0);**



1. Default constraints

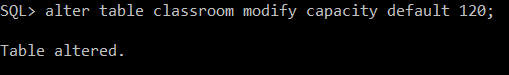
i. 1.5 credits in course table.

**alter table course modify credits default 1.5;**



ii. 120 capacity in classroom table.

**alter table classroom modify capacity default 120;**

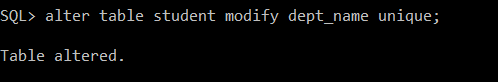


iii. 2019 (Don’t set value directly. Extract year from current date)

1. Unique constraints:

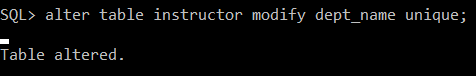
i. Dept\_name in student table.

**alter table student modify dept\_name unique;**



ii. Dept\_name in instructor table.

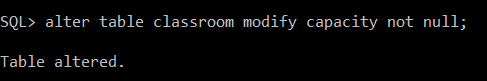
**alter table instructor modify dept\_name unique;**



1. NOT NULL

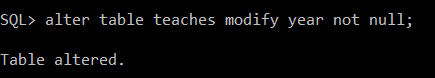
i. Capacity in classroom table.

**alter table classroom modify capacity not null;**



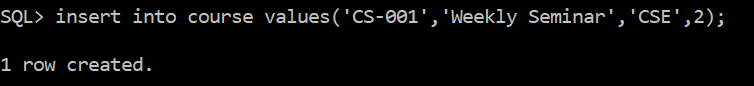
ii. Year in teaches table

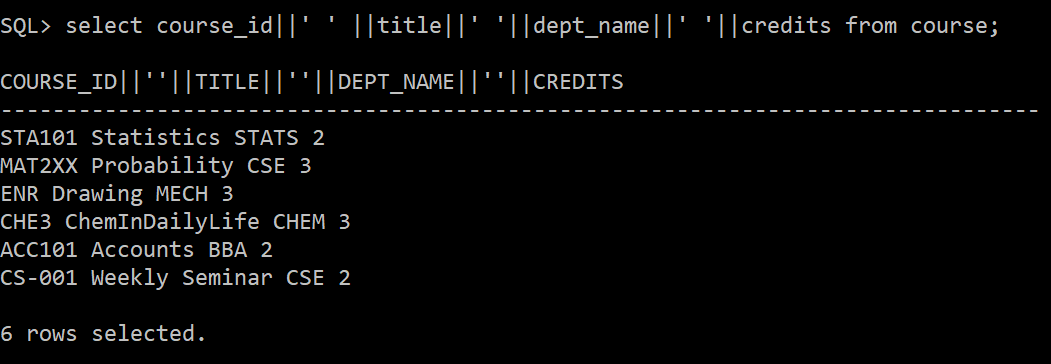
**alter table teaches modify year not null;**



1. Create a new course “CS-001”, titled “Weekly Seminar”, with 2 credits.

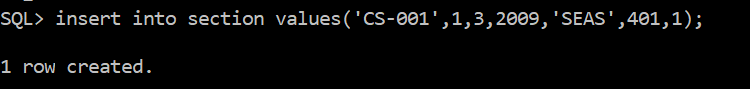
**insert into course values('CS-001','Weekly Seminar','CSE',2);**

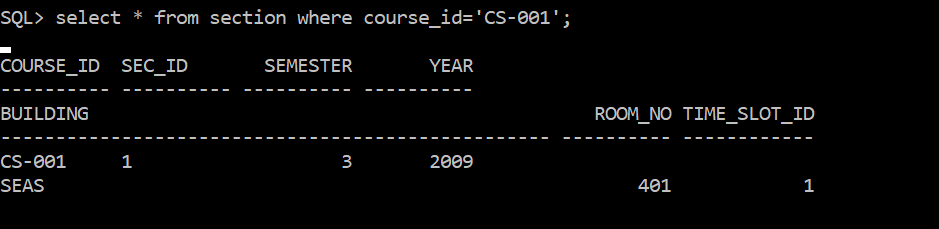




1. Create a section of the course “CS-001” in Autumn 2009, with sec\_id of 1

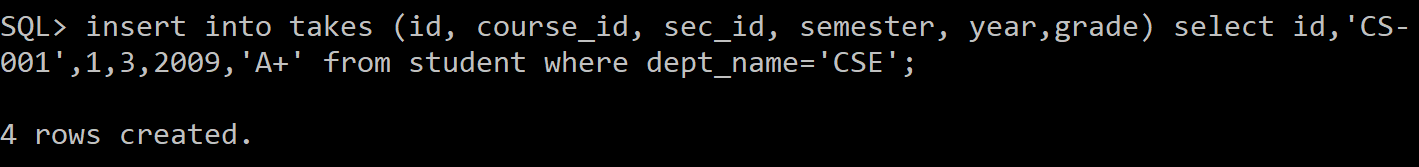
**insert into section values('CS-001',1,3,2009,'SEAS',401,1);**

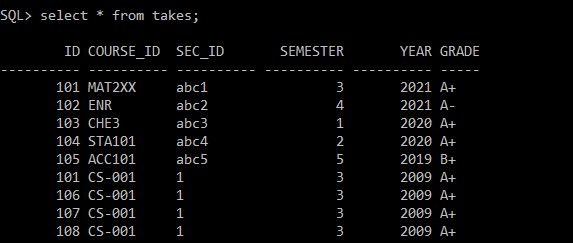
****

****

1. Enroll every student in the Comp. Sci. department in section 1.

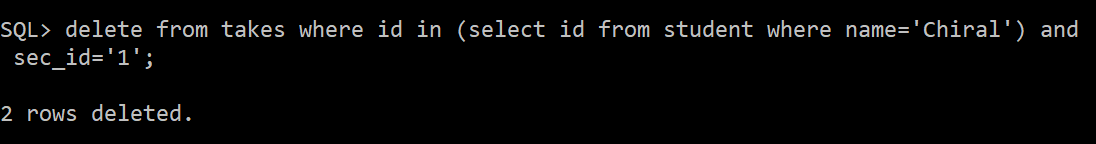
**insert into takes (id, course\_id, sec\_id, semester, year,grade) select id,'CS-001',1,3,2009,'A+' from student where dept\_name='CSE';**





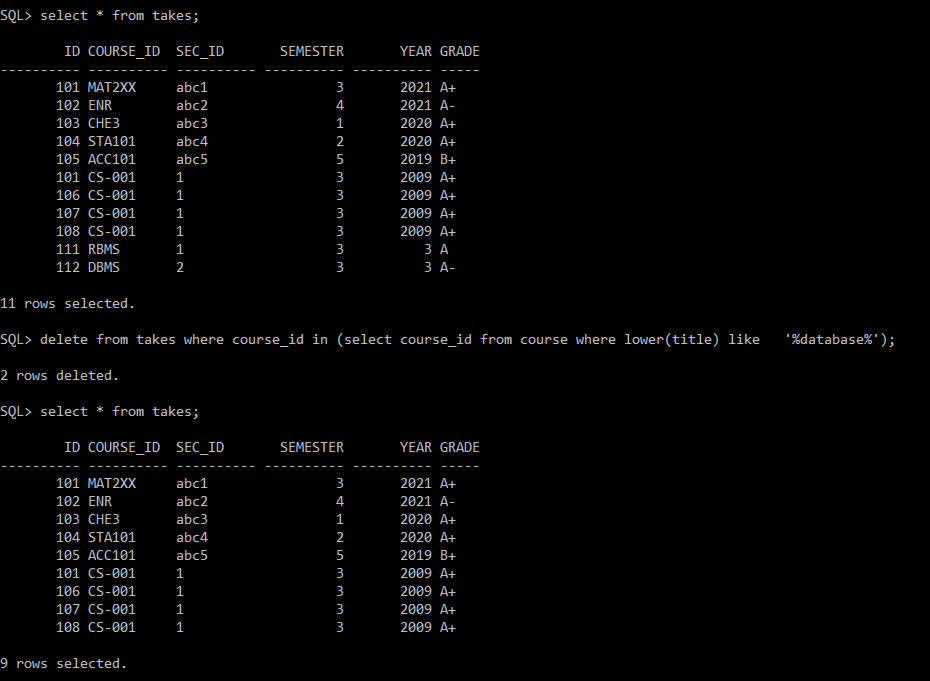
1. Delete enrollment in section 1 where student name is Chiral

**delete from takes where id in (select id from student where name='Chiral') and sec\_id='1';**



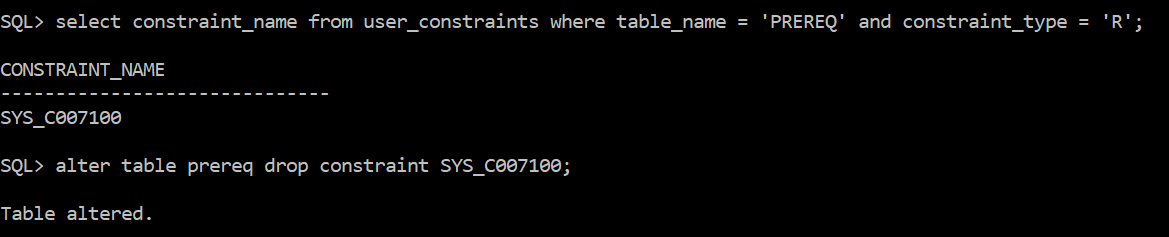
1. Delete all “takes” tuples corresponding to any section of any course with the word “database” as a part of the title, ignore case when matching the word with the title.

**delete from takes where course\_id in (select course\_id from course where lower(title) like '%database%');**



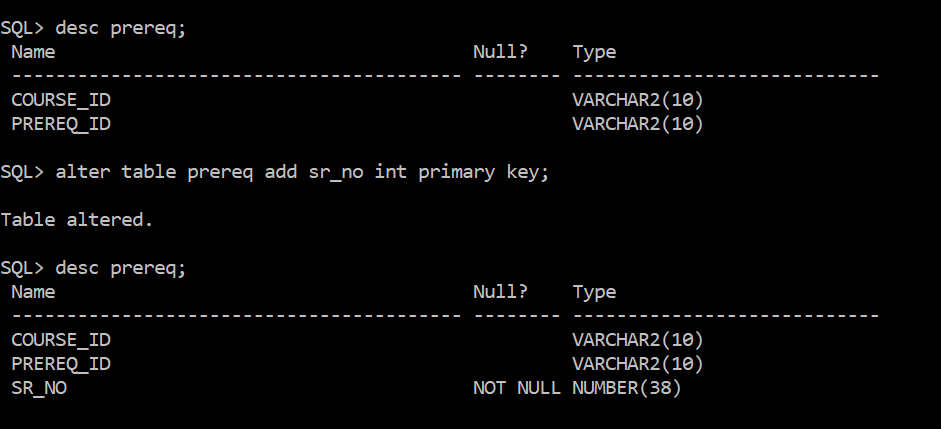
1. Drop “foreign key” constraint from “prereq” table.

* **select constraint\_name from user\_constraints where table\_name = 'PREREQ' and constraint\_type = 'R';**
* **alter table prereq drop constraint SYS\_C007100;**



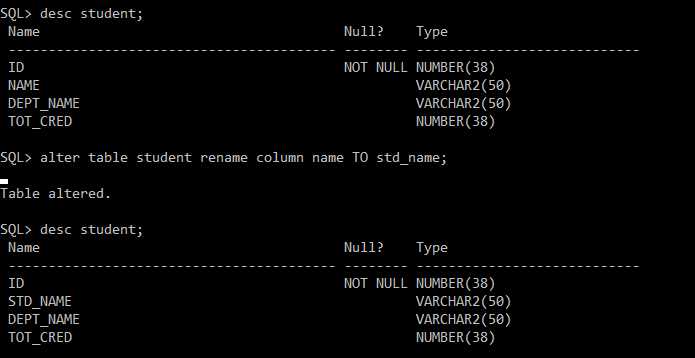
1. Add one field sr\_no “prereq” table and make it a primary key.

**alter table prereq add sr\_no int primary key;**



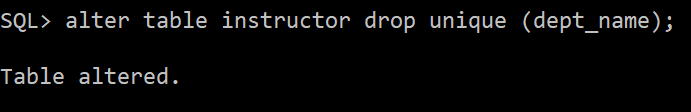
1. Rename “name” field of student table to “std\_name”.

**alter table student rename column name TO std\_name;**



1. Drop unique constraint from “dept\_name” field of instructor table.

**alter table instructor drop unique (dept\_name);**



1. Drop not null constraint from “capacity” field of “classroom” table.

**alter table classroom modify capacity null;**

