

WEEK 4

Create Borrower table

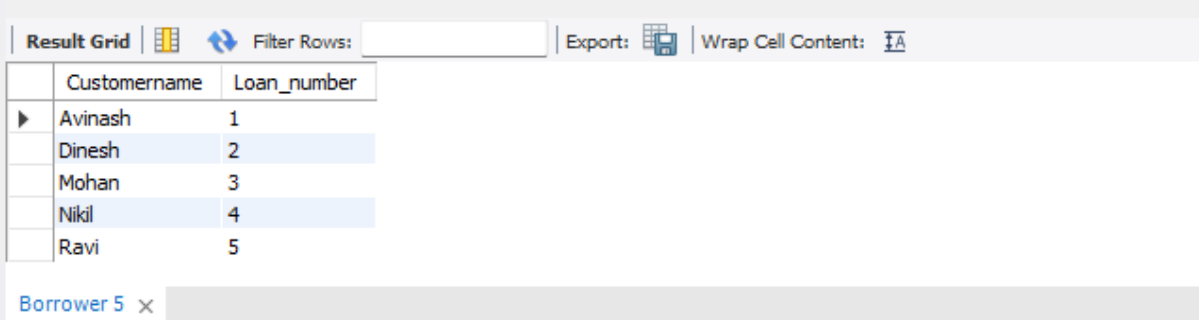
```
create table Borrower(  
Customer_name varchar(20),  
Loan_number int,  
foreign key(Customer_name) references  
bank_customer(customer_name),  
foreign key(Loan_number) references Loan(Loan_number)  
);
```

Insert values into the Borrower table. (INSERTION)

```
insert into Borrower values("Avinash",1,"Dinesh",2,"Mohan",3,"Nikil",4,"Ravi",5);
```

Select new table. (SELECTION)

```
select * from Borrower;
```



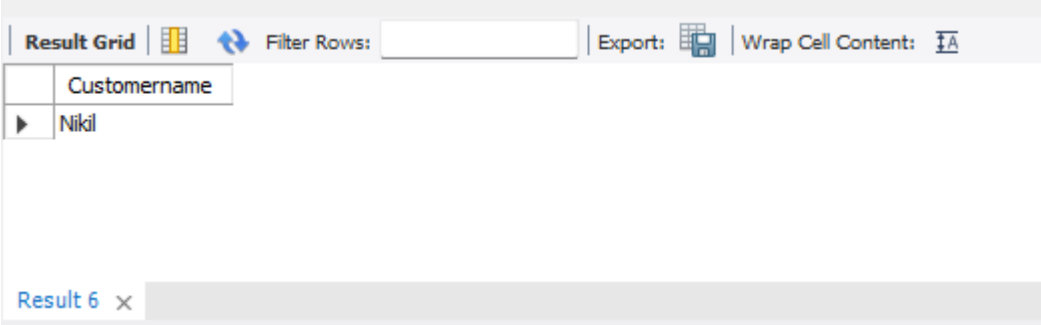
The screenshot shows a database application interface. At the top, there is a toolbar with 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content' options. Below the toolbar is a table with two columns: 'Customername' and 'Loan_number'. The table contains five rows of data. At the bottom, there is a tab labeled 'Borrower 5'.

	Customername	Loan_number
▶	Avinash	1
	Dinesh	2
	Mohan	3
	Nikil	4
	Ravi	5

QUERIES- TO DO:

- 1. Find all the customers who have an account at all the branches located in a specific city (Ex. delhi).**

```
select d.Customer_name from branch1 b, Depositer d,  
bank_account ba where  
b.Branch_city='Delhi' and d.Acc_no=ba.Acc_no and  
b.Branch_name=ba.Branch_name  
group by d.Customername having count(distinct  
b.Branch_name)= (select count(distinct b.Branch_name) from  
branch b where b.Branch_city='Delhi');
```



The screenshot shows a database query result grid. At the top, there are tabs for 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. Below the tabs, there is a table with one column labeled 'Customername' and one row containing the value 'Nikil'. At the bottom left, there is a label 'Result 6' with a close button 'x'.

Customername
Nikil

- 2. Find all customers who have a loan at the bank but do not have an account.**

```

select distinct b.Customer_name from Borrower b, Depositer1
where b.Customer_name NOT IN(
select d.Customer_name from Loan l, Depositer1 d, Borrower b
where l.Loan_number=b.Loan_number and
d.Customername=b.Customername

);

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Customername			
Mohan			

Result 7 x

3. Find all customers who have both an account and a loan at the Bangalore branch.

```

select distinct d.Customername from Depositer d
where d.Customername IN(
    select d.Customername from branch br, Depositer d,
    BankAccount ba
    where br.Branch_city='Bangalore' and
    br.Branch_name=ba.Branch_name
    and ba.accno=d.accno and Customername IN(
        select Customername from Borrower)
);

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Customername			
▶ Avinash			
□ Dinesh			

Depositer 8 x

4. Find the names of all branches that have greater assets than all branches located in Bangalore.

```
select b.Branch_name from Branch b
where b.assets > ALL (
select SUM(b.assets) from Branch b
where b.Branch_City='Bangalore' );
```

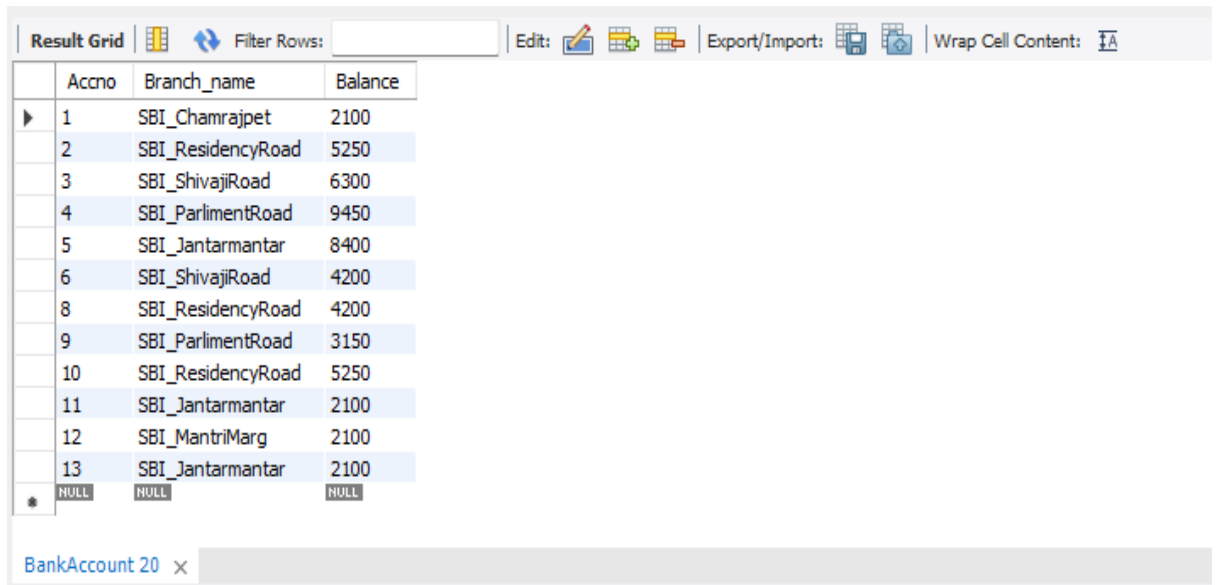
Result Grid	Filter Rows:	Edit:	Export/Import:	Wrap Cell Content:
Branch_name				
▶ SBI_MantriMarg				
✱ NULL				

Branch 9 x

5. Update the Balance of all accounts by 5%

```
UPDATE BankAccount set Balance=(Balance +
(Balance*0.05));
```

5));



	Accno	Branch_name	Balance
▶	1	SBI_Chamrajpet	2100
	2	SBI_ResidencyRoad	5250
	3	SBI_ShivajiRoad	6300
	4	SBI_ParliamentRoad	9450
	5	SBI_Jantarmantra	8400
	6	SBI_ShivajiRoad	4200
	8	SBI_ResidencyRoad	4200
	9	SBI_ParliamentRoad	3150
	10	SBI_ResidencyRoad	5250
	11	SBI_Jantarmantra	2100
	12	SBI_MantriMarg	2100
	13	SBI_Jantarmantra	2100
*	NULL	NULL	NULL

BankAccount 20 x

6. Demonstrate how you delete all account tuples at every branch located in a specific city (Ex. Bombay).

delete ba.* from BankAccount ba, branch b where
branch_city='Bombay' and ba.Branch_name=b.Branch_name;

select * from BankAccount;

Result Grid	Filter Rows:	Edit:	Export/Import:	Wrap Cell Content:
▶	1	SBI_Chamrajpet	2100	
	2	SBI_ResidencyRoad	5250	
	4	SBI_ParlimentRoad	9450	
	5	SBI_Jantarantar	8400	
	8	SBI_ResidencyRoad	4200	
	9	SBI_ParlimentRoad	3150	
	10	SBI_ResidencyRoad	5250	
	11	SBI_Jantarantar	2100	
	12	SBI_MantriMarg	2100	
	13	SBI_Jantarantar	2100	
*	NULL	NULL	NULL	