

# **20MCA134 ADVANCED DATABASE MANAGEMENT SYSTEMS LAB**

*Lab Report Submitted By*

**SALINI K B**

**Reg. No.: AJC21MCA-2091**

*In Partial fulfillment for the Award of the Degree Of*

**MASTER OF COMPUTER APPLICATIONS (2 Year)  
(MCA)**

**APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY**



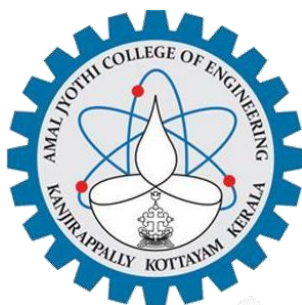
**AMAL JYOTHI COLLEGE OF ENGINEERING  
KANJIRAPPALLY**

[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE,  
Accredited by NAAC with 'A' grade. Koovapally, Kanjirappally, Kottayam, Kerala – 686518]

**2021-2023**

# **DEPARTMENT OF COMPUTER APPLICATIONS**

## **AMAL JYOTHI COLLEGE OF ENGINEERING KANJIRAPPALLY**



### **CERTIFICATE**

This is to certify that the lab report, “**20MCA134 ADVANCED DATABASE MANAGEMENT SYSTEMS LAB**” is the bonafide work of **SALINI K B (AJC21MCA-2091)** in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications under APJ Abdul Kalam Technological University during the year 2021-23.

**SR,Elsin chakkalakkal**

**Lab In-Charge**

**Rev.Fr.Dr.Rubin Thottupuram Jose**

**Head of the Department**

**Internal Examiner**

**External Examiner**

---

**CONTENT**

<b>Sl. No</b>	<b>Content</b>	<b>Date</b>	<b>Page No:</b>
1	To familiarize DDL commands - CREATE ,ALTER,DROP,TRUNCATE,RENAME	08-04-2022	4-11
2	To familiarize DML Commands SELECT,INSERT,UPDATE,DELETE	25-03-2022	12-20
3	To familiarize with set operations	19-04-2022	21-23
4	To familiarize with join or cartesian product	06-05-2022	24-32
5	To familiarize with Group by and Having clause	06-05-2022	33-39
6	Implementation of triggers	10-06-2022	40-42
7	Installation of MongoDB on Windows	24-05-2022	43-46
8	Designing Databases using NoSQL: MongoDB	24-05-2022	47-48
9	Query Processing : Performing CRUD operations with NoSQL database	03-06-2022	49-53
10	NoSQL and Front-End: PHP: Create a PHP form and insert data to mongodb	03-06-2022	54-56

**Program no:1**

**Aim:** To familiarize DDL commands -  
CREATE  
,ALTER,DROP,TRUNCATE,RENAME.

**Name:** salini kb

**Roll No:**33

**Batch:** MCA-B

**Date:** 08-04-2022

**Procedure****Questions**

1.Create a table emp1 with attributes empno number(4)as primary key, ename char(10),hiredate, salary, commission

insert 5 rows of data

```
CREATE DATABASE EMP;
```

```
CREATE TABLE emp1(EMPNO INT(4) PRIMARY KEY, ENAME CHAR(10),  
HIREDATE DATE, SALARY INT(5),COMMISSION INT(5));
```

```
INSERT INTO emp1(EMPNO,ENAME,HIREDATE,SALARY) VALUES  
(101,'RAMESH','1980-01-17',5000),
```

```
(102,'AJAY','1985-07-05',5000),
```

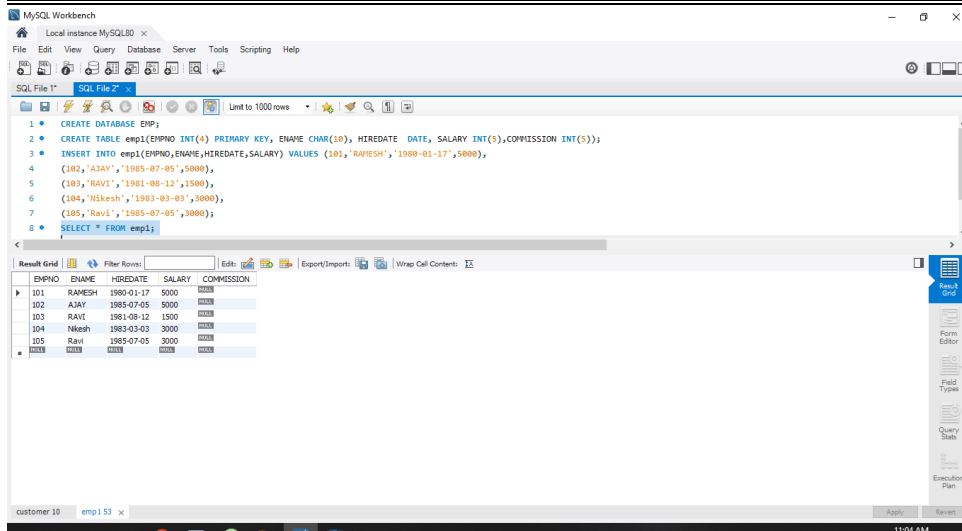
```
(103,'RAVI','1981-08-12',1500),
```

```
(104,'Nikesh','1983-03-03',3000),
```

```
(105,'Ravi','1985-07-05',3000);
```

```
SELECT * FROM emp1;
```

**Output Screenshot**



## 2.Modifying the structure of tables

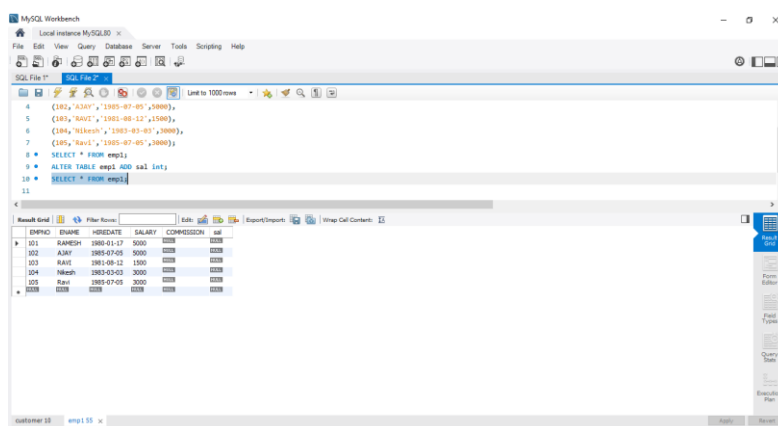
a.Add new columns: sal number(7,2)

```
SELECT * FROM emp1;
```

```
ALTER TABLE emp1 ADD sal int;
```

```
SELECT * FROM emp1;
```

## Output Screenshot

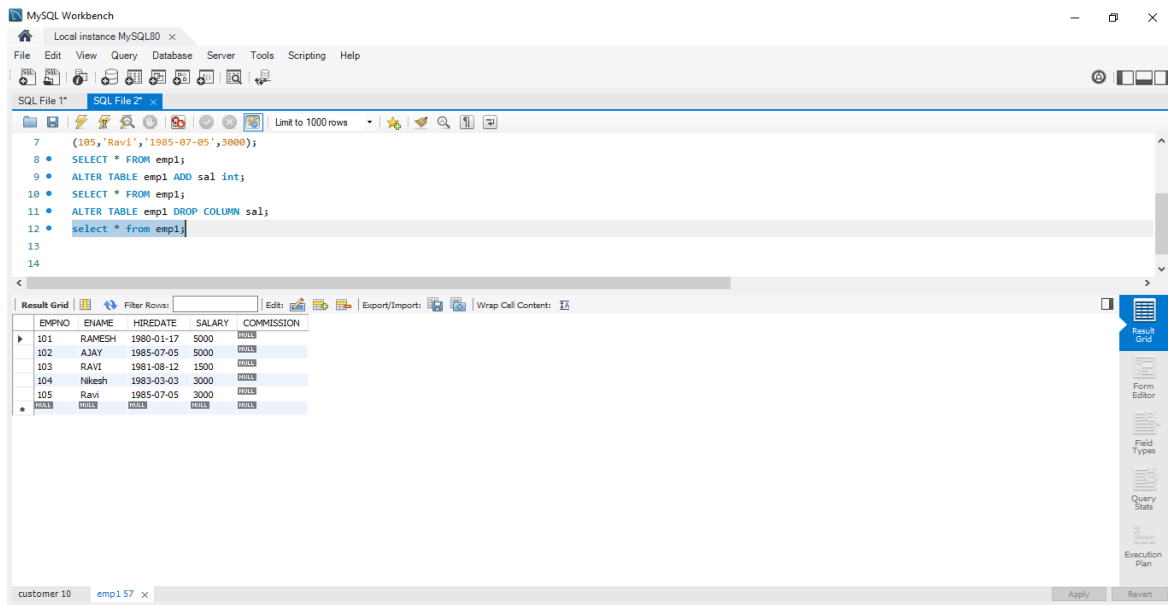


b.Dropping a column from a table: sal

```
ALTER TABLE emp1 DROP COLUMN sal;
```

```
select * from emp1;
```

## Output Screenshot

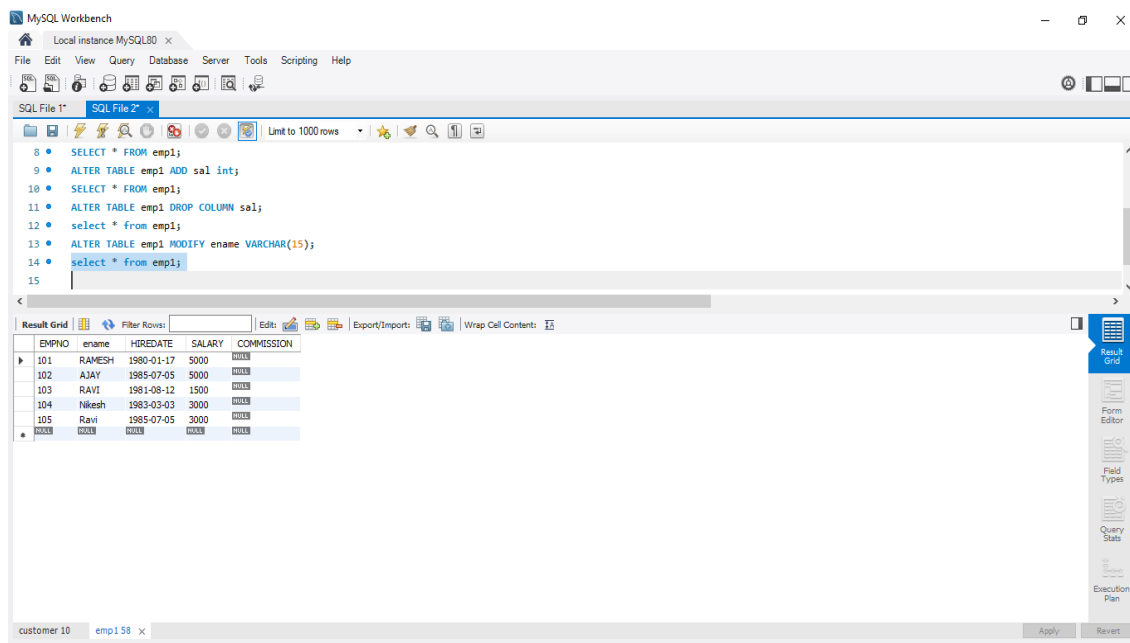


c.Modifying existing column :ename varchar2(15)

```
ALTER TABLE emp1 MODIFY ename VARCHAR(15);
```

```
select * from emp1;
```

## Output Screenshot

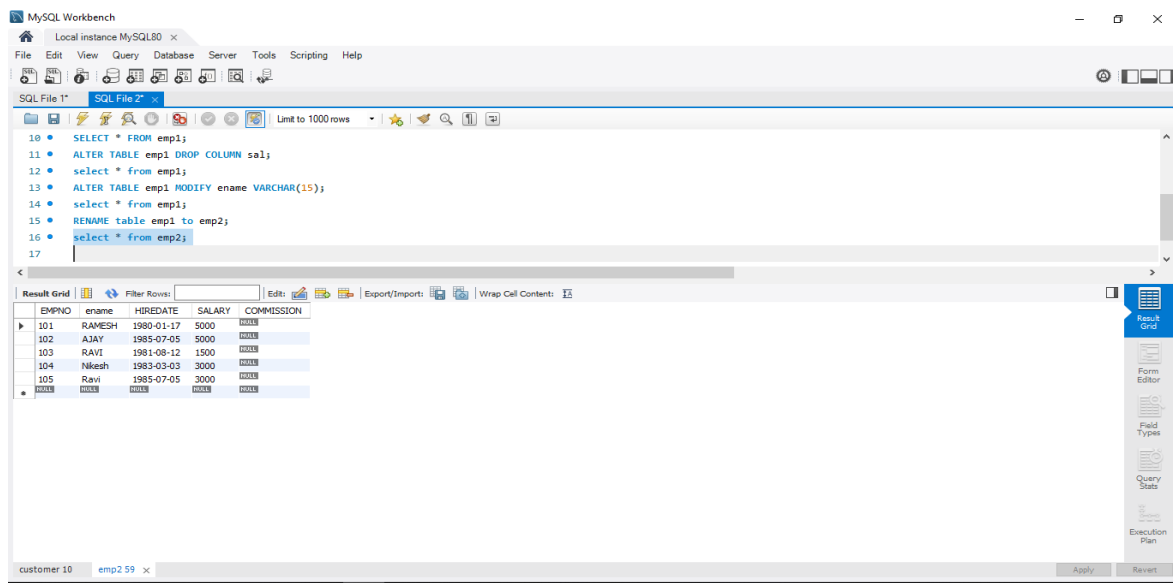


d.Renaming the tables: emp1 to emp2

```
RENAME table emp1 to emp2;
```

```
select * from emp2;
```

## Output Screenshot

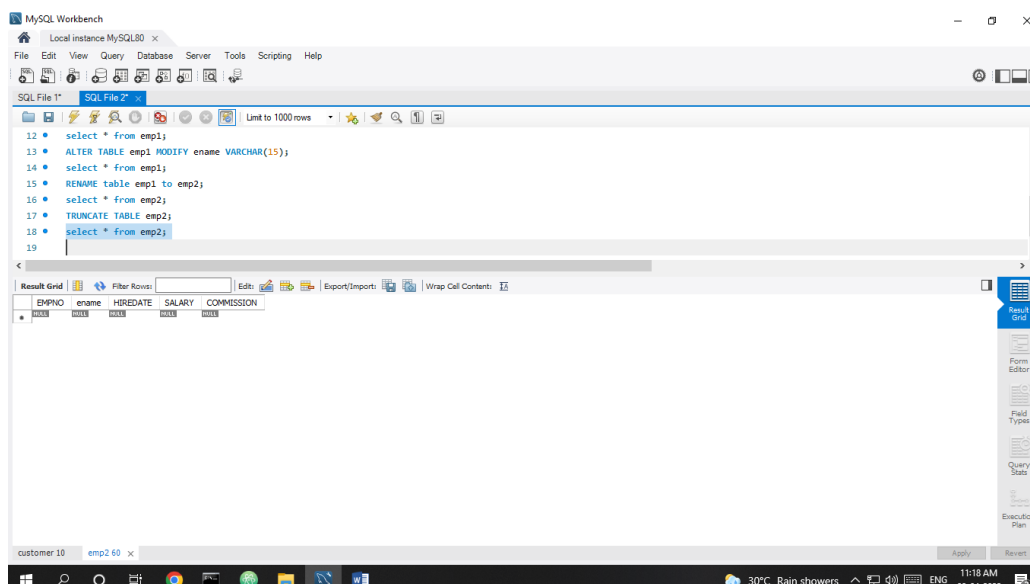


e.truncating the tables:emp2

```
TRUNCATE TABLE emp2;
```

```
select * from emp2;
```

## Output Screenshot



## f. Destroying tables:emp

DROP TABLE emp2;

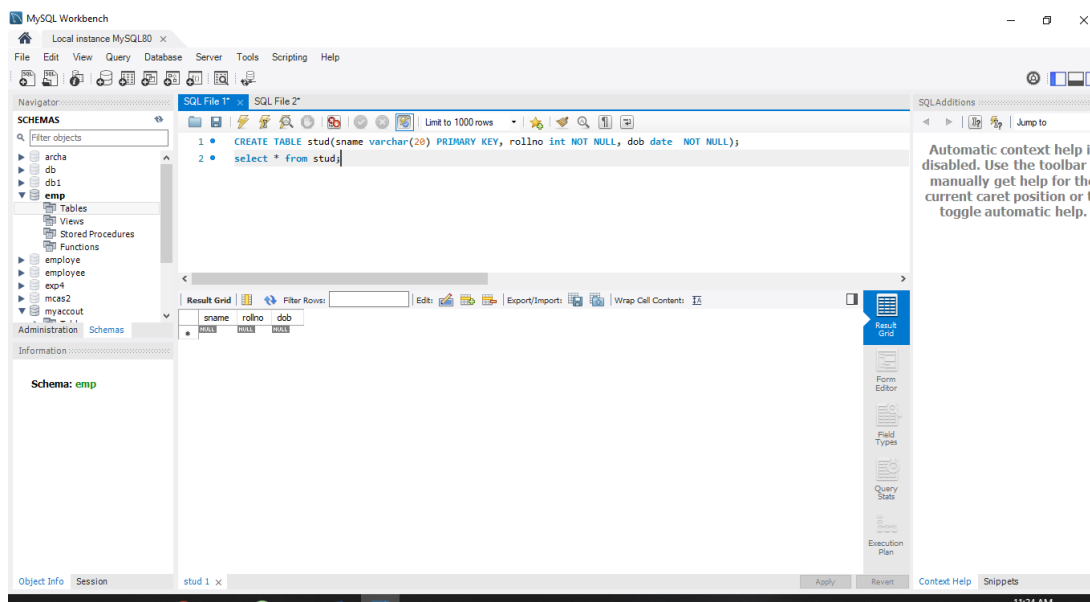
**Output Screenshot**

85	11:15:58	select * from emp2 LIMIT 0, 1000	5 row(s) returned	0.063 sec / 0.000 sec
86	11:17:44	TRUNCATE TABLE emp2	0 row(s) affected	0.985 sec
87	11:18:01	select * from emp2 LIMIT 0, 1000	0 row(s) returned	0.031 sec / 0.000 sec
88	11:19:15	DROP TABLE emp2	0 row(s) affected	0.594 sec
89	11:19:47	CREATE DATABASE EMP	Error Code: 1007. Can't create database 'emp'; database exists	0.000 sec

3. Create a table stud with sname varchar2(20) primary key, rollno number(10) not null, dob date not null

CREATE TABLE stud(sname varchar(20) PRIMARY KEY, rollno int NOT NULL, dob date NOT NULL);

select \* from stud;

**Output Screenshot**

4. Create a table student as regno number (6), mark number (3) check constraint (mark >=0 and mark

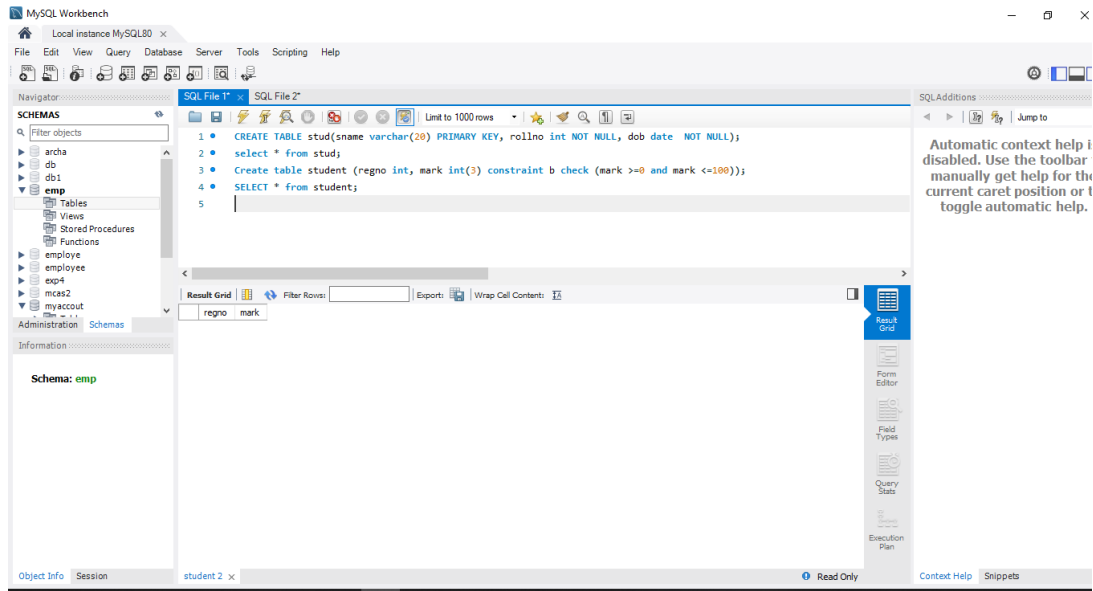
<=100));



Create table student (regno int, mark int(3) constraint b check (mark >=0 and mark <=100));

SELECT \* from student;

## Output Screenshot

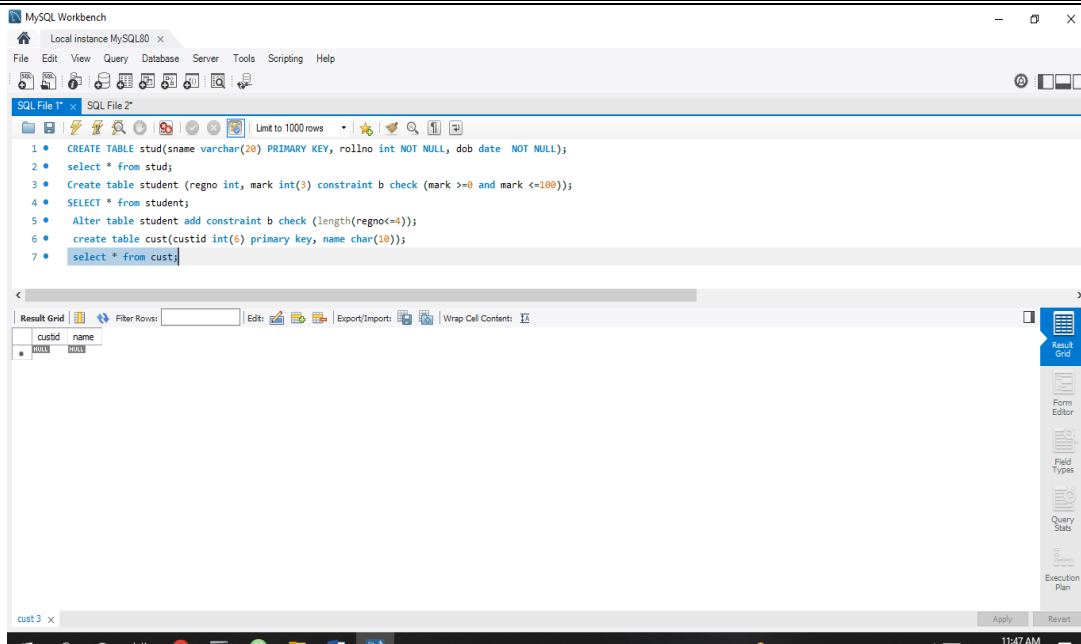


5.Create a table cust with(custid number(6) constraint unique, name char(10)

create table cust(custid int(6) primary key, name char(10));

select \* from cust;

## Output Screenshot



CREATE TABLE stud(sname varchar(20) PRIMARY KEY, rollno int NOT NULL, dob date NOT NULL);

select \* from stud;

Create table student (regno int, mark int(3) constraint b check (mark >=0 and mark <=100));

SELECT \* from student;

Alter table student add constraint b check (length(regno)<=4));

create table cust(custid int(6) primary key, name char(10));

select \* from cust;

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

SQL File 1' SQL File 2'

Output

#	Time	Action	Message	Duration / Fetch
1	11:23:47	CREATE TABLE studname varchar(20) PRIMARY KEY, rollno int NOT NULL, dob date NOT NULL	0 row(s) affected	0.375 sec
2	11:24:22	select * from stud LIMIT 0, 1000	0 row(s) returned	0.046 sec / 0.000
3	11:26:22	Create table student (regno int, mark int(3) constraint to check (mark >=0 and mark <=100))	0 row(s) affected, 1 warning(s): 1681 Integer display width is deprecated and will be removed in a future release.	0.688 sec
4	11:27:04	SELECT * from student LIMIT 0, 1000	0 row(s) returned	0.046 sec / 0.000
5	11:32:05	CREATE TABLE emp(EMPNO INT(4) PRIMARY KEY, ENAME CHAR(10), HIREDATE DATE, SALARY INT(5), ...	0 row(s) affected, 3 warning(s): 1681 Integer display width is deprecated and will be removed in a future release. 168...	0.500 sec
6	11:32:49	INSERT INTO emp(EMPNO,ENAME,HIREDATE,SALARY) VALUES(101,'RAMESH','1980-01-17',5000,null)	Error Code: 1136. Column count doesn't match value count at row 1	0.031 sec
7	11:38:49	Alter table student add constraint s2 check (length(regno)<=4)	Error Code: 3612. An expression of non-boolean type specified to a check constraint 's2'	0.000 sec
8	11:39:07	Alter table student add constraint s2 check (length(regno)<=4)	Error Code: 3612. An expression of non-boolean type specified to a check constraint 's2'	0.000 sec
9	11:43:35	Alter table student add constraint s2 check (length(regno)<=4)	Error Code: 3612. An expression of non-boolean type specified to a check constraint 's2'	0.000 sec
10	11:46:22	CREATE TABLE studname varchar(20) PRIMARY KEY, rollno int NOT NULL, dob date NOT NULL	Error Code: 1050. Table 'stud' already exists	0.000 sec
11	11:46:47	create table cust(custid number(5) primary key, Name char(10))	Error Code: 1054. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server v...	0.000 sec
12	11:47:21	create table cust(custid int(5) primary key, name char(10))	0 row(s) affected, 1 warning(s): 1681 Integer display width is deprecated and will be removed in a future release.	0.297 sec
13	11:47:39	select * from cust LIMIT 0, 1000	0 row(s) returned	0.031 sec / 0.000

---

**Program no:2****Aim:** To familiarize DML Commands SELECT,INSERT,UPDATE,DELETE**Name:** salini kb**Roll No:**33**Batch:** MCA-B**Date:** 25-03-2022**Question**

Create the following Tables and insert values

**Table 1: DEPOSIT**ACTNO VARCHAR2(5) PRIMARY KEY, FIRST LETTER MUST START WITH  
'D' CNAME VARCHAR2(15) FOREIGN KEY REFERENCES CUSTOMERBNAME VARCHAR2(20) FOREIGN KEY REFERENCES  
BRANCH AMOUNT NUMBER (8,2) NOT NULL, CANNOT BE  
0

ADATE DATE

**Table 2: BRANCH**

BNAME VARCHAR2(20) PRIMARY KEY

CITY VARCHAR2(30) NOT NULL , any one of NAGPUR, DELHI, BANGALORE,  
BOMBAY**Table 3: CUSTOMER**CNAME VARCHAR2(15) PRIMARY  
KEY CITY VARCHAR2(20) NOT  
NULL ,**Table 4: BORROW**LOANNO VARCHAR2(8) PRIMARY KEY / FIRST LETTER MUST START WITH  
'L' CNAME VARCHAR2(15) FOREIGN KEY REFERENCES CUSTOMERBNAME VARCHAR2(20) FOREIGN KEY REFERENCES  
BRANCH AMOUNT NUMBER(8,2) NOT NULL, CANNOT BE 0

1. List all data from table deposite
2. List all data from borrow
3. List all data from customer
4. List all data from branch

5. Give account no and amount of deposit
6. Give customer name and account no of depositors
7. Give name of customers
8. Give name of branches
9. Give name of borrowers
10. Give names of customer living in city Nagpur
11. Give names of depositors having amount greater than 4000
12. Give account date of Anil
13. Give name of all branches located in Bombay
14. Give name of borrower having loan number 1205
15. Give names of depositors having account at VRCE
16. Give names of all branches located in city Delhi
17. Give name of the customers who opened account date '1-12-96'
18. Give account no and deposit amount of customers having account opened between dates '1-12-96' and '1-5-96'
19. Give name of the city where branch KAROLBAGH is located
20. Give details of customer ANIL

### **Procedure**

CREATE DATABASE MYACCOUNT;

USE MYACCOUNT;

#### **BRANCH**

CREATE TABLE BRANCH(  
BNAME VARCHAR(20) PRIMARY KEY,  
CITY VARCHAR(30) NOT NULL );

---

```
alter table BRANCH add CITY Varchar(30) check( CITY in ( "NAGPUR", "DELHI", "BANGALORE", "BOMBAY"));
```

```
INSERT INTO BRANCH VALUES('VRCE','NAGPUR');
```

```
INSERT INTO BRANCH VALUES('AJNI','NAGPUR');
```

```
INSERT INTO BRANCH VALUES('KAROLBAGH','DELHI');
```

```
INSERT INTO BRANCH VALUES('CHANDNI','DELHI');
```

```
INSERT INTO BRANCH VALUES('DHARAMPETH','NAGPUR');
```

```
INSERT INTO BRANCH VALUES('MG ROAD','BANGALORE');
```

```
INSERT INTO BRANCH VALUES('ANDHERI','BOMBAY');
```

```
INSERT INTO BRANCH VALUES('NEHRU PALACE','DELHI');
```

```
INSERT INTO BRANCH VALUES('POWAI','BOMBAY');
```

## **CUSTOMER**

```
create table CUSTOMER
```

```
(CNAME VARCHAR(15) PRIMARY KEY ,
```

```
CITY VARCHAR(20) NOT NULL );
```

```
INSERT INTO CUSTOMER VALUES('ANIL','CALCUTTA');
```

```
INSERT INTO CUSTOMER VALUES('SUNIL','DELHI');
```

```
INSERT INTO CUSTOMER VALUES('MEHUL','BARODA');
```

```
INSERT INTO CUSTOMER VALUES('MANDAR','PATNA');
```

```
INSERT INTO CUSTOMER VALUES('MADHURI','NAGPUR');
```

```
INSERT INTO CUSTOMER VALUES('PRAMOD','NAGPUR');
```

```
INSERT INTO CUSTOMER VALUES('SANDIP','SURAT');
```

```
INSERT INTO CUSTOMER VALUES('SHIVANI','BOMBAY');
```

```
INSERT INTO CUSTOMER VALUES('KRANTI','BOMBAY');
```

```
INSERT INTO CUSTOMER VALUES('NAREN','BOMBAY');
```

## **BORROW**

---

```
CREATE TABLE BORROW (LOANNO VARCHAR(8) CHECK (LOANNO LIKE 'L%')
PRIMARY KEY,CNAME
```

```
VARCHAR(15) REFERENCES CUSTOMER(CNAME),BNAME VARCHAR(20)
REFERENCES
```

```
BRANCH(BNAME),AMOUNT FLOAT(8) CHECK (AMOUNT>0) NOT NULL);
```

```
INSERT INTO BORROW VALUES('L201','ANIL','VRCE',1000.00);
```

```
INSERT INTO BORROW VALUES('L206','MEHUL','AJNI',5000.00);
```

```
INSERT INTO BORROW VALUES('L311','SUNIL','DHARAMPETH',3000.00);
```

```
INSERT INTO BORROW VALUES('L321','MADHURI','ANDHERI',2000.00);
```

```
INSERT INTO BORROW VALUES('L371','PRAMOD','VIRAR',8000.00);
```

```
INSERT INTO BORROW VALUES('L481','KRANTI','NEHRU PLACE',3000.00);
```

## DEPOSIT

```
CREATE TABLE DEPOSIT (ACTNO VARCHAR(5) CHECK (ACTNO LIKE 'D%') PRIMARY
KEY ,CNAME
```

```
VARCHAR(15) REFERENCES CUSTOMER(CNAME),BNAME VARCHAR(20)
REFERENCES
```

```
BRANCH(BNAME),AMOUNT FLOAT(8) CHECK (AMOUNT>0) NOT NULL, ADATE
varchar(30));
```

```
INSERT INTO DEPOSIT VALUES('D100','ANIL','VRCE',1000.00,'1-MAR-95');
```

```
INSERT INTO DEPOSIT VALUES('D101','SUNIL','ANJNI',500.00,'4-JAN-96');
```

```
INSERT INTO DEPOSIT VALUES('D102','MEHUL','KAROLBAGH',3500.00,'17-NOV- 95');
```

```
INSERT INTO DEPOSIT VALUES('D104','MADHURI','CHANDNI',1200.00,'17-DEC-95');
```

```
INSERT INTO DEPOSIT VALUES('D105','PRAMOD','MG ROAD',3000.00,'27-MAR-96');
```

```
INSERT INTO DEPOSIT VALUES('D106','SANDIP','ANDHERI',2000.00,'31-MAR-96');
```

```
INSERT INTO DEPOSIT VALUES('D107','SHIVANI','VIRAR',1000.00,'5-SEP-95');
```

```
INSERT INTO DEPOSIT VALUES('D108','KRANTI','NEHRU PLACE',5000.00,'2-JUL- 95');
```

```
INSERT INTO DEPOSIT VALUES('D109','MINU','POWAI',7000.00,'10-AUG-95');
```

```
SELECT * FROM myaccount.branch;
```

```
SELECT * FROM myaccount.borrow;
```

```
SELECT * FROM myaccount.borrow;
```

```
SELECT * FROM myaccount.customer;
```

### Output Screenshot

	BNAME	CITY
	CHANDNI	DELHI
	DHARAMPETH	NAGPUR
	KAROLBAGH	DELHI
	MG ROAD	BANGALORE
	NEHRU PALACE	DELHI
	POWAI	BOMBAY
	VRCE	NAGPUR
*	NULL	NULL

	LOANNO	CNAME	BNAME	AMOUNT
	L201	ANIL	VRCE	1000
	L206	MEHUL	AJNI	5000
▶	L311	SUNIL	DHARAMPETH	3000
	L321	MADHURI	ANDHERI	2000
	L371	PRAMOD	VIRAR	8000
	L481	KRANTI	NEHRU PLACE	3000
*	NULL	NULL	NULL	NULL



	ACTNO	CNAME	BNAME	AMOUNT	ADATE
	D101	SUNIL	ANJNI	500	4-JAN-96
	D102	MEHUL	KAROLBAGH	3500	17-NOV-95
	D104	MADHURI	CHANDNI	1200	17-DEC-95
	D105	PRAMOD	MG ROAD	3000	27-MAR-96
	D106	SANDIP	ANDHERI	2000	31-MAR-96
	D107	SHIVANI	VIRAR	1000	5-SEP-95
	D108	KRANTI	NEHRU PLACE	5000	2-JUL-95
	D109	MINU	POWAI	7000	10-AUG-95
*	NULL	NULL	NULL	NULL	NULL

	CNAME	CITY
	MADHURI	NAGPUR
	MANDAR	PATNA
	MEHUL	BARODA
	NAREN	BOMBAY
	PRAMOD	NAGPUR
	SANDIP	SURAT
	SHIVANI	BOMBAY
	SUNIL	DELHI
*	NULL	NULL

1. SELECT \* FROM myaccount.deposit;

	ACTNO	CNAME	BNAME	AMOUNT	ADATE
▶	D100	ANIL	VRCE	1000	1-MAR-95
	D101	SUNIL	ANJNI	500	4-JAN-96
	D102	MEHUL	KAROLBAGH	3500	17-NOV-95
	D104	MADHURI	CHANDNI	1200	17-DEC-95
	D105	PRAMOD	MG ROAD	3000	27-MAR-96
	D106	SANDIP	ANDHERI	2000	31-MAR-96
	D107	SHIVANI	VIRAR	1000	5-SEP-95
	D108	KRANTI	NEHRU PLACE	5000	2-JUL-95
	D109	MINU	POWAI	7000	10-AUG-95
*	NULL	NULL	NULL	NULL	NULL

2. SELECT \* FROM myaccount.borrow;

	LOANNO	CNAME	BNAME	AMOUNT
▶	L201	ANIL	VRCE	1000
	L206	MEHUL	AJNI	5000
	L311	SUNIL	DHARAMPETH	3000
	L321	MADHURI	ANDHERI	2000
	L371	PRAMOD	VIRAR	8000
	L481	KRANTI	NEHRU PLACE	3000
*	NULL	NULL	NULL	NULL

3. SELECT \* FROM myaccount.customer;

	CNAME	CITY
▶	ANIL	CALCUTTA
	KRANTI	BOMBAY
	MADHURI	NAGPUR
	MANDAR	PATNA
	MEHUL	BARODA
	NAREN	BOMBAY
	PRAMOD	NAGPUR
	SANDIP	SURAT
	SHIVANI	BOMBAY
	SUNIL	DELHI
*	NULL	NULL

4. SELECT \* FROM myaccount.branch;

	BNAME	CITY
▶	AJNI	NAGPUR
	ANDHERI	BOMBAY
	CHANDNI	DELHI
	DHARAMPETH	NAGPUR
	KAROLBAGH	DELHI
	MG ROAD	BANGALORE
	NEHRU PALACE	DELHI
	POWAI	BOMBAY
	VRCE	NAGPUR
*	NULL	NULL

5. select ACTNO,AMOUNT from myaccount.deposit;

	ACTNO	AMOUNT
▶	D100	1000
	D101	500
	D102	3500
	D104	1200
	D105	3000
	D106	2000
	D107	1000
	D108	5000
	D109	7000
*	NULL	NULL

6. select CNAME ,ACTNO from myaccount.deposit;

	CNAME	ACTNO
▶	ANIL	D100
	SUNIL	D101
	MEHUL	D102
	MADHURI	D104
	PRAMOD	D105
	SANDIP	D106
	SHIVANI	D107
	KRANTI	D108
	MINU	D109
*	NULL	NULL

7. select CNAME from myaccount.customer;

CNAME
ANIL
KRANTI
MADHURI
MANDAR
MEHUL
NAREN
PRAMOD
SANDIP
SHIVANI
SUNIL
NULL

8. select BNAME from myaccount.branch;

BNAME
AJNI
ANDHERI
CHANDNI
DHARAMPETH
KAROLBAGH
MG ROAD
NEHRU PALACE
POWAI
VRCE
NULL

9. select BNAME from myaccount.borrow;

BNAME
VRCE
AJNI
DHARAMPETH
ANDHERI
VIRAR
NEHRU PLACE

10. select CNAME from myaccount.customer where city="Nagpur";

CNAME
MADHURI
PRAMOD
NULL

11. SELECT CNAME from myaccount.deposit where AMOUNT>4000;

CNAME
KRANTI
MINU

12. SELECT ADATE FROM myaccount.deposit where CNAME='ANIL';

ADATE
1-MAR-95

13. SELECT BNAME from myaccount.branch where CITY='Bombay';

BNAME
ANDHERI
POWAI
NULL

14. select BNAME from myaccount.borrow where LOANNO='L1205';

BNAME

15. SELECT CNAME from myaccount.deposit WHERE BNAME='VRCE';

CNAME
ANIL

16. SELECT BNAME from myaccount.branch WHERE CITY='Delhi';

BNAME
CHANDNI
KAROLBAGH
NEHRU PALACE
NULL

17. SELECT CNAME from myaccount.deposit WHERE ADATE='1-Mar-96';

CNAME

18. SELECT ACTNO,AMOUNT from myaccount.deposit WHERE ADATE BETWEEN '1-Dec-96' AND '1-May-96';

ACTNO	AMOUNT
D 100	1000
NULL	

19. SELECT CITY from myaccount.branch WHERE BNAME='KAROLBAGH';

CITY
DELHI

20. SELECT \* from myaccount.customer join borrow on customer.cname=borrow.cname join deposit on deposit.cname=borrow.cname WHERE customer.cname='ANIL';

CNAME	CITY	LOANNO	CNAME	BNAME	AMOUNT	ACTNO	CNAME	BNAME	AMOUNT	ADATE
ANIL	CALCUTTA	L201	ANIL	VRCE	1000	D 100	ANIL	VRCE	1000	1-MAR-95

**Program no:3****Aim:** To familiarize with set operations**Question****Name:** salini kb**Roll No:**33**Batch:** MCA-B**Date:** 19-04-2022

1. List all the customers who are depositors but not borrowers.
2. List all the customers who are both depositors and borrowers
3. List all the depositors having deposit in all the branches where Sunil is having Account
4. List all the customers living in city NAGPUR and having branch city BOMBAY or DELHI
5. List all the depositors living in city NAGPUR
6. List all the depositors living in the city NAGPUR and having branch in city BOMBAY
7. List the branch cities of Anil and Sunil
8. List the customers having deposit greater than 1000 and loan less than 10000.
9. List the cities of depositors having branch VRCE.
10. List the depositors having amount less than 1000 and living in the same city as Anil
11. List all the cities where branches of Anil and Sunil are located
12. List the amount for the depositors living in the city where Anil is living

**PROCEDURE**

1. SELECT CNAME FROM DEPOSIT  
WHERE NOT EXISTS (SELECT CNAME FROM BORROW);

CNAME
-------

2. SELECT CNAME FROM DEPOSIT UNION (SELECT CNAME FROM BORROW);

CNAME
ANIL
SUNIL
MEHUL
MADHURI
PRAMOD
SANDIP
SHIVANI
KRANTI
MINU

3. SELECT D1.CNAME FROM DEPOSIT D1 WHERE D1.BNAME IN (SELECT D2.BNAME FROM DEPOSIT D2 WHERE D2.CNAME = 'SUNIL');

CNAME
SUNIL

4. SELECT C1.CNAME FROM CUSTOMER C1,DEPOSIT D1, BRANCH B1 WHERE C1.CITY = 'NAGPUR' AND C1.CNAME = D1.CNAME AND D1.BNAME = B1.BNAME AND B1.CITY IN ('BOMBAY','DELHI');

CNAME
MADHURI

5. SELECT Distinct(CUSTOMER.CNAME) from CUSTOMER,DEPOSIT WHERE City='NAGPUR';

CNAME
PRAMOD
MADHURI

6. SELECT C1.CNAME FROM CUSTOMER C1,DEPOSIT D1, BRANCH B1 WHERE C1.CITY = 'NAGPUR' AND C1.CNAME = D1.CNAME AND D1.BNAME = B1.BNAME AND B1.CITY IN ('BOMBAY');

CNAME
-------

7. \_SELECT B1.CITY FROM DEPOSIT D1, BRANCH B1 WHERE D1.BNAME = B1.BNAME AND D1.CNAME IN ('SUNIL' ,'ANIL');

CITY
NAGPUR

8. \_SELECT DISTINCT D1.CNAME FROM deposit D1, borrow B1 WHERE D1.AMOUNT>1000 AND B1.AMOUNT<10000;

CNAME
MEHUL
MADHURI
PRAMOD
SANDIP
KRANTI
MINU

9. SELECT B1.CITY FROM deposit D1, branch B1 WHERE D1.BNAME=B1.BNAME AND B1.BNAME='VRCE';

CITY
NAGPUR

10. \_ SELECT D1.CNAME FROM deposit D1, customer C1 WHERE AMOUNT<1000 AND C1.CITY=(C1.CNAME='ANIL');

CNAME
SUNIL
SUNIL
SUNIL
SUNIL
SUNIL
SUNIL
SUNIL
SUNIL
SUNIL

11. SELECT B1.CITY FROM BRANCH B1 WHERE B1.BNAME IN (SELECT D1.BNAME FROM DEPOSIT D1 WHERE D1.CNAME IN ('ANIL','SUNIL'));

	CITY
▶	NAGPUR

12. SELECT DISTINCT(D1.CNAME),D1.AMOUNT ,C1.CITY FROM deposit D1, CUSTOMER C1, BRANCH B1 WHERE D1.CNAME=C1.CNAME AND C1.CITY IN(SELECT C2.CITY FROM customer C2 WHERE C2.CNAME='ANIL');

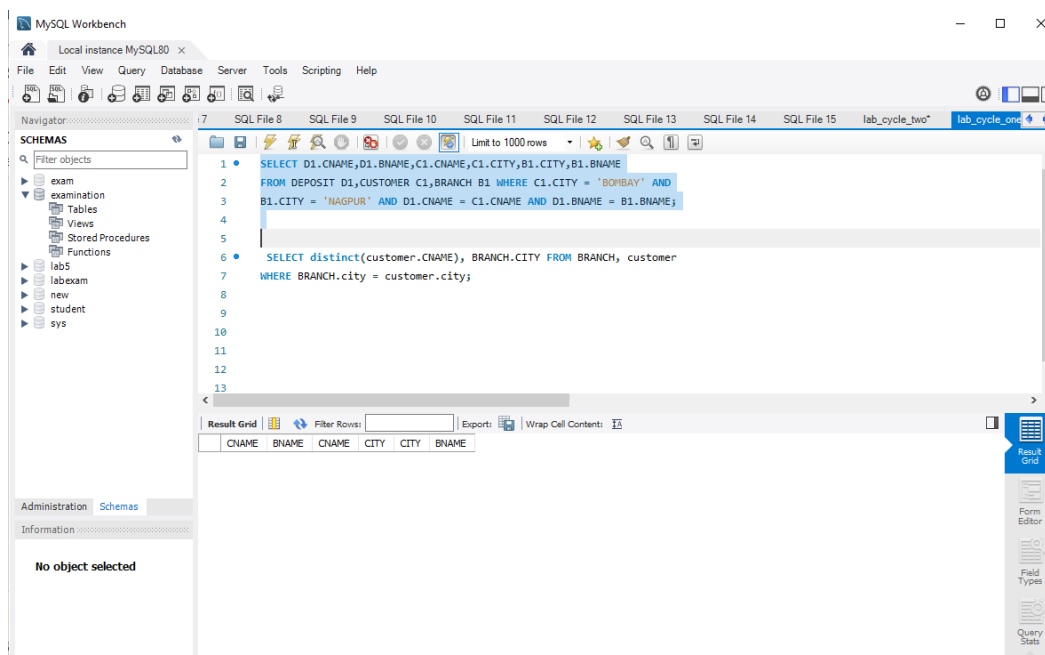
	CNAME	AMOUNT	CITY
▶	ANIL	1000	CALCUTTA

**Program no:4****Aim:** To familiarize with join or cartesian product

1. Give name of customers having living city BOMBAY and branch city NAGPUR

**Name:** salini kb**Roll No:**33**Batch:** MCA-B**Date:** 06-05-2022

```
SELECT D1.CNAME,D1.BNAME,C1.CNAME,C1.CITY,B1.CITY,B1.BNAME  
FROM DEPOSIT D1,CUSTOMER C1,BRANCH B1 WHERE C1.CITY = 'BOMBAY'  
AND  
B1.CITY = 'NAGPUR' AND D1.CNAME = C1.CNAME AND D1.BNAME =  
B1.BNAME;
```

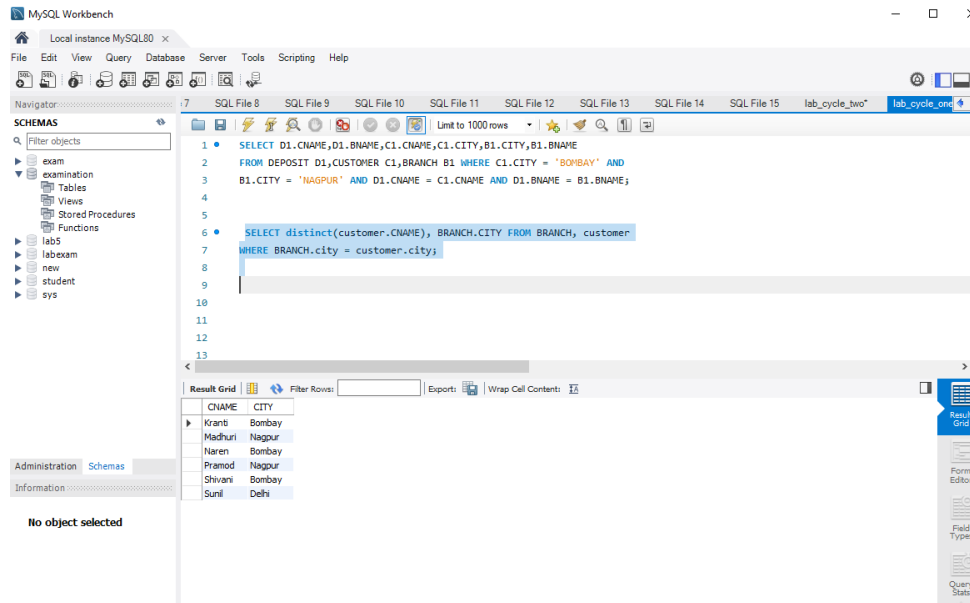
**OUTPUT**

2. Give names of customers having the same living city as their branch city



```
SELECT distinct(customer.CNAME), BRANCH.CITY FROM BRANCH, customer  
WHERE BRANCH.city = customer.city;
```

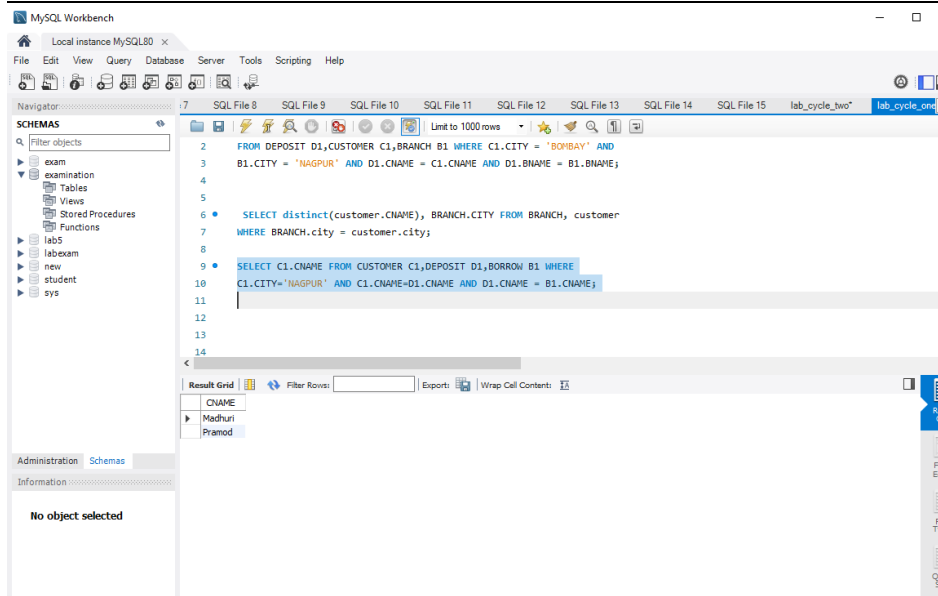
### OUTPUT



3. Give names of customers who are borrowers as well as depositors and having city

```
SELECT C1.CNAME FROM CUSTOMER C1,DEPOSIT D1,BORROW B1 WHERE  
C1.CITY='NAGPUR' AND C1.CNAME=D1.CNAME AND D1.CNAME = B1.CNAME;
```

### OUTPUT

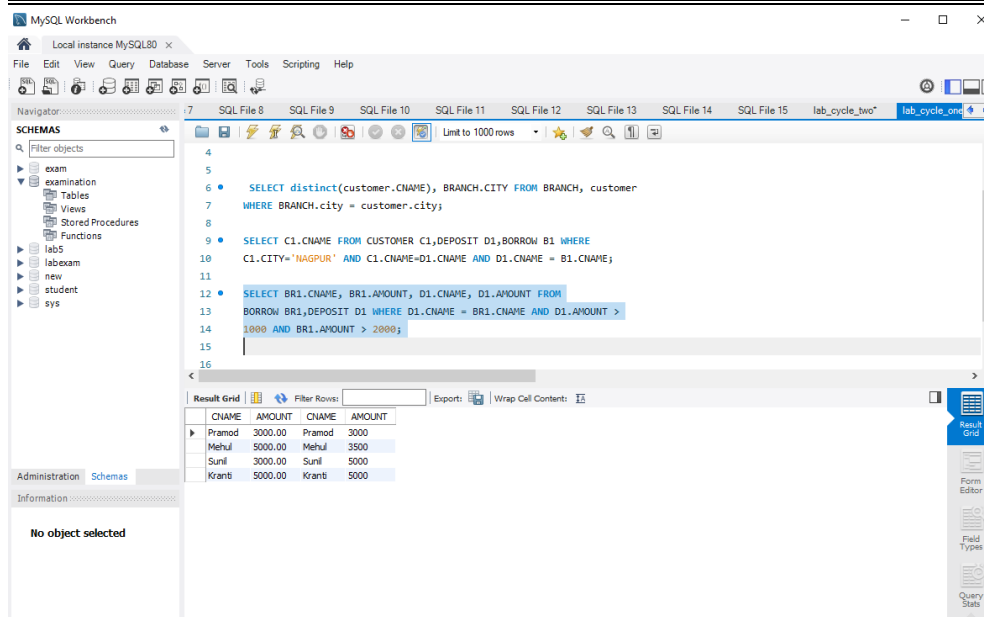


4. Give names of borrowers having deposit amount greater than 1000 and loan amount greater

than 2000.

```
SELECT BR1.CNAME, BR1.AMOUNT, D1.CNAME, D1.AMOUNT FROM
BORROW BR1,DEPOSIT D1 WHERE D1.CNAME = BR1.CNAME AND D1.AMOUNT
>
1000 AND BR1.AMOUNT > 2000;
```

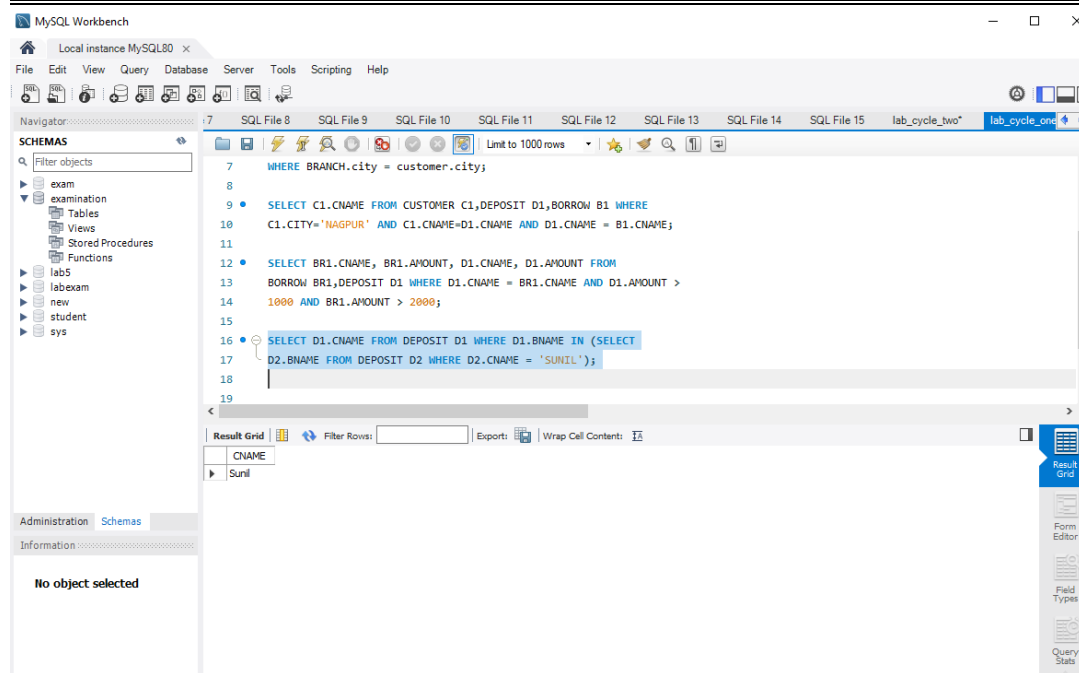
OUTPUT



5. Give names of depositors having the same branch as the branch of Sunil

SELECT D1.CNAME FROM DEPOSIT D1 WHERE D1.BNAME IN (SELECT  
D2.BNAME FROM DEPOSIT D2 WHERE D2.CNAME = 'SUNIL');

OUTPUT



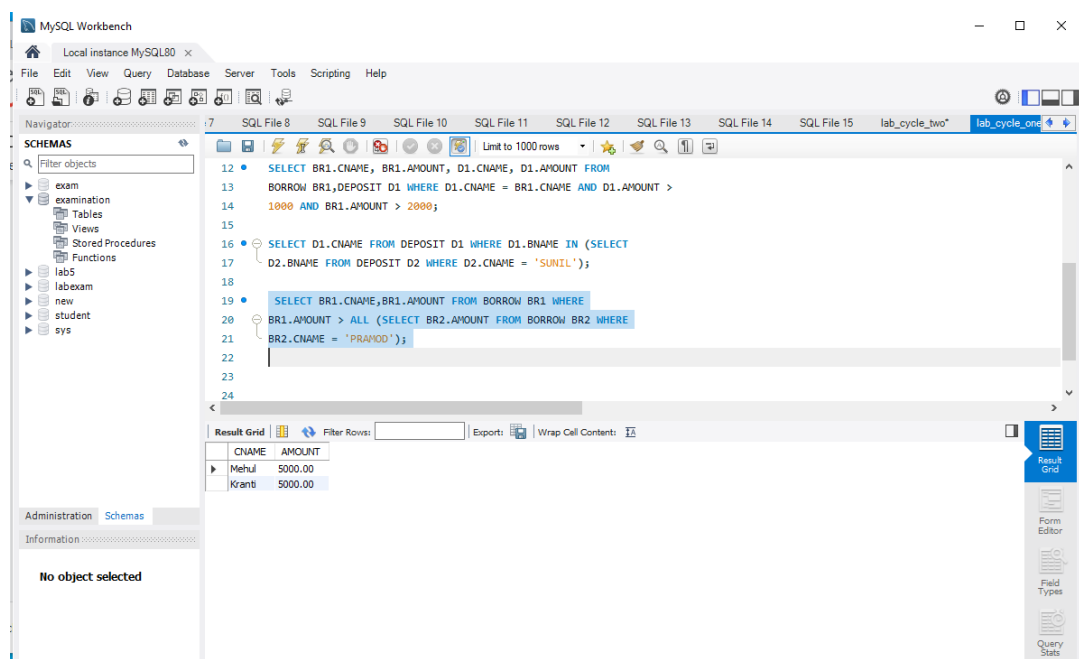
6. Give names of borrowers having loan amount greater than the loan amount of Pramod

```

SELECT BR1.CNAME,BR1.AMOUNT FROM BORROW BR1 WHERE
BR1.AMOUNT > ALL (SELECT BR2.AMOUNT FROM BORROW BR2 WHERE
BR2.CNAME = 'PRAMOD');

```

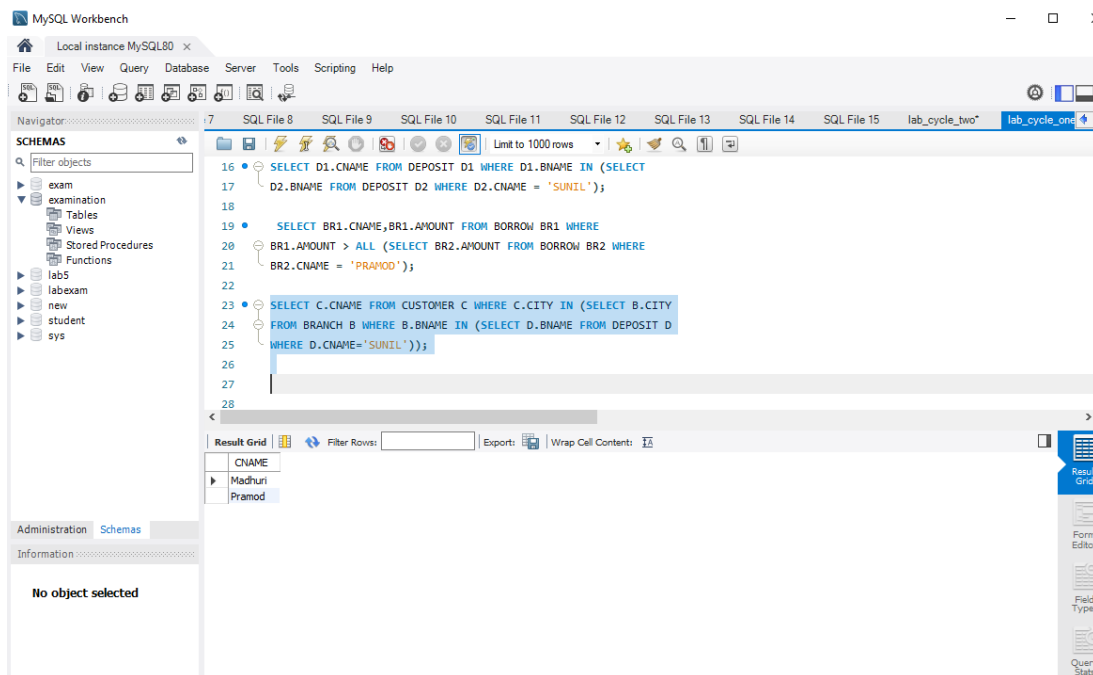
### OUTPUT



7. Give the name of the customer living in the city where branch of depositor Sunil is located.

```
SELECT C.CNAME FROM CUSTOMER C WHERE C.CITY IN (SELECT B.CITY
FROM BRANCH B WHERE B.BNAME IN (SELECT D.BNAME FROM DEPOSIT D
WHERE D.CNAME='SUNIL'));
```

### OUTPUT

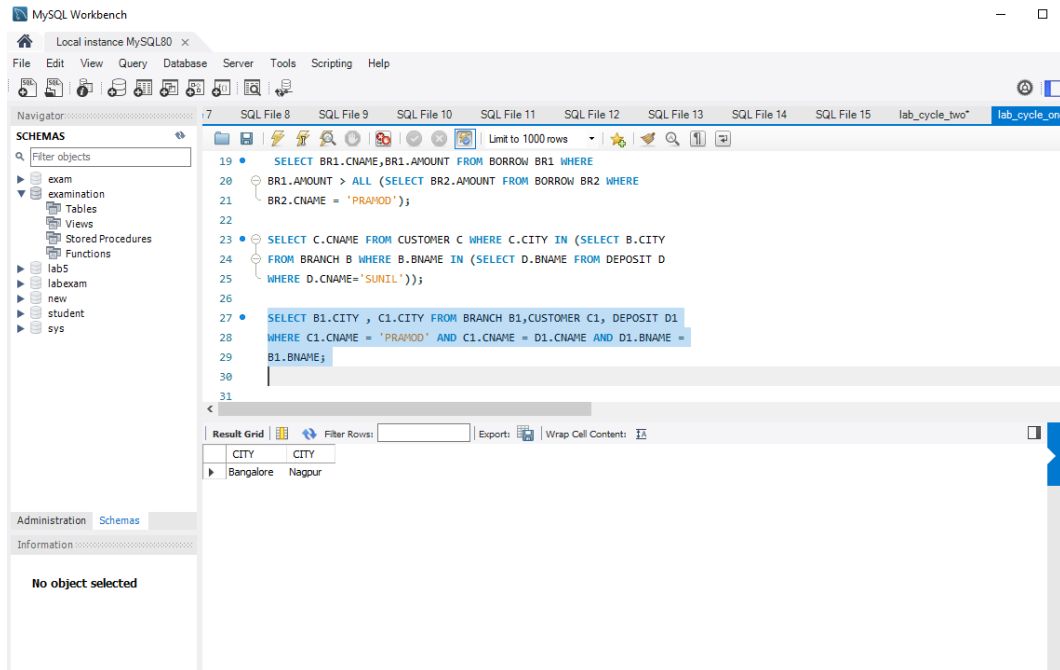


8. Give branch city and living city of Pramod

```
SELECT B1.CITY , C1.CITY FROM BRANCH B1,CUSTOMER C1, DEPOSIT D1
WHERE C1.CNAME = 'PRAMOD' AND C1.CNAME = D1.CNAME AND D1.BNAME =
```

B1.BNAME;

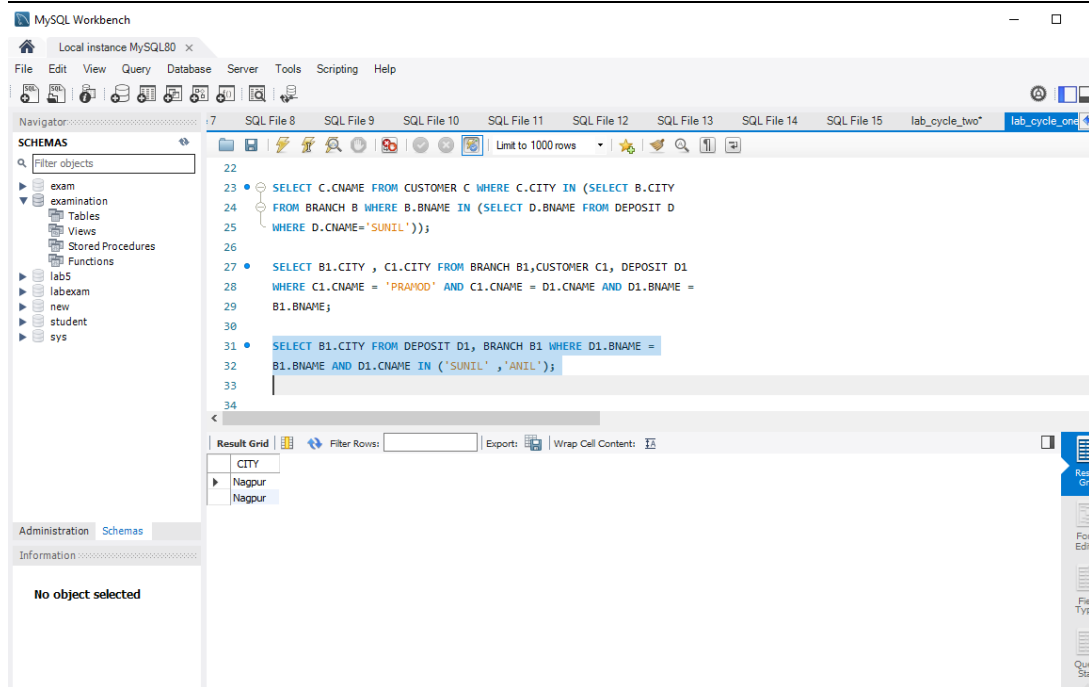
## OUTPUT



9. Give branch city of Sunil and branch city of Anil

SELECT B1.CITY FROM DEPOSIT D1, BRANCH B1 WHERE D1.BNAME =  
B1.BNAME AND D1.CNAME IN ('SUNIL', 'ANIL');

## OUTPUT



10. Give the living city of Anil and the living city of Sunil

```
SELECT C1.CNAME, C1.CITY FROM CUSTOMER C1 WHERE C1.CNAME =  
'ANIL' OR C1.CNAME = 'SUNIL';
```

### OUTPUT

The screenshot displays the MySQL Workbench interface. The left sidebar shows the 'SCHEMAS' tree with a search filter. The main editor window contains three SQL queries. The bottom query is selected and highlighted. Below the editor, the 'Result Grid' tab is active, showing a table with two columns: 'CNAME' and 'CITY'. The table contains two rows of data.

```
25 WHERE D.CNAME='SUNIL');
26
27 • SELECT B1.CITY , C1.CITY FROM BRANCH B1,CUSTOMER C1, DEPOSIT D1
28 WHERE C1.CNAME = 'PRAMOD' AND C1.CNAME = D1.CNAME AND D1.BNAME =
29 B1.BNAME;
30
31 • SELECT B1.CITY FROM DEPOSIT D1, BRANCH B1 WHERE D1.BNAME =
32 B1.BNAME AND D1.CNAME IN ('SUNIL' , 'ANIL');
33
34 • SELECT C1.CNAME, C1.CITY FROM CUSTOMER C1 WHERE C1.CNAME =
35 'ANIL' OR C1.CNAME = 'SUNIL';
36
37
```

CNAME	CITY
Anil	Calcutta
Sunil	Delhi



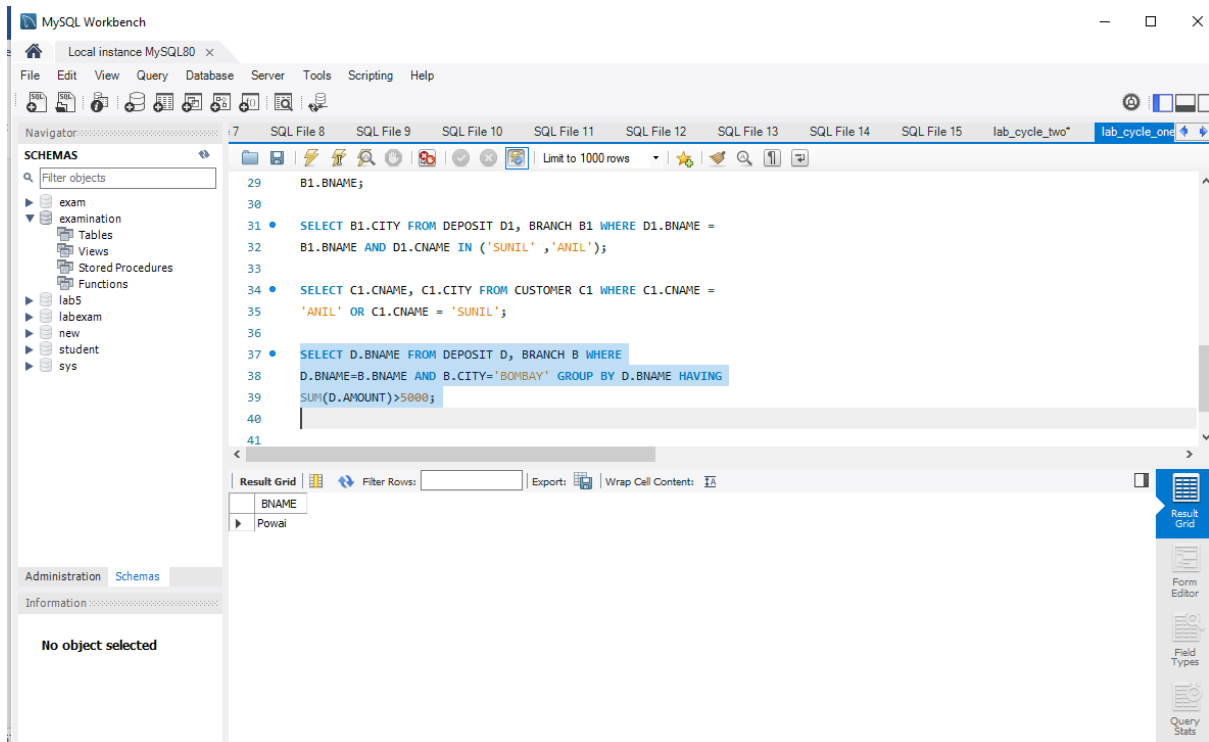
**Program no:5****Aim:** To familiarize with Group by and Having clause**Name:** salini kb**Roll No:**33**Batch:** MCA-B**Date:** 06-05-2022

1.List the branches having sum of deposit more than 5000.

```
SELECT D.BNAME FROM DEPOSIT D, BRANCH B  
WHERE
```

```
D.BNAME=B.BNAME AND B.CITY='BOMBAY' GROUP BY D.BNAME  
HAVING
```

```
SUM(D.AMOUNT)>5000;
```

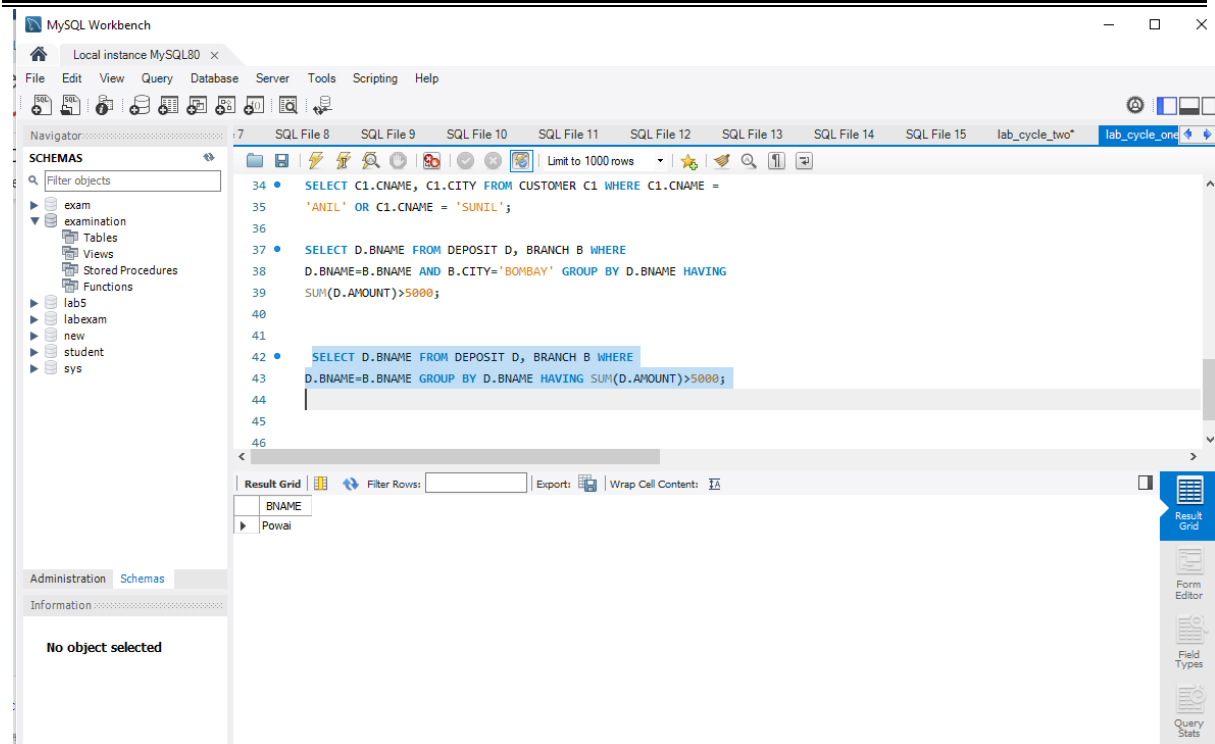
**OUTPUT**

2.List the branches having sum of deposit more than 5000 and located in city BOMBAY

```
SELECT D.BNAME FROM DEPOSIT D, BRANCH B WHERE
```

```
D.BNAME=B.BNAME GROUP BY D.BNAME HAVING  
SUM(D.AMOUNT)>5000;
```

**OUTPUT**



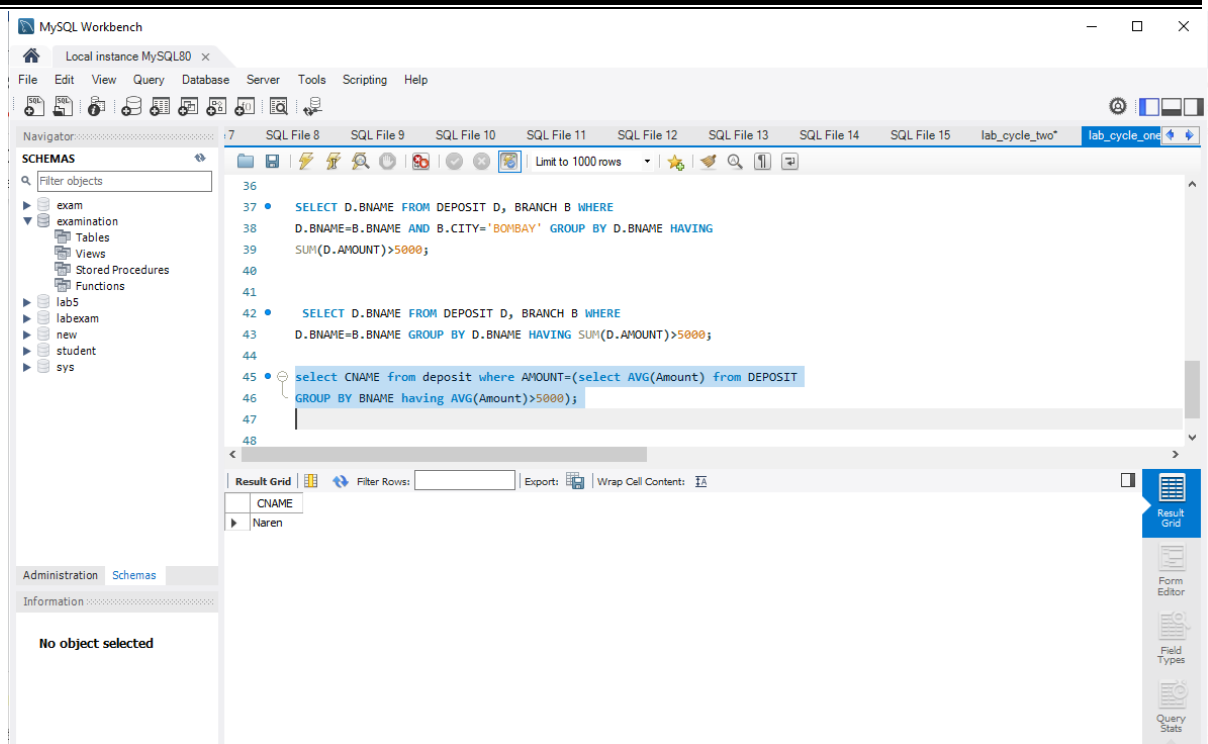
3. List the names of customers having deposited in the branches where the average deposit is

more than 5000.

select CNAME from deposit where AMOUNT=(select AVG(Amount) from DEPOSIT

GROUP BY BNAME having AVG(Amount)>5000);

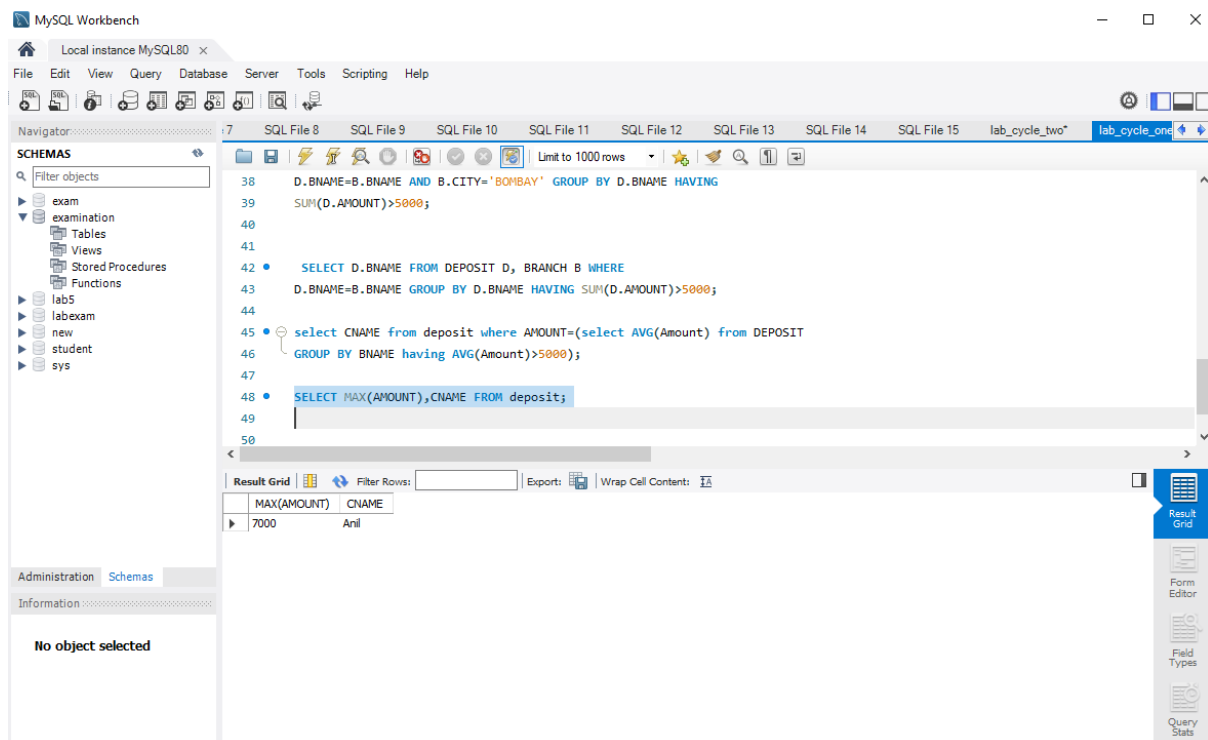
OUTPUT



4. List the names of customers having maximum deposit

SELECT MAX(AMOUNT),CNAME FROM deposit;

OUTPUT

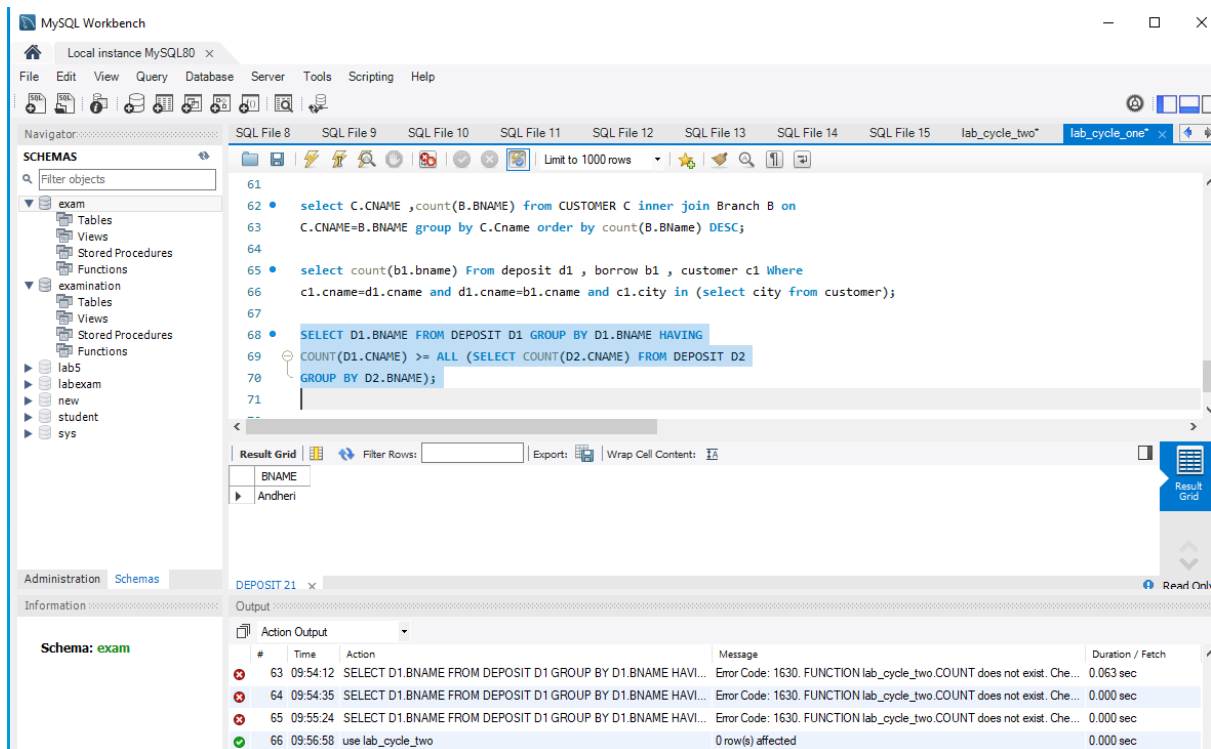


5. List the name of branch having highest number of depositors?

```

SELECT D1.BNAME FROM DEPOSIT D1 GROUP BY D1.BNAME HAVING
COUNT(D1.CNAME) >= ALL (SELECT COUNT(D2.CNAME) FROM DEPOSIT
D2
GROUP BY D2.BNAME);

```



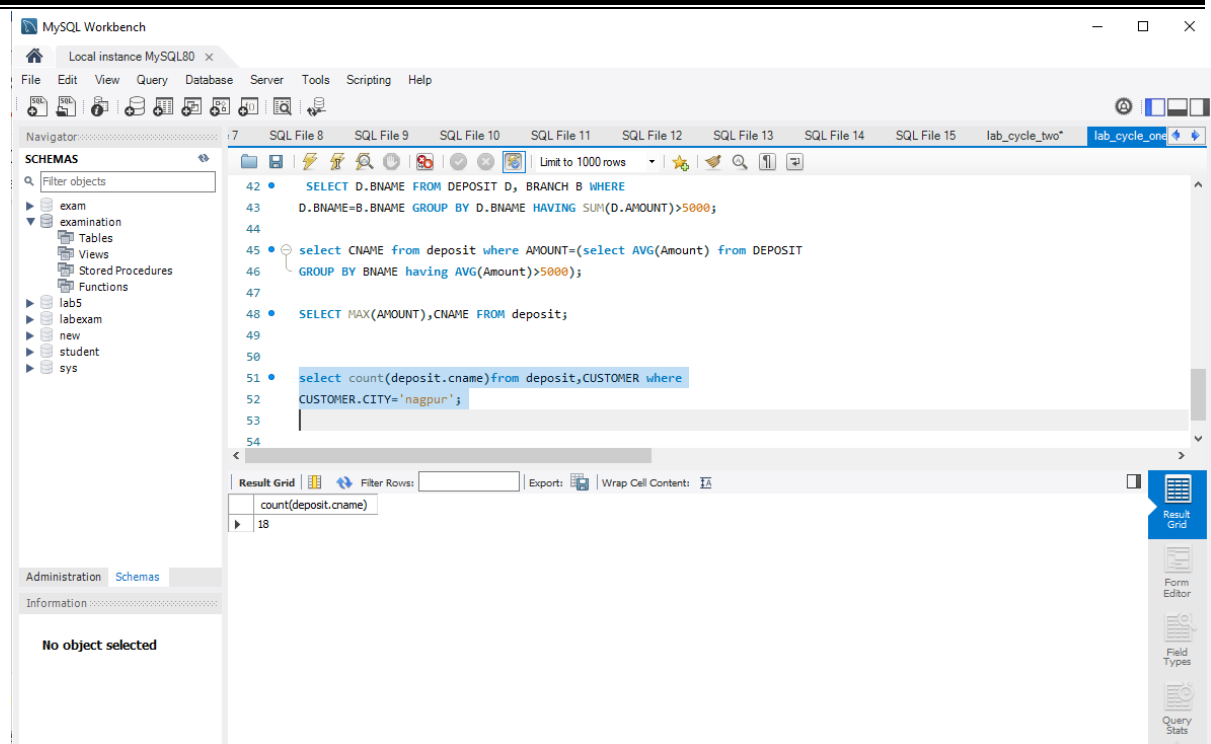
6.Count the number of depositors living in NAGPUR.

```

select count(deposit.cname)from deposit,CUSTOMER where
CUSTOMER.CITY='nagpur';

```

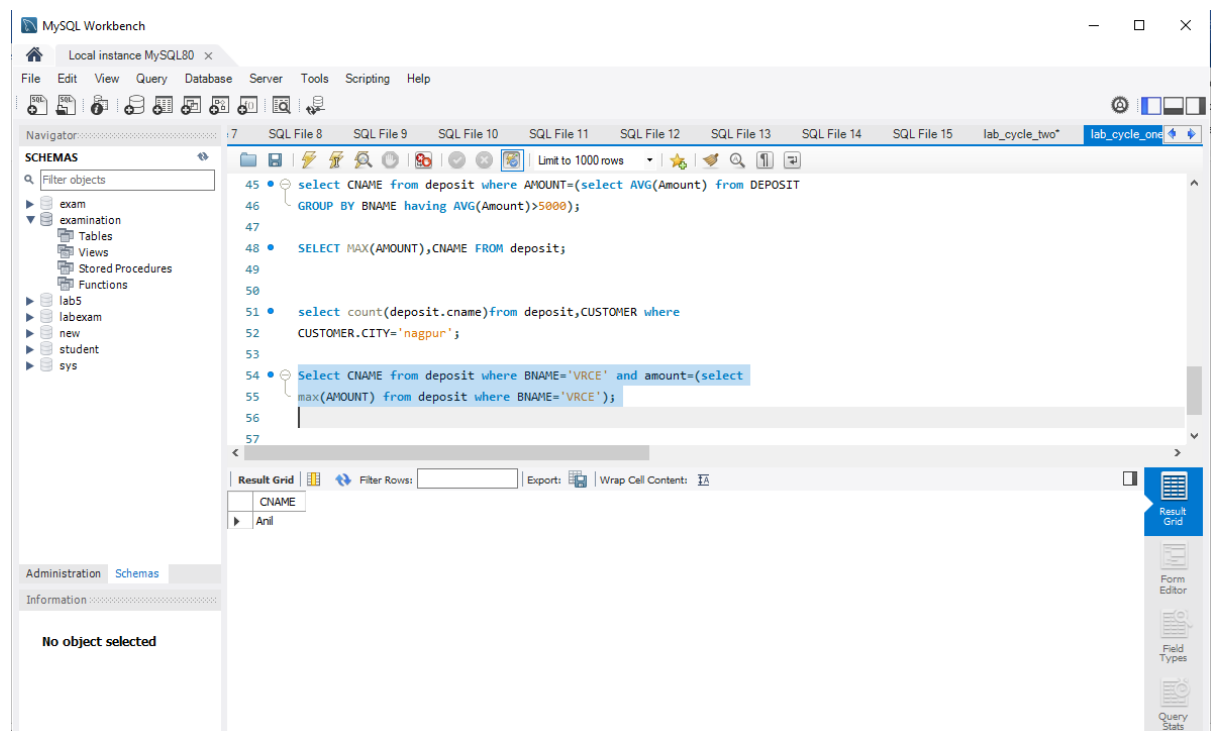
OUTPUT



7. Give names of customers in VRCE branch having more deposits than any other customer in same branch

Select CNAME from deposit where BNAME='VRCE' and amount=(select max(AMOUNT) from deposit where BNAME='VRCE');

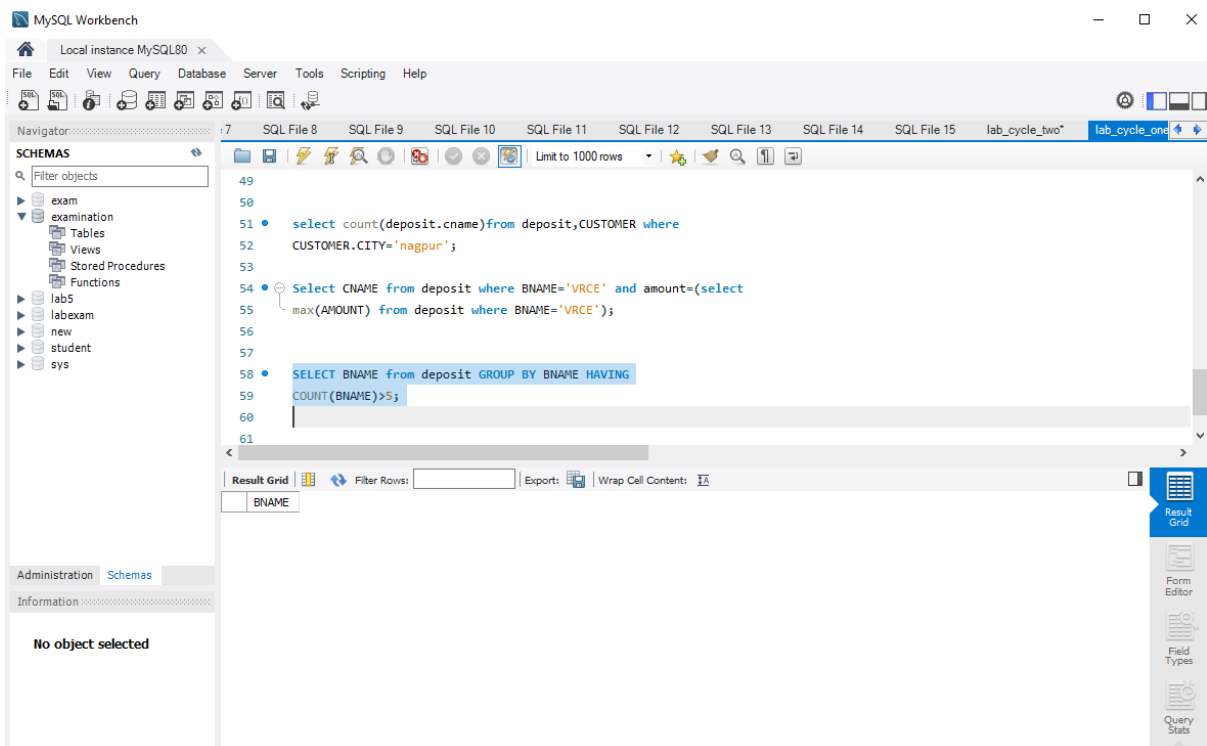
## OUTPUT



8. Give the names of branch where number of depositors is more than 5

SELECT BNAME from deposit GROUP BY BNAME HAVING  
COUNT(BNAME)>5;

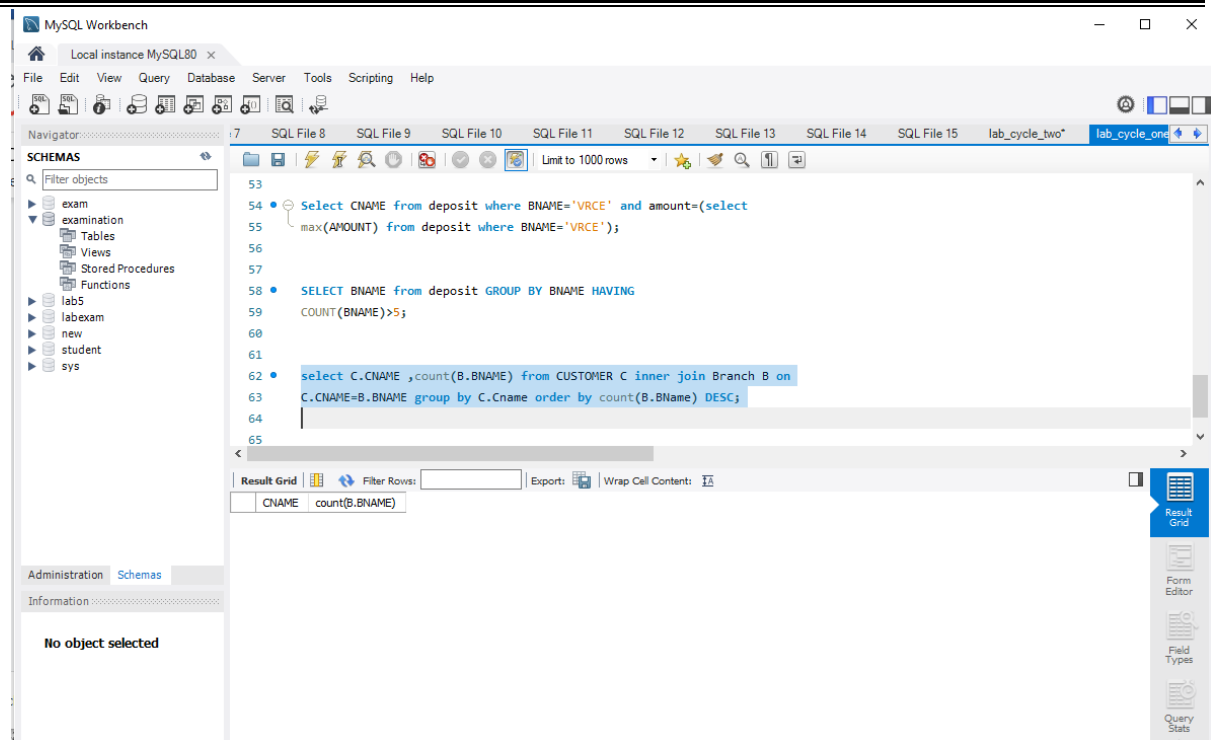
### OUTPUT



9. Give the names of cities in which the maximum number of branches are located

select C.CNAME ,count(B.BNAME) from CUSTOMER C inner join Branch B on  
C.CNAME=B.BNAME group by C.Cname order by count(B.BName) DESC;

### OUTPUT

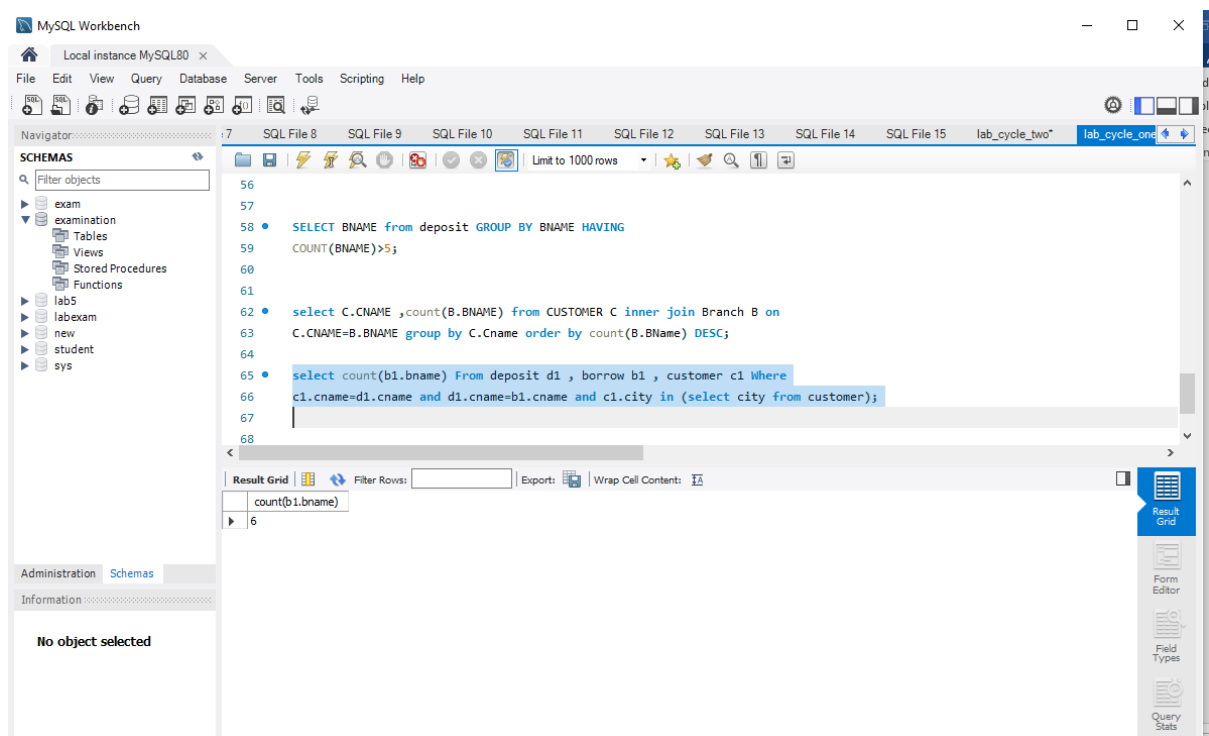


10.Count the number of customers living in the city where branch is located

select count(b1.bname) From deposit d1 , borrow b1 , customer c1 Where

c1.cname=d1.cname and d1.cname=b1.cname and c1.city in (select city from customer);

### OUTPUT



---

**Program no:6**

**Aim:** Implementation of triggers

Create a student table with fields id,name,subject1,subject2,subject3 and total,percentage. For each entry of row, update total marks and percentage using triggers in SQL

**Name:** salini kb

**Roll No:**33

**Batch:** MCA-B

**Date:** 10-06-2022

**Procedure**

step 1: start

step 2: create table student

step 3: create trigger on student table

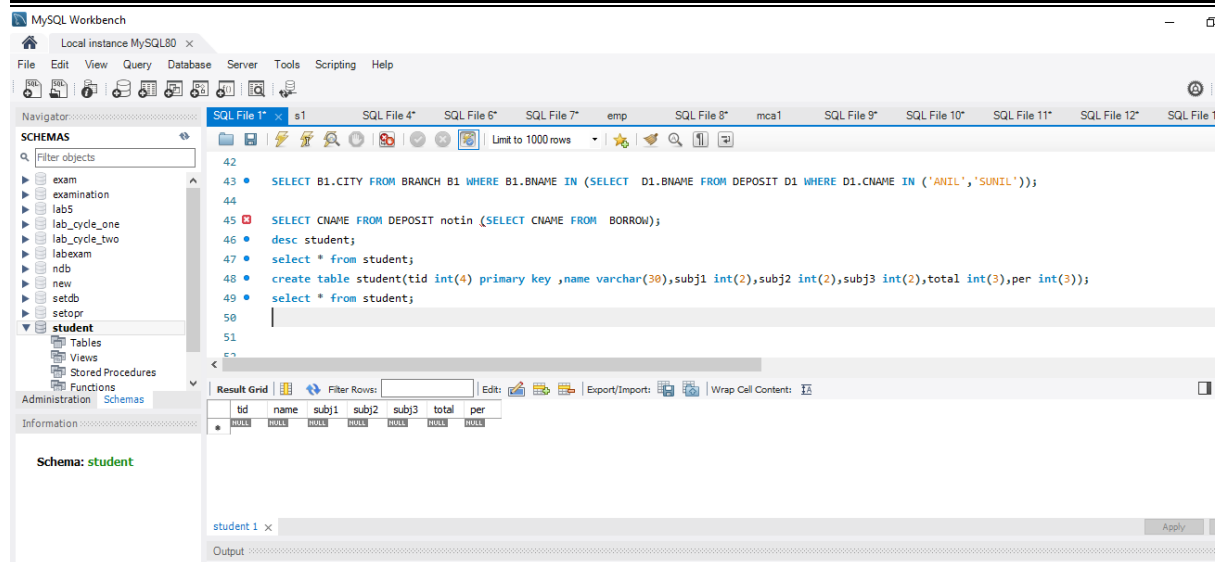
step 4: insert records

step 5: select the table to see if the triggers has been executed

step 6: stop

**Output Screenshot**



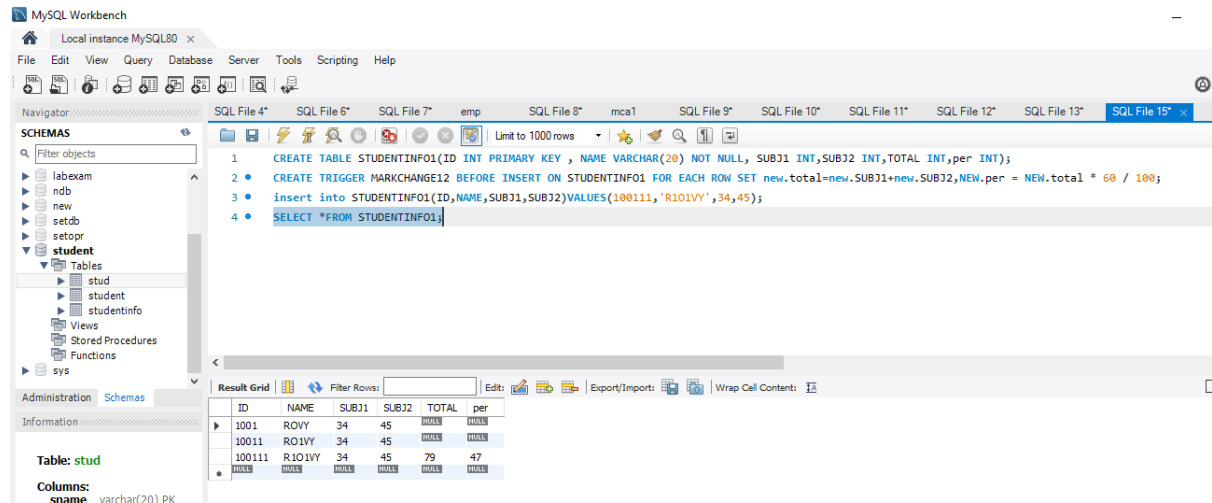


CREATE TABLE STUDENTINFO1(ID INT PRIMARY KEY , NAME VARCHAR(20) NOT NULL, SUBJ1 INT,SUBJ2 INT,TOTAL INT,per INT);

CREATE TRIGGER MARKCHANGE12 BEFORE INSERT ON STUDENTINFO1 FOR EACH ROW SET new.total=new.SUBJ1+new.SUBJ2,NEW.per = NEW.total \* 60 / 100;

insert into STUDENTINFO1(ID,NAME,SUBJ1,SUBJ2)VALUES(100111,'R101VY',34,45);

SELECT \*FROM STUDENTINFO1;



## Experiment No.: 2

### Aim

Create a Trigger for student table that will update another table shows the name, total marks and percentage .

### Procedure

CREATE TABLE MARKS(MARKID INT PRIMARY KEY AUTO\_INCREMENT,NAME VARCHAR(20),TOTAL\_MARKS INT);

CREATE TRIGGER MARK\_TRIGGER1 AFTER INSERT ON STUDENTINFO1 FOR EACH ROW

INSERT INTO MARKS(NAME,TOTAL\_MARKS) VALUES(new.NAME,new.TOTAL);

INSERT INTO STUDENTINFO1(ID,NAME,SUBJ1,SUBJ2)

values(1002,'SALU',35,42),(1003,'SAN',44,38);

SELECT \*FROM MARKS;

The screenshot displays the MySQL Workbench interface. The SQL Editor contains the following queries:

```

3 • insert into STUDENTINFO1(ID,NAME,SUBJ1,SUBJ2)VALUES(100111,'R101VY',34,45);
4 • SELECT *FROM STUDENTINFO1;
5
6 • CREATE TABLE MARKS(MARKID INT PRIMARY KEY AUTO_INCREMENT,NAME VARCHAR(20),TOTAL_MARKS INT);
7 • CREATE TRIGGER MARK_TRIGGER1 AFTER INSERT ON STUDENTINFO1 FOR EACH ROW
8   INSERT INTO MARKS(NAME,TOTAL_MARKS) VALUES(new.NAME,new.TOTAL);
9 • INSERT INTO STUDENTINFO1(ID,NAME,SUBJ1,SUBJ2)
10  values(1002,'SALU',35,42),(1003,'SAN',44,38);
11 • SELECT *FROM MARKS;
12

```

The Result Grid shows the output of the queries:

MARKID	NAME	TOTAL_MARKS
1	SALU	77
2	SAN	82

The Output pane shows the execution results of the queries:

#	Time	Action	Message	Duration / Fetch
73	11:50:45	CREATE TRIGGER MARK_TRIGGER1 AFTER INSERT ON STUDENTINFO1 FOR EACH ROW	0 row(s) affected	0.141 sec
74	11:50:50	INSERT INTO STUDENTINFO1(ID,NAME,SUBJ1,SUBJ2) values(1002,'SALU',35,42),(1003,'SAN',44,38);	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.203 sec
75	11:50:54	SELECT *FROM MARKS LIMIT 0. 1000	2 row(s) returned	0.000 sec / 0.000 sec

**Program no:7****Aim:** Installation of MongoDB on Windows

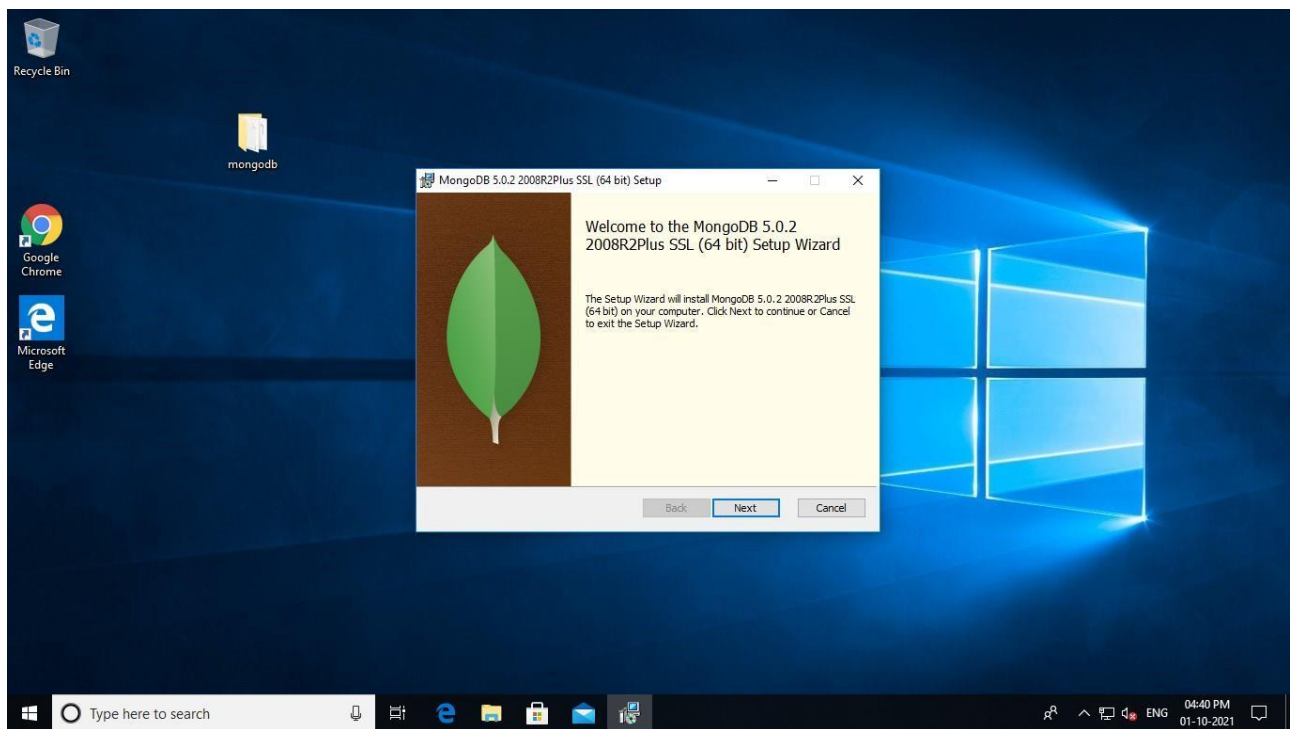
The installers for MongoDB are available in both the 32-bit and 64-bit format. The 32-bit installers are good for development and test environments. But for production environments you should use the 64-bit installers. Otherwise, you can be limited to the amount of data that can be stored within MongoDB.

**Name:** salini kb**Roll No:**33**Batch:** MCA-B**Date:** 24-05-2022**Download & Install MongoDB on Windows**

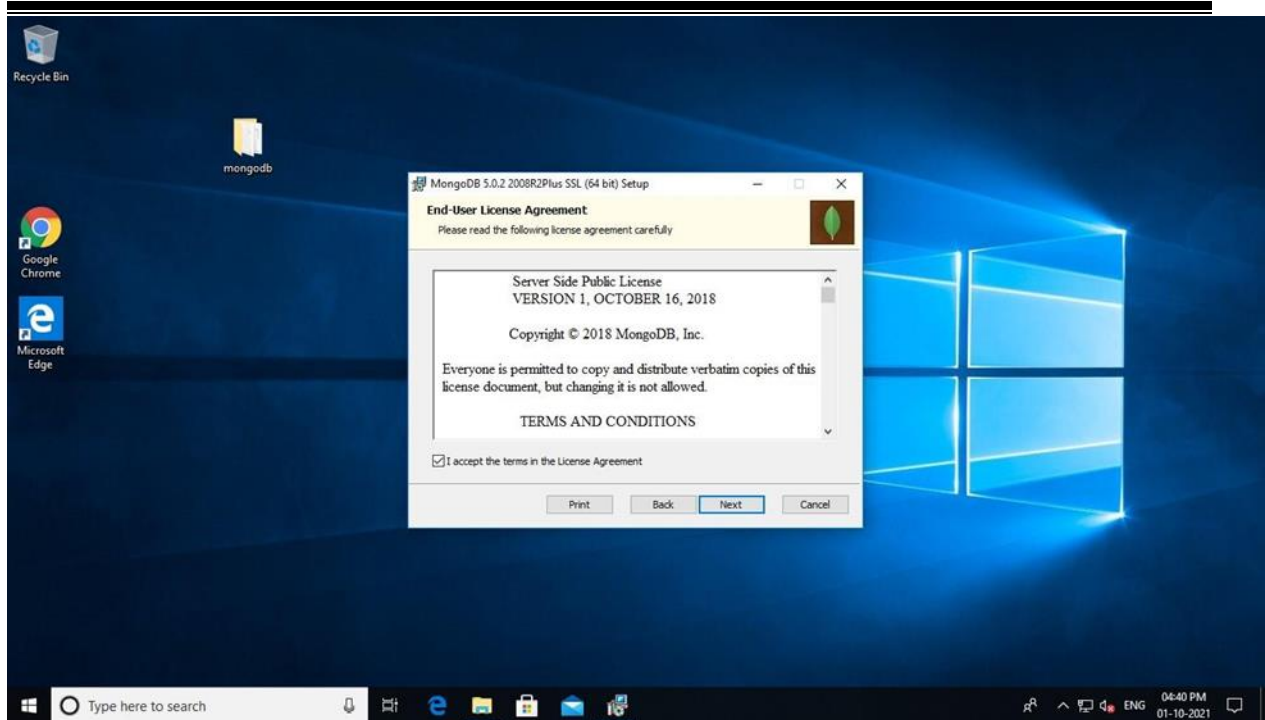
The following steps can be used to install MongoDB on Windows 10:

**Step 1:**

After downloading MongoDB, open the msi file and click next.

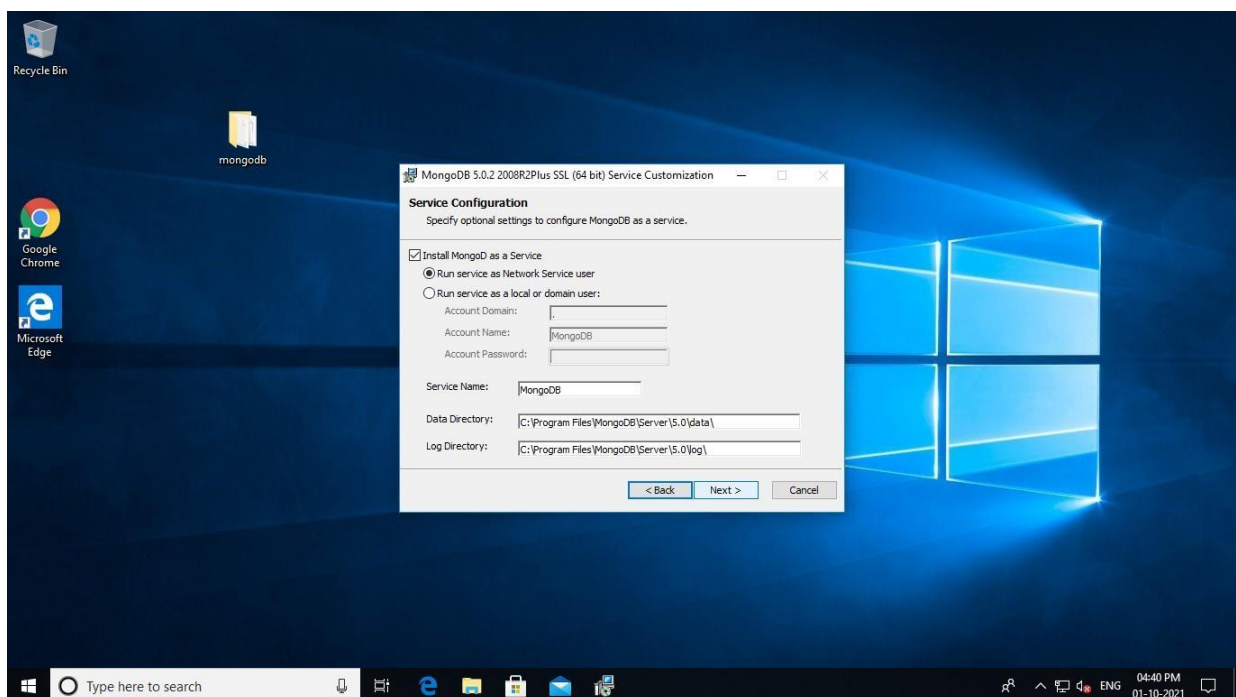
**Step 2:**

- a. Accept the End-User License Agreement
- b. Click Next

**Step 3:**

Click on the "complete" button to install all of the components. The custom option can be used to install selective components or if you want to change the location of the installation.

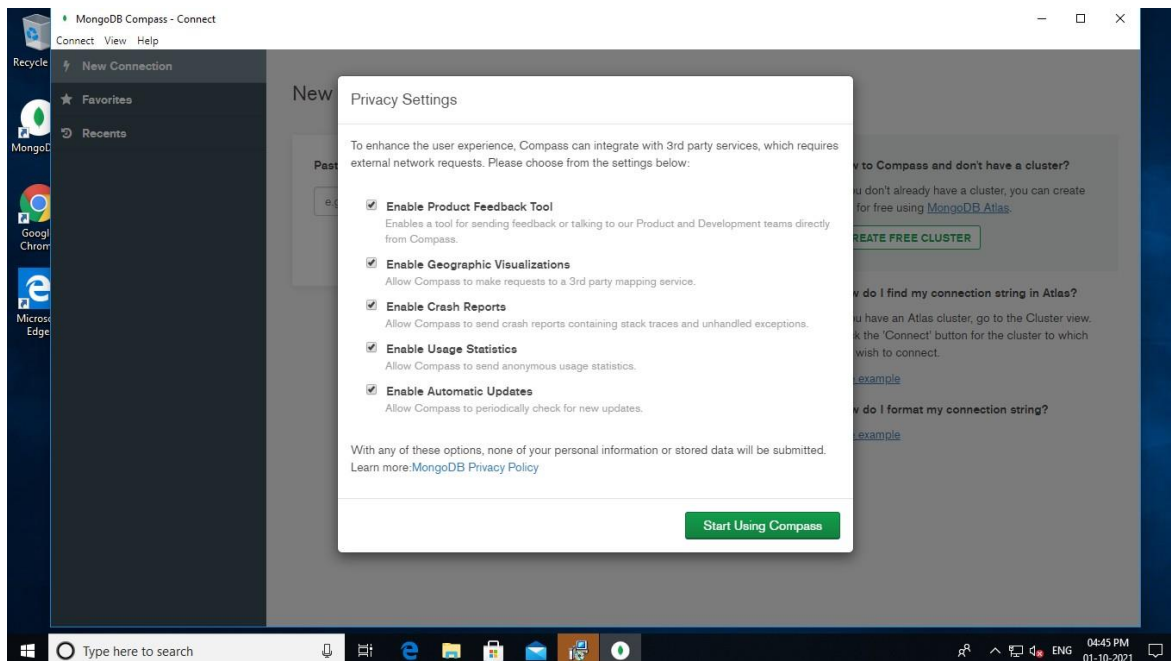
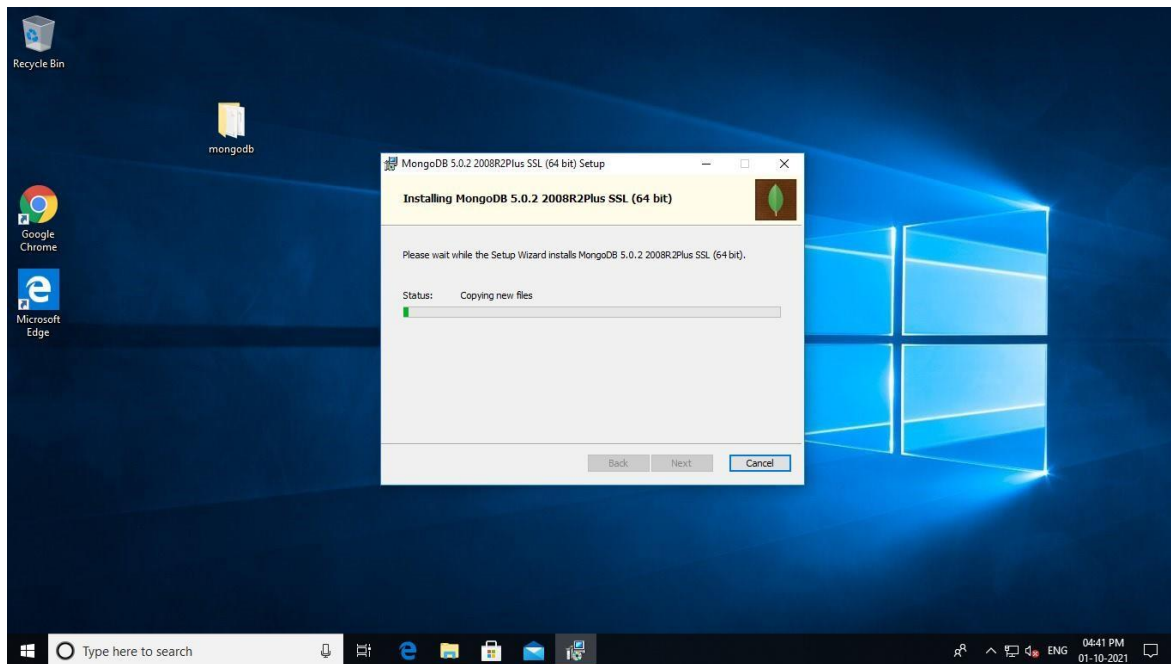
1. Select "Run service as Network Service user". Make a note of the data directory, we'll need this later.
2. Click Next

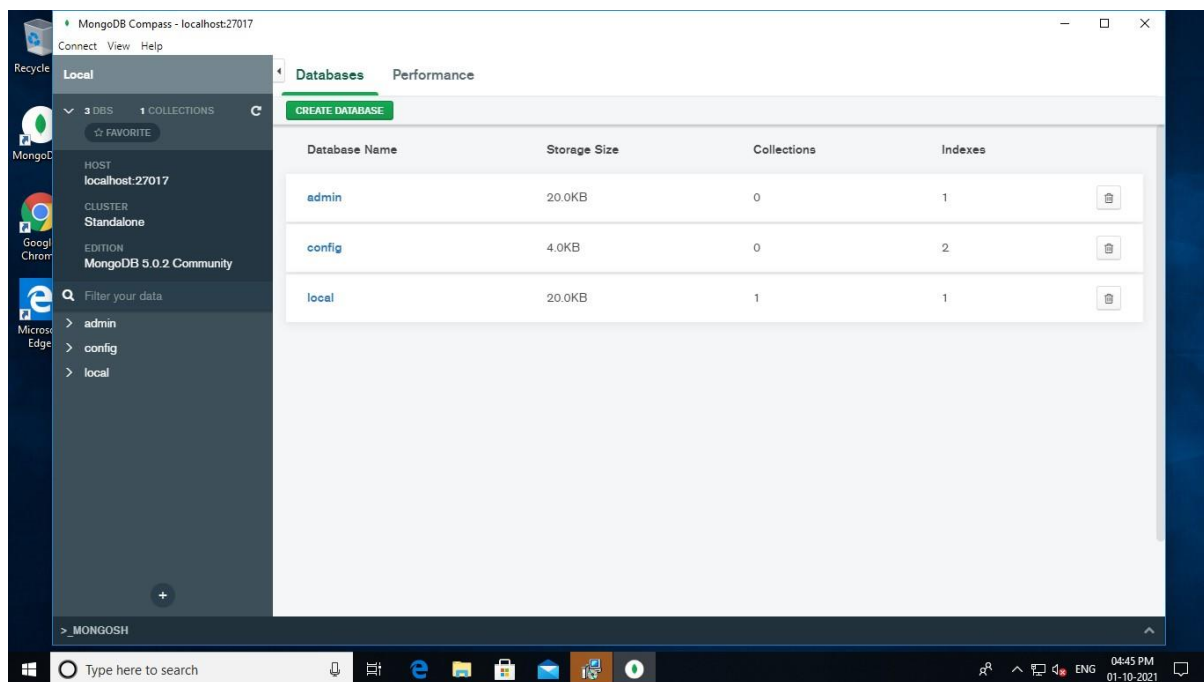
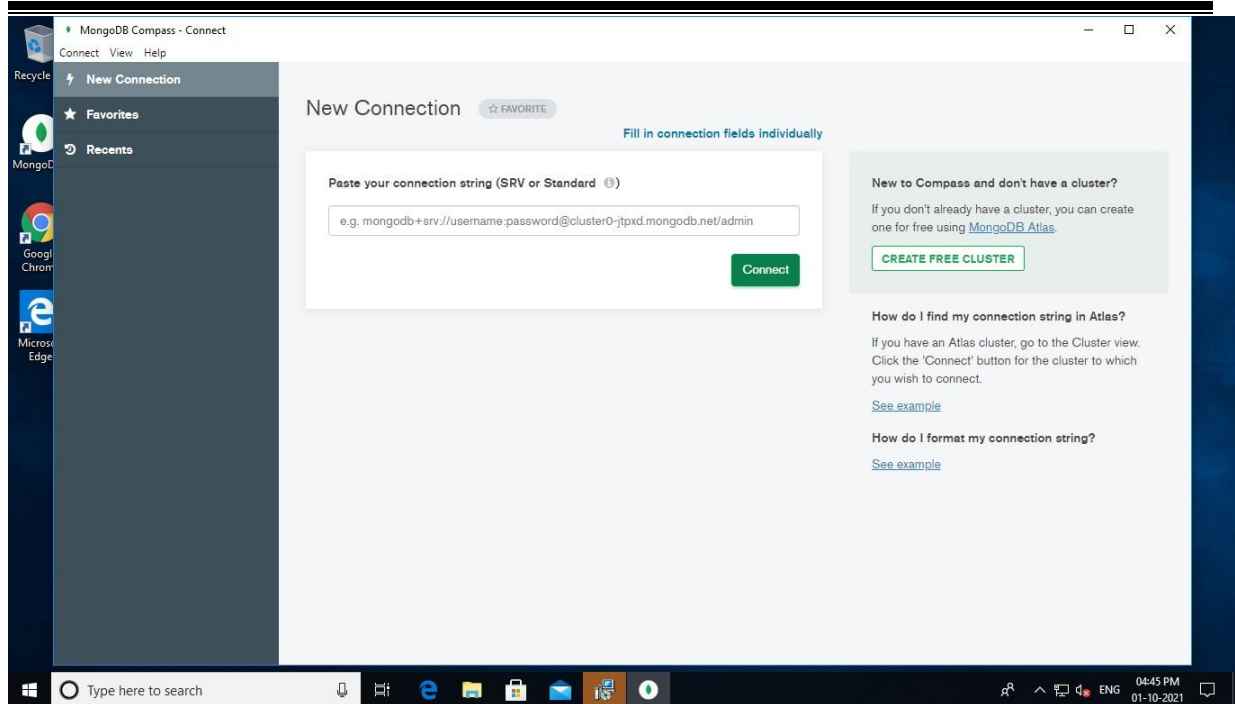
**Step 4:**

Click on the Install button to start the installation. Installation begins.

Click next once completed.

Click on the Finish button to complete the installation.







## Program no:8

### Aim: Designing Databases using NoSQL: MongoDB

Install PHP MongoDB Driver




### Procedure

Download correct mongodb driver for PHP, based on your PHP

version, Architecture and whether Thread Safety is enabled. You can get that information from the phpinfo page.

Do the following steps to install and configure MongoDB driver on Windows XAMPP Server.

1. Download the latest stable version of the PHP MongoDB driver from following URL  
<https://pecl.php.net/package/mongodb>.

Available Releases			
Version	State	Release Date	Downloads
<a href="#">1.3.2</a>	stable	2017-10-30	<a href="#">mongodb-1.3.2.tgz</a> (904.0kB) 
<a href="#">1.3.1</a>	stable	2017-10-16	<a href="#">mongodb-1.3.1.tgz</a> (904.0kB) 
<a href="#">1.3.0</a>	stable	2017-09-19	<a href="#">mongodb-1.3.0.tgz</a> (906.1kB) 

2. Extract the archive File.
3. Copy the php\_mongodb.dll file from the extracted folder to the PHP extension directory. this is usually the "C:\xampp\php\ext" folder in XAMPP Server.
4. Open the php.ini file inside your PHP installation (C:\xampp\php) and add the following line:

**extension=php\_mongodb.dll**

5. Save the file and close it. Restart the Apache web server.

Test MongoDB Connection From PHP Script

Let's write a very simple PHP program that creates a connection to the MongoDB server and dump the connection status.

Add following PHP code to your php script and access from the Web browser. The above PHP example will output something similar to:

```
<?php
```

```
$connection = new MongoDB\Driver\Manager("mongodb://localhost:27017");
var_dump($connection);
```

```
?>
```

### Result

```
object(MongoDB\Driver\Manager)#1 (2) { ["uri"]=> string(25) "mongodb://localhost:27017"
["cluster"]=> array(1) { [0]=> array(11) { ["host"]=> string(9) "localhost" ["port"]=> int(27017)
["type"]=> int(0) ["is_primary"]=> bool(false) ["is_secondary"]=> bool(false) ["is_arbiter"]=>
```

Name: salini kb

Roll No:33

Batch: MCA-B

Date: 24-05-2022

---

```
bool(false) ["is_hidden"]=> bool(false) ["is_passive"]=> bool(false) ["tags"]=> array(0) { }  
["last_is_master"]=> array(0) { } ["round_trip_time"]=> int(-1) } }
```



**Program no:9**

**Aim:** Query Processing : Performing CRUD operations with NoSQL database

```
use my_project_db
```

```
show dbs
```

```
db.createCollection("mydb")
```

```
db.createCollection("movie")
```

```
db.movie.insert({"name":"Avengers: Endgame"})
```

```
db.movie.insert(
```

```
[
```

```
  { "name": "Avengers: Infinity War" },
```

```
  { "name": "Avengers: Endgame" }
```

```
]
```

```
)
```

```
db.developers.insertMany(
```

```
[
```

```
  { _id: 20, devname: "John Wick", tools: "Visual Studio", born: 1948 },
```

```
  { _id: 21, devname: "Ganesh Roy", tools: "Net Beans", born: 1945 },
```

```
  { _id: 22, devname: "Deeksha Raul", tools: "Unity 3D", born: 1954 }
```

```
]
```

```
)
```

**Name:** salini kb

**Roll No:**33

**Batch:** MCA-B

**Date:** 03-06-2022

```

startup_log
> use my_project_db
switched to db my_project_db
> show dbs
AJCE      0.000GB
RECORD    0.000GB
admin     0.000GB
config    0.000GB
local     0.000GB
mca2023a  0.000GB
movie_db  0.000GB
> db.createCollection("mydb")
{ "ok" : 1 }
> db.createCollection("movie")
{ "ok" : 1 }
> db.movie.insert({"name":"Avengers: Endgame"})
WriteResult({ "nInserted" : 1 })
>
> db.movie.insert(
... [
...   { "name": "Avengers: Infinity War" },
...   { "name": "Avengers: Endgame" }
... ]
... )
uncaught exception: SyntaxError: illegal character :
@(shell):3:3
> db.movie.insert(
...   { "name": "Avengers: Infinity War" },
...   { "name": "Avengers: Endgame" }
... )
uncaught exception: SyntaxError: illegal character :
@(shell):3:3
>
> db.movie.insert({"name":"Avengers:infinitywar"})
WriteResult({ "nInserted" : 1 })
> db.movie.insert({"name":"Avengers:series"})
WriteResult({ "nInserted" : 1 })
> db.developers.insertMany(

```

```

> db.movie.insert(
...   { "name": "Avengers: Infinity War" },
...   { "name": "Avengers: Endgame" }
... )
uncaught exception: SyntaxError: illegal character :
@(shell):3:3
>
> db.movie.insert({"name":"Avengers:infinitywar"})
WriteResult({ "nInserted" : 1 })
> db.movie.insert({"name":"Avengers:series"})
WriteResult({ "nInserted" : 1 })
> db.developers.insertMany(
... [
...   { _id: 20, devname: "John Wick", tools: "Visual Studio", born: 1948 },
...   { _id: 21, devname: "Ganesh Roy", tools: "Net Beans", born: 1945 },
...   { _id: 22, devname: "Deeksha Raul", tools: "Unity 3D", born: 1954 }
... ]
... )
{ "acknowledged" : true, "insertedIds" : [ 20, 21, 22 ] }
>

```

**admin.table1** 5 DOCUMENTS

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER { field: 'value' } OPTIONS FIND RES

ADD DATA VIEW

Displaying documents 1 - 5 of 5

```

_id: ObjectId('62998d0a789203cafb10fcf7')
name: "salu"

_id: ObjectId('62998da5789203cafb10fcf8')
class: "mca"

_id: ObjectId('62998e1d789203cafb10fcf9')
name: "Avengers:Endgame"

_id: ObjectId('62998e2f789203cafb10fcfa')
name: "infinitywar"

_id: ObjectId('62998e4e789203cafb10fcfb')
name: "Avengers:series"

```

MongoDB Compass - localhost:27017/config.table2

localhost:27017 8 DBS 10 COLLECTIONS

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 5.0.8 Community

My Queries Databases

Filter your data

AIJCE RECORD admin table1 config table2 local

**config.table2** 3 DOCUMENTS 1 INDEXES

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER { field: 'value' } OPTIONS FIND RESET ...

ADD DATA VIEW

Displaying documents 1 - 3 of 3 REFRESH

```

_id: 20
devname: "johnwick"
tools: "visual studio"
born: 1948

_id: 21
devname: "ganeesh roy"
tools: "Netbeans"
born: 1954

_id: 22
devname: "Deeksha raul"
tools: "unity 3D"
born: 1945

```

## Read Operations

db.table1.find()

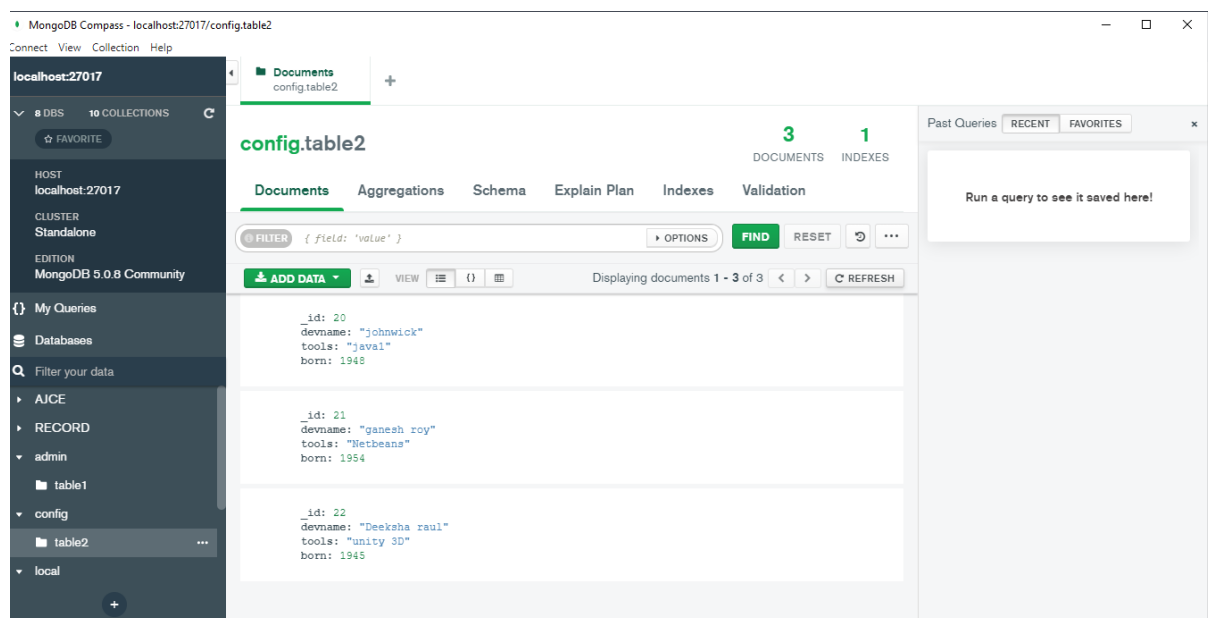
```
... )
WriteResult({ "nInserted" : 1 })
> db.table2.insert({_id:22, devname:"Deeksha raul",tools:"unity 3D",born:1945})
WriteResult({ "nInserted" : 1 })
> db.table1.find()
> db.table2.find()
{ "_id" : 20, "devname" : "johnwick", "tools" : "visual studio", "born" : 1948 }
{ "_id" : 21, "devname" : "ganesh roy", "tools" : "Netbeans", "born" : 1954 }
{ "_id" : 22, "devname" : "Deeksha raul", "tools" : "unity 3D", "born" : 1945 }
>
```

## Update

Use database name

Db.table2.update({"\_id":20,{set:{"tools":"java1"}})

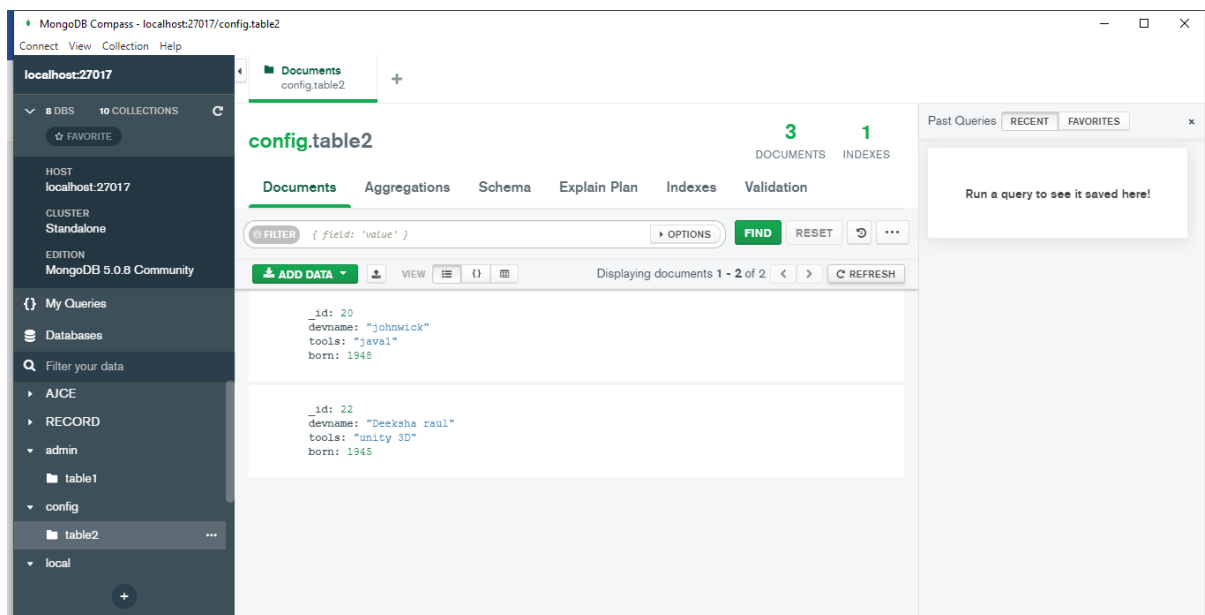
```
... {set: { "EmployeeName" : "NewMartin" } }},
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
> db.table2.update( {"_id" : 20}, {$set: { "tools" : "java1"}});
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
> db.table2.update( {"_id" : 20}, {$set: { "tools" : "java1"}});
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
> use config
switched to db config
> db.table2.update( {"_id" : 20}, {$set: { "tools" : "java1"}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```



## Delete

Db.table2.deleteOne({"\_id":21})

```
switched to db config
> db.table2.update( {"_id" : 20}, {$set: { "tools" : "java1"}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.table2.deleteOne({"_id":21})
{ "acknowledged" : true, "deletedCount" : 1 }
>
```



**Program no:10**

**Aim:** NoSQL and Front-End: PHP: Create a PHP form and insert data to mongodb

**Name:** salini kb

**Roll No:**33

**Batch:** MCA-B

**Date:** 03-06-2022

**Index.html**

```
<html>
<head>
<title>Document</title>
</head>
<body>
<h2>insert to mongo</h2>
<form action="insert.php" method="post">
<input type="text" name="name" placeholder="name">
<input type="number" name="rollno" placeholder="rollno">
<input type="password" name="password" placeholder="password">
<input type="text" name="firstname" placeholder="firstname">
<input type="submit" name="submit">
</form>
</body>
</html>
```

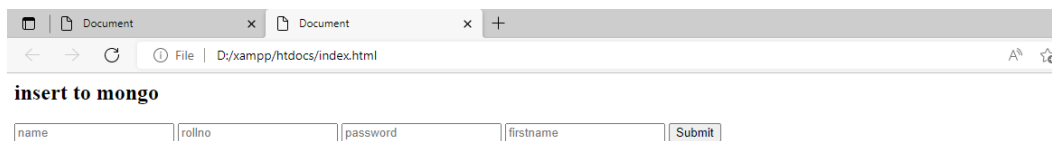
**Insert.php**

```
<?php
$mongo=new MongoDB\Driver\Manager("mongodb://localhost:27017");
if(isset($_POST["submit"])){
$name=$_POST["name"];
$first_name=$_POST["firstname"];
$rollno=$_POST["rollno"];
$passwd=$_POST["password"];
$writer=new MongoDB\Driver\Bulkwrite;
$writer-
>insert(["name"=>$name,"rollno"=>$rollno,"passwd"=>$passwd,"firstname"
=>$first_name]);
$mongo->executeBulkWrite('form124.insertion',$writer);
header("Location:success.html");
die();
}
```

?>

### Success.html

```
<html>
<head>
<title>Document</title>
</head>
<body>
<h2>successfully created</h2>
</body>
</html>
```



The screenshot shows a web browser window with two tabs, both titled 'Document'. The address bar shows the file path 'D:/xampp/htdocs/index.html'. Below the browser window, the text 'insert to mongo' is displayed. Underneath this text is a form with four input fields: 'name', 'rollno', 'password', and 'firstname', followed by a 'Submit' button.

