Aim: Write SQL queries using SQL Joins.

Objectives:

- Learn how to use SQL joins effectively to gather and connect information from various tables.
- Become skilled in using both inner and outer joins to analyze data thoroughly.
- To access one or more tables together through a select statement.

Tools Used: MySQL Workbench

Concepts:

SQL joins are used to combine rows from two or more tables based on a related column between them. They allow you to query and retrieve data from multiple tables simultaneously, creating a single result set that combines information from these tables. There are several types of SQL joins:

1. Inner Join: This type of join returns only the rows that have matching values in both tables. It essentially filters out rows that do not have a match in the other table. Inner joins are commonly used to combine related data.

2. **Left (Outer) Join:** A left join returns all the rows from the left table and the matched rows from the right table. If there are no matches in the right table, it will still return the rows from the left table with NULL values in the columns from the right table.

3. Right (Outer) Join: A right join is similar to a left join but returns all the rows from the right table and the matched rows from the left table. Rows from the left table with no match in the right table will have NULL values in the columns from the left table.

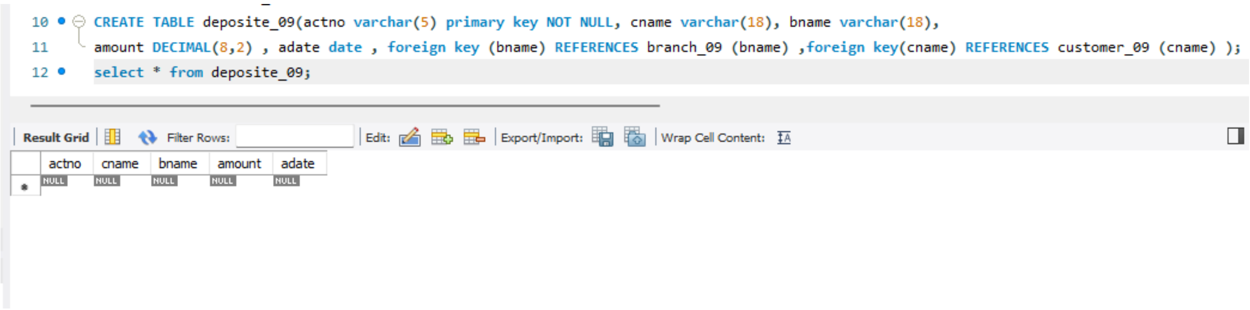
4. Full (Outer) Join: A full join returns all rows when there is a match in either the left or right table. It includes rows from both tables, filling in NULL values where there is no match.

5. Cross join : A cross join in SQL combines every row from one table with every row from another table, creating a result set that is the product of the number of rows in both tables.

Problem Statement:

1) CREATE table deposit_09 (actno varchar(5), cname varchar(18), bname varchar(18), amount number(8,2), adate date);
actno is primary key
bname and cname are foreign keys

Solution :



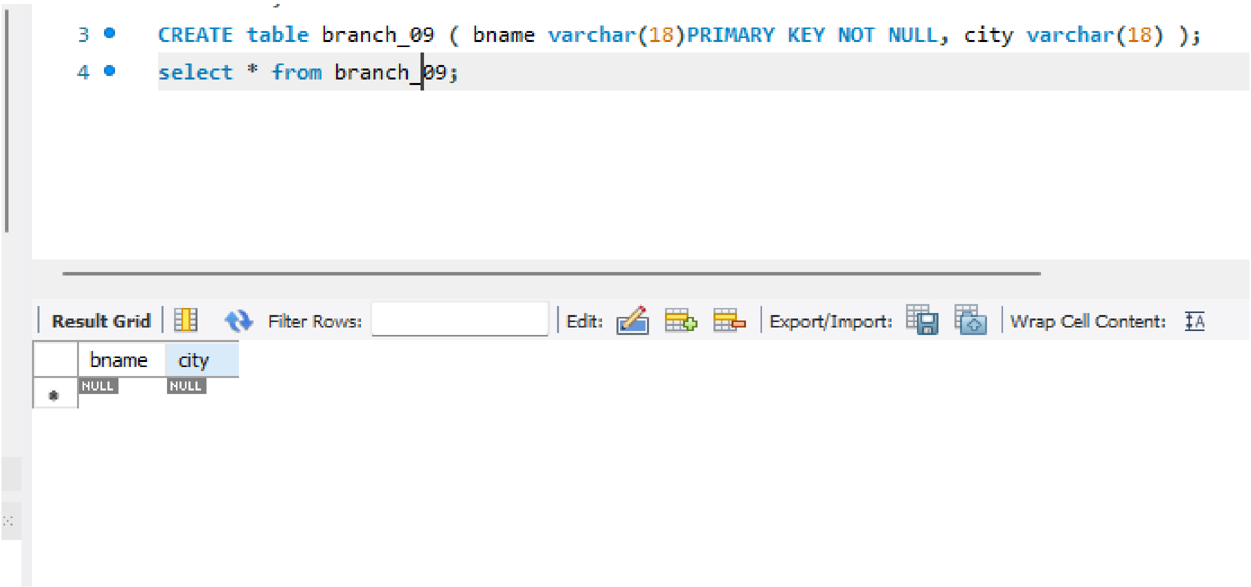
```
CREATE TABLE deposite_09(actno varchar(5) primary key NOT NULL, cname varchar(18), bname varchar(18),
amount DECIMAL(8,2) , adate date , foreign key (bname) REFERENCES branch_09 (bname) ,foreign key(cname) REFERENCES customer_09 (cname) );
select * from deposite_09;
```

	actno	cname	bname	amount	adate
*	NULL	NULL	NULL	NULL	NULL

Problem Statement:

2) CREATE table branch_09 (bname varchar(18), city varchar(18));
bname is a primary key

Solution :



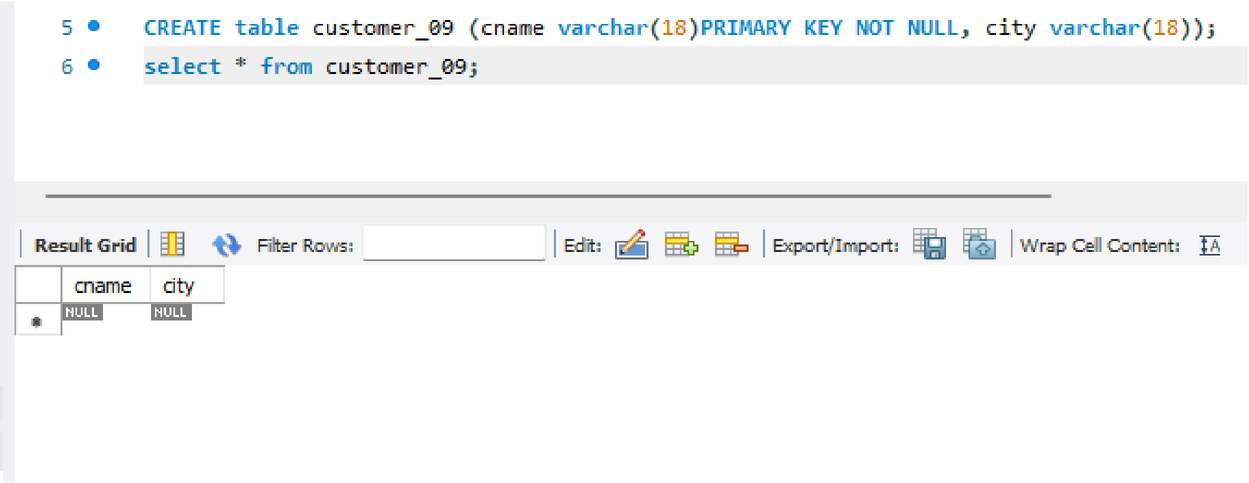
```
CREATE table branch_09 ( bname varchar(18)PRIMARY KEY NOT NULL, city varchar(18) );
select * from branch_09;
```

	bname	city
*	NULL	NULL

Problem Statement:

3) CREATE table customer_09 (cname varchar2(18), city varchar2(18));
cname is primary key

Solution :



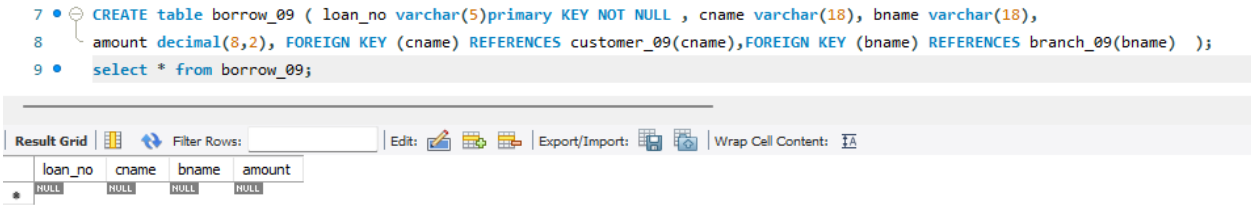
```
CREATE table customer_09 (cname varchar(18)PRIMARY KEY NOT NULL, city varchar(18));
select * from customer_09;
```

	cname	city
*	NULL	NULL

Problem Statement:

4) CREATE table borrow_09 (loan_no varchar2(5), cname varchar(18), bname varchar(18), amount number(8,2)); loan_no is primary key , bname and cname are foreign keys

Solution :



```
CREATE table borrow_09 ( loan_no varchar(5)primary KEY NOT NULL , cname varchar(18), bname varchar(18),
amount decimal(8,2), FOREIGN KEY (cname) REFERENCES customer_09(cname),FOREIGN KEY (bname) REFERENCES branch_09(bname)  );
select * from borrow_09;
```

	loan_no	cname	bname	amount
*	NULL	NULL	NULL	NULL

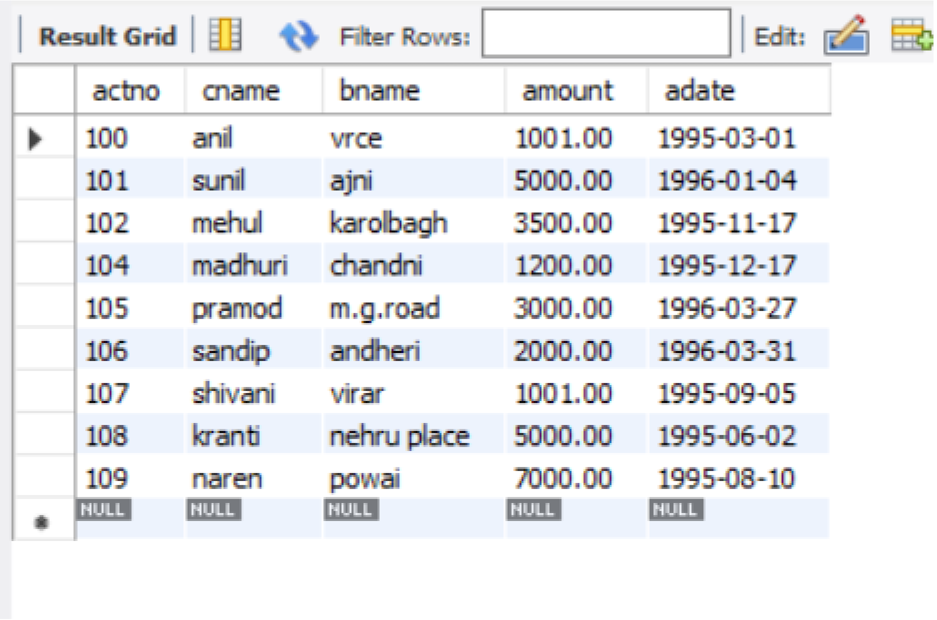
Basic Queries

Problem Statement:

1) **List all data from the deposit table.**

solution :

select * from deposite_09;



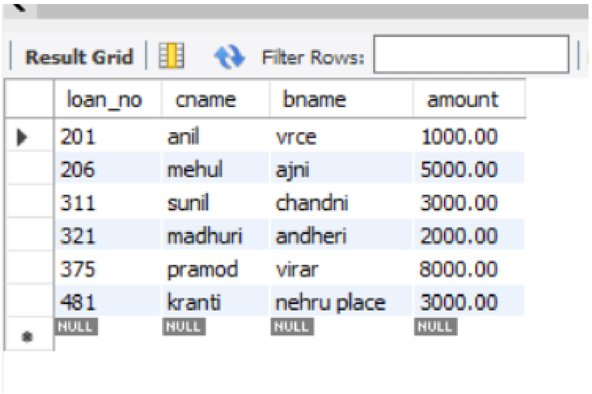
actno	cname	bname	amount	adate
100	anil	vrce	1001.00	1995-03-01
101	sunil	ajni	5000.00	1996-01-04
102	mehul	karolbagh	3500.00	1995-11-17
104	madhuri	chandni	1200.00	1995-12-17
105	pramod	m.g.road	3000.00	1996-03-27
106	sandip	andheri	2000.00	1996-03-31
107	shivani	virar	1001.00	1995-09-05
108	kranti	nehru place	5000.00	1995-06-02
109	naren	powai	7000.00	1995-08-10
NULL	NULL	NULL	NULL	NULL

Problem Statement:

2) List all data from the borrow table.

Solution :

select * from borrow_09;



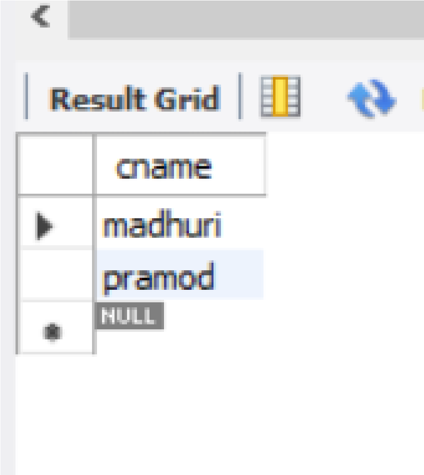
loan_no	cname	bname	amount
201	anil	vrce	1000.00
206	mehul	ajni	5000.00
311	sunil	chandni	3000.00
321	madhuri	andheri	2000.00
375	pramod	virar	8000.00
481	kranti	nehru place	3000.00
NULL	NULL	NULL	NULL

Problem Statement:

3) List names of customers living in Nagpur City.

Solution :

```
select cname from customer_09 where city = 'nagpur';
```



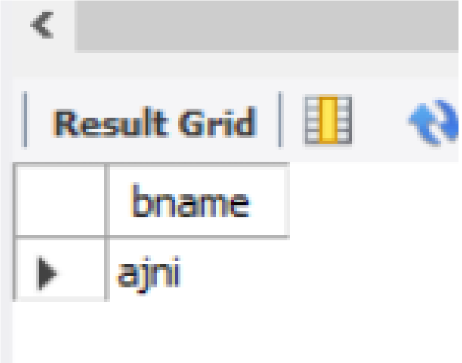
cname
madhuri
pramod
NULL

Problem Statement:

4) List names of borrowers having loan number 206.

Solution :

```
select bname from borrow_09 where loan_no=206;
```



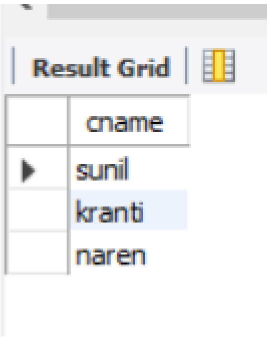
bname
ajni

Problem Statement:

5) List names of depositors having amount greater than 4000

Solution :

select cname from deposite_09 where amount>4000;



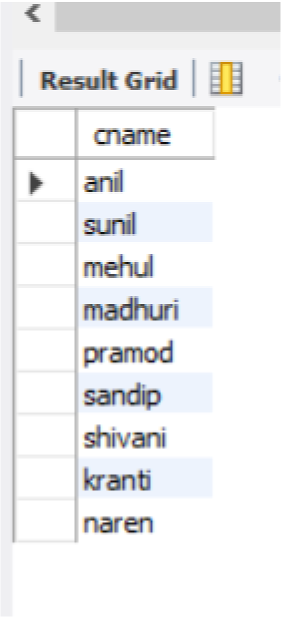
```
cname
sunil
kranti
naren
```

Problem Statement:

6) List names of customers who opened account after date 1/12/95.

Solution :

select cname from deposite_09 where adate > 1995-12-01;



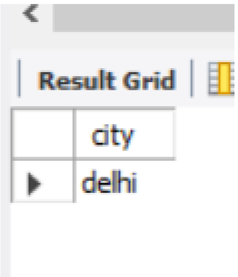
```
cname
anil
sunil
mehul
madhuri
pramod
sandip
shivani
kranti
naren
```

Problem Statement:

7) List name of the city where the Karol Bagh branch is located

Solution :

select city from branch_09 where bname = 'karolbagh';

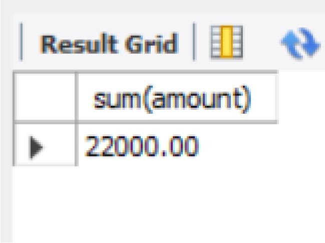


	city
▶	delhi

Problem Statement:

8) List total loan.

Solution :
select sum(amount) from borrow_09;

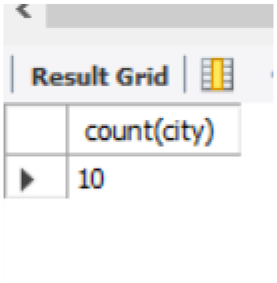


	sum(amount)
▶	22000.00

Problem Statement:

9) List total number of customer cities.

Solution :
select count(city) from customer_09;



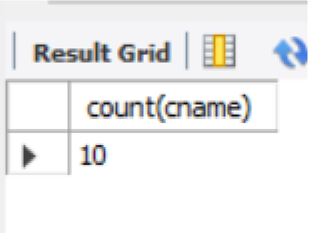
	count(city)
▶	10

Problem Statement:

10) Count total number of customers

Solution :

select count(cname) from customer_09;



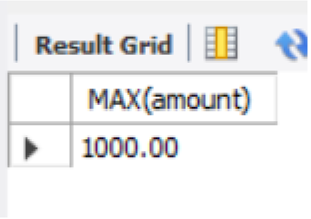
count(cname)
10

Problem Statement:

11) List maximum loan from VRCE branch

Solution :

select max(amount) from borrow_09 where bname = 'vrce';



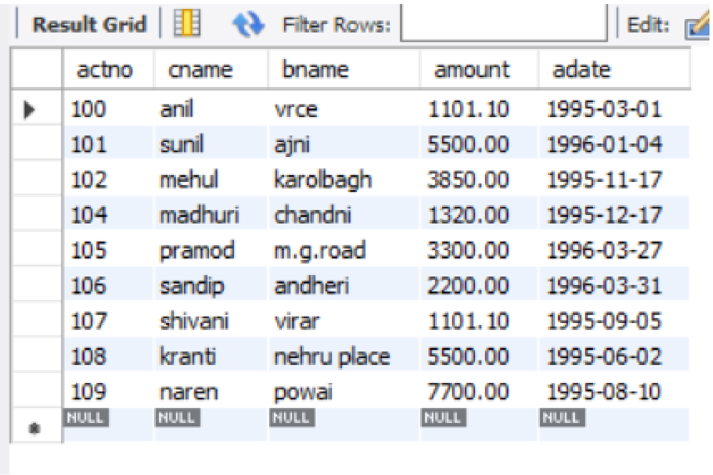
MAX(amount)
1000.00

Problem Statement:

12) Add 10% interest to all depositors.

Solution :

update deposite_09 set amount = amount * 1.10;



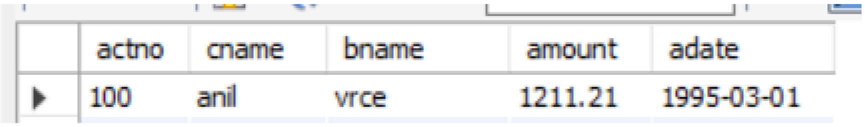
actno	cname	bname	amount	adate
100	anil	vrce	1101.10	1995-03-01
101	sunil	ajni	5500.00	1996-01-04
102	mehul	karolbagh	3850.00	1995-11-17
104	madhuri	chandni	1320.00	1995-12-17
105	pramod	m.g.road	3300.00	1996-03-27
106	sandip	andheri	2200.00	1996-03-31
107	shivani	virar	1101.10	1995-09-05
108	kranti	nehru place	5500.00	1995-06-02
109	naren	powai	7700.00	1995-08-10
NULL	NULL	NULL	NULL	NULL

Problem Statement:

13) Add 10% interest to all depositors having VRCE branch

Solution :

update deposite_09 set amount = amount * 1.10 where bname = 'vrce';



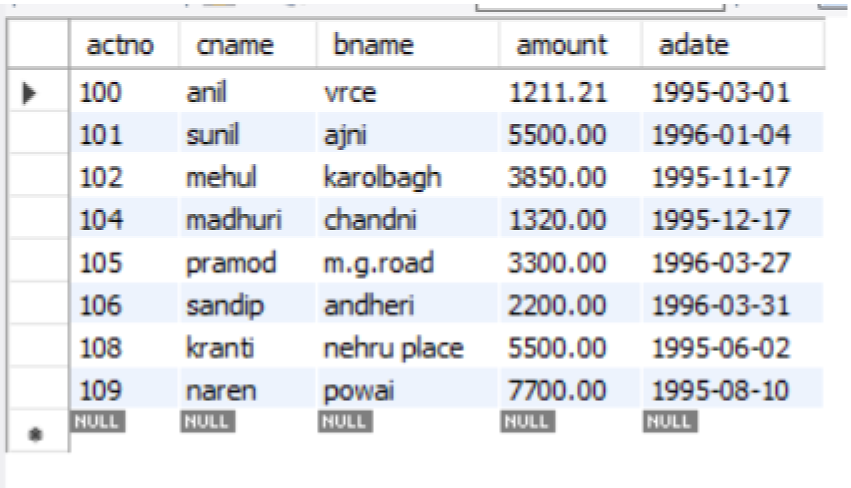
	actno	cname	bname	amount	adate
▶	100	anil	vrce	1211.21	1995-03-01

Problem Statement:

14) Delete depositors if the branch is Virar and the depositor name is Shivani.

Solution :

delete from deposite_09 where bname = 'virar' and cname = 'shivani';



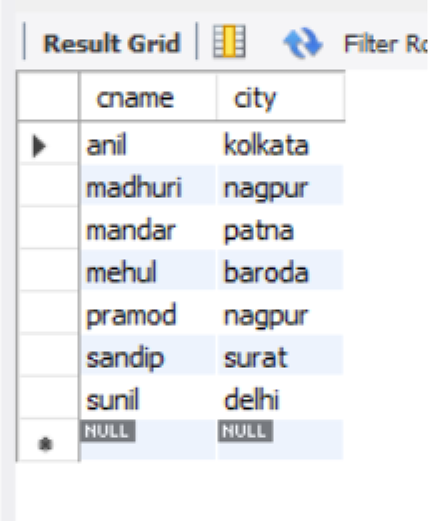
	actno	cname	bname	amount	adate
▶	100	anil	vrce	1211.21	1995-03-01
	101	sunil	ajni	5500.00	1996-01-04
	102	mehul	karolbagh	3850.00	1995-11-17
	104	madhuri	chandni	1320.00	1995-12-17
	105	pramod	m.g.road	3300.00	1996-03-27
	106	sandip	andheri	2200.00	1996-03-31
	108	kranti	nehru place	5500.00	1995-06-02
	109	naren	powai	7700.00	1995-08-10
*	NULL	NULL	NULL	NULL	NULL

Problem Statement:

15) Delete customers from Mumbai City.

Solution :

set foreign_key_checks = 0;
delete from customer_09 where city= 'mumbai';



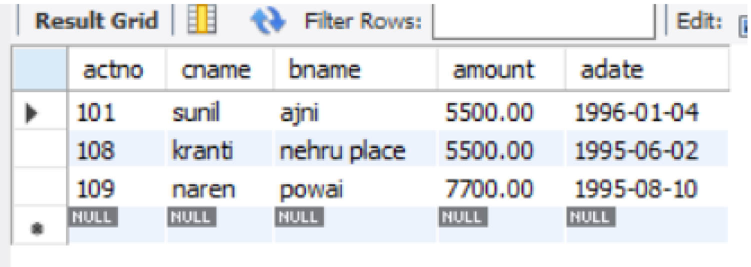
cname	city
anil	kolkata
madhuri	nagpur
mandar	patna
mehul	baroda
pramod	nagpur
sandip	surat
sunil	delhi
NULL	NULL

Problem Statement:

16) Delete depositor having deposit less than 5000

Solution :

delete from deposite_09 where amount < 5000;



actno	cname	bname	amount	adate
101	sunil	ajni	5500.00	1996-01-04
108	kranti	nehru place	5500.00	1995-06-02
109	naren	powai	7700.00	1995-08-10
NULL	NULL	NULL	NULL	NULL

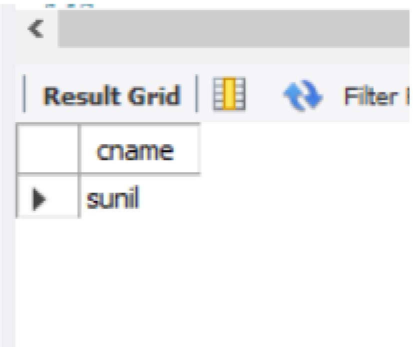
Questions on Joins

Problem Statement:

1. List names of depositors having same branch as the branch of SUNIL.

Solution :

```
select  d1.cname
from deposite_9 d1
inner join deposite_9 d2 on d1.bname = d2.bname
where d2.cname = 'sunil';
```



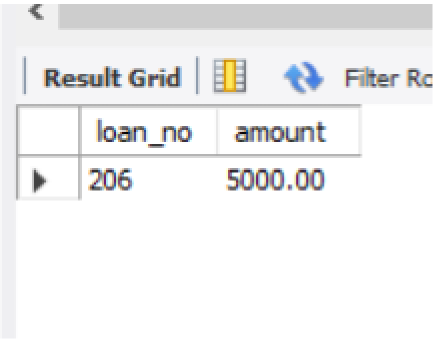
cname
sunil

Problem Statement:

2. List LoanNo and LoanAmount of borrowers having the same branch as the depositor SUNIL.

Solution :

```
select  b.loan_no, b.amount
from borrow_09 as b
inner join deposite_9 as d on b.bname = d.bname
where d.cname = 'sunil';
```



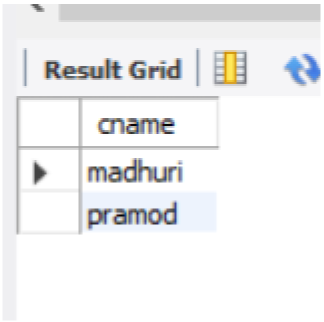
loan_no	amount
206	5000.00

Problem Statement:

3.List all depositors living in NAGPUR

Solution :

```
select c.cname from customer_09 as c
inner join deposite_09 as d on c.cname = d.cname where c.city = 'nagpur';
```



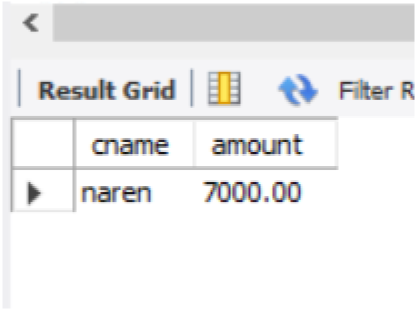
cname
madhuri
pramod

Problem Statement:

5. List names of customers having maximum deposite.

Solution :

```
select c.cname , amount from customer_09 as c
inner join deposite_09 as d on c.cname = d.cname
where d.amount = (select max(amount)from deposite_09);
```



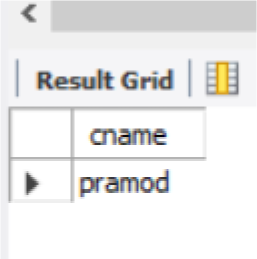
cname	amount
naren	7000.00

6. List names of customers having maximum deposit in the customers living in Nagpur.

Problem Statement:

```
select c.cname from customer_09 c
inner join deposite_09 as d on c.cname = d.cname where c.city = 'nagpur' and d.amount
= (
    select max(amount) from deposite_09 where cname in (select cname from
customer_09 where city = 'nagpur'));
```

Solution :



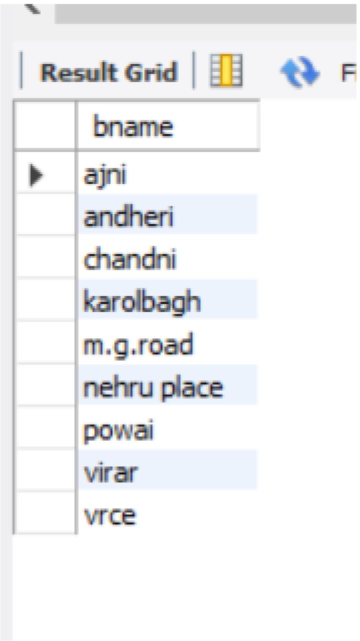
	cname
▶	pramod

Problem Statement:

7. List the names of branches having highest number of depositors.

Solution :

```
select b.bname from branch_09 as b
left join deposite_09 d on b.bname = d.bname
group by b.bname having count(distinct d.cname) = (select max(depositor_count)
    from (select count(distinct d2.cname) as depositor_count from deposite_09 d2 group
by d2.bname) as branch_depositor_counts);
```



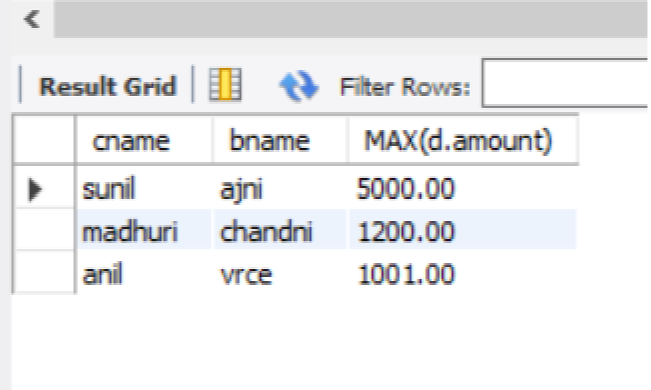
	bname
▶	ajni
	andheri
	chandni
	karolbagh
	m.g.road
	nehru place
	powai
	virar
	vrce

Problem Statement:

8. List the highest deposit of the city where branch of Sunil is located.

Solution:

```
select d.cname, d.bname, MAX(d.amount) from deposite_09 as d
JOIN (select distinct bname from branch_09 where city in (select city from branch_09
where bname in (select bname from deposite_09 where cname = 'sunil' ))) AS b ON
d.bname = b.bname GROUP BY d.cname, d.bname;
```



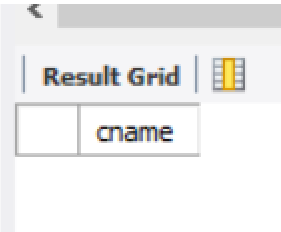
cname	bname	MAX(d.amount)
sunil	ajni	5000.00
madhuri	chandni	1200.00
anil	vrce	1001.00

Problem Statement:

9. List the names of customers having more deposit than the average deposit in their respective branches.

Solution:

```
select distinct c.cname from customer_09 as c
inner join deposite_09 d on c.cname = d.cname
inner join ( select d.bname, avg(d.amount) as branch_avg_deposit from deposite_09 as
d
   group by d.bname) as avg_deposits on d.bname = avg_deposits.bname
where d.amount > avg_deposits.branch_avg_deposit;
```



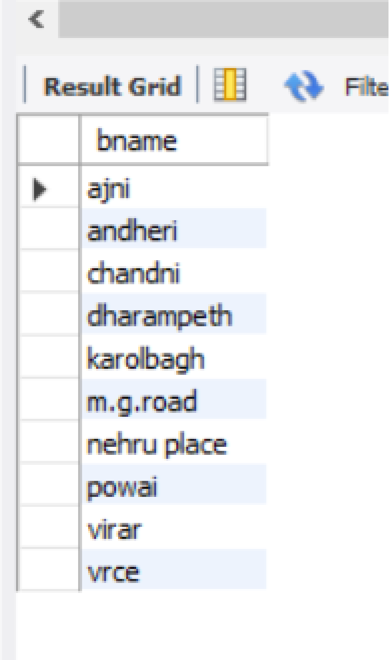
cname

Problem Statement:

10. List the names of branches where number of depositors less than 2.

Solution:

```
select b.bname from branch_09 as b left join deposite_09 as d on b.bname = d.bname
group by b.bname having count(distinct d.cname) < 2;
```



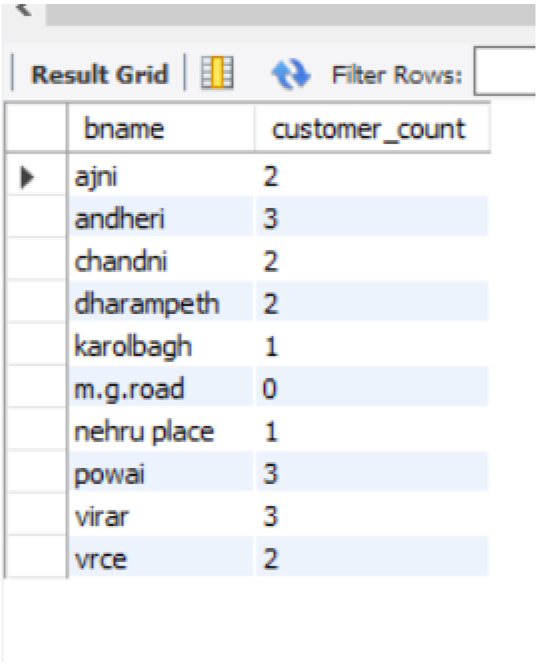
bname
ajni
andheri
chandni
dharampeth
karolbagh
m.g.road
nehru place
powai
virar
vrce

Problem Statement:

11. Count the number of customers living in the city where branch is located.

Solution:

```
select b.bname, count(c.cname) as customer_count from branch_09 as b
left join customer_09 as c on b.city = c.city group by b.bname;
```



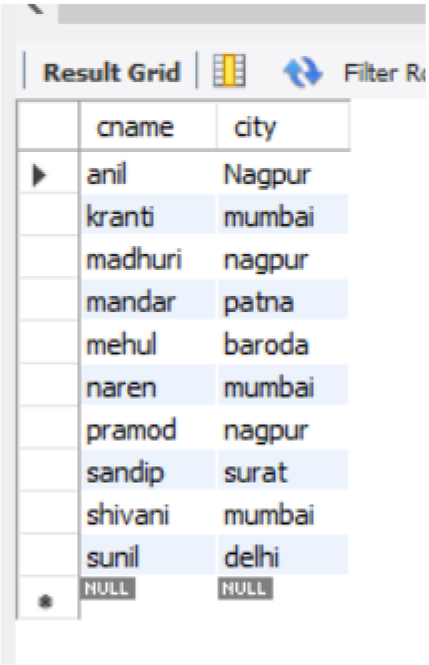
bname	customer_count
ajni	2
andheri	3
chandni	2
dharampeth	2
karolbagh	1
m.g.road	0
nehru place	1
powai	3
virar	3
vrce	2

Problem Statement:

12. Change the living city of the VRCE branch borrowers to Nagpur.

Solution:

```
update customer_09 as c join borrow_09 as b on c.cname = b.cname
set c.city = 'nagpur' where b.bname = 'vrce';
```



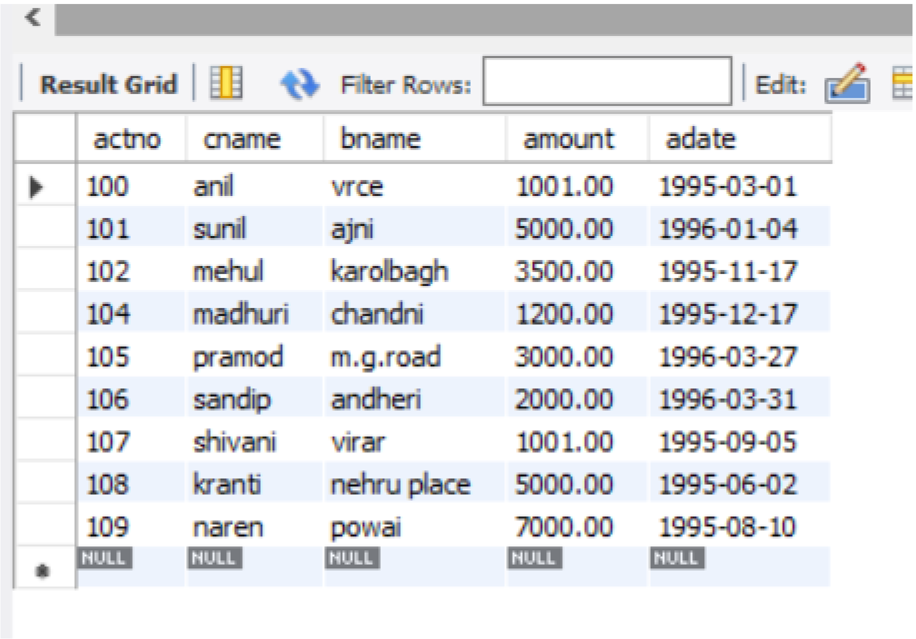
cname	city
anil	Nagpur
kranti	mumbai
madhuri	nagpur
mandar	patna
mehul	baroda
naren	mumbai
pramod	nagpur
sandip	surat
shivani	mumbai
sunil	delhi
NULL	NULL

Problem Statement:

14. Transfer Rs. 100 from account Anil to account Sunil if both are having the same branch.

Solution:

```
update deposite_09 as da join deposite_09 as ds on da.bname = ds.bname
set da.amount = da.amount - 100, ds.amount = ds.amount + 100
where da.cname = 'anil' and ds.cname = 'sunil';
```



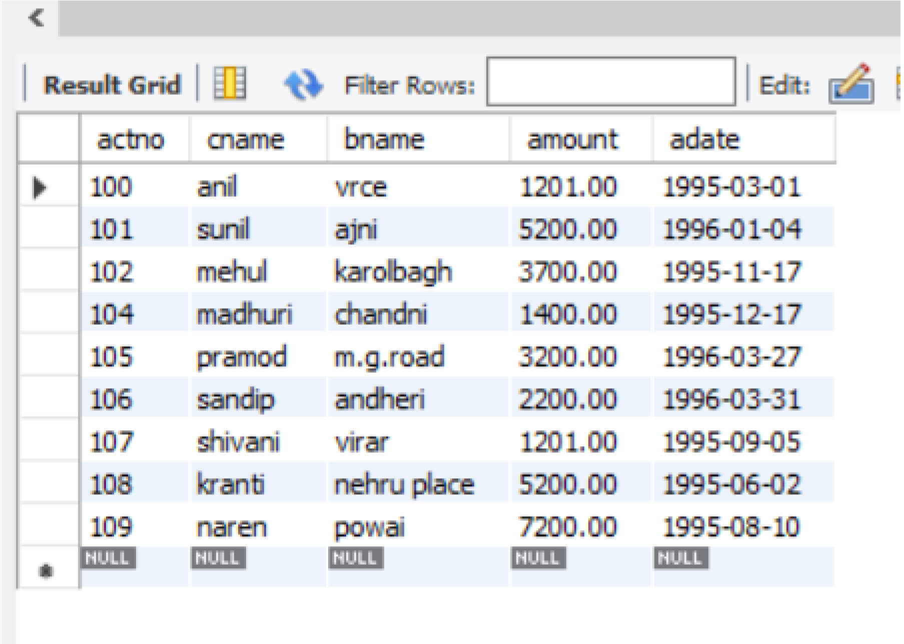
actno	cname	bname	amount	adate
100	anil	vrce	1001.00	1995-03-01
101	sunil	ajni	5000.00	1996-01-04
102	mehul	karolbagh	3500.00	1995-11-17
104	madhuri	chandni	1200.00	1995-12-17
105	pramod	m.g.road	3000.00	1996-03-27
106	sandip	andheri	2000.00	1996-03-31
107	shivani	virar	1001.00	1995-09-05
108	kranti	nehru place	5000.00	1995-06-02
109	naren	powai	7000.00	1995-08-10
NULL	NULL	NULL	NULL	NULL

Problem Statement:

15. Add Rs. 100 to the account of all those depositors who are having the highest deposit amount in their respective branches.

Solution:

```
update deposite_09 as d1
join ( select bname, max(amount) as max_amount from deposite_09 group by bname)
as d2 on d1.bname = d2.bname and d1.amount = d2.max_amount
set d1.amount = d1.amount + 100;
```



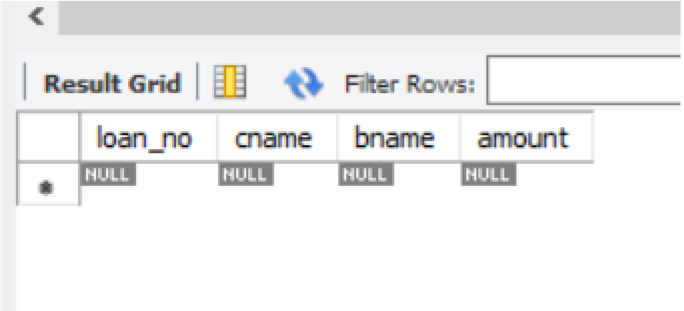
actno	cname	bname	amount	adate
100	anil	vrce	1201.00	1995-03-01
101	sunil	ajni	5200.00	1996-01-04
102	mehul	karolbagh	3700.00	1995-11-17
104	madhuri	chandni	1400.00	1995-12-17
105	pramod	m.g.road	3200.00	1996-03-27
106	sandip	andheri	2200.00	1996-03-31
107	shivani	virar	1201.00	1995-09-05
108	kranti	nehru place	5200.00	1995-06-02
109	naren	powai	7200.00	1995-08-10
NULL	NULL	NULL	NULL	NULL

Problem Statement:

18. Delete borrower of branches having minimum number of customers.

Solution:

```
delete from borrow_09 where bname in (
   select bname from ( select bname, count(distinct cname) as customer_count from
customer_09 group by bname
having customer_count = (select min(customer_count)from (select bname,
count(distinct cname) as customer_count
        from customer_09 group by bname) as branch_customers)) as min_branches);
```



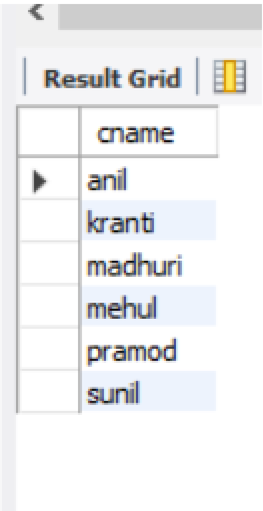
loan_no	cname	bname	amount
NULL	NULL	NULL	NULL

Problem Statement:

19. List names of customers who are depositors as well as borrowers.

Solution:

select distinct c.cname from customer_09 as c inner join deposite_09 as d on c.cname = d.cname
inner join borrow_09 as b on c.cname = b.cname;



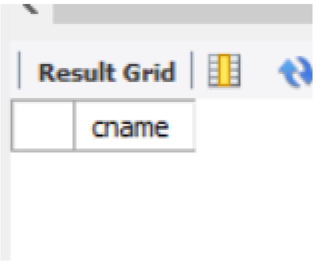
cname
anil
kranti
madhuri
mehul
pramod
sunil

Problem Statement:

21. List the depositors having the same living city as Sunil and the same branch city as Anil

Solution:

select distinct d.cname from deposite_09 d
inner join customer_09 as s on d.cname = s.cname and s.cname = 'sunil' inner join customer_09 as a on d.cname = a.cname and a.cname = 'anil'
where d.bname in (select distinct b.bname from branch_09 b where b.city = s.city);



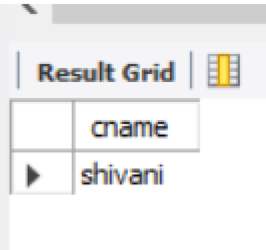
cname

Problem Statement:

22. List the depositors having amount less than 5000 and living in the city as Shivani.

Solution:

```
select distinct d.cname from deposite_09 as d
inner join customer_09 as s on d.cname = s.cname and s.cname = 'shivani' where
d.amount < 5000;
```



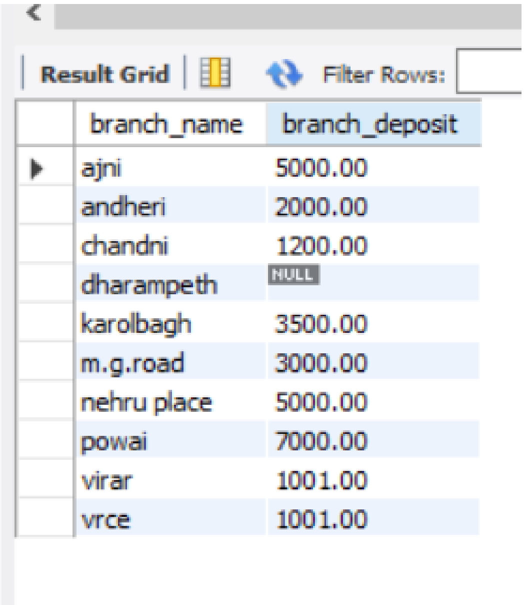
cname
shivani

Problem Statement:

24. List the branch name and branch wise deposit.

Solution:

```
select b.bname as branch_name, sum(d.amount) as branch_deposit from branch_09 b
left join deposite_09 d on b.bname = d.bname group by b.bname;
```



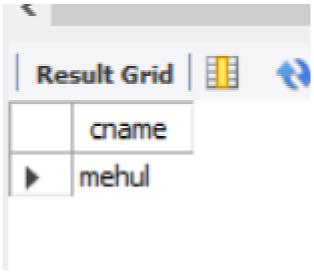
branch_name	branch_deposit
ajni	5000.00
andheri	2000.00
chandni	1200.00
dharampeth	NULL
karolbagh	3500.00
m.g.road	3000.00
nehru place	5000.00
powai	7000.00
virar	1001.00
vrce	1001.00

Problem Statement:

26. List names of depositors who has third highest amount.

Solution:

```
select d.cname
from deposite_09 d
join ( select distinct d1.amount as deposit_amount from deposite_09 d1
   left join deposite_09 d2 on d1.amount < d2.amount group by d1.amount
   having count(distinct d2.amount) = 2) as third_highest on d.amount =
third_highest.deposit_amount;
```



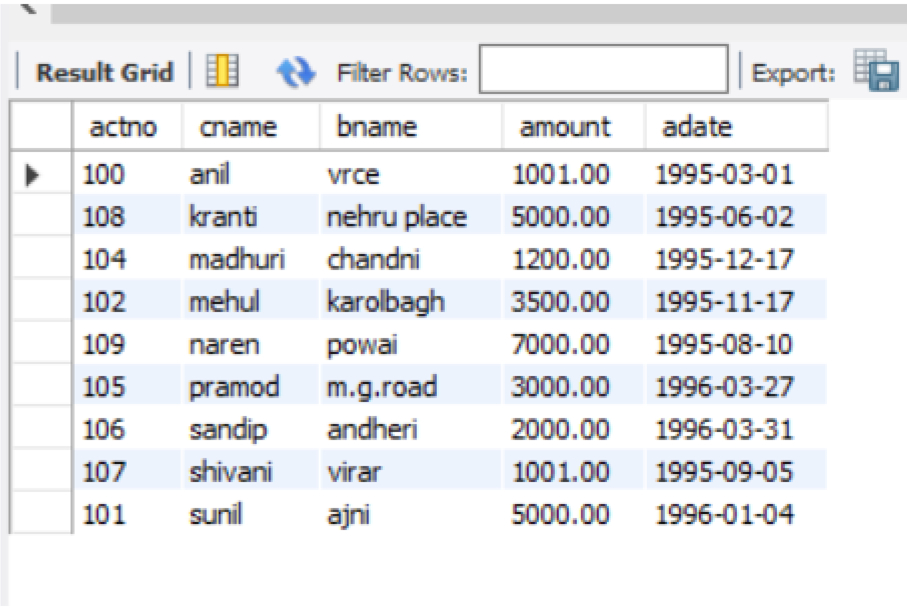
cname
mehul

Problem Statement:

27. List details of depositors according to ascending order of customer names.

Solution:

```
select d.actno, c.cname, d.bname, d.amount, d.adate from deposite_09 d inner join
customer_09 c on d.cname = c.cname
order by c.cname asc;
```



actno	cname	bname	amount	adate
100	anil	vrce	1001.00	1995-03-01
108	kranti	nehru place	5000.00	1995-06-02
104	madhuri	chandni	1200.00	1995-12-17
102	mehul	karolbagh	3500.00	1995-11-17
109	naren	powai	7000.00	1995-08-10
105	pramod	m.g.road	3000.00	1996-03-27
106	sandip	andheri	2000.00	1996-03-31
107	shivani	virar	1001.00	1995-09-05
101	sunil	ajni	5000.00	1996-01-04

Observation :

In this practical , I understand the use of SQL joins query which helps to retrieve data from two or more tables from the database at same time.