"Shri Gajanan Maharaj Prasanna"
P. R. Pote Patil Edu. & Welf. Trust's, Group of Institutions,
College of Engineering & Management, Amravati
Pote Estate, Pote Patil Road, Kathora, Amravati. (Maharashtra)
0721-2530342, 44, 2530089 Fax No.: 0721- 2530341 Email: prpotepatilcollege@gmail.com, website: www.prpatilcollege.org
(Recognized by AICTE, New Delhi & Affiliated to SGBAU, Amravati)

Department of Computer Science & Engineering

Certificate

This is to certify that Mr./Ms. **Krunal Gangadharrao Dhote**

of **V th** Semester of Bachelor of Engineering in **Computer Science and Engineering** of P. R. Pote (Patil) College of Engineering & Management, Amravati, has completed the term work satisfactory in subject **DBMS** for the academic year 2021- 2022 as prescribed in the curriculum.

Place: Amravati **PRN No: 198480093**

Date: Dec. 13, 2021 **Roll No: 334**

Head of the Department

Subject Teacher

Practical 1 : To Study the DBMS, RDBMS and Design the Entity Relationship Diagram

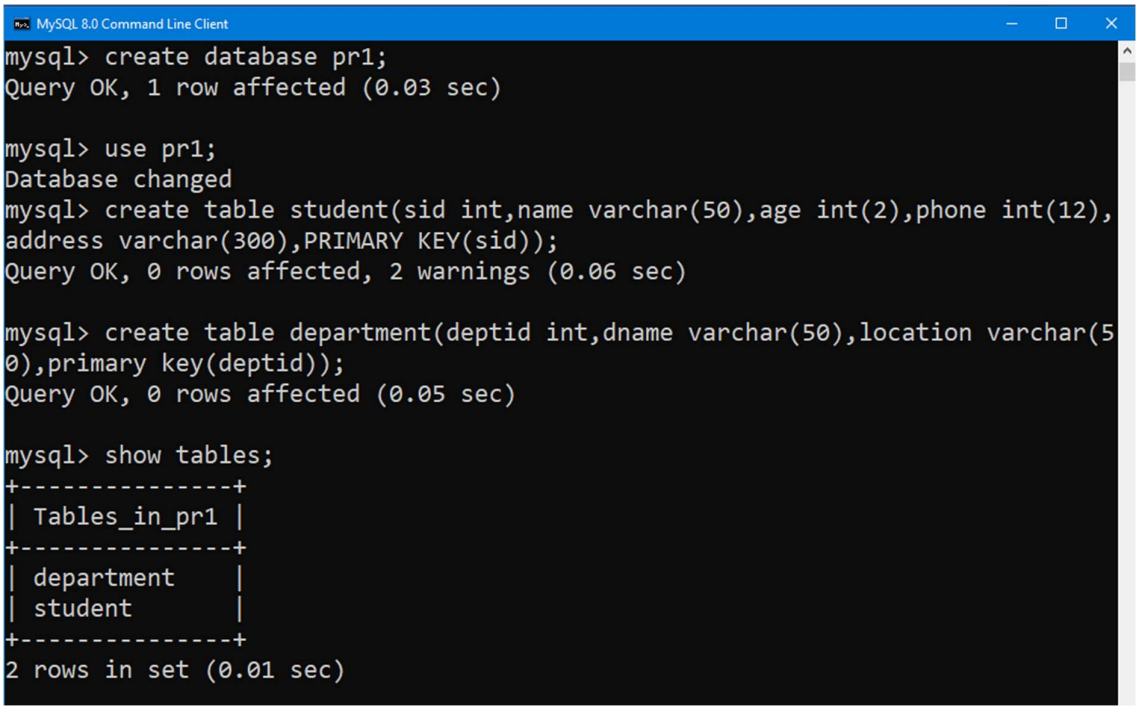
Practical 2 : To Implement Data Definition Language and Data Manipulation Language

Create table to store information of students and their departmentsas follows:

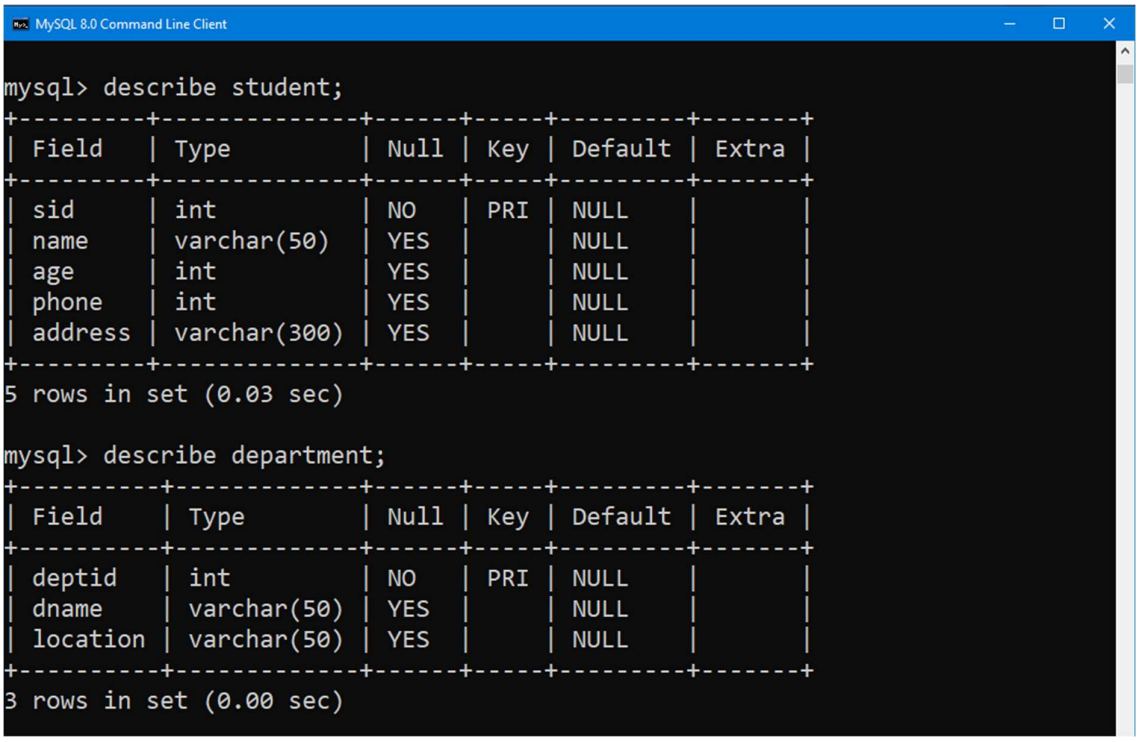
Student (id, name, age, phone, address, deptid)

Department (deptid, dname, location)

a) Create a table with several attributes and different constarints.

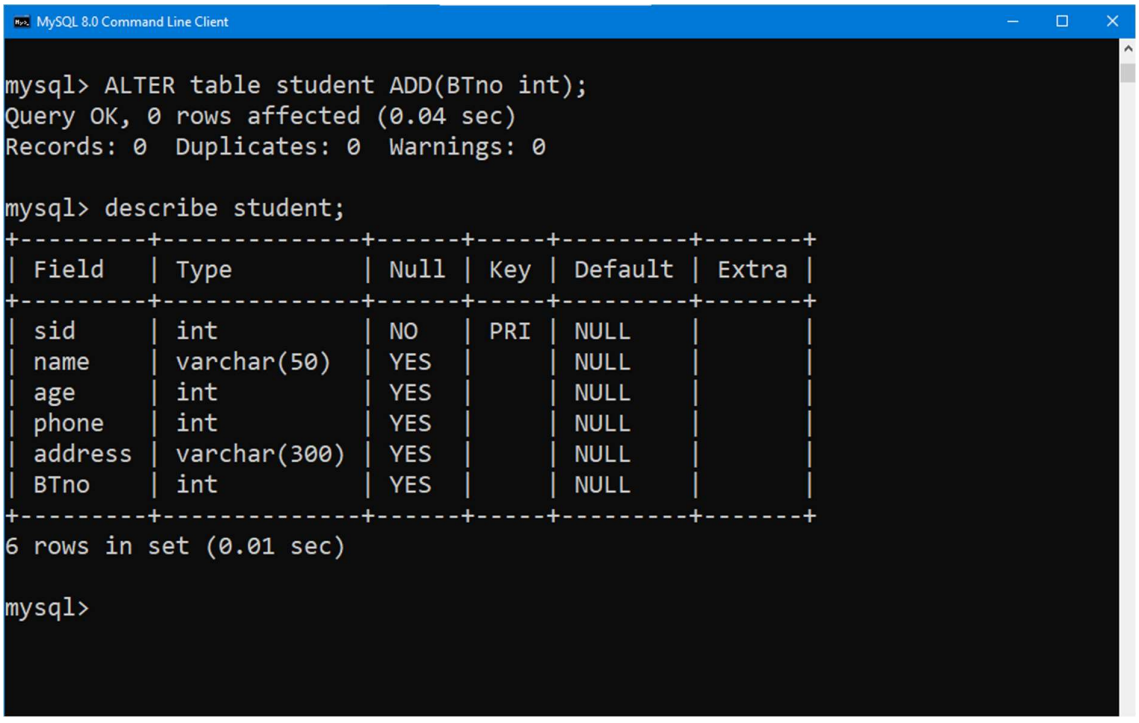


b) Describe the schema of the table

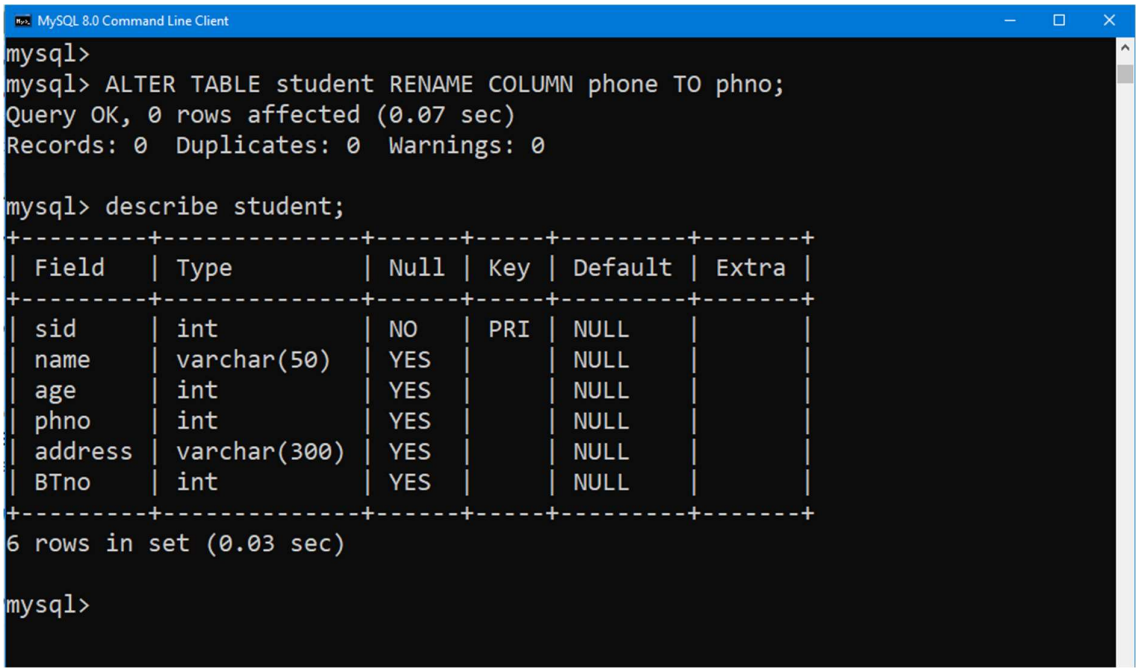


c) Add a column BTno in students table.

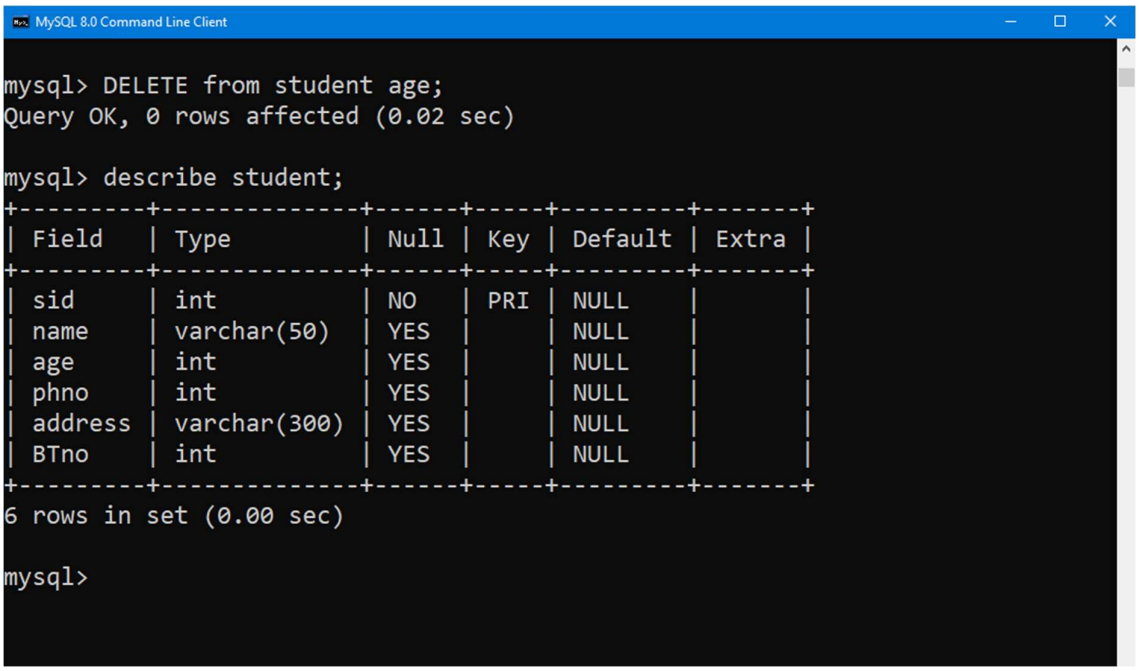
d) Change the data type of BTno.



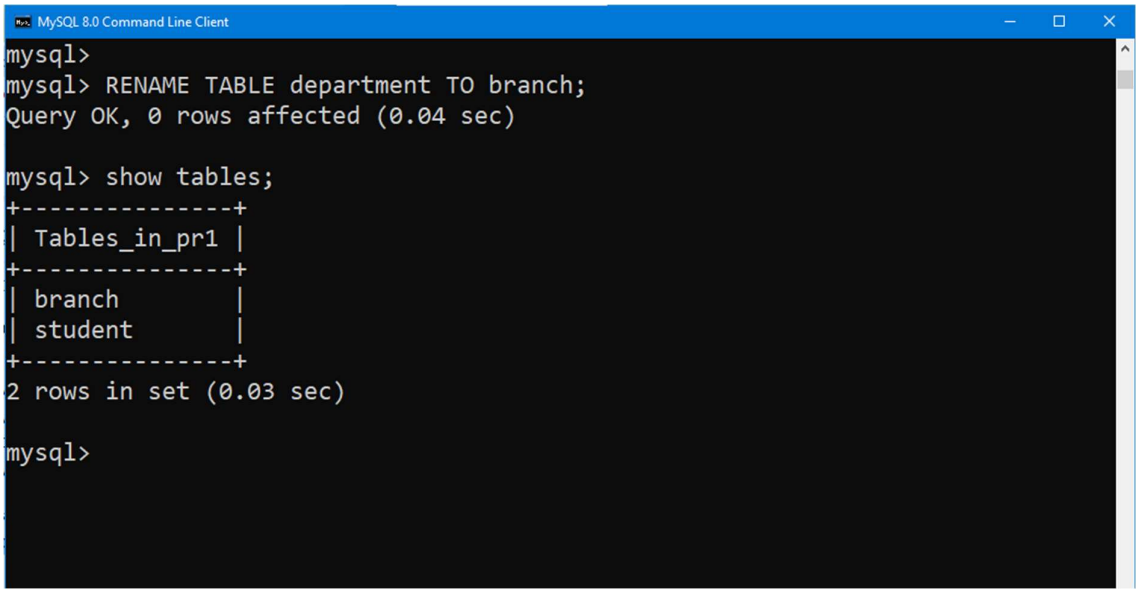
e) Rename the column phone to phno.



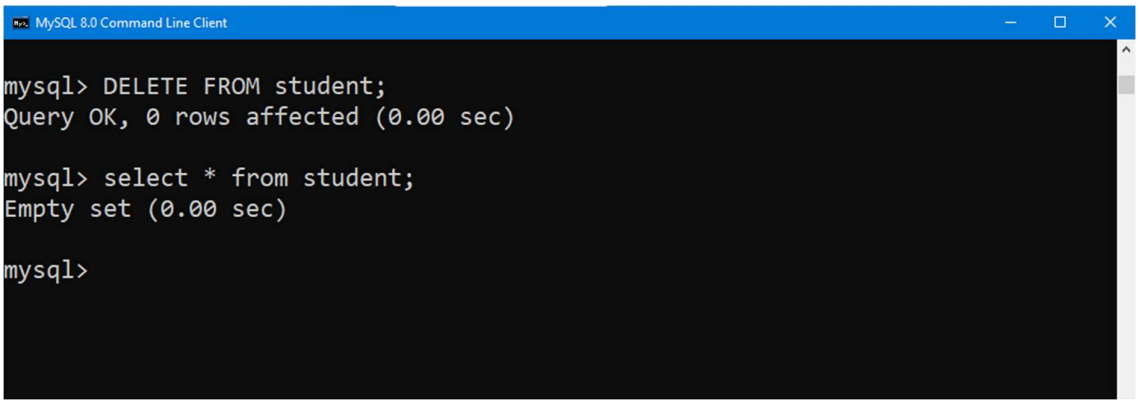
f) Delete the column age.



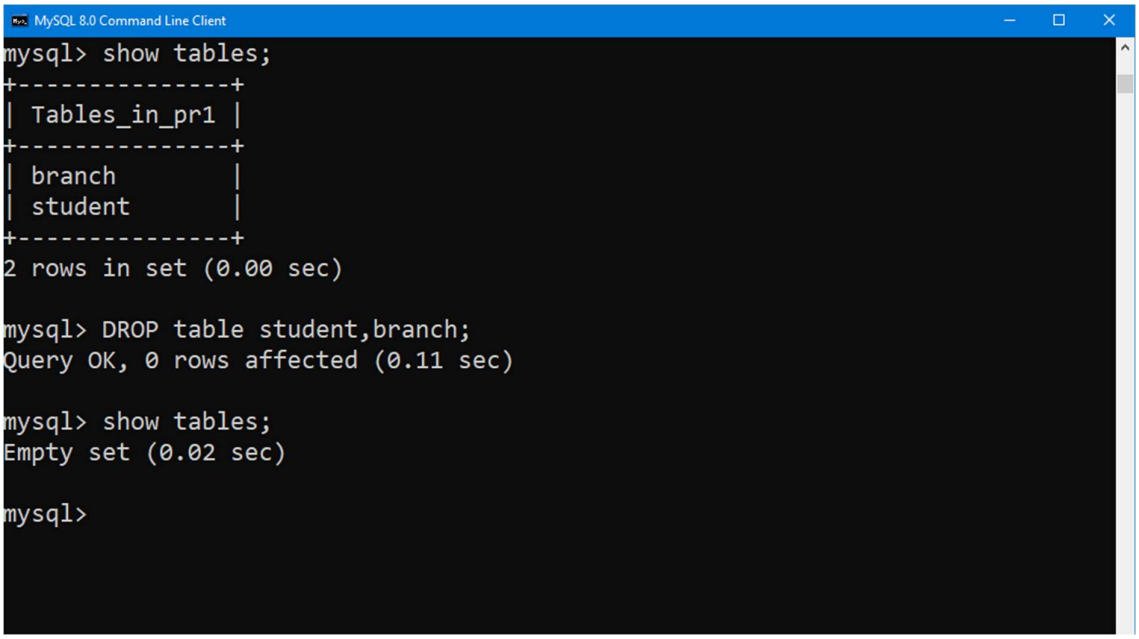
g) Rename the table department as Branch



h) Delete the table data.



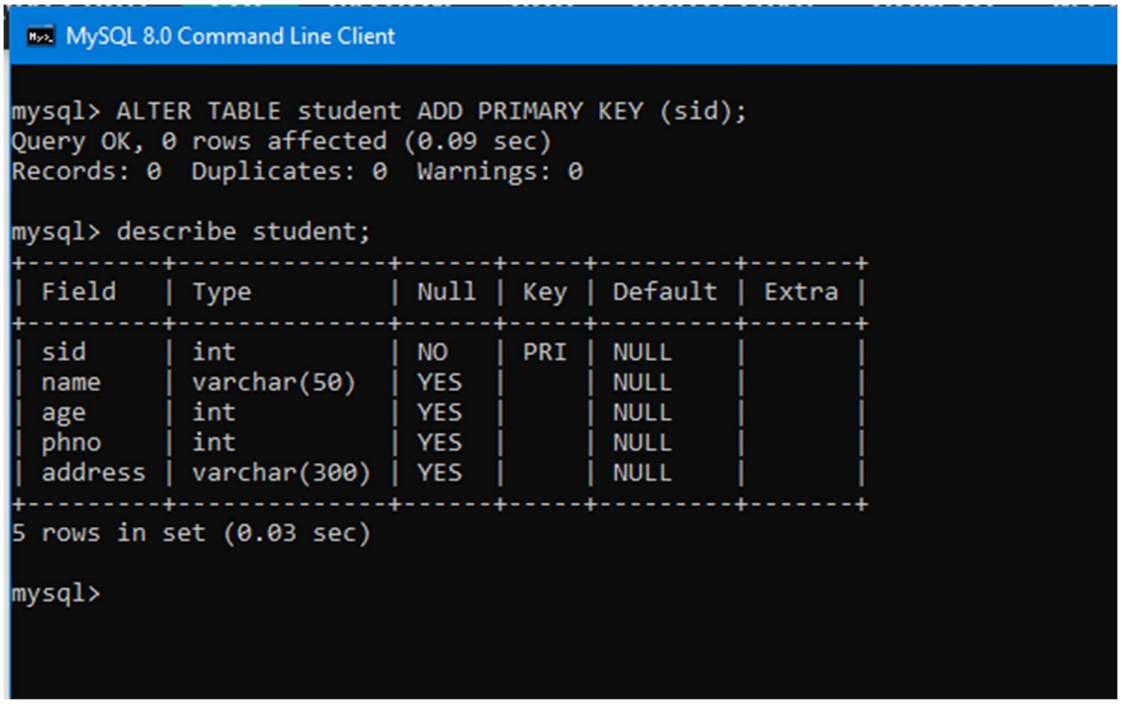
i) Delete the table schema



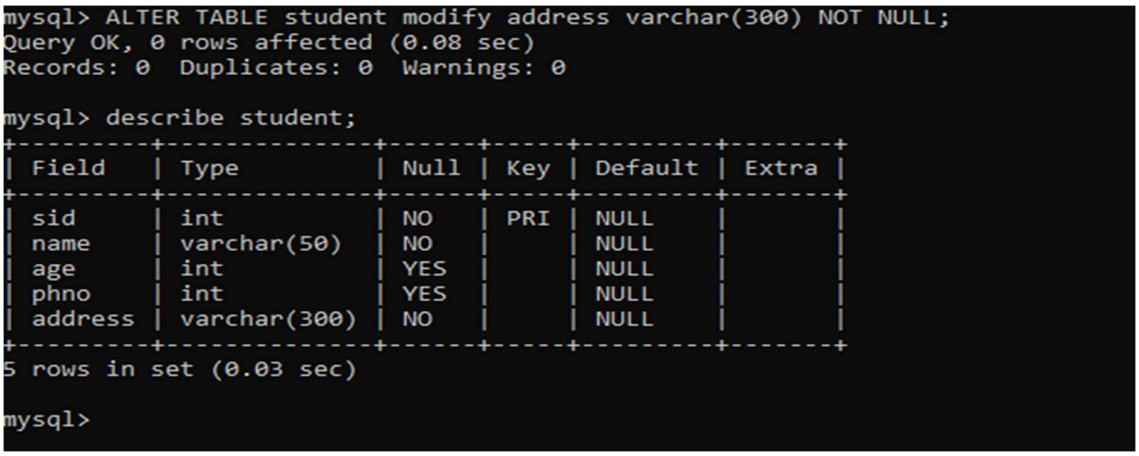
Practical 3 : To Implement various types of integrity constraints in SQL

Add the actual data of the students and department in the tables created in practical 2 and perform the following Queries.

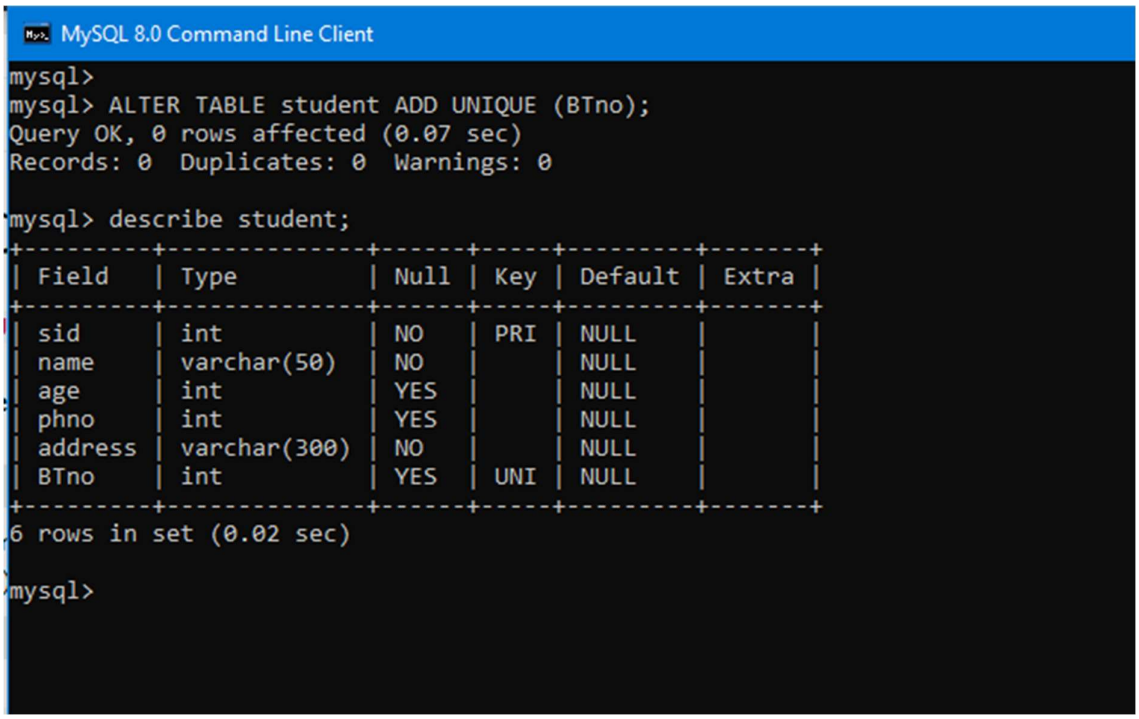
a) Make Id of Student a primary key.



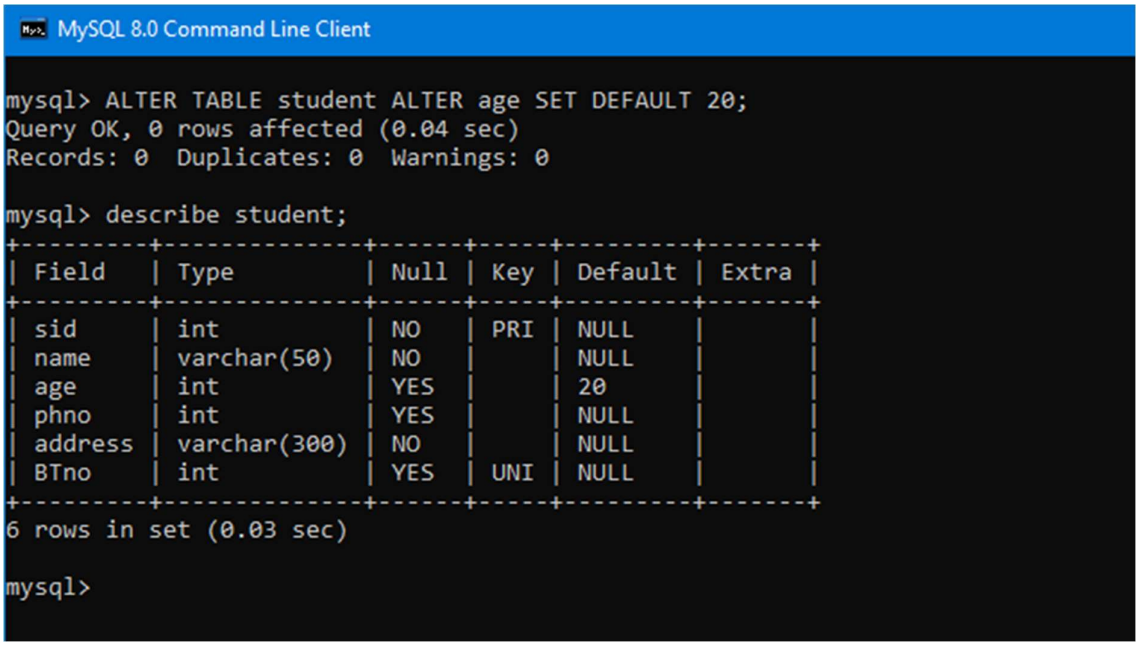
b) Make the Name and address field as not null.



c) Impose constraint on BTno such that values are unique.



d) Insert default value of age as 20



e) Insert multiple rows in the table one by one

```
mysql>
mysql>
mysql> INSERT INTO student values(334,'Krunal G. Dhote',20,9307871334,"At Post Tembhurkheda Ta. Warud Dist. Amravati",89);
Query OK, 1 row affected (0.02 sec)

mysql> select * from student;
+-----+----------------+------+------------+---------------------------------------------+------+
| sid | name           | age  | phno       | address                                     | BTno |
+-----+----------------+------+------------+---------------------------------------------+------+
| 334 | Krunal G. Dhote |   20 | 9307871334 | At Post Tembhurkheda Ta. Warud Dist. Amravati |   89 |
+-----+----------------+------+------------+---------------------------------------------+------+
1 row in set (0.00 sec)

mysql>
```

f) Insert multiple rows using single query.

```
mysql>
mysql>
mysql> INSERT INTO student (sid,name,age,phno,address,BTno)  values (302,'Achal V. Kale',19,9373223047,"Amravati",02),(338,'Nikhil D. Kothe',22,9309823466,'Udapur',03);

Query OK, 2 rows affected (0.03 sec)
Records: 2  Duplicates: 0  Warnings: 0

mysql> select * from student;
+-----+----------------+------+-----------+---------------------------------------------+------+
| sid | name           | age  | phno      | address                                     | BTno |
+-----+----------------+------+-----------+---------------------------------------------+------+
| 302 | Achal V. Kale  |   19 | 9373223047 | Amravati                                   |    2 |
| 334 | Krunal G. Dhote |  20 | 9307871334 | At Post Tembhurkheda Ta. Warud Dist. Amravati |  89 |
| 338 | Nikhil D. Kothe |  22 | 9309823466 | Udapur                                     |    3 |
+-----+----------------+------+-----------+---------------------------------------------+------+
3 rows in set (0.02 sec)

mysql>
```

g) Insert some rows skipping values of some attributes.

```
mysql> INSERT INTO student (sid,name,age,address)  values (341,'PrajjwalWatane',22,"Amravati");
Query OK, 1 row affected (0.01 sec)

mysql> select * from student;
+-----+----------------+------+------------+---------------------------------------------+------+
| sid | name           | age  | phno       | address                                     | BTno |
+-----+----------------+------+------------+---------------------------------------------+------+
| 302 | Achal V. Kale  |   19 | 9373223047 | Amravati                                    |    2 |
| 334 | Krunal G. Dhote |  20 | 9307871334 | At Post Tembhurkheda Ta. Warud Dist. Amravati |  89 |
| 338 | Nikhil D. Kothe |  22 | 9309823466 | Udapur                                      |    3 |
| 341 | PrajjwalWatane  |  22 | NULL       | Amravati                                    | NULL |
+-----+----------------+------+------------+---------------------------------------------+------+
4 rows in set (0.02 sec)

mysql>
```

h) Display the complete data entered into the tables.

```
mysql> select * from student;
+-----+-----------------+------+------------+-----------------------------------------------+------+
| sid | name            | age  | phno       | address                                       | BTno |
+-----+-----------------+------+------------+-----------------------------------------------+------+
| 302 | Achal V. Kale   |   19 | 9373223047 | Amravati                                      |    2 |
| 334 | Krunal G. Dhote |   20 | 9307871334 | At Post Tembhurkheda Ta. Warud Dist. Amravati |   89 |
| 338 | Nikhil D. Kothe |   22 | 9309823466 | Udapur                                        |    3 |
| 341 | PrajjwalWatane  |   22 | NULL       | Amravati                                      | NULL |
+-----+-----------------+------+------------+-----------------------------------------------+------+
4 rows in set (0.02 sec)

mysql>
```

i) Update student name whose id is 341

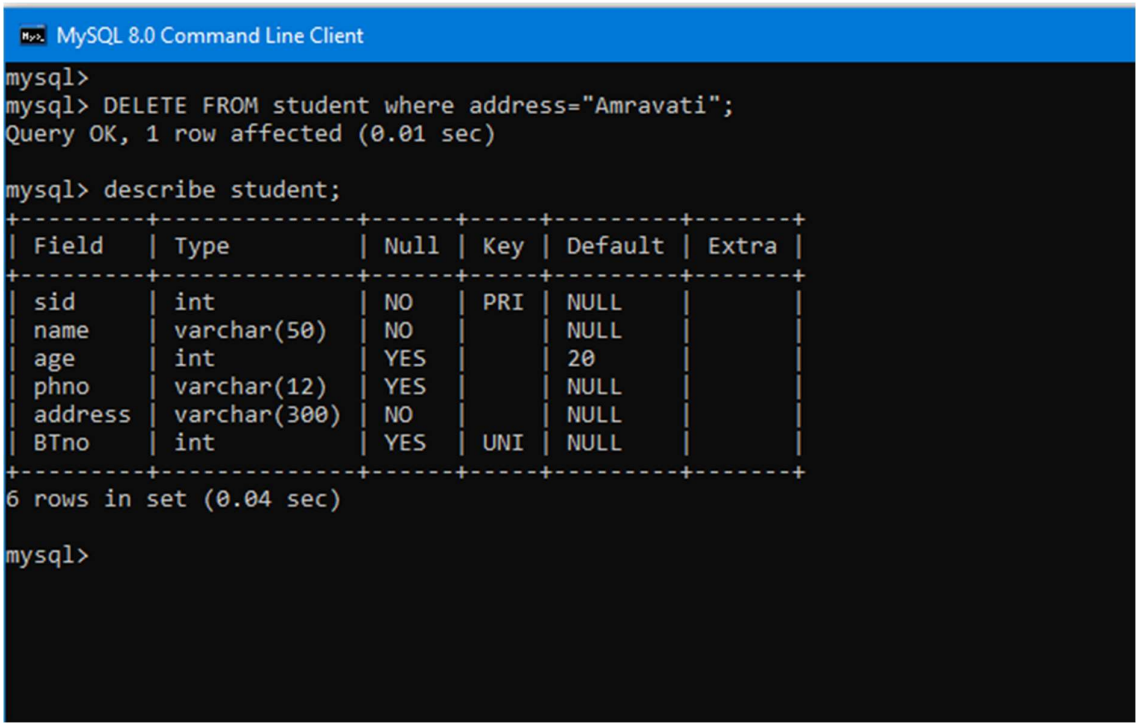
```
MySQL 8.0 Command Line Client

mysql> use pr2;
Database changed
mysql> update student set name="Prajjwal G. Watane" where sid=341;
Query OK, 1 row affected (0.03 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from student;
+-----+-------------------+------+------------+-----------------------------------------------+------+
| sid | name              | age  | phno       | address                                       | BTno |
+-----+-------------------+------+------------+-----------------------------------------------+------+
| 302 | Achal V. Kale     |   19 | 9373223047 | Amravati                                      |    2 |
| 334 | Krunal G. Dhote   |   20 | 9307871334 | At Post Tembhurkheda Ta. Warud Dist. Amravati |   89 |
| 338 | Nikhil D. Kothe   |   22 | 9309823466 | Udapur                                        |    3 |
| 341 | Prajjwal G. Watane |   22 | NULL       | Amravati                                      | NULL |
+-----+-------------------+------+------------+-----------------------------------------------+------+
4 rows in set (0.00 sec)

mysql>
```

j) Change address and phone no of student with name XYZ.

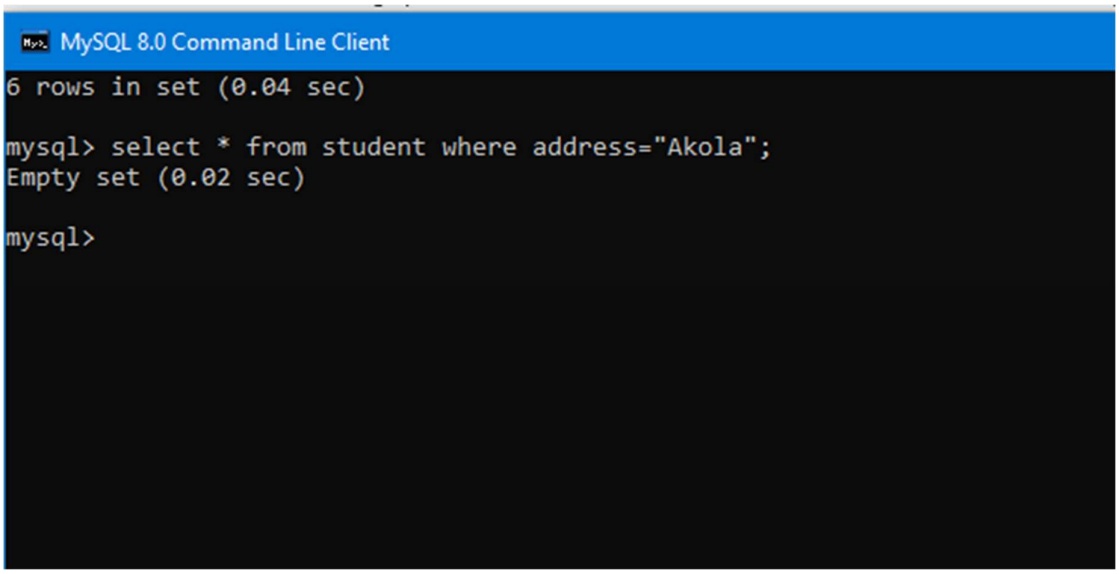
```
PYTHON  PHP  BOOTSTRAP  HOW TO  W3.CSS  JAVA  JQUERY  C++

MySQL 8.0 Command Line Client

ysql>
ysql>
ysql> update student set address="AMT",phno=919373223047 where name="Achal V. Kale";
uery OK, 1 row affected (0.02 sec)
ows matched: 1  Changed: 1  Warnings: 0

ysql> select * from student;
-----+-------------------+------+-------------+-----------------------------------------------+------+
 sid | name              | age  | phno        | address                                       | BTno |
-----+-------------------+------+-------------+-----------------------------------------------+------+
 302 | Achal V. Kale     |   19 | 919373223047 | AMT                                          |    2 |
 334 | Krunal G. Dhote   |   20 | 9307871334  | At Post Tembhurkheda Ta. Warud Dist. Amravati |   89 |
 338 | Nikhil D. Kothe   |   22 | 9309823466  | Udapur                                        |    3 |
 341 | Prajjwal G. Watane |   22 | NULL        | Amravati                                      | NULL |
-----+-------------------+------+-------------+-----------------------------------------------+------+
 rows in set (0.02 sec)

ysql>
```

k) Delete the phone no of student 23.

l) Delete the students who belong to Amravati city.



m) Display all students who belong to city Akola.



n) Display the student names with their department names.

Practical 4 : To study operators, expressions, functions and sub queries in SQL

MySQL 8.0 Command Line Client

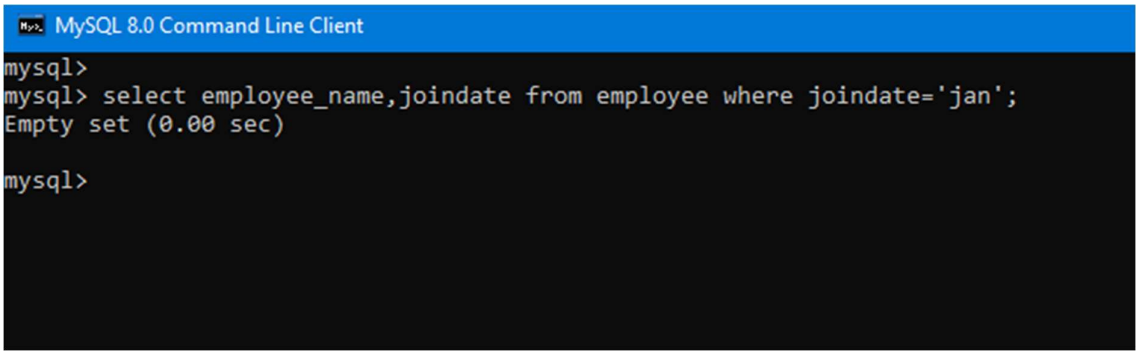
```
mysql>
mysql> select * from employee;
+-------+---------------+---------+---------+------+----------+--------+------+
| empno | employee_name | address | manager | post | joindate | salary | age  |
+-------+---------------+---------+---------+------+----------+--------+------+
|     1 | A             | Delhi   | Z       | AB   | 5 apr    |  20000 |   21 |
|     2 | B             | Delhi   | Z       | CD   | 5 apr    |  21000 |   21 |
|     3 | C             | Mumbai  | Y       | AB   | 9 oct    |  50000 |   21 |
+-------+---------------+---------+---------+------+----------+--------+------+
3 rows in set (0.00 sec)

mysql> select * from department;
+---------+-----------+----------+
| Dept_id | dept_name | location |
+---------+-----------+----------+
|       1 | CSE       | AMT      |
|       2 | CIVIL     | AMT      |
+---------+-----------+----------+
2 rows in set (0.00 sec)
```

1) Find out the difference between maximum and minimum salary of employee.

MySQL 8.0 Command Line Client

```
mysql> select salary from employee;
+--------+
| salary |
+--------+
|  20000 |
|  21000 |
|  50000 |
+--------+
3 rows in set (0.02 sec)

mysql> SELECT MAX(salary) - MIN(salary) DIFFERENCE FROM employee;
+------------+
| DIFFERENCE |
+------------+
|      30000 |
+------------+
1 row in set (0.02 sec)

mysql>
```

2) Display name and date of joining of employee who join in month of January.



3) Find out the most experienced employee.

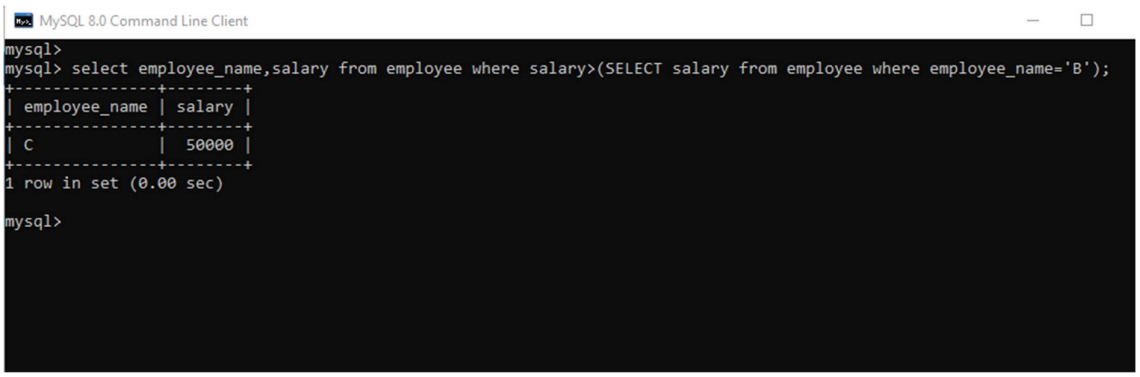
4) Display employee name in small letter.

5) Calculate experience of each employee and print with employee name.

6) Display the months between 1 June 2010 & 1 August 2012.

7) List all jobs available in employee table.

8) List the employee name and salary whose salary greater than salary of any name.



Practical 5: To Implement SELECT command with different clauses and SET operators in SQL.

1) List all employees who work as clerk analyst or both.

2) List all employee who work in both dept 30, 40.

3) List all employees who work as analyst but not managers.

4) List number of employees who belong to same job.

5) Display total salary spent for each job.

6) List all employees in alphabetical order.

7) Display manager name and no. of employee who has more than 1 employee working under him. 8) Display all employees in order starting from largest salary.

9) Display no. of employees working in each department along with department name.

```
mysql> select * from emplyee where post='cleark' union all select * from emplyee where post='analyst';
+-------+----------+----------+---------+---------+------------+--------+------+---------+
| empID | emp_name | address  | manger  | post    | JOD        | salary | age  | deptid  |
+-------+----------+----------+---------+---------+------------+--------+------+---------+
|   198 | samir    | amravati | megha   | cleark  | 2010-03-01 |  25000 |   30 |      40 |
|   129 | tanishk  | mumbai   | nidhi   | cleark  | 2018-04-05 |  30000 |   21 |      30 |
|   156 | revati   | pune     | megha   | analyst | 2014-05-21 |  27500 |   35 |      40 |
+-------+----------+----------+---------+---------+------------+--------+------+---------+
3 rows in set (0.01 sec)

mysql> select post,count(emp_name) from emplyee group by post;
+--------------------+-----------------+
| post               | count(emp_name) |
+--------------------+-----------------+
| analyst            |               1 |
| cleark             |               2 |
| frontend dev       |               1 |
| manager            |               1 |
| senior engineer    |               1 |
| system-adminstrator|               1 |
| web devloper       |               1 |
+--------------------+-----------------+
7 rows in set (0.04 sec)

mysql> select post ,sum(salary) from emplyee group by post;
+--------------------+-------------+
| post               | sum(salary) |
+--------------------+-------------+
| analyst            |       27500 |
| cleark             |       55000 |
| frontend dev       |       30000 |
| manager            |       60000 |
| senior engineer    |       60000 |
| system-adminstrator|       25000 |
| web devloper       |       27500 |
+--------------------+-------------+
7 rows in set (0.02 sec)
```

```
mysql> select emp_name from emplyee order by emp_name;
+----------+
| emp_name |
+----------+
| avanti   |
| ragini   |
| revati   |
| riddhi   |
| rohit    |
| samir    |
| tanishk  |
| vansh    |
+----------+
8 rows in set (0.01 sec)

mysql>
```

Practical 6 : To Implement various types of joins in SQL

1) Display id, post, deptname and dept location of employee using equijoin, innerjoin, natural join with USING and ON clause.

2) Display id, name, location having deptid > its dept using non equijoin.

3) Display employee name manager name its department with job of manager using self join.

4) Perform Cartesian product of employee and department table using cross join.

5) Display all the information of employee and department using left right and full outer join.

```
mysql> SELECT empID ,post, dep_name,location from emplyee INNER JOIN departments on emplyee.dept_id=depar
tments.deptid;
+-------+---------------------+---------------------+--------------------+
| empID | post                | dep_name            | location           |
+-------+---------------------+---------------------+--------------------+
|   102 | system-adminstrator | DBS department      | wing B             |
|   152 | web devloper        | DBS department      | wing B             |
|   125 | frontend dev        | management department | second floor     |
|   161 | senior engineer     | management department | second floor     |
|   198 | cleark              | DBS department      | wing B             |
|   156 | analyst             | DBS department      | wing B             |
|   129 | cleark              | AI department       | wing C second floor |
|   167 | manager             | management department | second floor     |
+-------+---------------------+---------------------+--------------------+
8 rows in set (0.04 sec)
```

```
mysql> select empID,post dep_name,location from emplyee INNER JOIN departments using(deptid);
+-------+---------------------+---------------------+
| empID | dep_name            | location            |
+-------+---------------------+---------------------+
|   102 | system-adminstrator | wing B              |
|   152 | web devloper        | wing B              |
|   125 | frontend dev        | second floor        |
|   161 | senior engineer     | second floor        |
|   198 | cleark              | wing B              |
|   156 | analyst             | wing B              |
|   129 | cleark              | wing C second floor |
|   167 | manager             | second floor        |
+-------+---------------------+---------------------+
8 rows in set (0.00 sec)

mysql> select empID,post,dep_name,location from emplyee NATURAL JOIN departments;
+-------+---------------------+---------------------+--------------------+
| empID | post                | dep_name            | location           |
+-------+---------------------+---------------------+--------------------+
|   102 | system-adminstrator | DBS department      | wing B             |
|   152 | web devloper        | DBS department      | wing B             |
|   125 | frontend dev        | management department | second floor     |
|   161 | senior engineer     | management department | second floor     |
|   198 | cleark              | DBS department      | wing B             |
|   156 | analyst             | DBS department      | wing B             |
|   129 | cleark              | AI department       | wing C second floor |
|   167 | manager             | management department | second floor     |
+-------+---------------------+---------------------+--------------------+
8 rows in set (0.00 sec)
```

```
mysql> select empID,post,dep_name,location from emplyee INNER JOIN departments on emplyee.deptid>departme
nts.deptid;
+-------+---------------------+---------------------+--------------------+
| empID | post                | dep_name            | location           |
+-------+---------------------+---------------------+--------------------+
|   102 | system-adminstrator | web dev department  | first floor        |
|   102 | system-adminstrator | management department | second floor     |
|   102 | system-adminstrator | AI department       | wing C second floor |
|   152 | web devloper        | web dev department  | first floor        |
|   152 | web devloper        | management department | second floor     |
|   152 | web devloper        | AI department       | wing C second floor |
|   125 | frontend dev        | web dev department  | first floor        |
|   161 | senior engineer     | web dev department  | first floor        |
|   198 | cleark              | web dev department  | first floor        |
|   198 | cleark              | management department | second floor     |
|   198 | cleark              | AI department       | wing C second floor |
|   156 | analyst             | web dev department  | first floor        |
|   156 | analyst             | management department | second floor     |
|   156 | analyst             | AI department       | wing C second floor |
|   129 | cleark              | web dev department  | first floor        |
|   129 | cleark              | management department | second floor     |
|   167 | manager             | web dev department  | first floor        |
+-------+---------------------+---------------------+--------------------+
17 rows in set (0.00 sec)
```

```
mysql> select empID,post,dep_name,location from emplyee CROSS JOIN departments;
+-------+--------------------+----------------------+--------------------+
| empID | post               | dep_name             | location           |
+-------+--------------------+----------------------+--------------------+
|   102 | system-adminstrator | web dev department   | first floor        |
|   102 | system-adminstrator | management department | second floor       |
|   102 | system-adminstrator | AI department        | wing C second floor |
|   102 | system-adminstrator | DBS department       | wing B             |
|   152 | web devloper       | web dev department   | first floor        |
|   152 | web devloper       | management department | second floor       |
|   152 | web devloper       | AI department        | wing C second floor |
|   152 | web devloper       | DBS department       | wing B             |
|   125 | frontend dev       | web dev department   | first floor        |
|   125 | frontend dev       | management department | second floor       |
|   125 | frontend dev       | AI department        | wing C second floor |
|   125 | frontend dev       | DBS department       | wing B             |
|   161 | senior engineer    | web dev department   | first floor        |
|   161 | senior engineer    | management department | second floor       |
|   161 | senior engineer    | AI department        | wing C second floor |
|   161 | senior engineer    | DBS department       | wing B             |
|   198 | cleark             | web dev department   | first floor        |
|   198 | cleark             | management department | second floor       |
|   198 | cleark             | AI department        | wing C second floor |
|   198 | cleark             | DBS department       | wing B             |
|   156 | analyst            | web dev department   | first floor        |
|   156 | analyst            | management department | second floor       |
|   156 | analyst            | AI department        | wing C second floor |
|   156 | analyst            | DBS department       | wing B             |
|   129 | cleark             | web dev department   | first floor        |
|   129 | cleark             | management department | second floor       |
|   129 | cleark             | AI department        | wing C second floor |
|   129 | cleark             | DBS department       | wing B             |
|   167 | manager            | web dev department   | first floor        |
|   167 | manager            | management department | second floor       |
|   167 | manager            | AI department        | wing C second floor |
|   167 | manager            | DBS department       | wing B             |
+-------+--------------------+----------------------+--------------------+
32 rows in set (0.00 sec)
```

```
mysql> select empID,post,dep_name,location from emplyee LEFT JOIN departments on emplyee.deptid=departmen
ts.deptid;
+-------+--------------------+----------------------+--------------------+
| empID | post               | dep_name             | location           |
+-------+--------------------+----------------------+--------------------+
|   125 | frontend dev       | management department | second floor       |
|   161 | senior engineer    | management department | second floor       |
|   167 | manager            | management department | second floor       |
|   129 | cleark             | AI department        | wing C second floor |
|   102 | system-adminstrator | DBS department       | wing B             |
|   152 | web devloper       | DBS department       | wing B             |
|   198 | cleark             | DBS department       | wing B             |
|   156 | analyst            | DBS department       | wing B             |
+-------+--------------------+----------------------+--------------------+
8 rows in set (0.00 sec)

mysql> select empID,post ,dep_name,location from emplyee RIGHT JOIN departments on emplyee.deptid=departm
ents.deptid;
+-------+--------------------+----------------------+--------------------+
| empID | post               | dep_name             | location           |
+-------+--------------------+----------------------+--------------------+
|   102 | system-adminstrator | DBS department       | wing B             |
|   152 | web devloper       | DBS department       | wing B             |
|   125 | frontend dev       | management department | second floor       |
|   161 | senior engineer    | management department | second floor       |
|   198 | cleark             | DBS department       | wing B             |
|   156 | analyst            | DBS department       | wing B             |
|   129 | cleark             | AI department        | wing C second floor |
|   167 | manager            | management department | second floor       |
|  NULL | NULL               | web dev department   | first floor        |
+-------+--------------------+----------------------+--------------------+
9 rows in set (0.00 sec)
```

Prcatical 7 : To Study and Implement VIEWS in SQL

1) Create a view called manager which stores the information about the entire employee who is manager.

3) Create a view called manager which stores the information about the employee and their respective department name, deptname, job and location.

4) Display information of all the view.

5) Display the manager who works in ‘10’ department.

6) Increment salary of a manager whose age is greater than 50 by 10%.

7) Show the updated information in the view and in the logical relation.

8) Update a dept name of emp who work in CSE to IT dept and show the result of a view as well as logical relation.

9) Drop the view.

```
mysql> CREATE VIEW managerview as select empID,emp_name,age,post,address,deptid,salary from emplyee where
 manger=' not null';
Query OK, 0 rows affected (0.16 sec)

mysql> create view empdata as select empID,dep_name,location ,post from emplyee,departments where emplyee
.deptid=departments.deptid;
Query OK, 0 rows affected (0.04 sec)

mysql> select * from managerview;
Empty set (0.03 sec)

mysql> select * from empdata;
+-------+----------------------+--------------------+--------------------+
| empID | dep_name             | location           | post               |
+-------+----------------------+--------------------+--------------------+
|   102 | DBS department       | wing B             | system-adminstrator |
|   152 | DBS department       | wing B             | web devloper       |
|   125 | management department | second floor       | frontend dev       |
|   161 | management department | second floor       | senior engineer    |
|   198 | DBS department       | wing B             | cleark             |
|   156 | DBS department       | wing B             | analyst            |
|   129 | AI department        | wing C second floor | cleark             |
|   167 | management department | second floor       | manager            |
+-------+----------------------+--------------------+--------------------+
8 rows in set (0.00 sec)
```


```
mysql> CREATE VIEW managerview as select empID,emp_name,age,post,address,deptid,salary from emplyee where
 manger=' not null';
Query OK, 0 rows affected (0.16 sec)

mysql> create view empdata as select empID,dep_name,location ,post from emplyee,departments where emplyee
.deptid=departments.deptid;
Query OK, 0 rows affected (0.04 sec)

mysql> select * from managerview;
Empty set (0.03 sec)

mysql> select * from empdata;
+-------+----------------------+--------------------+--------------------+
| empID | dep_name             | location           | post               |
+-------+----------------------+--------------------+--------------------+
|   102 | DBS department       | wing B             | system-adminstrator |
|   152 | DBS department       | wing B             | web devloper       |
|   125 | management department | second floor       | frontend dev       |
|   161 | management department | second floor       | senior engineer    |
|   198 | DBS department       | wing B             | cleark             |
|   156 | DBS department       | wing B             | analyst            |
|   129 | AI department        | wing C second floor | cleark             |
|   167 | management department | second floor       | manager            |
+-------+----------------------+--------------------+--------------------+
8 rows in set (0.00 sec)
```


```
mysql> select emp_name from managersview where deptid=10;
Empty set (0.00 sec)

mysql> select emp_name from managersview where deptid=20;
+----------+
| emp_name |
+----------+
| vansh    |
| riddhi   |
| avanti   |
+----------+
3 rows in set (0.00 sec)

mysql> update managersview set salary=salary*1.1 where age>50;
Query OK, 0 rows affected (0.07 sec)
Rows matched: 0  Changed: 0  Warnings: 0

mysql> update managersview set salary=salary*1.1 where age>30;
Query OK, 2 rows affected (0.07 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> select * from managersview;
+-------+----------+--------------------+------+--------+--------+
| empID | emp_name | post               | age  | salary | deptid |
+-------+----------+--------------------+------+--------+--------+
|   102 | ragini   | system-adminstrator |   30 |  25000 |     40 |
|   152 | rohit    | web devloper       |   35 |  27500 |     40 |
|   125 | vansh    | frontend dev       |   21 |  30000 |     20 |
|   161 | riddhi   | senior engineer    |   28 |  60000 |     20 |
|   198 | samir    | cleark             |   30 |  25000 |     40 |
|   156 | revati   | analyst            |   35 |  27500 |     40 |
|   129 | tanishk  | cleark             |   21 |  30000 |     30 |
|   167 | avanti   | manager            |   28 |  60000 |     20 |
+-------+----------+--------------------+------+--------+--------+
8 rows in set (0.00 sec)
```

```
mysql> select * from emplyee;
+-------+----------+----------+--------+--------------------+------------+--------+------+--------+
| empID | emp_name | address  | manger | post               | JOD        | salary | age  | deptid |
+-------+----------+----------+--------+--------------------+------------+--------+------+--------+
|   102 | ragini   | amravati | megha  | system-adminstrator| 2010-01-01 |  25000 |   30 |     40 |
|   152 | rohit    | pune     | megha  | web devloper       | 2014-05-21 |  27500 |   35 |     40 |
|   125 | vansh    | mumbai   | nidhi  | frontend dev       | 2018-04-05 |  30000 |   21 |     20 |
|   161 | riddhi   | amravati | harish | senior engineer    | 2020-02-14 |  60000 |   28 |     20 |
|   198 | samir    | amravati | megha  | cleark             | 2010-03-01 |  25000 |   30 |     40 |
|   156 | revati   | pune     | megha  | analyst            | 2014-05-21 |  27500 |   35 |     40 |
|   129 | tanishk  | mumbai   | nidhi  | cleark             | 2018-04-05 |  30000 |   21 |     30 |
|   167 | avanti   | amravati | harish | manager            | 2020-02-14 |  60000 |   28 |     20 |
+-------+----------+----------+--------+--------------------+------------+--------+------+--------+
8 rows in set (0.00 sec)

mysql> update empdata set dep_name='IT' where dep_name='cse';
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0  Changed: 0  Warnings: 0

mysql> update empdata set dep_name='IT dept' where dep_name='management department';
Query OK, 1 row affected (0.08 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from empdata;
+-------+---------------+--------------------+--------------------+
| empID | dep_name      | location           | post               |
+-------+---------------+--------------------+--------------------+
|   102 | DBS department| wing B             | system-adminstrator|
|   152 | DBS department| wing B             | web devloper       |
|   125 | IT dept       | second floor       | frontend dev       |
|   161 | IT dept       | second floor       | senior engineer    |
|   198 | DBS department| wing B             | cleark             |
|   156 | DBS department| wing B             | analyst            |
|   129 | AI department | wing C second floor| cleark             |
|   167 | IT dept       | second floor       | manager            |
+-------+---------------+--------------------+--------------------+
8 rows in set (0.00 sec)

mysql> drop managersview;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL
 server version for the right syntax to use near 'managersview' at line 1
mysql> drop view managerview;
Query OK, 0 rows affected (0.01 sec)
```

Practical 8 : To Study and Implement Triggers in SQL

1) Create trigger to update to product history table when the price of product is updated in the product table.

2) Fire the trigger by executing respective queries.

```
mysql> drop table history;
Query OK, 0 rows affected (0.31 sec)

mysql> create table product(pid int primary key,pname varchar(20),supplier varchar(20),unitprice numeric(
3,2));
Query OK, 0 rows affected (0.33 sec)

mysql> create table history(pid int primary key,pname varchar(20),supplier varchar(20),unitprice numeric(
3,2),logdate date);
Query OK, 0 rows affected (0.39 sec)
```

```
mysql> Delimiter /
mysql> CREATE TRIGGER mytrig BEFORE UPDATE on Product
    -> FOR EACH ROW
    -> BEGIN
    -> INSERT into HISTORY values( OLD.pid, OLD.pname, OLD.supplier,
    -> OLD.UnitPrice, SYSDATE( ));
    -> END
    -> /
Query OK, 0 rows affected (0.19 sec)
```

```
mysql> update product set unitprice=5.5 where pname='halkefulke';/
Query OK, 1 row affected (0.08 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from history;
    -> /
+-----+------------+----------+-----------+------------+
| pid | pname      | supplier | unitprice | logdate    |
+-----+------------+----------+-----------+------------+
|  20 | halkefulke | haldiram |      2.00 | 2021-12-20 |
+-----+------------+----------+-----------+------------+
1 row in set (0.00 sec)
```