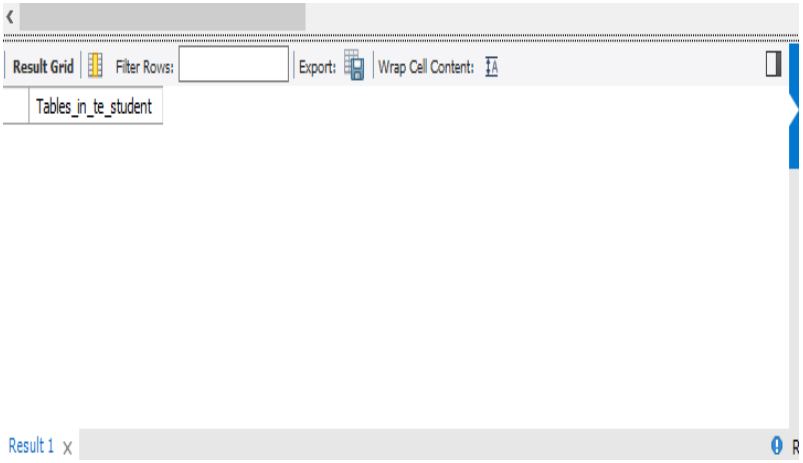
EXPERIMENT NO : 2

Samarth Hanji TE59

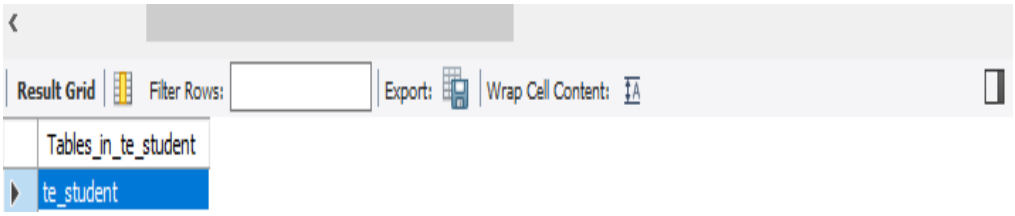
create database TE_student;

use TE_student;

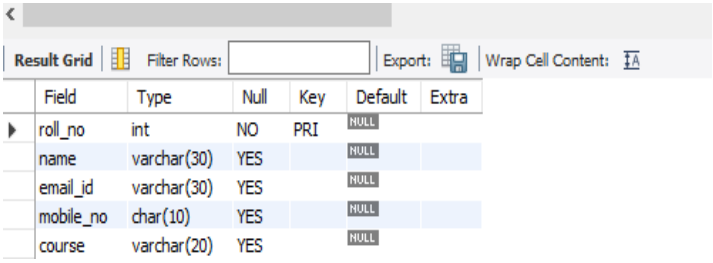
show tables;



create table TE_student(roll_no int(5) primary key, name varchar(30), email_id varchar(30), mobile_no char(10), course varchar(20));



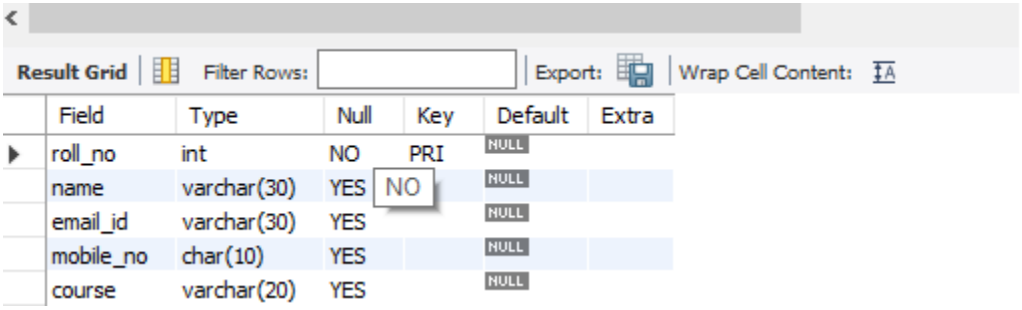
desc TE_student;



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
roll_no	int	NO	PRI	NULL	
name	varchar(30)	YES		NULL	
email_id	varchar(30)	YES		NULL	
mobile_no	char(10)	YES		NULL	
course	varchar(20)	YES		NULL	

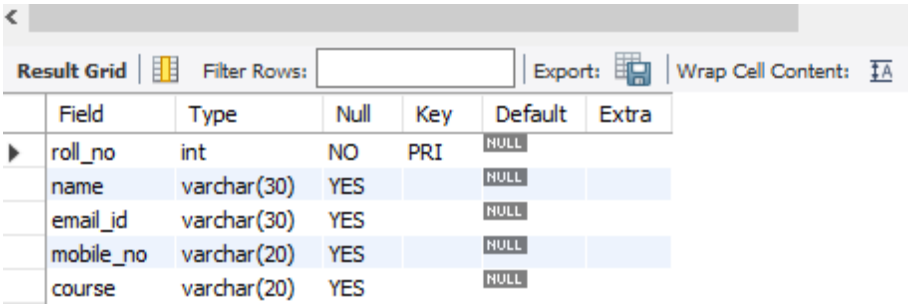
alter table TE_student drop address;



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
roll_no	int	NO	PRI	NULL	
name	varchar(30)	YES	NO	NULL	
email_id	varchar(30)	YES		NULL	
mobile_no	char(10)	YES		NULL	
course	varchar(20)	YES		NULL	

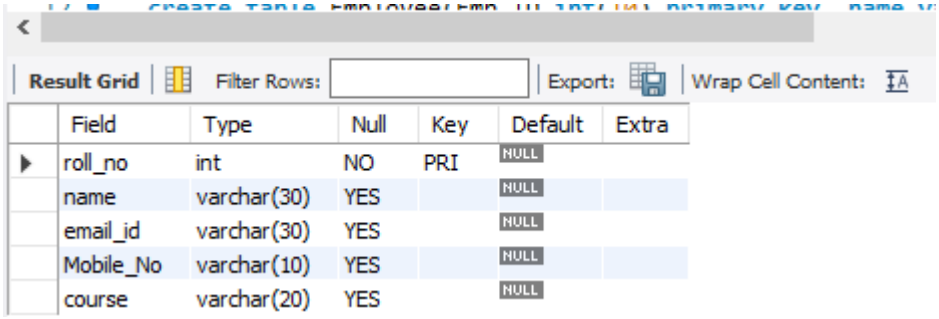
alter table TE_student modify mobile_no varchar(20);



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
roll_no	int	NO	PRI	NULL	
name	varchar(30)	YES		NULL	
email_id	varchar(30)	YES		NULL	
mobile_no	varchar(20)	YES		NULL	
course	varchar(20)	YES		NULL	

alter table TE_student change mobile_no Mobile_No varchar(10);



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

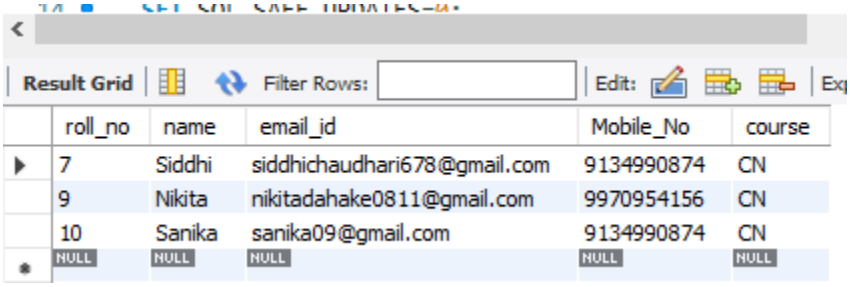
Field	Type	Null	Key	Default	Extra
roll_no	int	NO	PRI	NULL	
name	varchar(30)	YES		NULL	
email_id	varchar(30)	YES		NULL	
Mobile_No	varchar(10)	YES		NULL	
course	varchar(20)	YES		NULL	

insert into TE_student values(7,'Siddhi','siddhichaudhari678@gmail.com','9134990874','CN');

insert into TE_student values(9,'Nikita','nikitadahake0811@gmail.com','9970954156','CN');

insert into TE_student values(10,'Sanika','sanika09@gmail.com','9134990874','CN');

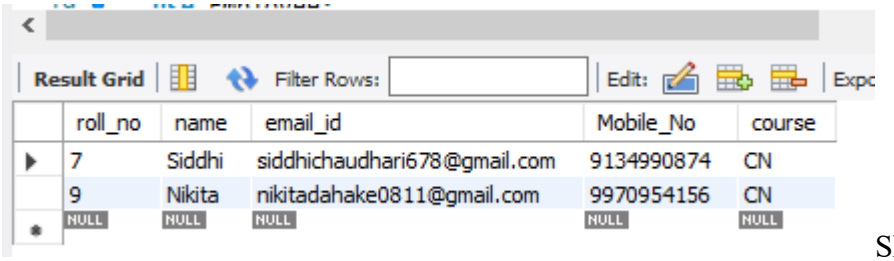
select *from TE_student;



roll_no	name	email_id	Mobile_No	course
7	Siddhi	siddhichaudhari678@gmail.com	9134990874	CN
9	Nikita	nikitadahake0811@gmail.com	9970954156	CN
10	Sanika	sanika09@gmail.com	9134990874	CN
NULL	NULL	NULL	NULL	NULL

delete from TE_student where name='sanika';

select *from TE_student;

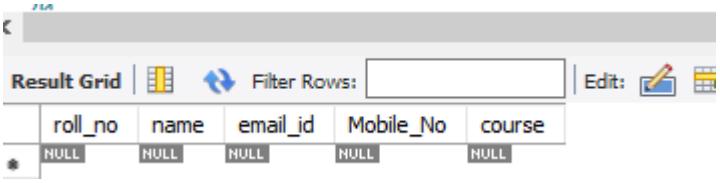


roll_no	name	email_id	Mobile_No	course
7	Siddhi	siddhichaudhari678@gmail.com	9134990874	CN
9	Nikita	nikitadahake0811@gmail.com	9970954156	CN
NULL	NULL	NULL	NULL	NULL

SET SQL_SAFE_UPDATES=0;

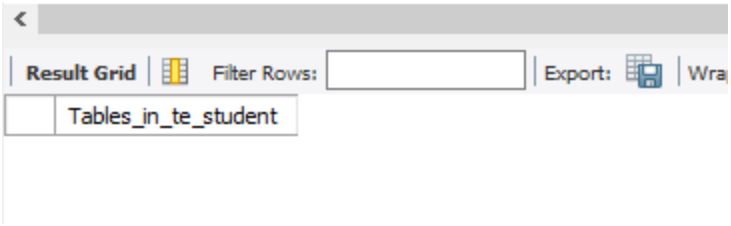
truncate table TE_student;

select *from TE_student;

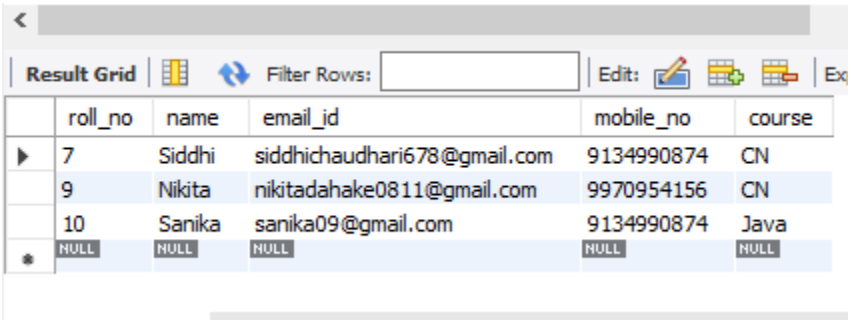


roll_no	name	email_id	Mobile_No	course
NULL	NULL	NULL	NULL	NULL

Drop table TE_student;

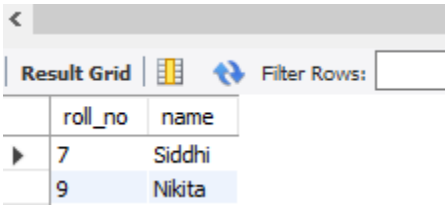


#Again create table and insert values



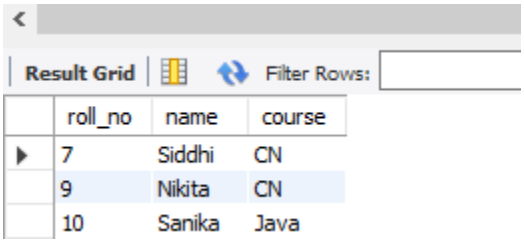
create view v1 as select roll_no,name from TE_student where course='CN';

select*from v1;



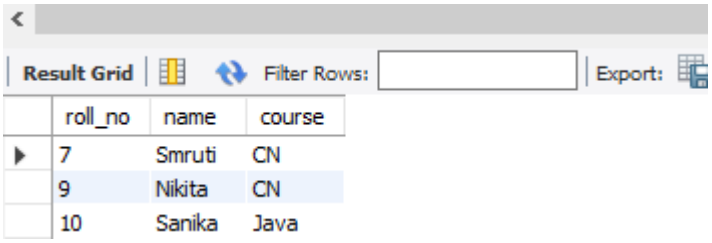
create view v2 as select roll_no,name,course from TE_student;

select*from v2;



update v2 set name ='Smruti' where roll_no=7;

select*from v2;

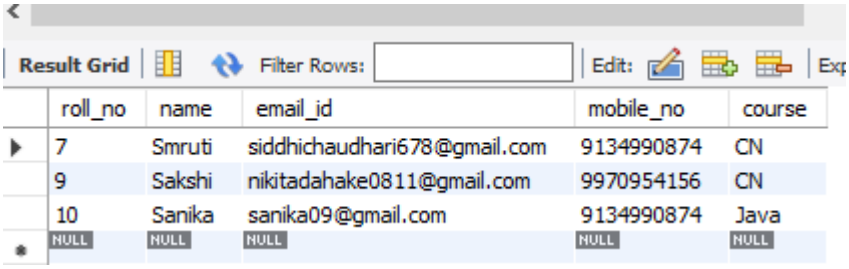


Result Grid | Filter Rows: | Export:

roll_no	name	course
7	Smruti	CN
9	Nikita	CN
10	Sanika	Java

update TE_student set name ='Sakshi' where roll_no=9;

select *from TE_student;



Result Grid | Filter Rows: | Edit: | Exp

roll_no	name	email_id	mobile_no	course
7	Smruti	siddhichaudhari678@gmail.com	9134990874	CN
9	Sakshi	nikitadahake0811@gmail.com	9970954156	CN
10	Sanika	sanika09@gmail.com	9134990874	Java
NULL	NULL	NULL	NULL	NULL

create table studinfo(roll_no integer(10) Primary key AUTO_INCREMENT, Name varchar(30), Mobile_no varchar(10));

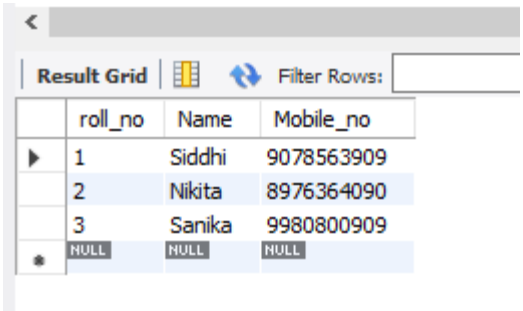
desc studinfo;

insert into studinfo(Name,Mobile_no) values('Siddhi','9078563909');

insert into studinfo(Name,Mobile_no) values('Nikita','8976364090');

insert into studinfo(Name,Mobile_no) values('Sanika','9980800909');

select *from studinfo;



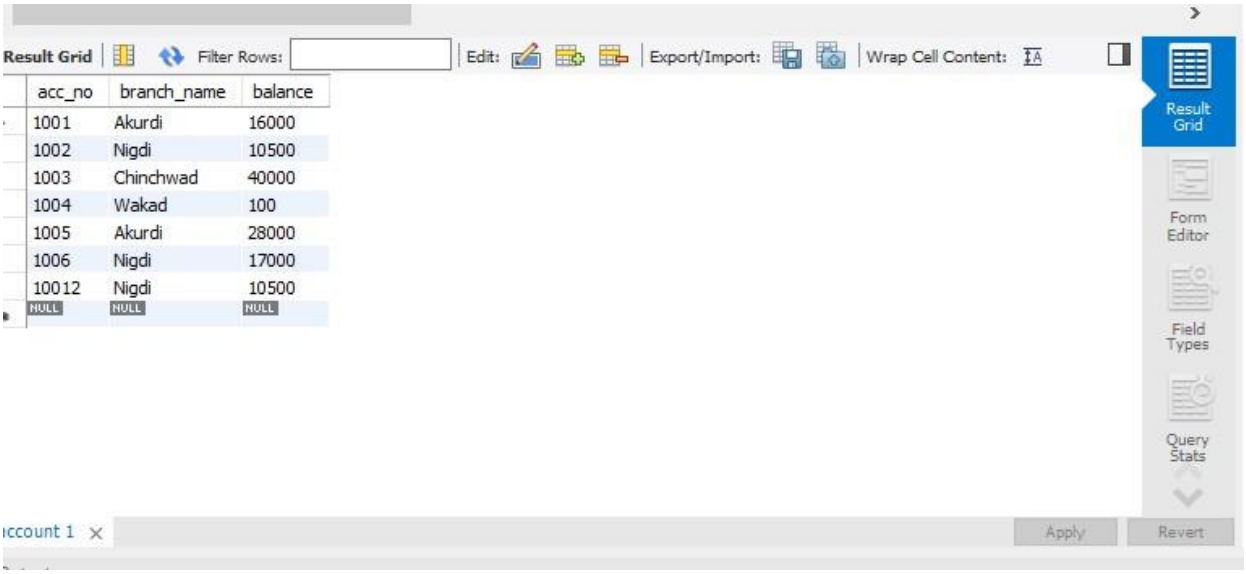
Result Grid | Filter Rows:

roll_no	Name	Mobile_no
1	Siddhi	9078563909
2	Nikita	8976364090
3	Sanika	9980800909
NULL	NULL	NULL

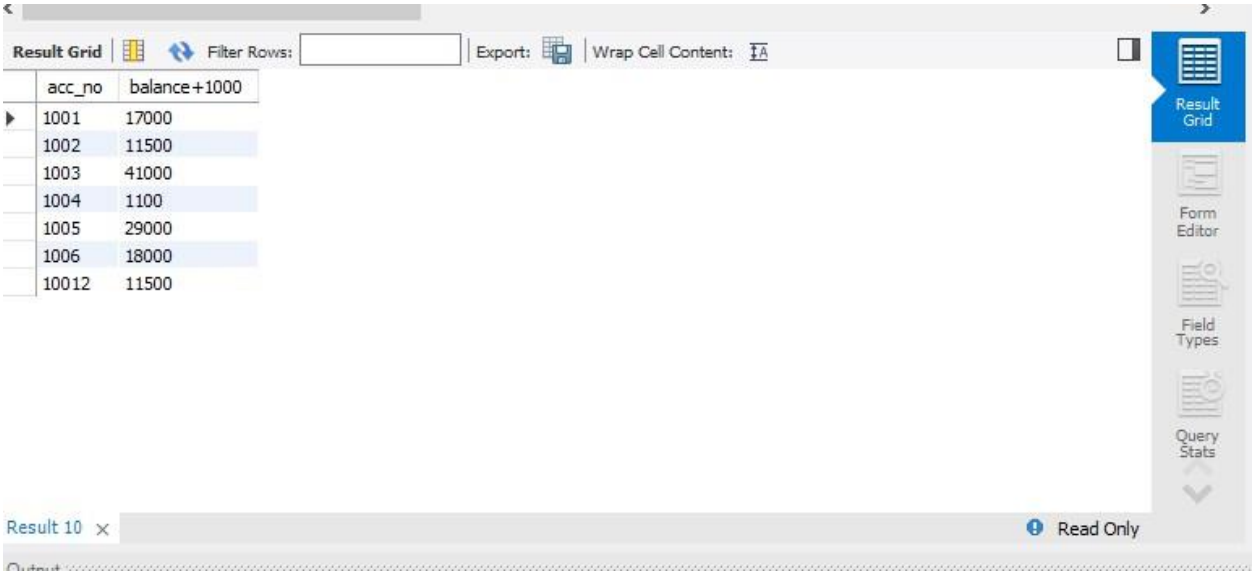
EXPERIMENT NO 3

Samarth Hanji TE59

```
create database bank;
use bank;
create table account(acc_no int(10) primary key ,branch_name varchar(20), balance int(20));
insert into account values( 1001,'Akurdi', 15000+1000);
insert into account values( 1002,'Nigdi', 11000-500);
insert into account values( 1003,'Chinchwad',20000*2);
insert into account values( 1004,'Wakad', 10000/100);
insert into account values( 1005,'Akurdi', 14000*2);
insert into account values( 1006,'Nigdi', 17000);
 select * from account;
```

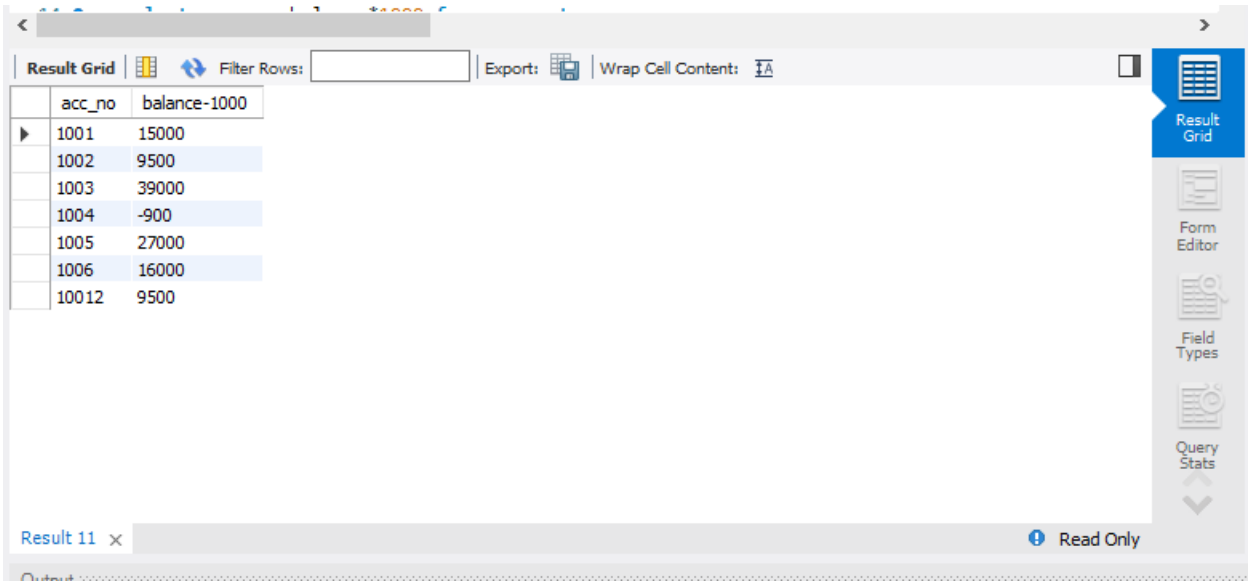



```
select acc_no, balance+1000 from account;
```



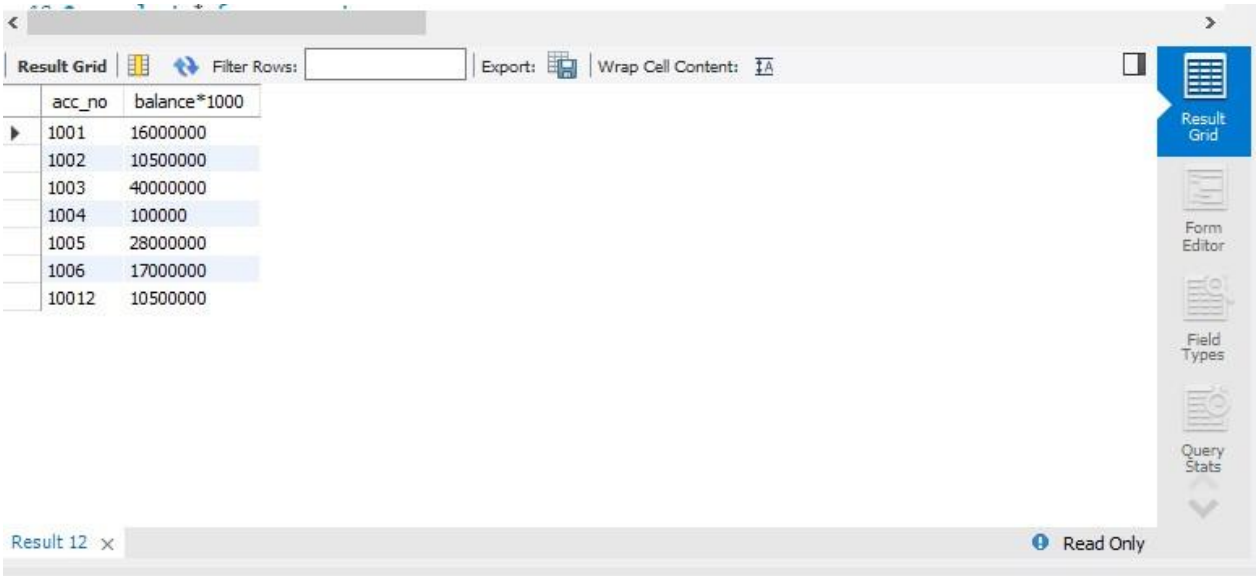
acc_no	balance+1000
1001	17000
1002	11500
1003	41000
1004	1100
1005	29000
1006	18000
10012	11500

select acc_no, balance-1000 from account;

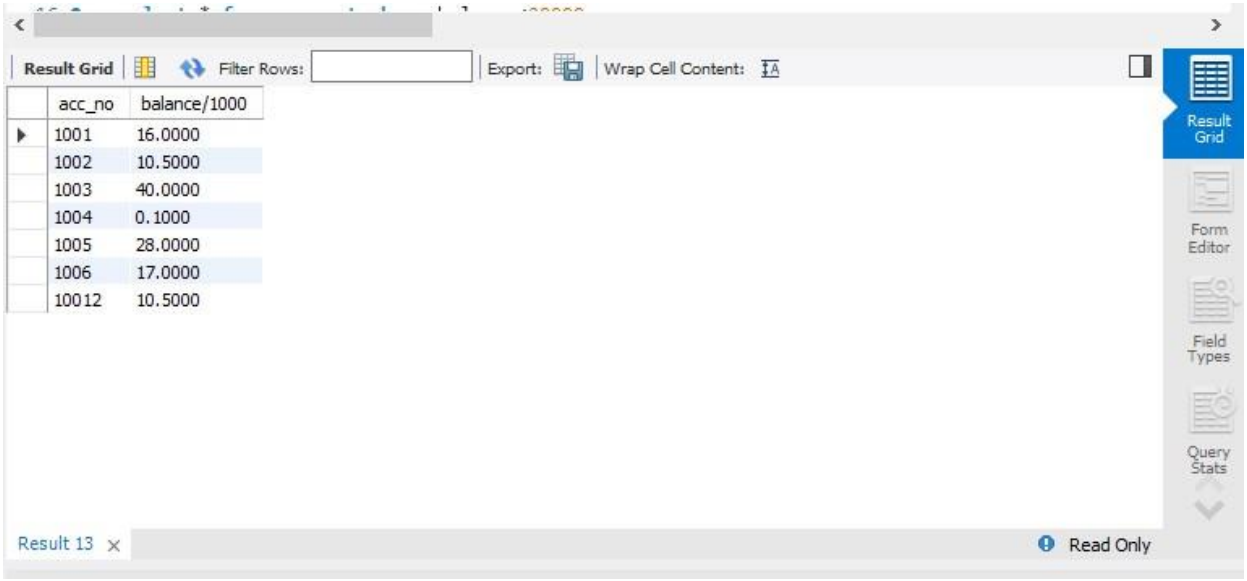


acc_no	balance-1000
1001	15000
1002	9500
1003	39000
1004	-900
1005	27000
1006	16000
10012	9500

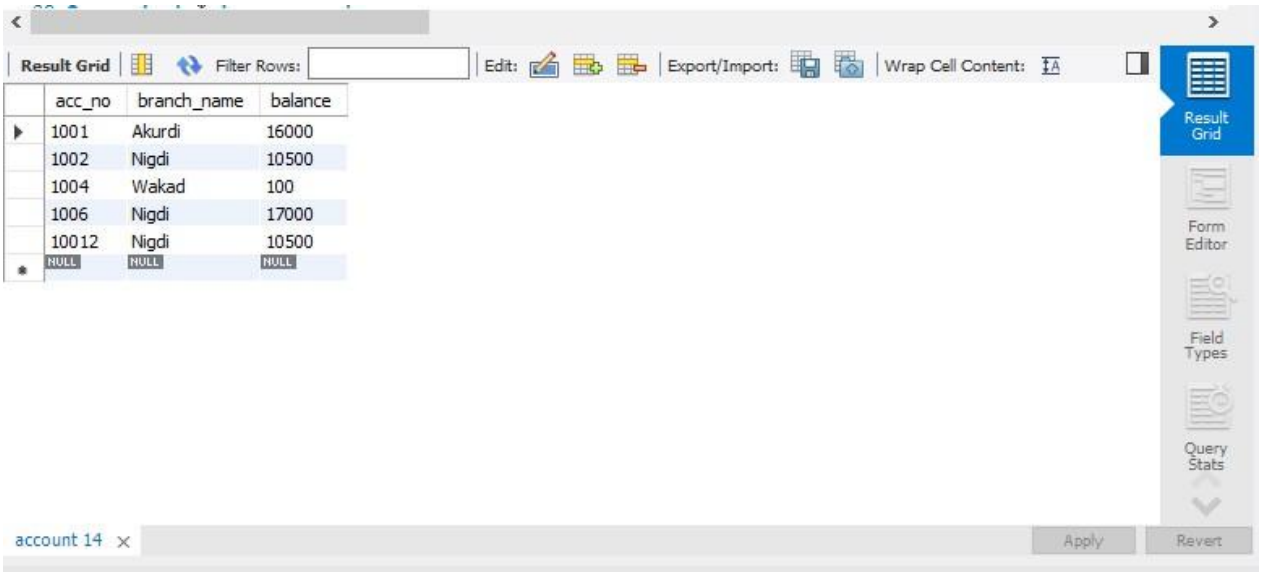
select acc_no, balance*1000 from account;



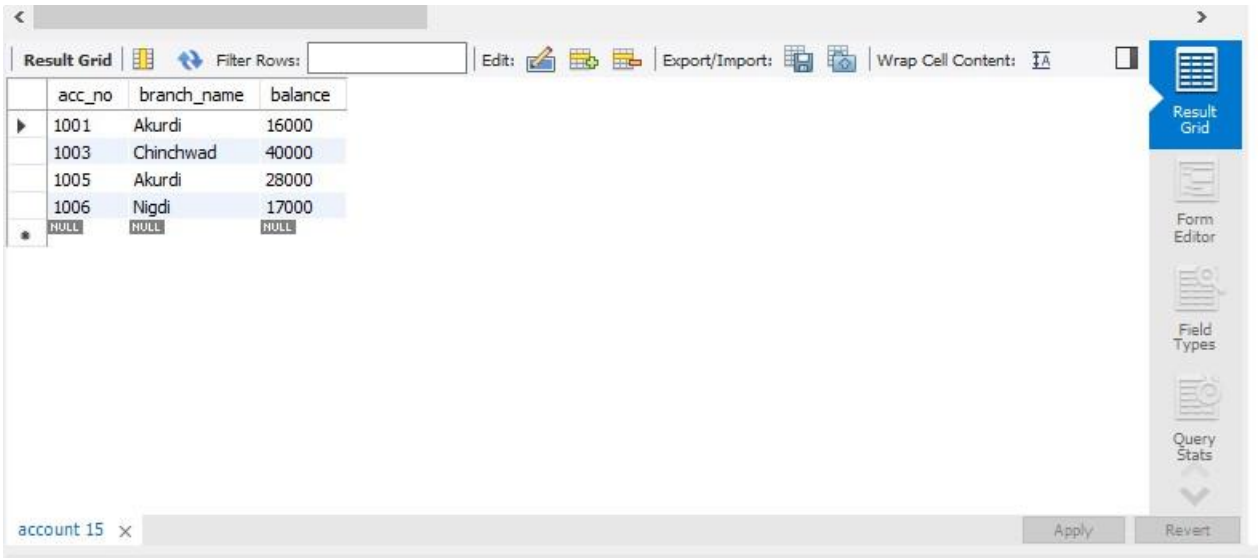
select acc_no, balance/1000 from account;



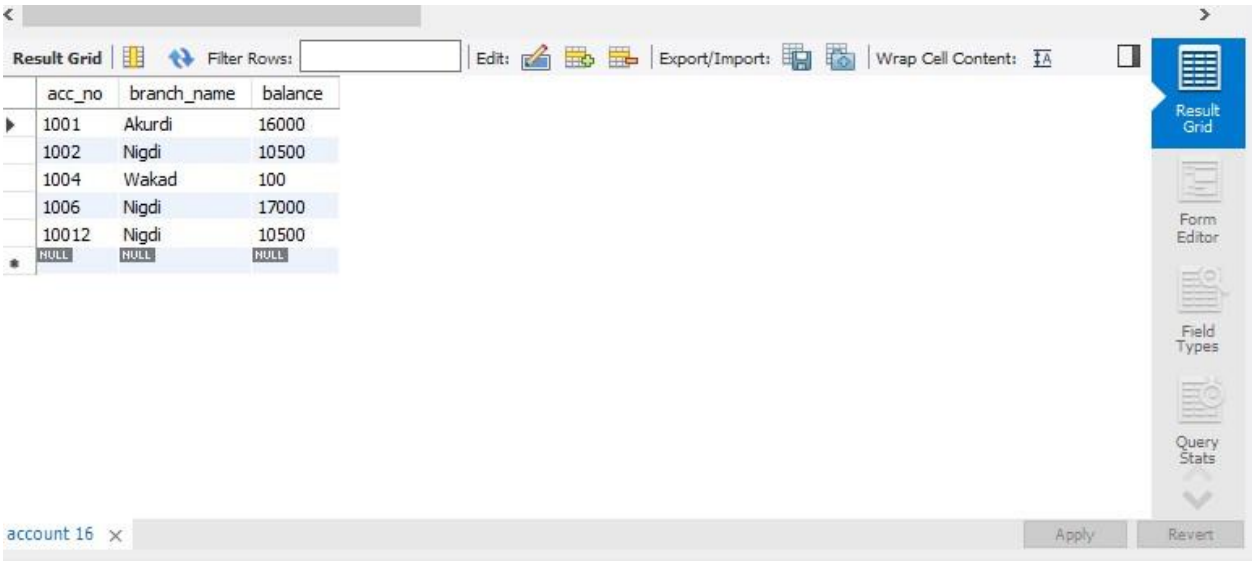
select * from account where balance<20000 ;



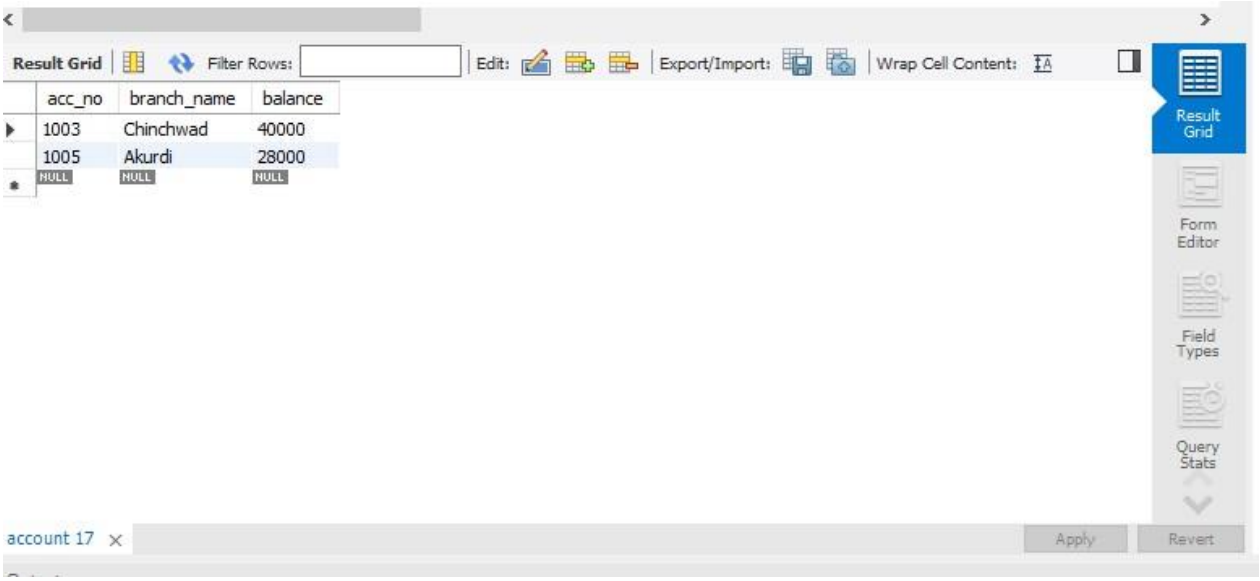
select * from account where balance>15000;



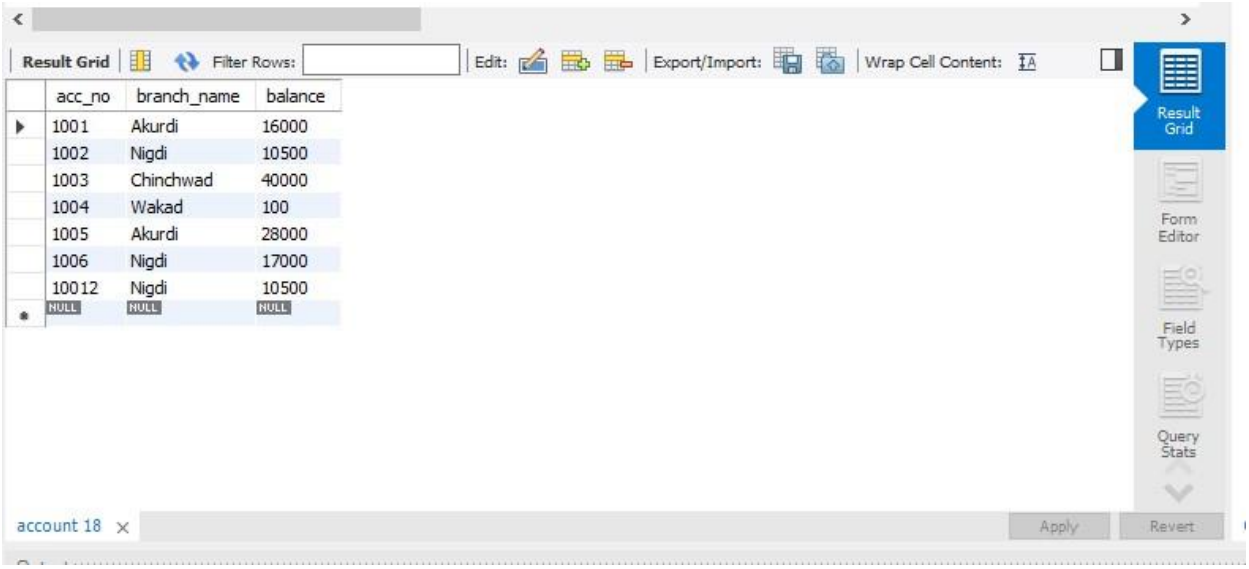
select * from account where balance <= 20000;



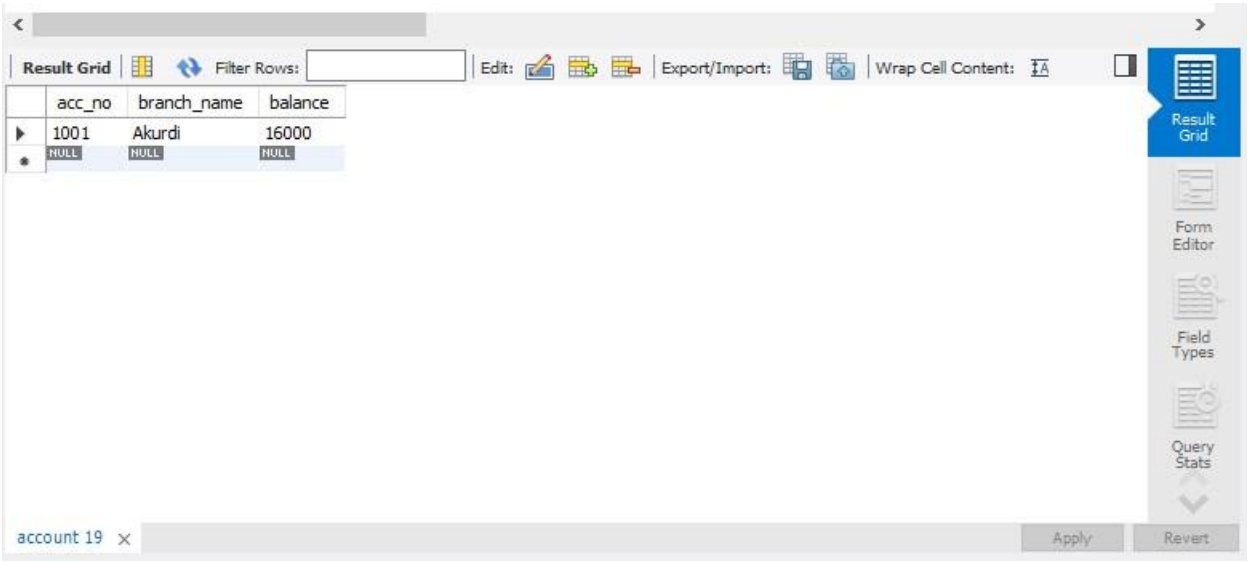
select * from account where balance>= 20000;



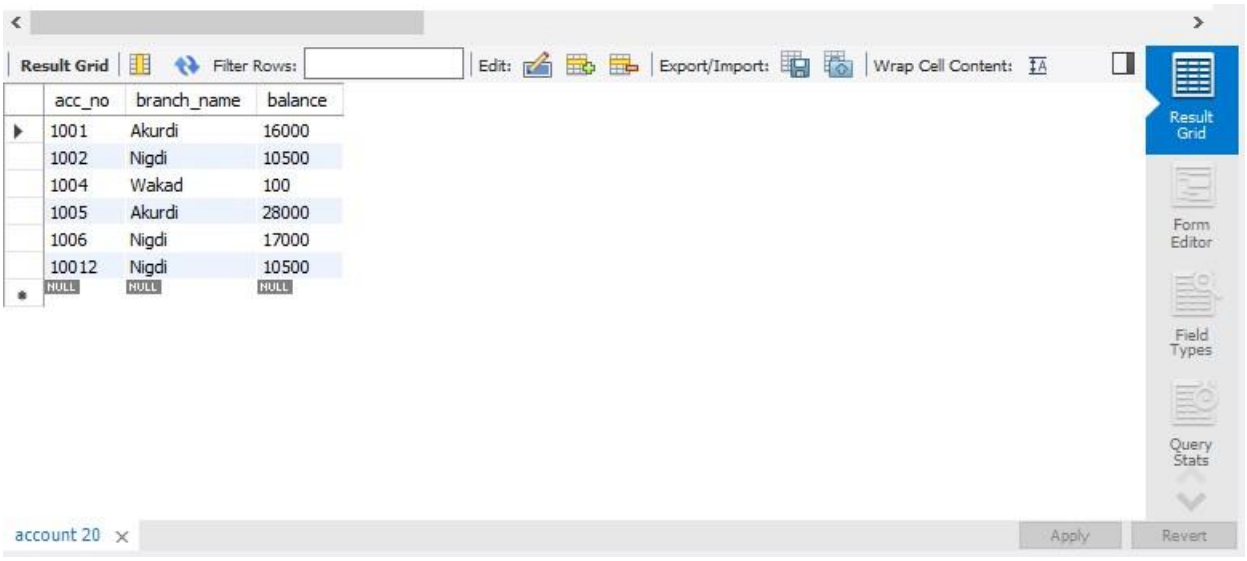
select * from account where balance != 20000;



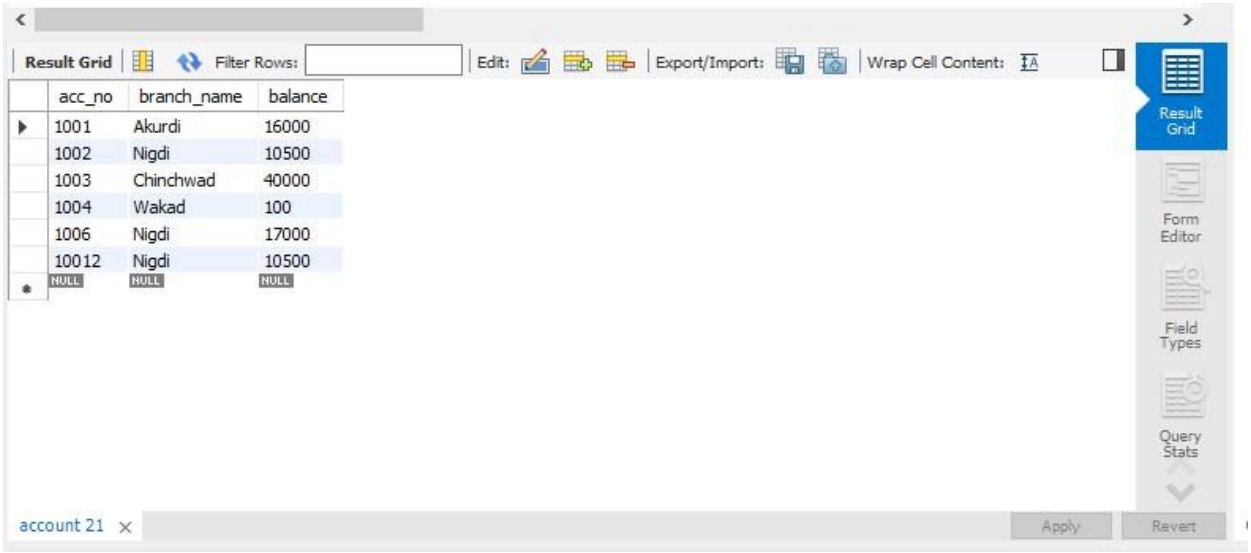
select * from account where balance<20000 AND branch_name = 'Akurdi';



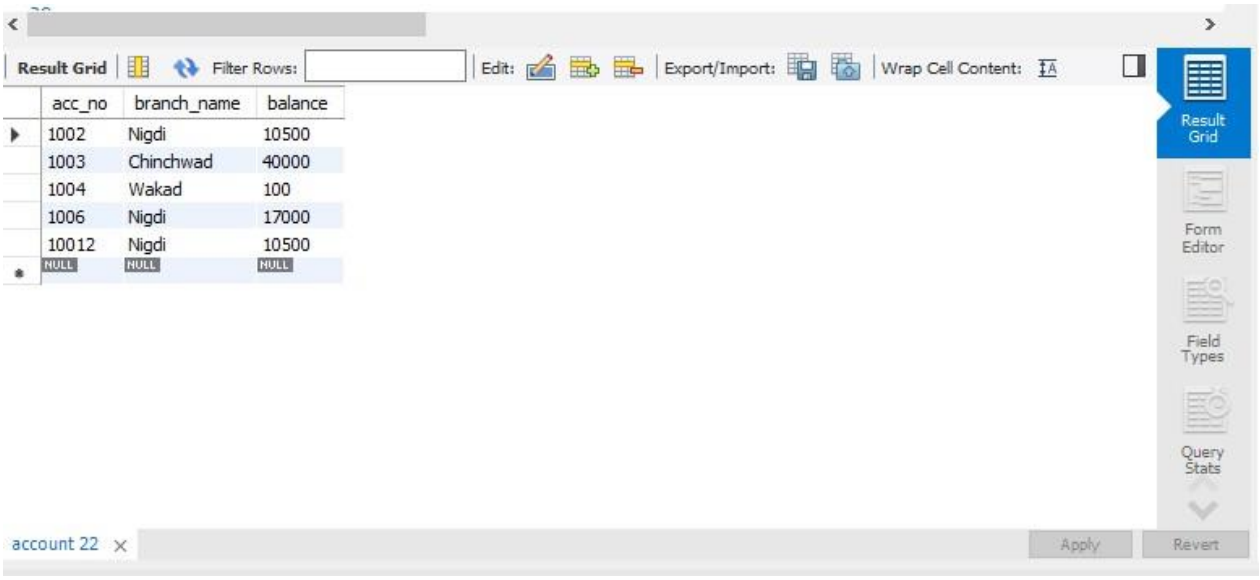
select * from account where balance<20000 OR branch_name = 'Akurdi';



select * from account where balance<20000 OR NOT branch_name = 'Akurdi';

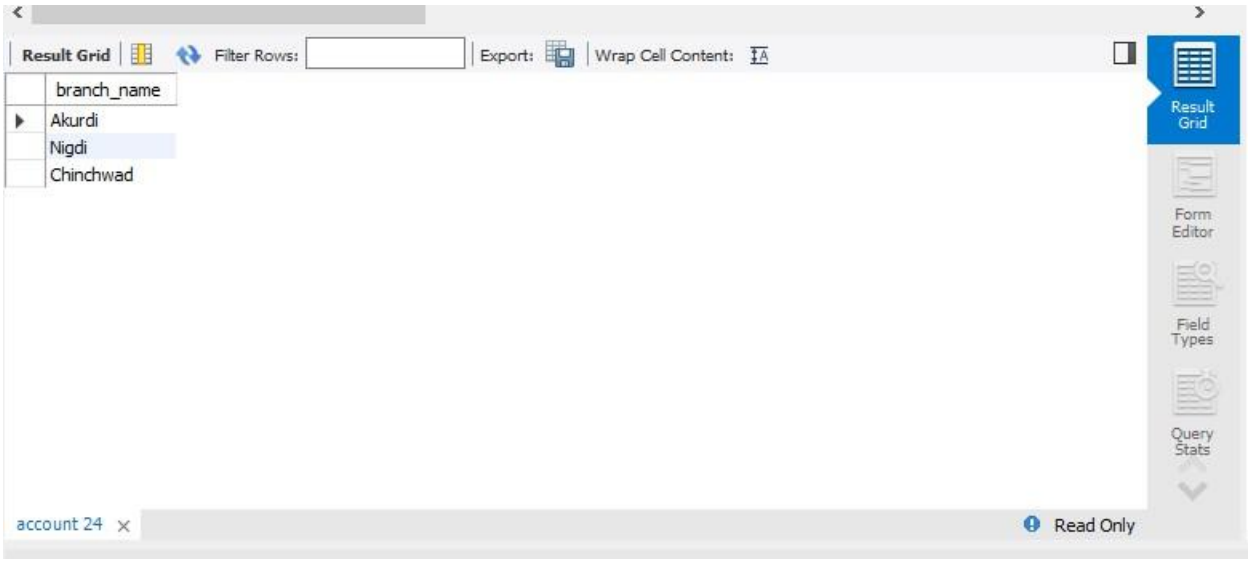


select * from account where NOT branch_name = 'Akurdi';

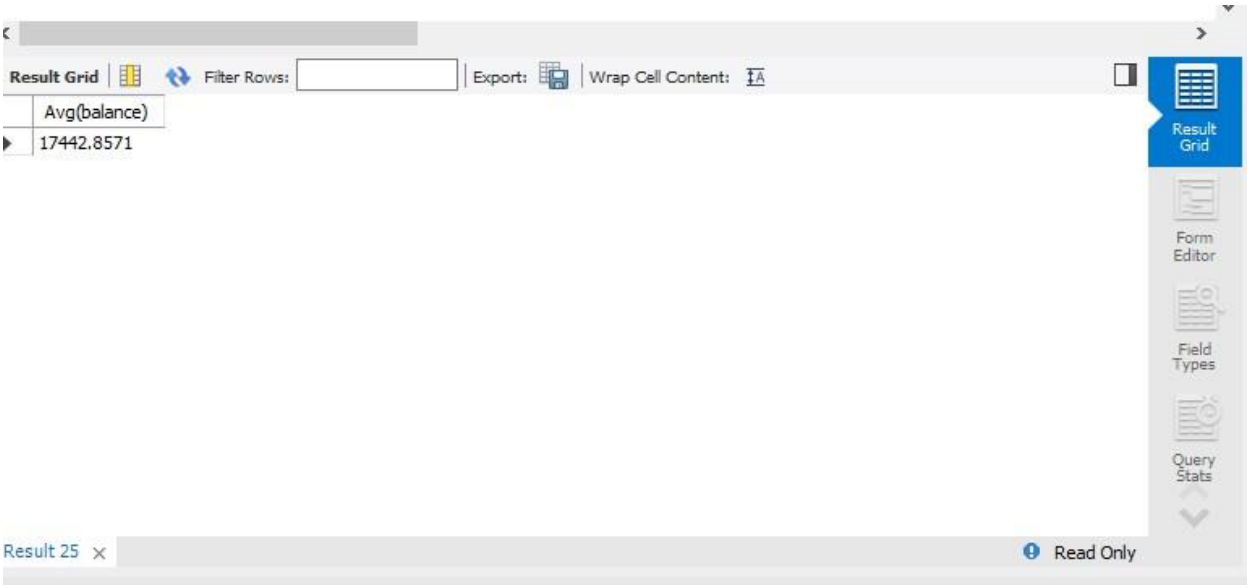


#Find the branches where average account balance >12000.

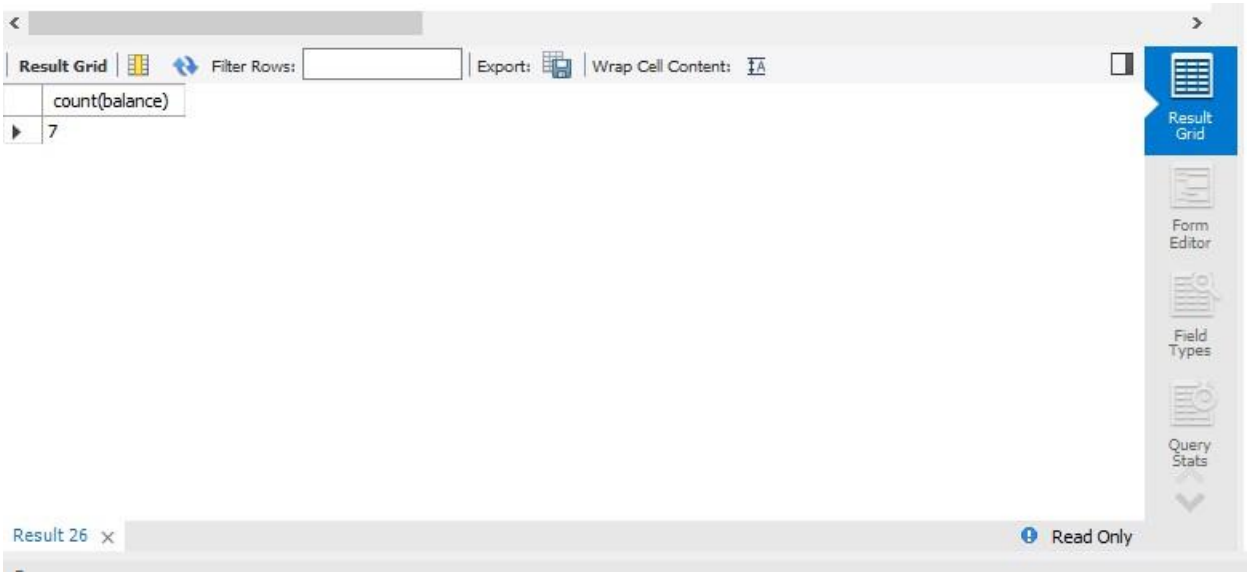
select branch_name from account group by branch_name having avg(balance)>12000;



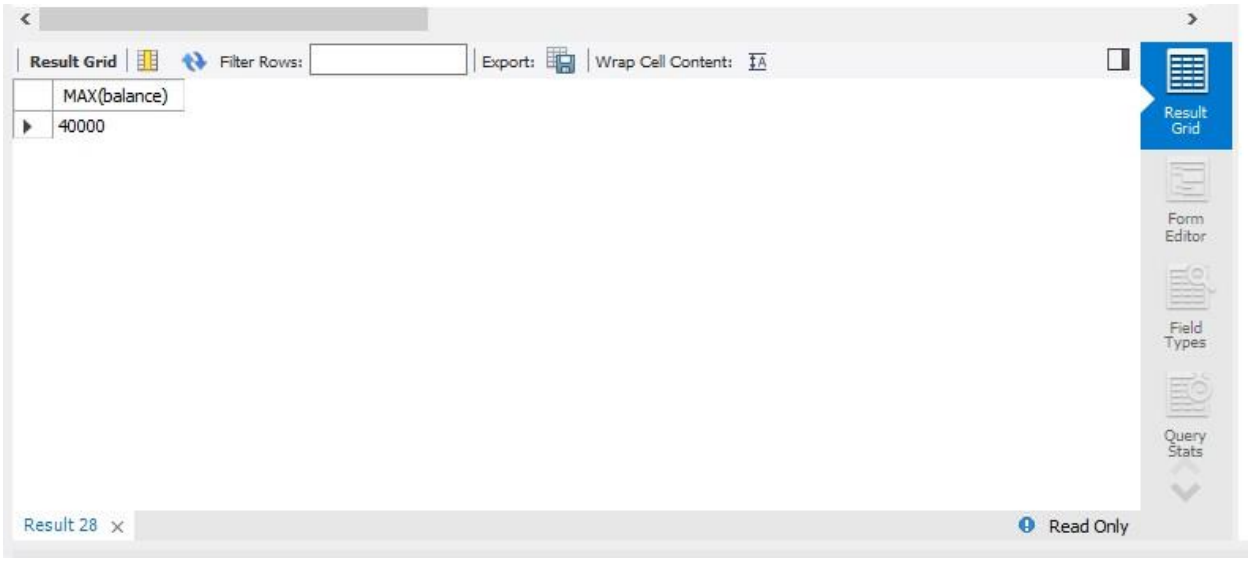
select Avg(balance) from account;



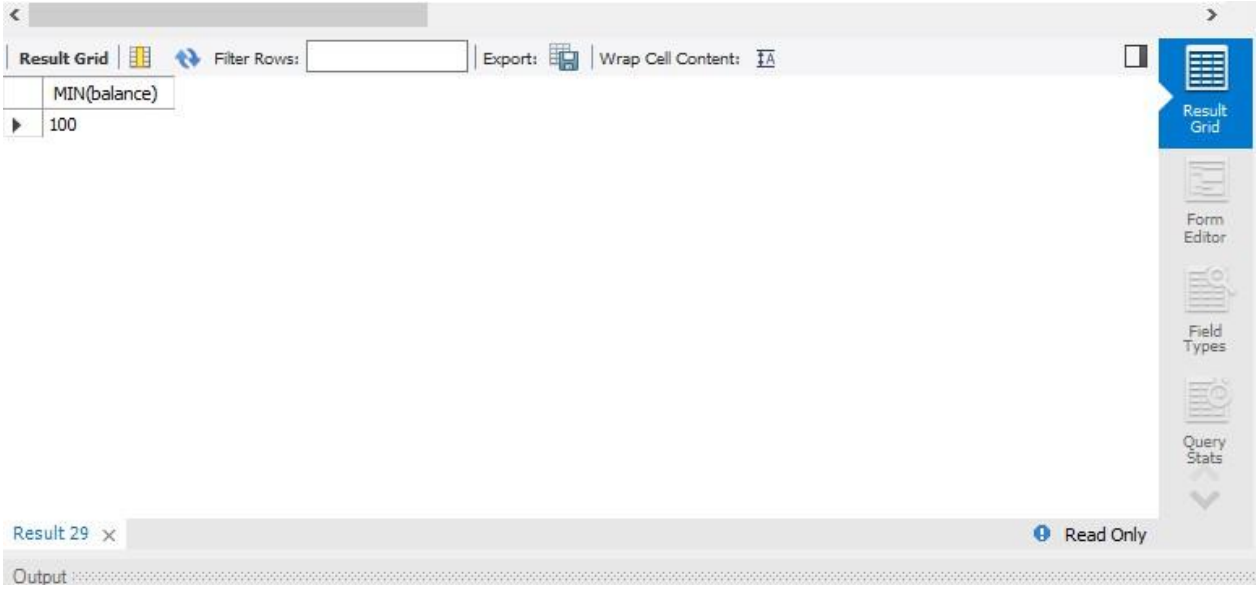
select count(balance) from account;



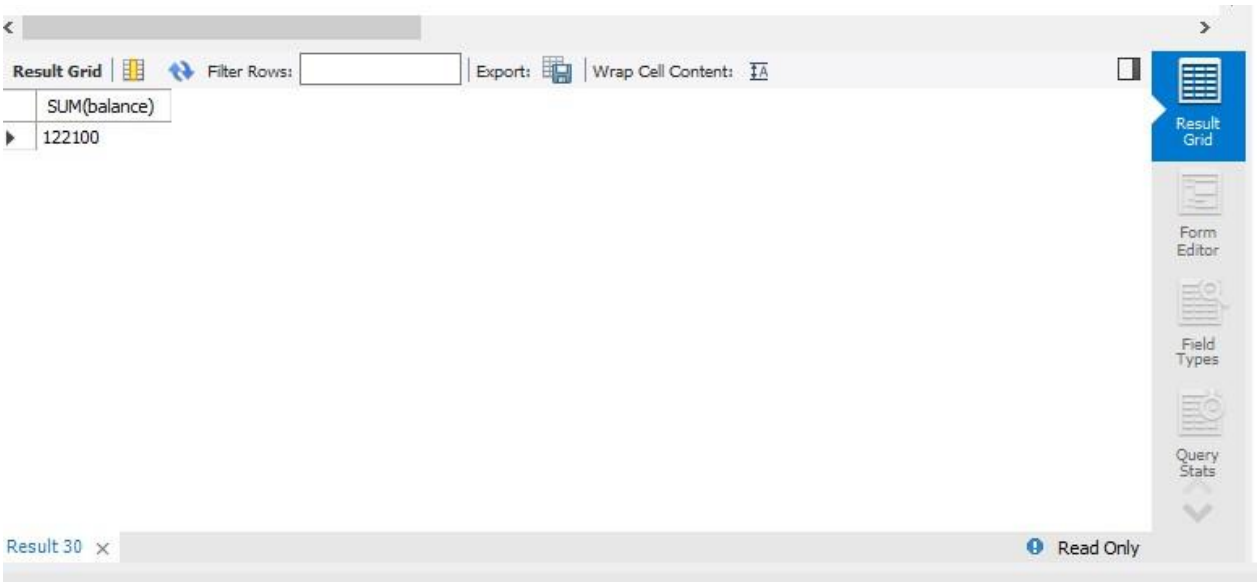
select MAX(balance) from account;



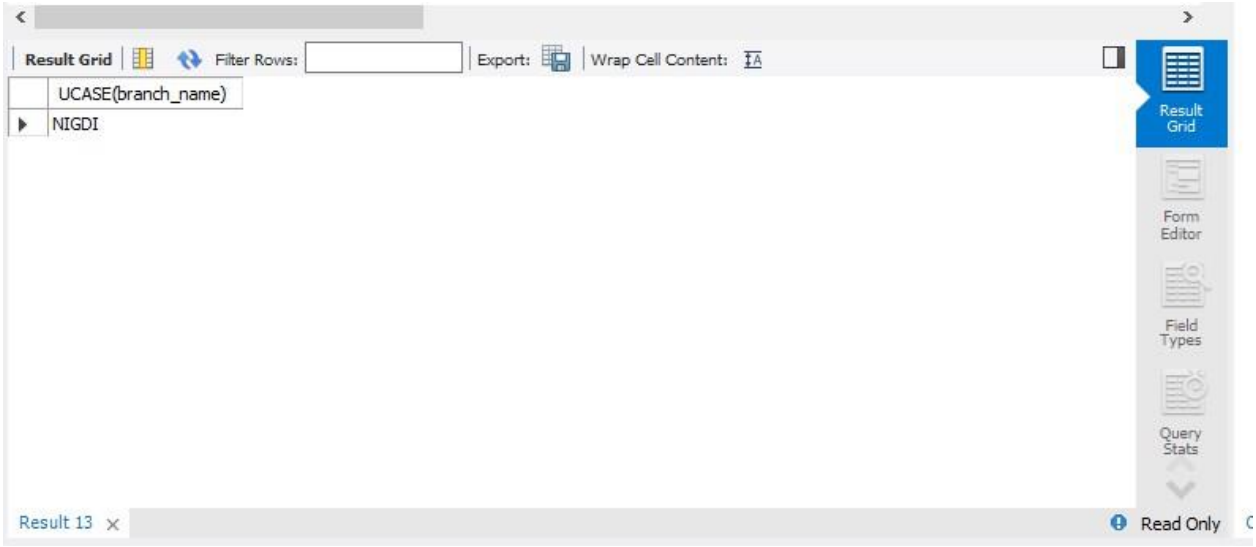
select MIN(balance) from account;



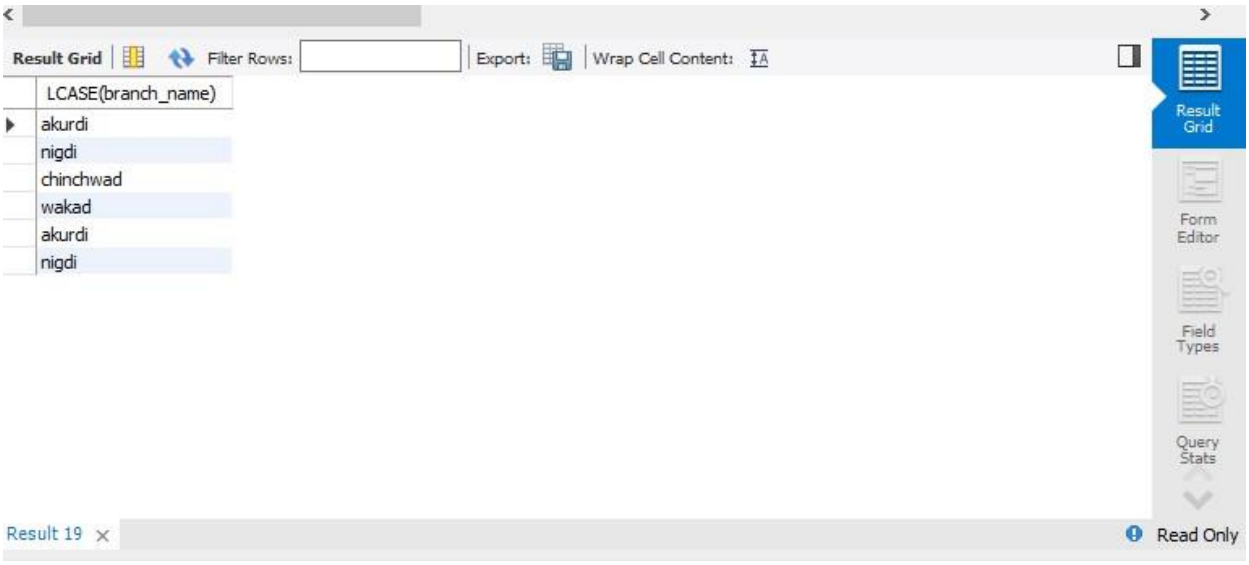
select SUM(balance) from account;



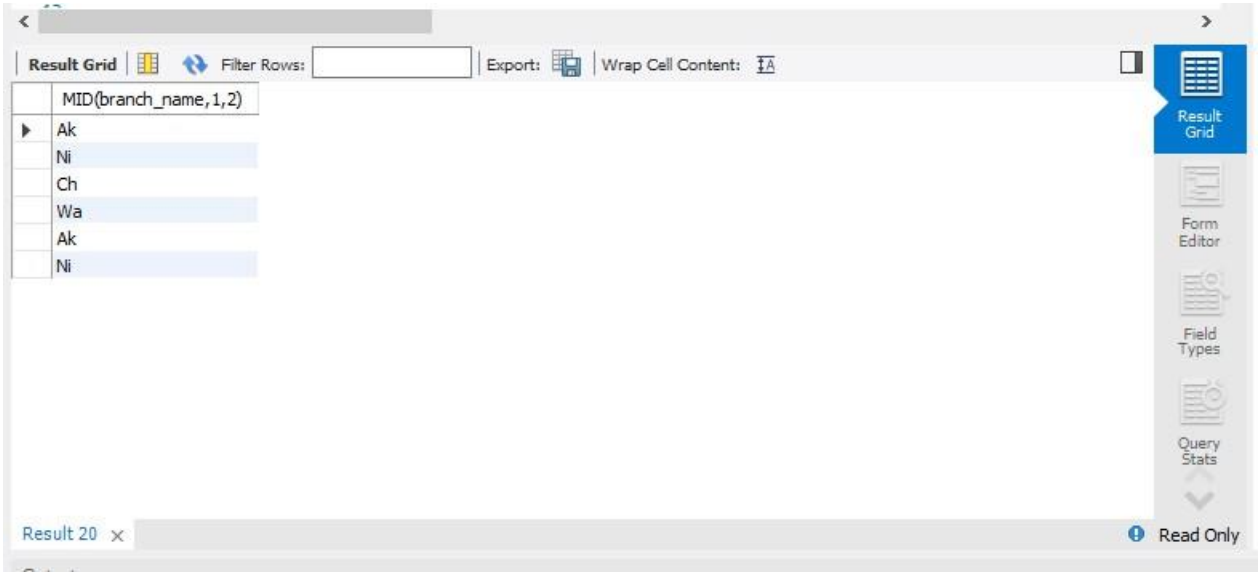
SELECT UCASE(branch_name) FROM account;



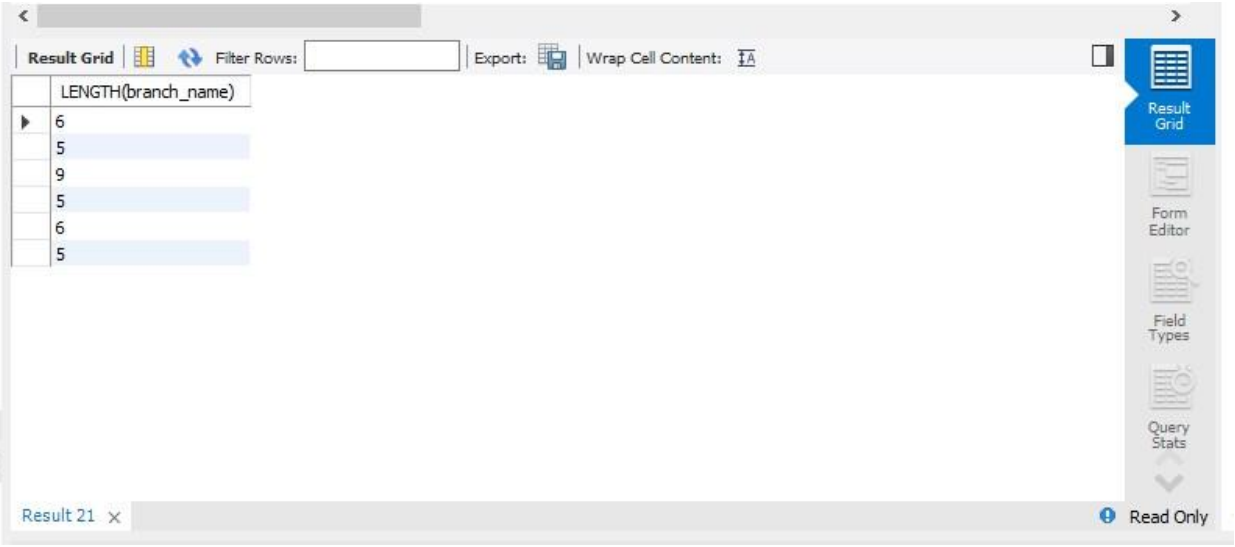
SELECT LCASE(branch_name) FROM account;



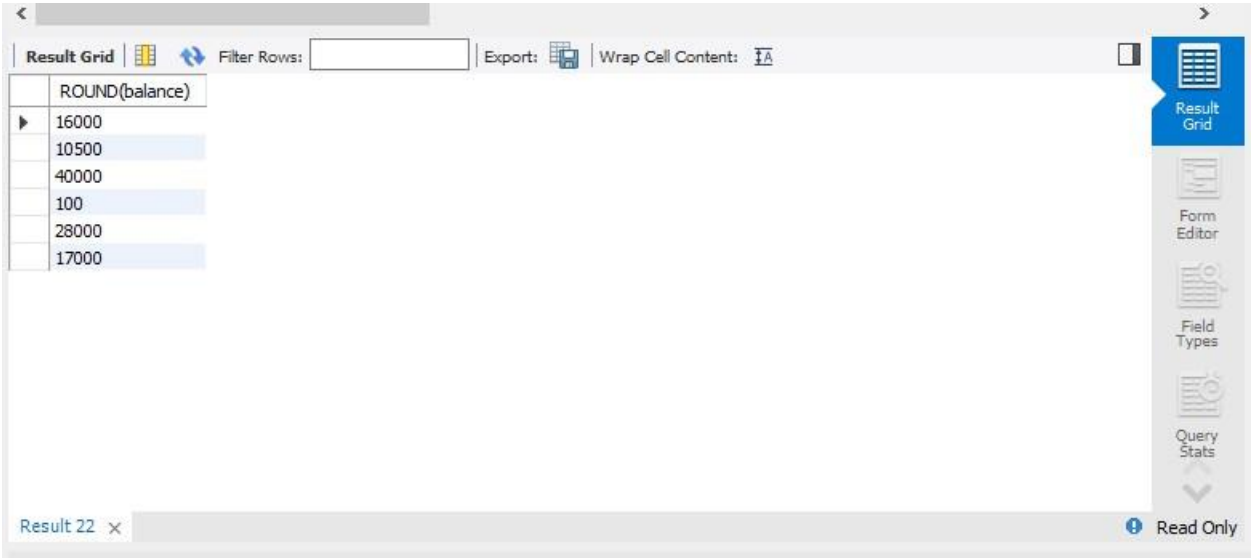
SELECT MID(branch_name,1,2) FROM account;



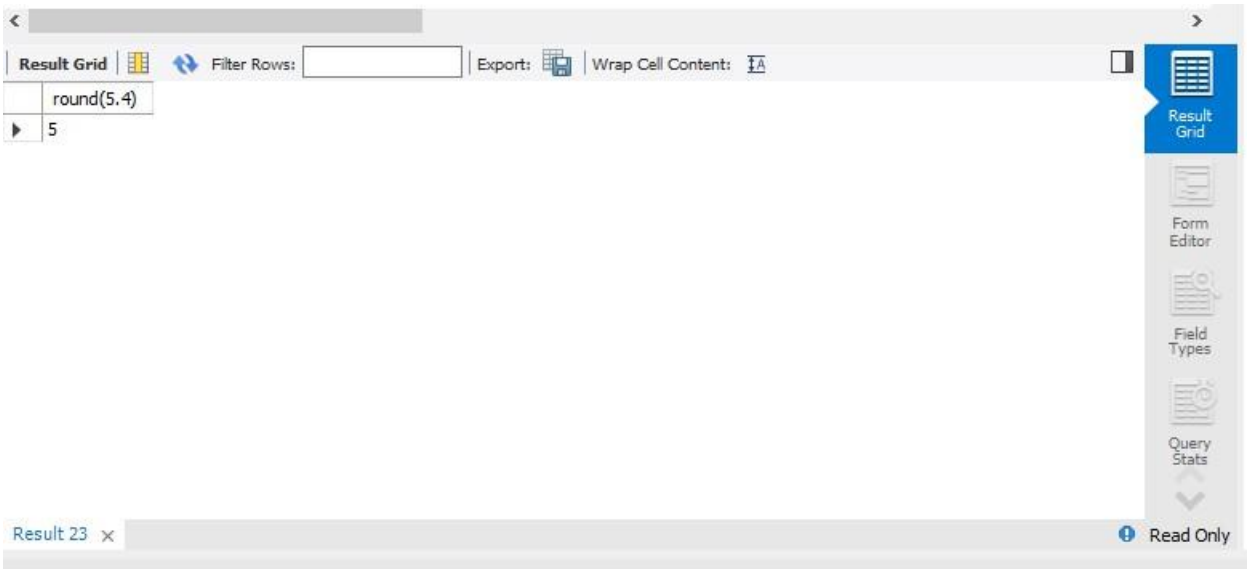
SELECT LENGTH(branch_name) FROM account;



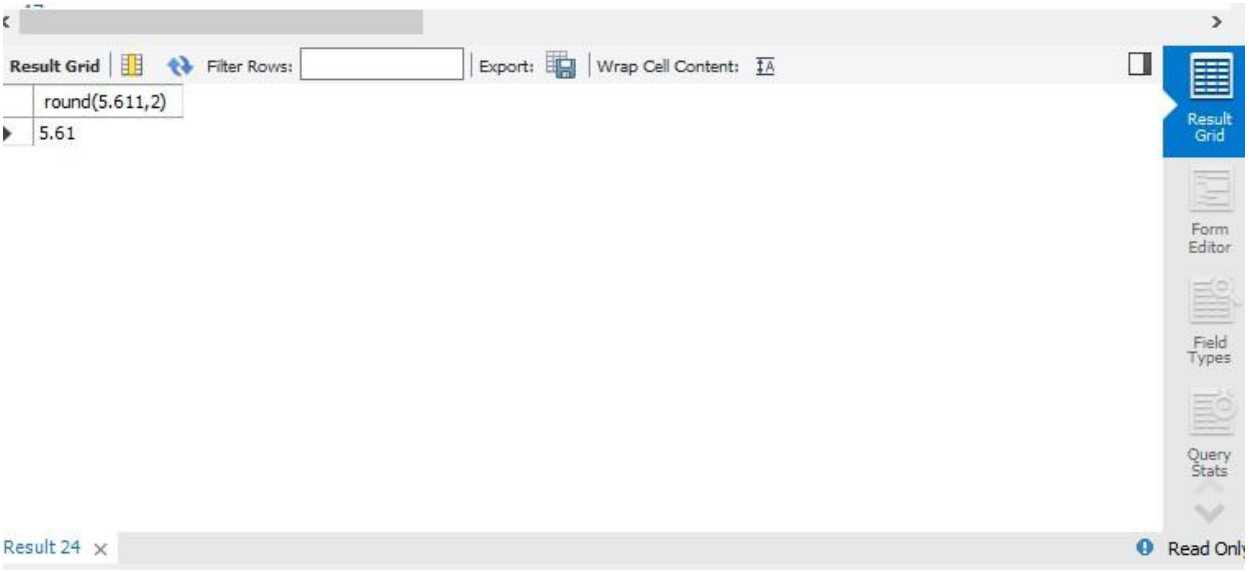
SELECT ROUND(balance) FROM account;



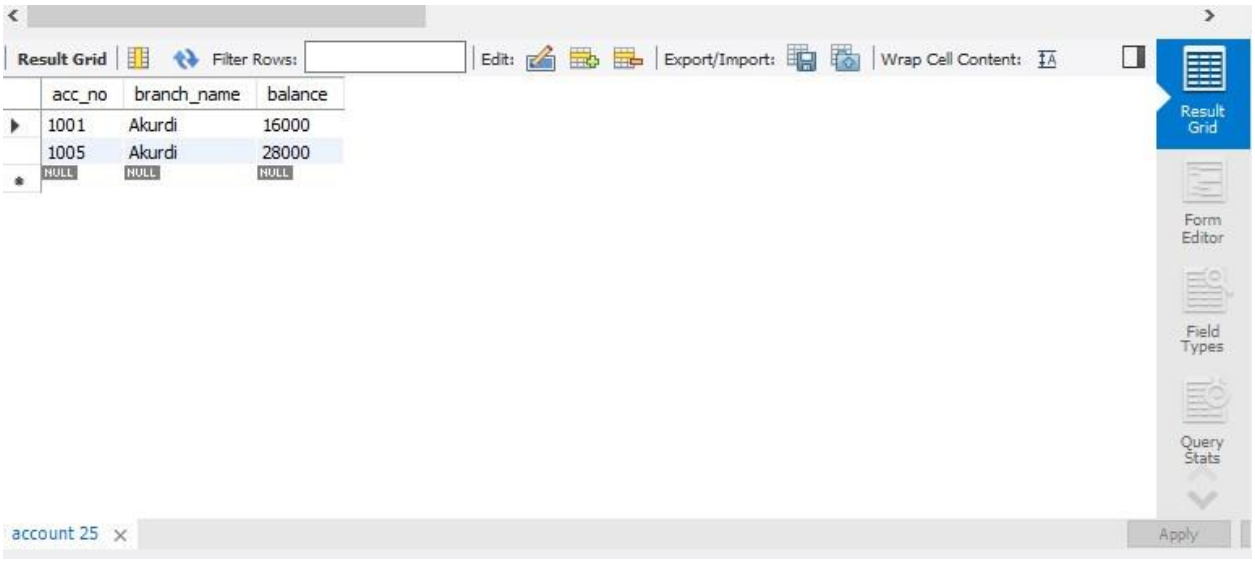
select round(5.4);



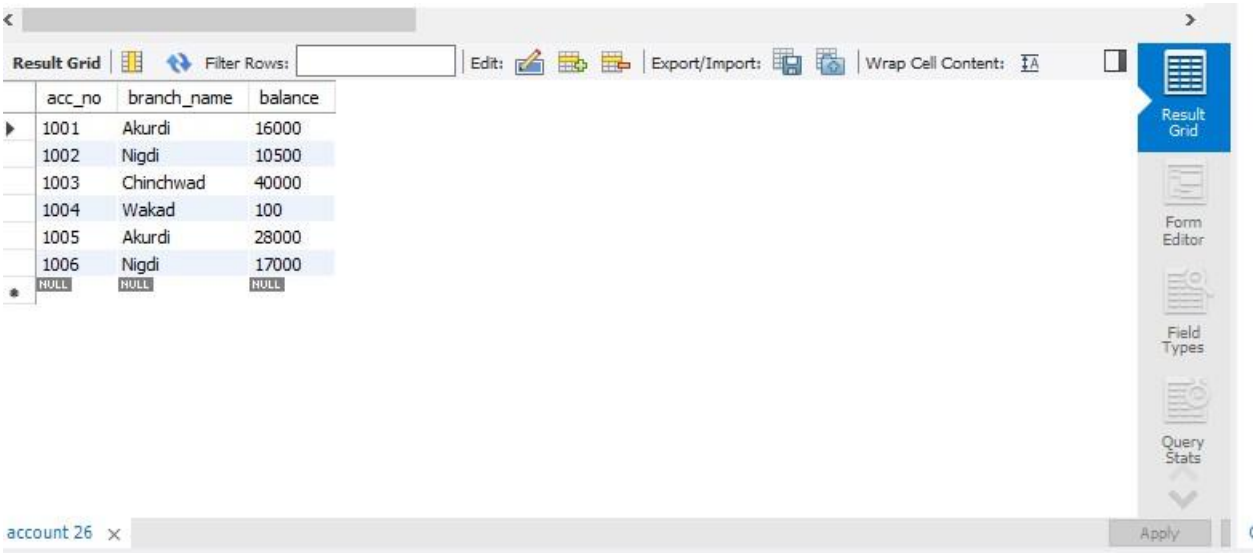
select round(5.611,2);



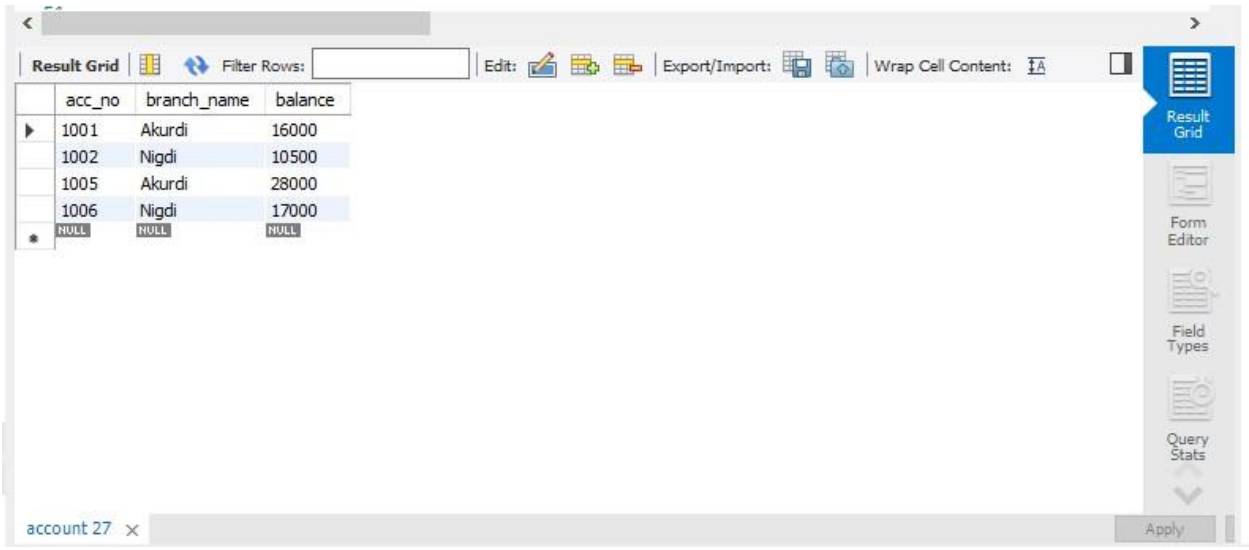
SELECT * FROM account WHERE branch_name LIKE 'ak%';



SELECT * FROM account WHERE branch_name LIKE '%d%';

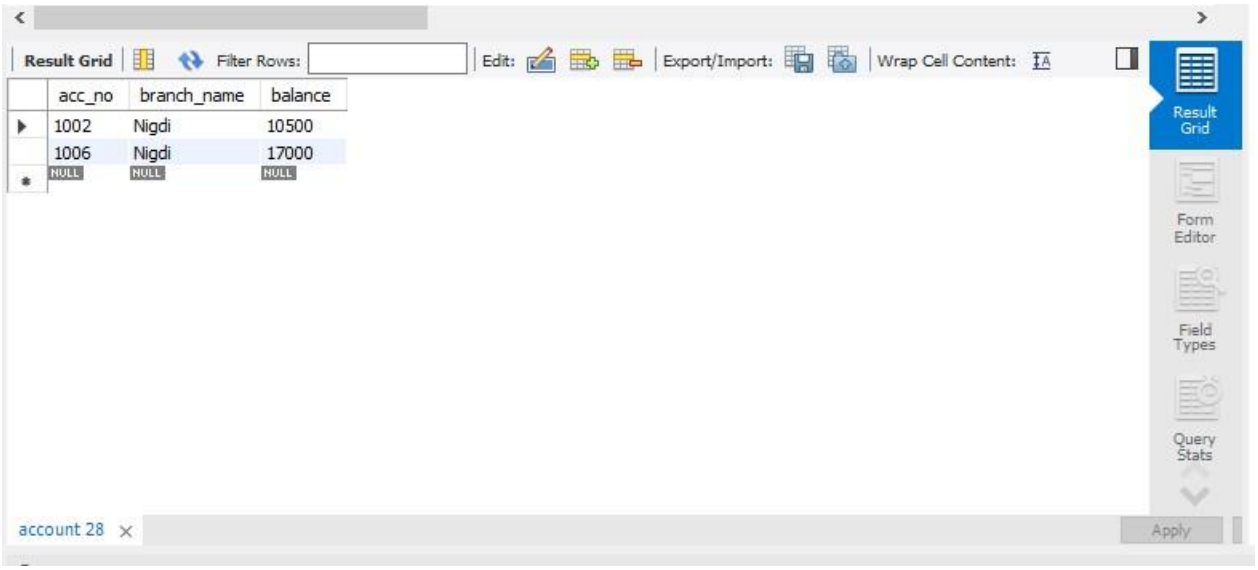


SELECT * FROM account WHERE branch_name LIKE '%di';



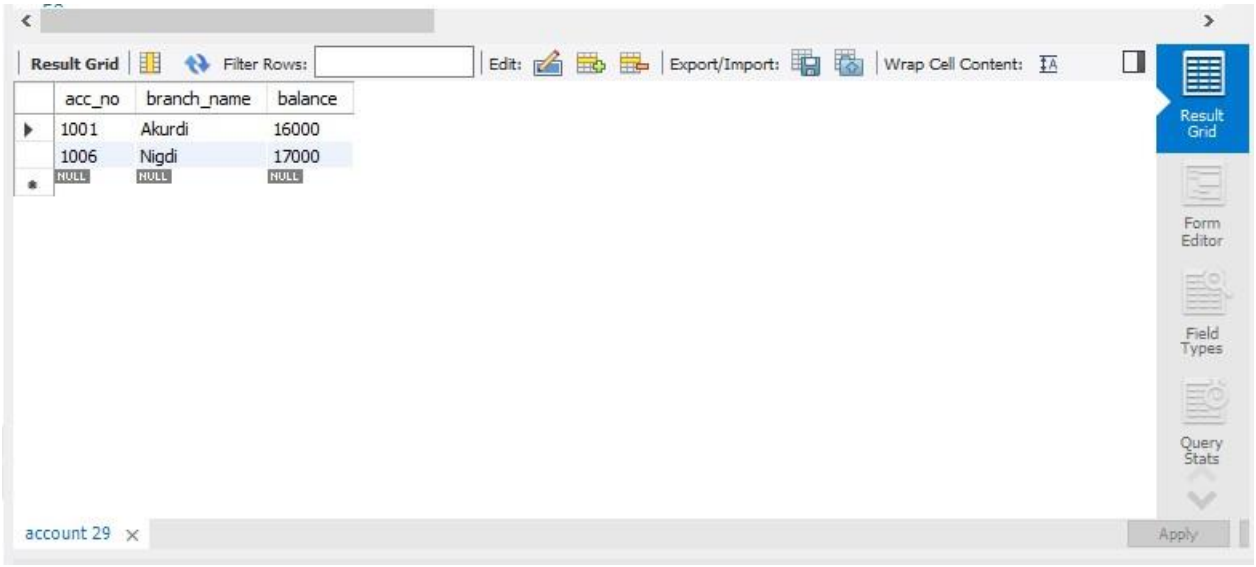
#Q.Display the accounts from Akrudi and Nigdi branch

select * from account where branch_name in('Akrudi','Nigdi');



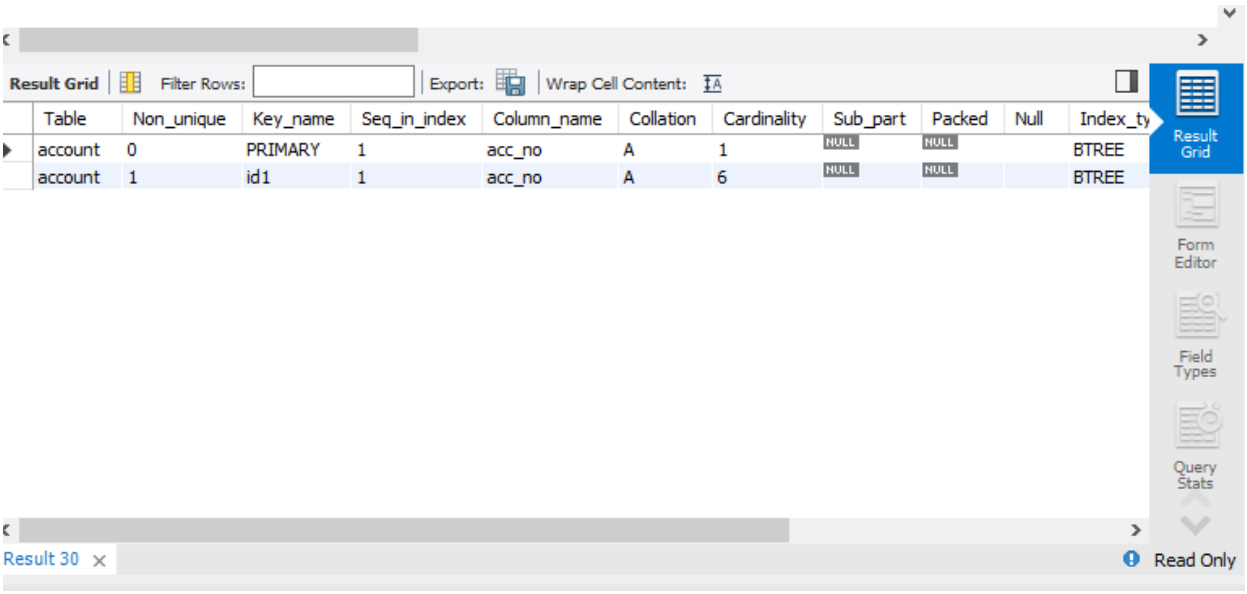
#Q.Display the account details having balance between 15000 and 20000

select * from account where balance between 15000 and 20000;



create index id1 on account (acc_no);

show indexes from account;



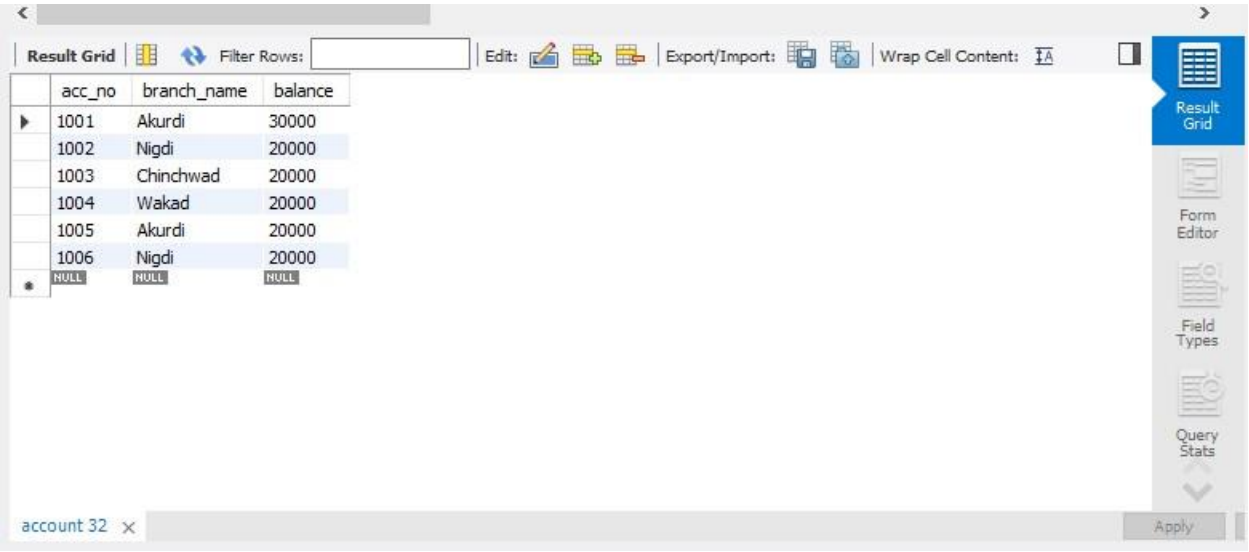
#update

SET SQL_SAFE_UPDATES=0;

update account set balance ='20000';

update account set balance='30000' where acc_no=1001;

select * from account;



alter table account add(loan varchar(10));

update account set loan =10000 where acc_no=1001;

update account set loan =20000 where acc_no=1002;

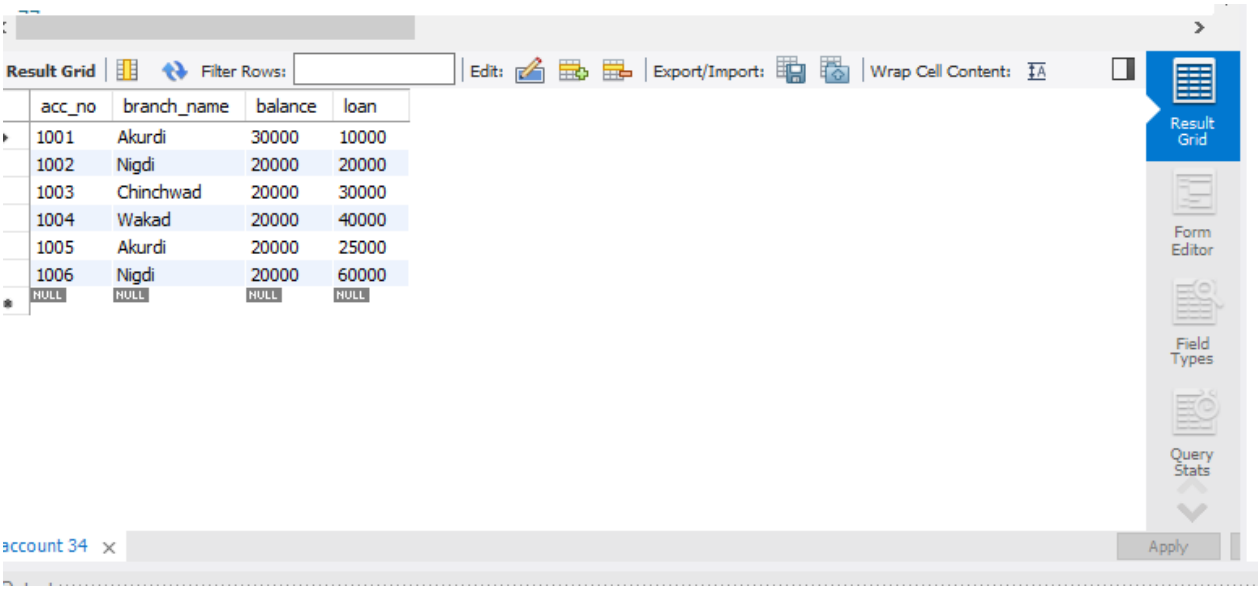
update account set loan =30000 where acc_no=1003;

update account set loan =40000 where acc_no=1004;

update account set loan =25000 where acc_no=1005;

update account set loan =60000 where acc_no=1006;

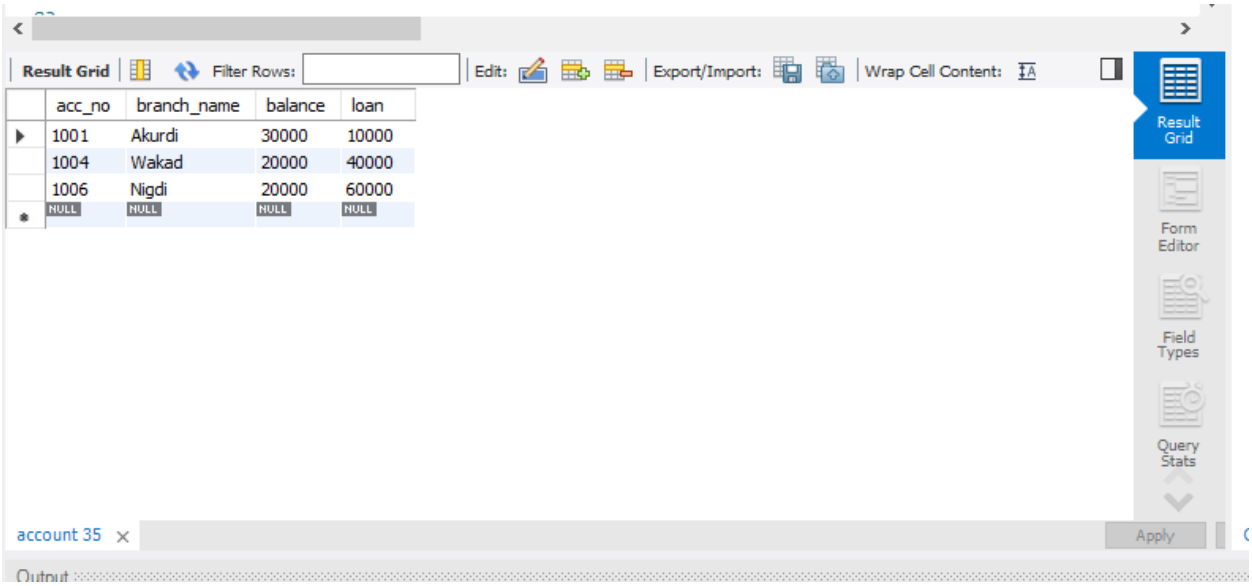
select * from account;



#Q14. DELETE all loans with loan amount between 10000 and 40000.

delete from account where loan>10000 and loan<40000;

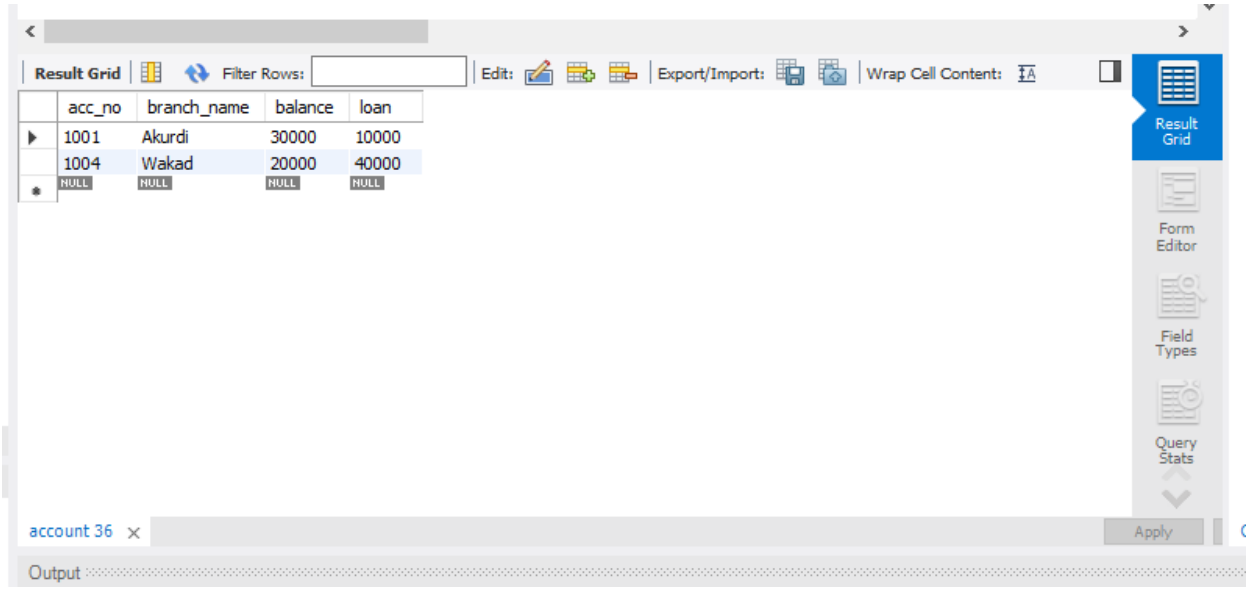
select * from account;



#Q15. DELETE all loans at every branch located in Nigdi.

delete from account where branch_name='Nigdi';

select * from account;



EXPERIMENT NO.4

Samarth Hanji TE59

```
mysql> show databases;
+--------------+
| Database     |
+--------------+
| information_schema |
| mysql        |
| performance_schema |
| student      |
| sys          |
+--------------+
5 rows in set (0.01 sec)

mysql> create database assign;
Query OK, 1 row affected (0.02 sec)

mysql> show databases;
+--------------+
| Database     |
+--------------+
| assign       |
| bank         |
| electronics  |
| estudent     |
| information_schema |
| mysql        |
| performance_schema |
```

```
| student            |
| sys                |
+--------------------+
9 rows in set (0.00 sec)

mysql> use assign;
Database changed

mysql> create table emp1(empno integer, ename varchar(15), joindate date, esal integer, primary key(empno));
Query OK, 0 rows affected (0.05 sec)

mysql> desc emp1;
+--------+-----------+--------+-----+---------+-------+
| Field  | Type      | Null   | Key | Default | Extra |
+--------+-----------+--------+-----+---------+-------+
| empno  | int       | NO     | PRI | NULL    |       |
| ename  | varchar(15) | YES  |     | NULL    |       |
| joindate | date    | YES    |     | NULL    |       |
| esal   | int       | YES    |     | NULL    |       |
+--------+-----------+--------+-----+---------+-------+
4 rows in set (0.01 sec)

mysql> insert into emp1 values
    -> (1,"Suresh",'2005-12-25',4000),
    -> (2,"Mahesh",'2020-10-14',5000),
    -> (3,"Ramesh",'2019-06-20',3000);
Query OK, 3 rows affected (0.02 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

```
mysql> insert into emp1 values
    -> (4,"Shashank",'2015-10-05',2000),
    -> (5,"Binod",'2015-05-24',1000),
    -> (6,"Mohit",'2009-04-10',300);
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> select * from emp1;
+-----+------------+----------+------+
| empno | ename    | joindate   | esal |
+-----+------------+----------+------+
|     1 | Suresh   | 2005-12-25 | 4000 |
|     2 | Mahesh   | 2020-10-14 | 5000 |
|     3 | Ramesh   | 2019-06-20 | 3000 |
|     4 | Shashank | 2015-10-05 | 2000 |
|     5 | Binod    | 2015-05-24 | 1000 |
|     6 | Mohit    | 2009-04-10 |  300 |
+-----+------------+----------+------+
6 rows in set (0.00 sec)

mysql> insert into emp1 (empno) values (11);
Query OK, 1 row affected (0.00 sec)

mysql> select * from emp1;
+-----+------------+----------+------+
| empno | ename    | joindate   | esal |
+-----+------------+----------+------+
```

```
|    1 | Suresh   | 2005-12-25 | 4000 |
|    2 | Mahesh   | 2020-10-14 | 5000 |
|    3 | Ramesh   | 2019-06-20 | 3000 |
|    4 | Shashank | 2015-10-05 | 2000 |
|    5 | Binod    | 2015-05-24 | 1000 |
|    6 | Mohit    | 2009-04-10 |  300 |
|   11 | NULL     | NULL       | NULL |
+------+----------+------------+------+
7 rows in set (0.00 sec)

mysql> create table emp2(empno integer, ename varchar(15), joindate date, esal integer, primary key(empno));
Query OK, 0 rows affected (0.01 sec)

mysql> show tables;
+------------+
| Tables_in_assign |
+------------+
| emp1       |
| emp2       |
+------------+
2 rows in set (0.00 sec)

mysql> insert into emp2 select * from emp1;
Query OK, 7 rows affected (0.00 sec)
Records: 7  Duplicates: 0  Warnings: 0

mysql> select * from emp2;
+------+----------+------------+------+
```

```
| empno | ename    | joindate   | esal |
+-----+----------+--------+----+
|     1 | Suresh   | 2005-12-25 | 4000 |
|     2 | Mahesh   | 2020-10-14 | 5000 |
|     3 | Ramesh   | 2019-06-20 | 3000 |
|     4 | Shashank | 2015-10-05 | 2000 |
|     5 | Binod    | 2015-05-24 | 1000 |
|     6 | Mohit    | 2009-04-10 |  300 |
|    11 | NULL     | NULL       | NULL |
+-----+----------+--------+----+
7 rows in set (0.00 sec)

mysql> select ename, joindate from emp2;
+-------+--------+
| ename    | joindate   |
+-------+--------+
| Suresh   | 2005-12-25 |
| Mahesh   | 2020-10-14 |
| Ramesh   | 2019-06-20 |
| Shashank | 2015-10-05 |
| Binod    | 2015-05-24 |
| Mohit    | 2009-04-10 |
| NULL     | NULL       |
+-------+--------+
7 rows in set (0.00 sec)

mysql> select ename, esal from emp2 where esal > 2000;
+-----+----+
```

```
| ename  | esal |
+-----+----+
| Suresh | 4000 |
| Mahesh | 5000 |
| Ramesh | 3000 |
+-----+----+
3 rows in set (0.00 sec)

mysql> select ename, esal from emp2 where esal < 2000;
+-----+-------+
| ename | esal |
+-----+-------+
| Binod | 1000 |
| Mohit |  300 |
+-----+-------+
2 rows in set (0.00 sec)

mysql> select ename, esal from emp2 where esal >= 2000;
+-------+----+
| ename    | esal |
+-------+----+
| Suresh   | 4000 |
| Mahesh   | 5000 |
| Ramesh   | 3000 |
| Shashank | 2000 |
+-------+----+
4 rows in set (0.00 sec)
```

```
mysql> select ename, esal from emp2 where esal <= 2000;
+-------+----+
| ename    | esal |
+-------+----+
| Shashank | 2000 |
| Binod    | 1000 |
| Mohit    |  300 |
+-------+----+
3 rows in set (0.00 sec)

mysql> select distinct(esal) from emp2;
+----+
| esal |
+----+
| 4000 |
| 5000 |
| 3000 |
| 2000 |
| 1000 |
|  300 |
| NULL |
+----+
7 rows in set (0.00 sec)

mysql> select * from emp2 order by joindate;
+-----+------------+--------+----+
| empno | ename    | joindate   | esal |
+-----+------------+--------+----+
```

```
|    11 | NULL     | NULL       | NULL |
|     1 | Suresh   | 2005-12-25 | 4000 |
|     5 | Binod    | 2015-05-24 | 1000 |
|     4 | Shashank | 2015-10-05 | 2000 |
|     3 | Ramesh   | 2019-06-20 | 3000 |
|     2 | Mahesh   | 2020-10-14 | 5000 |
|     6 | Mohit    | 2009-04-10 |  300 |
+-------+----------+------------+------+
7 rows in set (0.00 sec)

mysql> update emp2 set esal=10000 where ename="Suresh";
Query OK, 1 row affected (0.00 sec)

mysql> select * from emp2;
+-------+----------+------------+-------+
| empno | ename    | joindate   | esal  |
+-------+----------+------------+-------+
|     1 | Suresh   | 2005-12-25 | 10000 |
|     2 | Mahesh   | 2020-10-14 |  5000 |
|     3 | Ramesh   | 2019-06-20 |  3000 |
|     4 | Shashank | 2015-10-05 |  2000 |
|     5 | Binod    | 2015-05-24 |  1000 |
|     6 | Mohit    | 2009-04-10 |   300 |
|    11 | NULL     | NULL       |  NULL |
+-------+----------+------------+-------+
7 rows in set (0.00 sec)

mysql> delete from emp2 where ename="Suresh";
```

Query OK, 1 row affected (0.00 sec)

```
mysql> select * from emp2;
+-----+------------+----------+------+
| empno | ename    | joindate   | esal |
+-----+------------+----------+------+
|     2 | Mahesh   | 2020-10-14 | 5000 |
|     3 | Ramesh   | 2019-06-20 | 3000 |
|     4 | Shashank | 2015-10-05 | 2000 |
|     5 | Binod    | 2015-05-24 | 1000 |
|     6 | Mohit    | 2009-04-10 |  300 |
|    11 | NULL     | NULL       | NULL |
+-----+------------+----------+------+
6 rows in set (0.00 sec)

mysql> select * from emp1 where esal between 2000 and 5000;
+-----+------------+----------+------+
| empno | ename    | joindate   | esal |
+-----+------------+----------+------+
|     1 | Suresh   | 2005-12-25 | 4000 |
|     2 | Mahesh   | 2020-10-14 | 5000 |
|     3 | Ramesh   | 2019-06-20 | 3000 |
|     4 | Shashank | 2015-10-05 | 2000 |
+-----+------------+----------+------+
4 rows in set (0.00 sec)

mysql> select * from emp1 where esal not in (2000,5000);
+-----+---------+----------+-------+
```

| empno | ename | joindate | esal |

+-----+---------+----------+-------+

| 1 | Suresh | 2005-12-25 | 4000 |

| 3 | Ramesh | 2019-06-20 | 3000 |

| 5 | Binod | 2015-05-24 | 1000 |

| 6 | Mohit | 2009-04-10 | 300 |

+-----+---------+----------+-------+

4 rows in set (0.00 sec)

EXPERIMENT NO: 05

Samarth Hanji TE59

create database pune_bank; use

pune_bank;

create table branch(branch_name varchar(20) primary key, branch_city varchar(20), assets int(20));

insert into branch values('Akurdi','Pune',200000); insert

into branch values('Nigadi','PCMC',300000); insert into

branch values('Wakad','Pune',100000); insert into

branch values('Chinchwad','PCMC',400000); insert into

branch values('Sangavi','Pune',230000); select * from

branch;

branch_name	branch_city	assets
Akurdi	Pune	200000
Chinchwad	PCMC	400000
Nigadi	PCMC	300000
Sangavi	Pune	230000
Wakad	Pune	100000
NULL	NULL	NULL

create table account(acc_no int(10) primary key, branch_name varchar(20), balance int(20), constraint FK_S foreign key(branch_name) references branch(branch_name) on delete cascade);

insert into account values(1001,'Akurdi',15000);

insert into account values(1002,'Nigadi',11000);

insert into account values(1003,'Chinchwad',20000);

insert into account values(1004,'Wakad',10000);

insert into account values(1005,'Akurdi',14000);

insert into account values(1006,'Nigadi',17000);

select * from account;

acc_no	branch_name	balance
1001	Akurdi	15000
1002	Nigadi	11000
1003	Chinchwad	20000
1004	Wakad	10000
1005	Akurdi	14000
1006	Nigadi	17000

create table loan(loan_no int(20) primary key, branch_name varchar(20), amount int(20), constraint FK_6 foreign key(branch_name) references branch(branch_name) on delete cascade);

insert into loan values(2001,'Akurdi',2000);

insert into loan values(2002,'Nigadi',1200);

insert into loan values(2003,'Akurdi',1400);

insert into loan values(2004,'Wakad',1350);

insert into loan values(2005,'Chinchwad',1490);

insert into loan values(2006,'Akurdi',12300);

insert into loan values(2007,'Akurdi',14000);

select * from loan;

loan_no	branch_name	amount
2001	Akurdi	2000
2002	Nigadi	1200
2003	Akurdi	1400
2004	Wakad	1350
2005	Chinchwad	1490
2006	Akurdi	12300
2007	Akurdi	14000
NULL	NULL	NULL

create table customer(cust_name varchar(20) primary key, cust_street varchar(20), cust_city varchar(20));

insert into customer values('Rutuja','JM road','Pune'); insert

into customer values('Alka','Senapati road','Pune');

insert into customer values('Samiksha','Savedi road','PCMC');

into customer values('Mahima','Pipeline road','PCMC

insert into customer values('Ayushi','FC road','Pune');

insert into customer values('Priti','Camp road','PCMC');

select * from customer

cust_name	cust_street	cust_city
Alka	Senapati road	Pune
Ayushi	FC road	Pune
Mahima	Pipeline road	PCMC
Priti	Camp road	PCMC
Rutuja	JM road	Pune
Samiksha	Savedi road	PCMC
Trupti	JLakshmi road	Pune
NULL	NULL	NULL

create table depositor(cust_name varchar(20), acc_no integer(10), constraint FK_1 foreign key(cust_name) references customer(cust_name) on delete cascade, constraint FK_2 foreign key (acc_no) references account(acc_no) on delete cascade);

insert into depositor values('Rutuja',1005);

insert into depositor values('Trupti',1002); insert

into depositor values('Samiksha',1004); select *

from depositor;

	cust_name	acc_no
▶	Trupti	1002
	Rutuja	1005
	Samiksha	1004

create table borrower(cust_name varchar(20), loan_no integer(10), constraint FK_3 foreign key (cust_name) references customer(cust_name) on delete cascade, constraint FK_4 foreign key (loan_no) references loan(loan_no) on delete cascade);

insert into borrower values('Mahima',2005);

insert into borrower values('Trupti',2002);

insert into borrower values('Rutuja',2004);

insert into borrower values('Ayushi',2006);

insert into borrower values('Priti',2007); select

* from borrower;

	cust_name	loan_no
▶	Mahima	2005
	Trupti	2002
	Rutuja	2004
	Ayushi	2006
	Priti	2007

#cross join

select * from borrower b,loan l;

select * from borrower b cross join loan l;

	cust_nam	loan_no	loan_no	branch_na	amount
1	Priti	2007	2001	Akurdi	2000
2	Ayushi	2006	2001	Akurdi	2000
3	Rutuja	2004	2001	Akurdi	2000
4	Trupti	2002	2001	Akurdi	2000
5	Mahima	2005	2001	Akurdi	2000
6	Priti	2007	2002	Nigadi	1200
7	Ayushi	2006	2002	Nigadi	1200
8	Rutuja	2004	2002	Nigadi	1200
9	Trupti	2002	2002	Nigadi	1200
10	Mahima	2005	2002	Nigadi	1200
11	Priti	2007	2003	Akurdi	1400
12	Ayushi	2006	2003	Akurdi	1400
13	Rutuja	2004	2003	Akurdi	1400
14	Trupti	2002	2003	Akurdi	1400
15	Mahima	2005	2003	Akurdi	1400
16	Priti	2007	2004	Wakad	1350
17	Ayushi	2006	2004	Wakad	1350
18	Rutuja	2004	2004	Wakad	1350

#Natural join

select * from borrower b,loan l where b.loan_no = l.loan_no;

	cust_name	loan_no	loan_no	branch_name	amount
▶	Mahima	2005	2005	Chinchwad	1490
	Trupti	2002	2002	Nigadi	1200
	Rutuja	2004	2004	Wakad	1350
	Ayushi	2006	2006	Akurdi	12300
	Priti	2007	2007	Akurdi	14000

select b.cust_name, l.loan_no, l.amount from borrower b, loan l where b.loan_no = l.loan_no;

cust_name	loan_no	amount
Mahima	2005	1490
Trupti	2002	1200
Rutuja	2004	1350
Ayushi	2006	12300
Priti	2007	14000

#inner join

select b.cust_name, b.loan_no, l.amount from borrower b inner join loan l on b.loan_no=l.loan_no;

cust_name	loan_no	amount
Mahima	2005	1490
Trupti	2002	1200
Rutuja	2004	1350
Ayushi	2006	12300
Priti	2007	14000

select b.cust_name from borrower b inner join loan l on b.loan_no=l.loan_no where l.branch_name='Akurdi' order by b.cust_name;

cust_name
Ayushi
Priti

select branch_name, count(branch_name) from account a inner join depositor d on a.acc_no=d.acc_no group by branch_name;

branch_name	count(branch_name)
Nigadi	1
Wakad	1
Akurdi	1

right outer join

select b.cust_name,l.loan_no, l.amount from borrower b right join loan l on b.loan_no=l.loan_no;

cust_name	loan_no	amount
NULL	2001	2000
Trupti	2002	1200
NULL	2003	1400
Rutuja	2004	1350
Mahima	2005	1490
Ayushi	2006	12300
Priti	2007	14000

#left outer join

select b.cust_name, l.loan_no,l.amount from borrower b left join loan l on b.loan_no=l.loan_no;

cust_name	loan_no	amount
Mahima	2005	1490
Trupti	2002	1200
Rutuja	2004	1350
Ayushi	2006	12300
Priti	2007	14000

#full join

select * from borrower b left join loan l on b.loan_no=l.loan_no union select * from borrower b right join loan l on b.loan_no=l.loan_no;

Result Grid | Filter Rows: | Export:

cust_name	loan_no	loan_no	branch_name	amount
Mahima	2005	2005	Chinchwad	1490
Trupti	2002	2002	Nigadi	1200
Rutuja	2004	2004	Wakad	1350
Ayushi	2006	2006	Akurdi	12300
Priti	2007	2007	Akurdi	14000
NULL	NULL	2001	Akurdi	2000
NULL	NULL	2003	Akurdi	1400

EXPERIMENT NO:06

Samarth Hanji TE59

```
create database library;

use library;

create table borrower(Roll_no int primary key, name varchar(20), Date_of_issue Date,
Name_of_book varchar(20), Status char(20));

insert into borrower values(1, 'abt','2025-09-01','SEPM','I');

insert into borrower values(2, 'xyz','2025-08-17','OOP','I');

insert into borrower values(3, 'pqr','2025-06-29','DBMS','I');

insert into borrower values(4, 'def','2025-08-30','DSA','I');

insert into borrower values(5, 'lmn','2025-09-15','ADS','I');

create table fine(Roll_no int(10),foreign key(Roll_no) references borrower(Roll_no),
Return_Date Date, Amount int(10));

select * from borrower;
```

Roll_no	name	Date_of_issue	Name_of_book	Status
1	abt	2025-09-01	SEPM	I
2	xyz	2025-08-17	OOP	I
3	pqr	2025-06-29	DBMS	I
4	def	2025-08-30	DSA	I
5	lmn	2025-09-15	ADS	I
NULL	NULL	NULL	NULL	NULL

```
create table fine(Roll_no int(10),foreign key(Roll_no) references borrower(Roll_no),
Return_Date Date, Amount int(10));


delimiter $

create procedure calc_fine_lib(in roll int)

begin

declare fine1 int;

declare noofdays int;

declare issuedate date;

declare exit handler for sqlexception select 'create table definition';
```

```
select Date_of_issue into issuedate from borrower where Roll_no=roll;
select datediff(curdate(),issuedate) into noofdays;
 if noofdays>15 and noofdays<=30 then
 set fine1=noofdays*5;
 insert into fine values(roll,curdate(),fine1);
 elseif noofdays>30 then
 set fine1=((noofdays-30)*50) + 30*5;
 insert into fine values(roll,curdate(),fine1);
 else
 insert into fine values(roll,curdate(),0);
 end if;
 update borrower set status='R' where Roll_no=roll;
 end$
 delimiter ;
call   calc_fine_lib(1);
call   calc_fine_lib(2);
call   calc_fine_lib(3);
call   calc_fine_lib(4);
call calc_fine_lib(5);
```

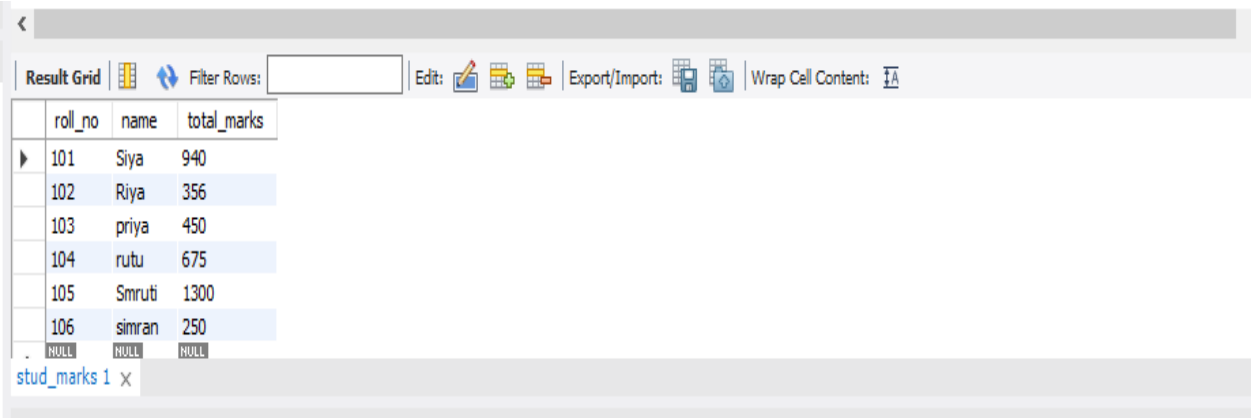
```
 select * from fine;
```

Roll_no	Return_Date	Amount
1	2025-09-17	80
2	2025-09-17	200
3	2025-09-17	2650
4	2025-09-17	90
5	2025-09-17	0

EXPERIMENT NO.8

Samarth Hanji TE59

```
create database exam;
use exam;
create table stud_marks(roll_no int primary key, name char(10),total_marks int);
insert into stud_marks values(101,'Siya',940);
insert into stud_marks values(102,'Riya',356);
insert into stud_marks values(103,'priya',450);
insert into stud_marks values(104,'rutu',675);
insert into stud_marks values(105,'Smruti',1300);
insert into stud_marks values(106,'simran',250);
select*from stud_marks;
```

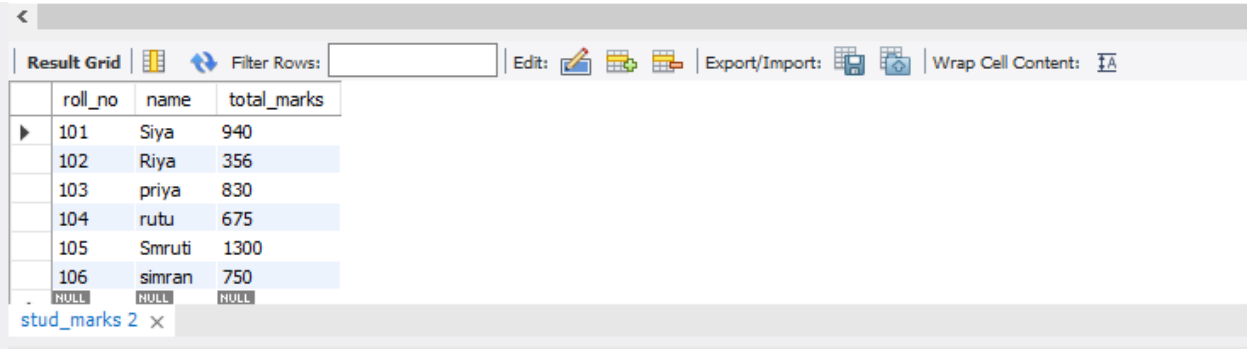


Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

roll_no	name	total_marks
101	Siya	940
102	Riya	356
103	priya	450
104	rutu	675
105	Smruti	1300
106	simran	250
NULL	NULL	NULL

stud_marks 1

```
update stud_marks SET total_marks=830 WHERE roll_no=103;
update stud_marks SET total_marks=750 WHERE roll_no=106;
```



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

roll_no	name	total_marks
101	Siya	940
102	Riya	356
103	priya	830
104	rutu	675
105	Smruti	1300
106	simran	750
NULL	NULL	NULL

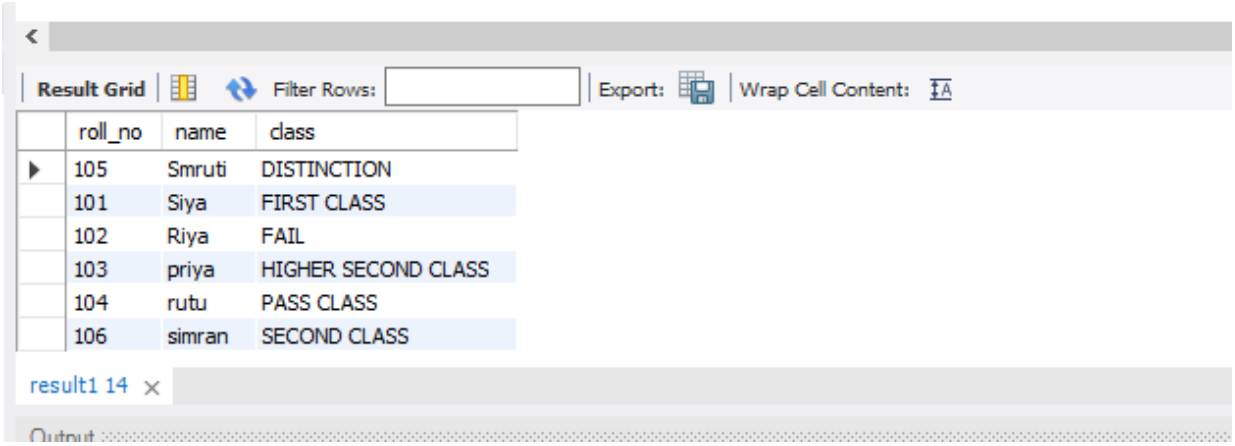
stud_marks 2

```
create table result1(roll_no int, name char(30),class char(50));
```

```
delimiter $$
create procedure proc_grade1(in marks int, out class char(50))
begin
if marks<=1500 and marks>=990 then set class='DISTINCTION';
end if;
if marks<=989 and marks>=900 then set class='FIRST CLASS';
end if;
if marks<=899 and marks>=825 then set class='HIGHER SECOND CLASS';
end if;
if marks<=824 and marks>=750 then set class='SECOND CLASS';
end if;
if marks<=749 and marks>=650 then set class= 'PASS CLASS';
end if;
if marks< 650 then set class='FAIL';
end if;
end $
delimiter ;

delimiter $$
create function find_result1(roll_in int) returns int deterministic
begin
declare fmarks int;
declare grade char(10);
declare stud_name char(10);
select stud_marks.total_marks,stud_marks.name into fmarks, stud_name from
stud_marks where stud_marks.roll_no=roll_in;
call proc_grade1(fmarks,@grade);
insert into result1 values(roll_in,stud_name,@grade);
```

```
return roll_in;
end $
delimiter ;
select find_result1(105);
select*from result1;
select find_result1(101);
select*from result1;
select find_result1(102);
select*from result1;
select find_result1(103);
select*from result1;
select find_result1(104);
select*from result1;
select find_result1(106);
select*from result1;
```

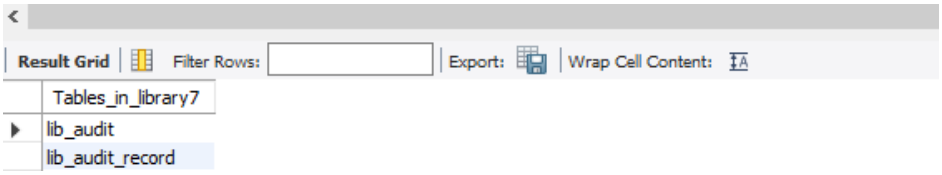


roll_no	name	class
105	Smruti	DISTINCTION
101	Siya	FIRST CLASS
102	Riya	FAIL
103	priya	HIGHER SECOND CLASS
104	rutu	PASS CLASS
106	simran	SECOND CLASS

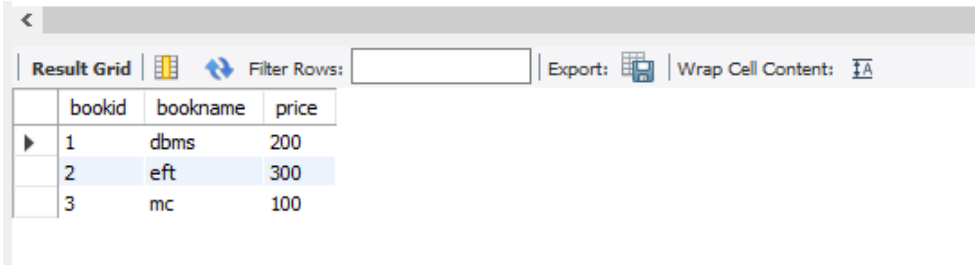
EXPERIMENT NO.9

Samarth Hanji TE59

```
create database library7;
use library7;
create table lib_audit(bookid int, bookname varchar(20),price int);
create table lib_audit_record(bookid int, bookname varchar(20),price int);
show tables;
```




```
insert into lib_audit values(1,'dbms',200);
insert into lib_audit values(2,'eft',300);
insert into lib_audit values(3,'mc',100);
select *from lib_audit;
```




```
delimiter $
create trigger before_delete_lib_audit before delete on lib_audit for each row begin
insert into lib_audit_record values(old.bookid,old.bookname,old.price);
end $
delimiter ;

select*from lib_audit_record;
set SQL_SAFE_UPDATES =0;
```

delete from lib_audit where bookid=3;

bookid	bookname	price
3	mc	100

insert into lib_audit values (4,'CS',350);

delete from lib_audit where bookname='CS';

select*from lib_audit_record;

bookid	bookname	price
3	mc	100
4	CS	350

#update

delimiter $

create trigger before_update_lib_audit before update on lib_audit for each row begin

insert into lib_audit_record values(old.bookid,old.bookname,old.price);

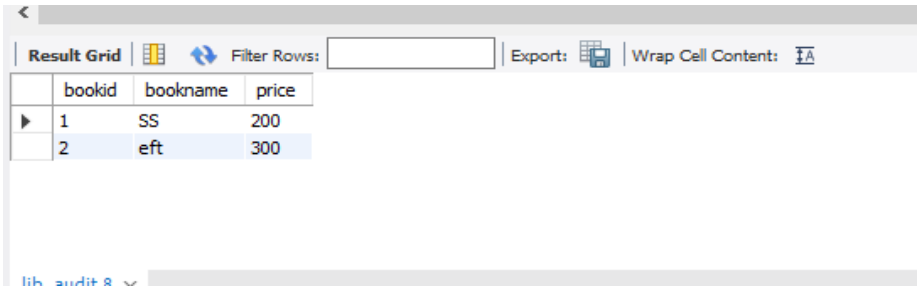
end $

delimiter ;

update lib_audit set bookname ='SS' where bookid=1;

select *from lib_audit;

select*from lib_audit_record;



bookid	bookname	price
1	SS	200
2	eft	300