Lab cycle 1

Date :10/02/2025

Experiment No: 1

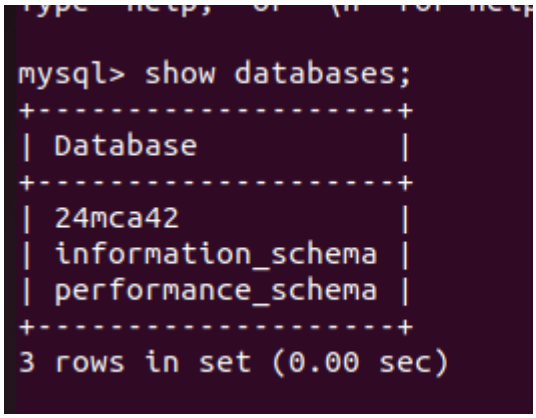
AIM : Familiarization of DDL Commands

Data Definition Language (DDL) - These SQL commands are used for creating, modifying, and dropping the structure of database objects. The commands are CREATE, ALTER, DROP, RENAME, and TRUNCATE.

A. Consider the database for a college. Write SQL commands to implement the following:

1.Create database

SQL : CREATE DATABASE college;



2.Select the current database.

SQL : USE college;

3.Create the following table:

a) Student (roll_no integer, name varchar, dob date, address text, phone_no varchar, blood_grp varchar)

SQL : CREATE TABLE student(roll_no int,name varchar(225),dob date,address text,phone_no varchar(225),blood_grp varchar(225));

b) Course (Course_id integer, Course_name varchar, course_duration integer)

SQL : CREATE TABLE course(course_id int,course_name varchar(225),course_duration int);

4.List all tables in the current database.

SQL : SHOW tables;

OUTPUT :

```
Database changed
mysql> show tables;
+--------------------+
| Tables_in_24mca42 |
+--------------------+
| course             |
| student            |
+--------------------+
2 rows in set (0.01 sec)
```

5.Display the structure of the Student table.

SQL : DESC student;

OUTPUT :

```
mysql> desc student;
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| roll_no   | int          | YES  |     | NULL    |       |
| name      | varchar(223) | YES  |     | NULL    |       |
| dob       | date         | YES  |     | NULL    |       |
| address   | text         | YES  |     | NULL    |       |
| phone_no  | varchar(223) | YES  |     | NULL    |       |
| blood_grp | varchar(223) | YES  |     | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
6 rows in set (0.00 sec)
```

6.Drop the column blood_grp from Student table.

SQL : ALTER TABLE student DROP COLUMN blood_grp;

OUTPUT :

```
mysql> alter table student drop column blood_grp;
Query OK, 0 rows affected (0.24 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc student;
+----------+--------------+------+-----+---------+-------+
| Field    | Type         | Null | Key | Default | Extra |
+----------+--------------+------+-----+---------+-------+
|          | nt           | YES  |     | NULL    |       |
|          | varchar(223) | YES  |     | NULL    |       |
| dob      | date         | YES  |     | NULL    |       |
| address  | text         | YES  |     | NULL    |       |
| phone_no | int          | YES  |     | NULL    |       |
| adar_no  | int          | YES  |     | NULL    |       |
+----------+--------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

LibreOffice Writer

7.Add a new column Adar_no with domain number to the table Student.

SQL : ALTER TABLE student ADD COLUMN adar_no int;

OUTPUT :

```
mysql> alter table student add column adar_no int;
Query OK, 0 rows affected (0.21 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc student;
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| roll_no   | int          | YES  |     | NULL    |       |
| name      | varchar(223) | YES  |     | NULL    |       |
| dob       | date         | YES  |     | NULL    |       |
| address   | text         | YES  |     | NULL    |       |
| phone_no  | varchar(223) | YES  |     | NULL    |       |
| blood_grp | varchar(223) | YES  |     | NULL    |       |
| adar_no   | int          | YES  |     | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
7 rows in set (0.00 sec)
```

8.Change the datatype of phone_no from varchar to int.

SQL : ALTER TABLE student MODIFY phone_no int;

OUTPUT :

```
mysql> alter table student modify phone_no int;
Image Viewer rows affected (0.83 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc student;
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| roll_no   | int          | YES  |     | NULL    |       |
| name      | varchar(223) | YES  |     | NULL    |       |
| dob       | date         | YES  |     | NULL    |       |
| address   | text         | YES  |     | NULL    |       |
| phone_no  | int          | YES  |     | NULL    |       |
| blood_grp | varchar(223) | YES  |     | NULL    |       |
| adar_no   | int          | YES  |     | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
7 rows in set (0.00 sec)
```

9. Drop the tables.

SQL : DROP TABLE student;

10.Delete the database.

SQL : DROP DATABASE college;

B. Consider the database for an organization. Write SQL commands to implement the following:

1. Create a database

SQL : CREATE DATABASE organization;

2. Select the current database

SQL : USE organization;

3. Create the following tables:

a) Employee (emp_no varchar, emp_name varchar, dob date, address text, mobile_no integer, dept_no varchar, salary integer)

SQL : CREATE TABLE employee(emp_no varchar(225),emp_name varchar(225),dob date,address text,mobile_no int,dep_no varchar,salary int);

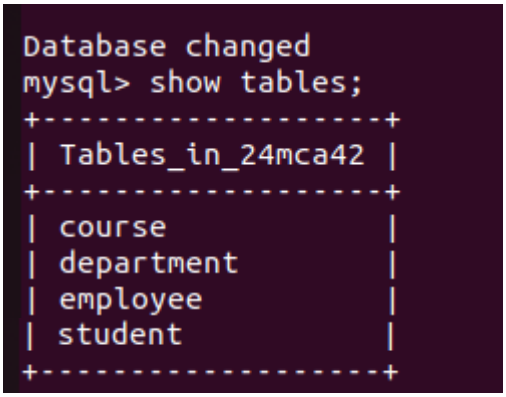
b) Department (dept_no varchar, dept_name varchar, location varchar)

SQL : CREATE TABLE department(dep_no varchar(223),dep_name varchar(223),location varchar(223));

4. List all tables in the current database.

SQL : SHOW tables;

OUTPUT :



5. Display the structure of the Employee table and Department table.

SQL : DESC employee;

OUTPUT :

```
mysql> create table employee(emp_no  varchar(224),emp_name varchar(223),dob date,address text,mobile_no int,dep_no varchar(223),salary int);
Query OK, 0 rows affected (0.31 sec)

mysql> desc employee;
+----------+--------------+------+-----+---------+-------+
| Field    | Type         | Null | Key | Default | Extra |
+----------+--------------+------+-----+---------+-------+
| emp_no   | varchar(224) | YES  |     | NULL    |       |
| emp_name | varchar(223) | YES  |     | NULL    |       |
| dob      | date         | YES  |     | NULL    |       |
| address  | text         | YES  |     | NULL    |       |
| mobile_no | int         | YES  |     | NULL    |       |
| dep_no   | varchar(223) | YES  |     | NULL    |       |
| salary   | int          | YES  |     | NULL    |       |
+----------+--------------+------+-----+---------+-------+
7 rows in set (0.00 sec)
```

SQL : DESC department;

OUTPUT :

```
mysql> create table department(dep_no  varchar(224),dep_name varchar(223),location varchar(223));
Query OK, 0 rows affected (0.31 sec)

mysql> desc department;
+----------+--------------+------+-----+---------+-------+
| Field    | Type         | Null | Key | Default | Extra |
+----------+--------------+------+-----+---------+-------+
| dep_no   | varchar(224) | YES  |     | NULL    |       |
| dep_name | varchar(223) | YES  |     | NULL    |       |
| location | varchar(223) | YES  |     | NULL    |       |
+----------+--------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

6. Add a new column ‘Designation’ to the table Employee.

SQL : ALTER TABLE employee ADD COLUMN designation varchar(223);

DESC employee;

OUTPUT :

```
mysql> alter table employee add column designation varchar(223);
Query OK, 0 rows affected (0.20 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc employee;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| emp_no      | varchar(224) | YES  |     | NULL    |       |
| emp_name    | varchar(223) | YES  |     | NULL    |       |
| dob         | date        | YES  |     | NULL    |       |
| address     | text        | YES  |     | NULL    |       |
| mobile_no   | int         | YES  |     | NULL    |       |
| dep_no      | varchar(223) | YES  |     | NULL    |       |
| salary      | int         | YES  |     | NULL    |       |
| designation | varchar(223) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
8 rows in set (0.01 sec)
```

7. Drop the column ‘location’ from Department table.

SQL : ALTER TABLE department DROP COLUMN location;

DESC department;

OUTPUT :

```
mysql> desc department;
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| dep_no   | varchar(224) | YES  |     | NULL    |       |
| dep_name | varchar(223) | YES  |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

Lab cycle 1

Date :10/02/2025

Experiment No: 2

AIM: Familiarization of SQL Constraints.

1. Create new table Persons with attributes PersonID (integer, PRIMARY KEY), Name (varchar , NOT NULL), Aadhar (Number, NOT NULL, UNIQUE), Age (integer , CHECK>18).

SQL : CREATE TABLE person(personID int PRIMARY KEY, name varchar(223) NOT NULL,aadhar int NOT NULL,aadhar int NOT NULL UNIQUE,age int check(age>18));

2. CREATE TABLE Orders with attributes OrderID (PRIMARY KEY),OrderNumber(NOT NULL) and PersonID(set FOREIGN KEY on attribute PersonID referencing the column PersonId of Person table)

SQL : CREATE TABLE orders(orderID int PRIMARY KEY,ordernumber int NOT NULL,personID int,FOREIGN KEY(personID) REFERENCES person(personID));

3. Display the structure of Persons tables.

SQL : DESC persons;

OUTPUT :

```
mysql> create table persons(personID int PRIMARY KEY,name varchar(223) NOT NULL,aadhar int NOT NULL UNIQUE,age int check(age>18));
Query OK, 0 rows affected (0.35 sec)

mysql> desc persons;
+----------+--------------+------+-----+---------+-------+
| Field    | Type         | Null | Key | Default | Extra |
+----------+--------------+------+-----+---------+-------+
| personID | int          | NO   | PRI | NULL    |       |
| name     | varchar(223) | NO   |     | NULL    |       |
| aadhar   | int          | NO   | UNI | NULL    |       |
| age      | int          | YES  |     | NULL    |       |
+----------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

4. Display the structure of Orders tables.

SQL : DESC orders;

OUTPUT :

```
mysql> create table orders(orderID int PRIMARY KEY,ordernumber int  NOT NULL,personID int,FOREIGN KEY(personID) REFERENCES persons(personID));
Query OK, 0 rows affected (0.76 sec)

mysql> desc orders;
+------------+------+------+-----+---------+-------+
| Field      | Type | Null | Key | Default | Extra |
+------------+------+------+-----+---------+-------+
| orderID    | int  | NO   | PRI | NULL    |       |
| ordernumber | int  | NO   |     | NULL    |       |
| personID   | int  | YES  | MUL | NULL    |       |
+------------+------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

5. Add emp_no as the primary key of the table Employee.

SQL : ALTER TABLE employee MODIFY emp_no varchar(223) PRIMARY KEY;

DESC employee;

OUTPUT :

```
mysql> alter table employee modify emp_no varchar(223) primary key;
Query OK, 0 rows affected (0.70 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc employee;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| emp_no      | varchar(223) | NO   | PRI | NULL    |       |
| emp_name    | varchar(223) | YES  |     | NULL    |       |
| dob         | date         | YES  |     | NULL    |       |
| address     | text         | YES  |     | NULL    |       |
| mobile_no   | int          | YES  |     | NULL    |       |
| dep_no      | varchar(223) | YES  |     | NULL    |       |
| salary      | int          | YES  |     | NULL    |       |
| designation | varchar(223) | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
8 rows in set (0.00 sec)
```

6. Add dept_no as the primary key of the table Department.

SQL : ALTER TABLE department MODIFY dep_no varchar(224) PRIMARY KEY;

DESC department;

OUTPUT :

```
mysql> alter table department modify dep_no varchar(223) primary key;
Query OK, 0 rows affected (0.65 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc department;
+----------+--------------+------+-----+---------+-------+
| Field    | Type         | Null | Key | Default | Extra |
+----------+--------------+------+-----+---------+-------+
| dep_no   | varchar(223) | NO   | PRI | NULL    |       |
| dep_name | varchar(223) | YES  |     | NULL    |       |
+----------+--------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

7. Add dept_no in Employee table as the foreign key reference to the table Department with on delete cascade.

SQL : ALTER TABLE employee ADD constraint FK_department FOREIGN

KEY(dep_no) REFERENCE department(dep_no);

DESC employee;

OUTPUT :

```
mysql> alter table employee  add constraint FK_departmentno foreign key(dep_no) references department(dep_no);
Query OK, 0 rows affected (1.11 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc employee;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| emp_no      | varchar(223) | NO   | PRI | NULL    |       |
| emp_name    | varchar(223) | YES  |     | NULL    |       |
| dob         | date         | YES  |     | NULL    |       |
| address     | text         | YES  |     | NULL    |       |
| mobile_no   | int          | YES  |     | NULL    |       |
| dep_no      | varchar(223) | YES  | MUL | NULL    |       |
| salary      | int          | YES  |     | NULL    |       |
| designation | varchar(223) | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
8 rows in set (0.00 sec)

mysql> desc orders;
+-------------+------+------+-----+---------+-------+
| Field       | Type | Null | Key | Default | Extra |
+-------------+------+------+-----+---------+-------+
| orderID     | int  | NO   | PRI | NULL    |       |
| ordernumber | int  | NO   |     | NULL    |       |
| personID    | int  | YES  | MUL | NULL    |       |
+-------------+------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

8. Drop the primary key of the table Orders.

SQL : ALTER TABLE orders drop PRIMARY KEY;

DESC orders;

OUTPUT :

```
mysql> alter table orders drop primary key;
Query OK, 0 rows affected (1.81 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc orders;
+-------------+------+------+-----+---------+-------+
| Field       | Type | Null | Key | Default | Extra |
+-------------+------+------+-----+---------+-------+
| orderID     | int  | NO   |     | NULL    |       |
| ordernumber | int  | NO   |     | NULL    |       |
| personID    | int  | YES  | MUL | NULL    |       |
+-------------+------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

Lab cycle 1

Date :20/02/2025

Experiment No: 3

Familiarization of DML Commands.

1. Add at least 10 rows into the table Employee and Department.

SQL:

```
INSERT  INTO employees
VALUES ('emp01', 'Ancy', '2003-09-09', 'kerala',   '9946235795', 'D01',200000, 'manager');

INSERT INTO department
VALUES('D01', 'finance');
```

2. Display all the records from the above tables.

SQL:
```
SELECT *
FROM employee;
```

OUTPUT:

```
select * from employee' at line 1
mysql>  select * from employee;
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp_no | emp_name | dob        | address | mobile_no   | dep_no | salary | designation        |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp01  | Ancy     | 2003-09-09 | kerala  | 9946235798  | D01    | 200000 | manager            |
| emp02  | Ammu     | 2003-08-07 | Goa     | 9946235790  | D02    | 290000 | manager            |
| emp03  | Arun     | 2003-04-12 | shimla  | 9946555790  | D03    |   7000 | computer_assistant |
| emp04  | john     | 2003-04-12 | dubai   | 9946095790  | D01    |  20000 | computer_assistant |
| emp05  | sani     | 2003-04-14 | delhi   | 994633790   | D01    |   4000 | manager            |
| emp06  | anju     | 2003-04-10 | kerala  | 994643790   | D02    |   2500 | manager            |
| emp07  | zool     | 2003-08-12 | dubai   | 9946090890  | D09    | 150000 | computer_assistant |
| emp08  | piku     | 2003-08-11 | kerala  | 994603890   | D03    |   3000 | computer_assistant |
| emp09  | anamika  | 2003-08-23 | kerela  | 43343337453 | D2     | 500000 |  manager           |
| emp10  | osheen   | 2003-08-03 | kerela  | 4943937453  | D5     |   7000 |  manager           |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
10 rows in set (0.00 sec)
```

SQL:
```
SELECT *
FROM department;
```

OUTPUT:

```
mysql> select * from department;
+--------+----------+
| dep_no | dep_name |
+--------+----------+
| D01    | finance  |
| D02    | finance  |
| D03    | finance  |
| D04    | HR       |
| D05    | HR       |
| D06    | HR       |
| D07    | finance  |
| D08    | health   |
| D09    | health   |
| D10    | HR       |
+--------+----------+
10 rows in set (0.00 sec)
```

3. Display the emp_no and name of employees from department no ‘D02’.

SQL: SELECT emp_no,emp_name
FROM employee
WHERE dep_no='D02';

OUTPUT:

```
mysql> select emp_no,emp_name from employee where dep_no='D02';
+--------+----------+
| emp_no | emp_name |
+--------+----------+
| emp02  | Ammu     |
| emp06  | anju     |
+--------+----------+
2 rows in set (0.00 sec)
```

4. Display emp_no, emp_name , designation, deptno and salary of employees in the descending order of salary.

SQL: SELECT emp_no,emp_name,designation,dep_no,salary
FROM employee
ORDER BY salary DESC;

OUTPUT:

```
ysql> select emp_no,emp_name,designation,dep_no,salary from employee order by salary desc;
--------+----------+--------------------+--------+--------+
 emp_no | emp_name | designation        | dep_no | salary |
--------+----------+--------------------+--------+--------+
 emp09  | anamika  |  manager           | D2     | 500000 |
 emp02  | Ammu     | manager            | D02    | 290000 |
 emp01  | Ancy     | manager            | D01    | 200000 |
 emp07  | zool     | computer_assistant | D09    | 150000 |
 emp04  | john     | computer_assistant | D01    |  20000 |
 emp03  | Arun     | computer_assistant | D03    |   7000 |
 emp10  | osheen   |  manager           | D5     |   7000 |
 emp05  | sani     | manager            | D01    |   4000 |
 emp08  | piku     | computer_assistant | D03    |   3000 |
 emp06  | anju     | manager            | D02    |   2500 |
--------+----------+--------------------+--------+--------+
10 rows in set (0.00 sec)
```

5. Display the emp_no , name of employees whose salary is between 2000 and 5000

SQL: SELECT emp_no, emp_name
FROM employee
WHERE salary
BETWEEN 2000 AND 5000;

OUTPUT:

```
mysql> select emp_no,emp_name from employee where salary between 2000 and 5000;
+--------+----------+
| emp_no | emp_name |
+--------+----------+
| emp05  | sani     |
| emp06  | anju     |
| emp08  | piku     |
+--------+----------+
3 rows in set (0.00 sec)
```

6. Display the designations without duplicate values.

SQL: SELECT DISTINCT designation
FROM employee;

OUTPUT:

```
mysql> select distinct designation from employee;
+--------------------+
| designation        |
+--------------------+
| manager            |
| computer_assistant |
|  manager           |
+--------------------+
3 rows in set (0.01 sec)
```

7. Change the salary of employees to 45000 whose designation is 'Manager'.

SQL: UPDATE employee
SET salary = 45000
WHERE designation ='manager';

SELECT *
FROM employee;

OUTPUT:

```
mysql> update employee set salary = 45000 where designation='manager';
Query OK, 4 rows affected (0.06 sec)
Rows matched: 4  Changed: 4  Warnings: 0
```

```
mysql> ^C
mysql> update employee set salary = 45000 where designation=' manager';
Query OK, 2 rows affected (0.06 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> select * from employee;
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp_no | emp_name | dob        | address | mobile_no   | dep_no | salary | designation        |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp01  | Ancy     | 2003-09-09 | kerala  | 9946235798  | D01    |  45000 | manager            |
| emp02  | Ammu     | 2003-08-07 | Goa     | 9946235790  | D02    |  45000 | manager            |
| emp03  | Arun     | 2003-04-12 | shimla  | 9946555790  | D03    |   7000 | computer_assistant |
| emp04  | john     | 2003-04-12 | dubai   | 9946095790  | D01    |  20000 | computer_assistant |
| emp05  | sani     | 2003-04-14 | delhi   | 994633790   | D01    |  45000 | manager            |
| emp06  | anju     | 2003-04-10 | kerala  | 994643790   | D02    |  45000 | manager            |
| emp07  | zool     | 2003-08-12 | dubai   | 9946090890  | D09    | 150000 | computer_assistant |
| emp08  | piku     | 2003-08-11 | kerala  | 994603890   | D03    |   3000 | computer_assistant |
| emp09  | anamika  | 2003-08-23 | kerela  | 43343337453 | D2     |  45000 |  manager           |
| emp10  | osheen   | 2003-08-03 | kerela  | 4943937453  | D5     |  45000 |  manager           |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
10 rows in set (0.00 sec)
```

8. Change the mobile number of employees named John.

SQL: UPDATE employee
SET mobile_no='12345678910'
WHERE emp_name='john';

SELECT *
FROM employee;

OUTPUT:

```
mysql> update employee set mobile_no='1234567891' where emp_name='john';
Query OK, 1 row affected (0.07 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from employee;
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp_no | emp_name | dob        | address | mobile_no   | dep_no | salary | designation        |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp01  | Ancy     | 2003-09-09 | kerala  | 9946235798  | D01    |  45000 | manager            |
| emp02  | Ammu     | 2003-08-07 | Goa     | 9946235790  | D02    |  45000 | manager            |
| emp03  | Arun     | 2003-04-12 | shimla  | 9946555790  | D03    |   7000 | computer_assistant |
| emp04  | john     | 2003-04-12 | dubai   | 1234567891  | D01    |  20000 | computer_assistant |
| emp05  | sani     | 2003-04-14 | delhi   | 994633790   | D01    |  45000 | manager            |
| emp06  | anju     | 2003-04-10 | kerala  | 994643790   | D02    |  45000 | manager            |
| emp07  | zool     | 2003-08-12 | dubai   | 9946090890  | D09    | 150000 | computer_assistant |
| emp08  | piku     | 2003-08-11 | kerala  | 994603890   | D03    |   3000 | computer_assistant |
| emp09  | anamika  | 2003-08-23 | kerela  | 43343337453 | D2     |  45000 |  manager           |
| emp10  | osheen   | 2003-08-03 | kerela  | 4943937453  | D5     |  45000 |  manager           |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
10 rows in set (0.00 sec)
```

9. Delete all employees whose salary is equal to Rs.7000.

SQL: DELET FROM employee
WHERE salary ='7000';

SELECT *
FROM employee;

OUTPUT:

```
mysql> delete from employee where salary ='7000';
Query OK, 1 row affected (0.05 sec)

mysql> select * from employee;
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp_no | emp_name | dob        | address | mobile_no   | dep_no | salary | designation        |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp01  | Ancy     | 2003-09-09 | kerala  | 9946235798  | D01    |  45000 | manager            |
| emp02  | Ammu     | 2003-08-07 | Goa     | 9946235790  | D02    |  45000 | manager            |
| emp04  | john     | 2003-04-12 | dubai   | 1234567891  | D01    |  20000 | computer_assistant |
| emp05  | sani     | 2003-04-14 | delhi   | 994633790   | D01    |  45000 | manager            |
| emp06  | anju     | 2003-04-10 | kerala  | 994643790   | D02    |  45000 | manager            |
| emp07  | zool     | 2003-08-12 | dubai   | 9946090890  | D09    | 150000 | computer_assistant |
| emp08  | piku     | 2003-08-11 | kerala  | 994603890   | D03    |   3000 | computer_assistant |
| emp09  | anamika  | 2003-08-23 | kerela  | 43343337453 | D2     |  45000 |  manager           |
| emp10  | osheen   | 2003-08-03 | kerela  | 4943937453  | D5     |  45000 |  manager           |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
9 rows in set (0.00 sec)
```

10. Retrieve the name, mobile number of all employees whose name start with “A”.

SQL: SELECT emp_no,mobile_no
FROM employee
WHERE emp_name LIKE 'A%';

```
mysql> select emp_name,mobile_no from employee where emp_name like 'A%';
+----------+-------------+
| emp_name | mobile_no   |
+----------+-------------+
| Ancy     | 9946235798  |
| Ammu     | 9946235790  |
| anju     | 994643790   |
| anamika  | 43343337453 |
+----------+-------------+
4 rows in set (0.00 sec)
```

11. Display the details of the employee whose name has at least three characters and salary greater than 20000.

SQL: SELECT *
FROM employee
WHERE LENGTH(emp_name)>=3 AND salary > 20000;

OUTPUT:

```
mysql> select * from employee where length(emp_name)>=3 and salary >20000;
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp_no | emp_name | dob        | address | mobile_no   | dep_no | salary | designation        |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
| emp01  | Ancy     | 2003-09-09 | kerala  | 9946235798  | D01    |  45000 | manager            |
| emp02  | Ammu     | 2003-08-07 | Goa     | 9946235790  | D02    |  45000 | manager            |
| emp05  | sani     | 2003-04-14 | delhi   | 994633790   | D01    |  45000 | manager            |
| emp06  | anju     | 2003-04-10 | kerala  | 994643790   | D02    |  45000 | manager            |
| emp07  | zool     | 2003-08-12 | dubai   | 9946090890  | D09    | 150000 | computer_assistant |
| emp09  | anamika  | 2003-08-23 | kerela  | 43343337453 | D2     |  45000 |  manager           |
| emp10  | osheen   | 2003-08-03 | kerela  | 4943937453  | D5     |  45000 |  manager           |
+--------+----------+------------+---------+-------------+--------+--------+--------------------+
7 rows in set (0.00 sec)

mysql>
```

12. Display the details of employees with empid ‘emp1’, ‘emp2’ and ‘emp6’.

SQL: SELECT *
FROM employee
WHERE emp_no in ('emp01', 'emp02', 'emp06');

OUTPUT:

```
mysql> select * from employee where emp_no in ('emp01','emp02','emp06');
+--------+----------+------------+---------+------------+--------+--------+-------------+
| emp_no | emp_name | dob        | address | mobile_no  | dep_no | salary | designation |
+--------+----------+------------+---------+------------+--------+--------+-------------+
| emp01  | Ancy     | 2003-09-09 | kerala  | 9946235798 | D01    |  45000 | manager     |
| emp02  | Ammu     | 2003-08-07 | Goa     | 9946235790 | D02    |  45000 | manager     |
| emp06  | anju     | 2003-04-10 | kerala  | 994643790  | D02    |  45000 | manager     |
+--------+----------+------------+---------+------------+--------+--------+-------------+
3 rows in set (0.00 sec)
```

13. Display employee name and employee id of those who have salary between 120000 and 300000.

SQL: SELECT *
FROM employee
WHERE salary
BETWEEN 120000 AND 300000;

OUTPUT:

```
mysql> select * from employee where salary between 120000 and 300000;
+--------+----------+------------+---------+------------+--------+--------+---------------
| emp_no | emp_name | dob        | address | mobile_no  | dep_no | salary | designation
+--------+----------+------------+---------+------------+--------+--------+---------------
| emp07  | zool     | 2003-08-12 | dubai   | 9946090890 | D09    | 150000 | computer_assis
+--------+----------+------------+---------+------------+--------+--------+---------------
1 row in set (0.00 sec)
```

14. Display the details of employees whose designation is ‘Manager’ or ‘Computer Assistant’.

SQL: SELECT *
FROM employee
WHERE designation ='manager' OR 'computer_assistant';

OUTPUT:

```
mysql> select * from employee where designation='manager' or 'computer_assistant';
+--------+----------+------------+---------+------------+--------+--------+-------------+
| emp_no | emp_name | dob        | address | mobile_no  | dep_no | salary | designation |
+--------+----------+------------+---------+------------+--------+--------+-------------+
| emp01  | Ancy     | 2003-09-09 | kerala  | 9946235798 | D01    |  45000 | manager     |
| emp02  | Ammu     | 2003-08-07 | Goa     | 9946235790 | D02    |  45000 | manager     |
| emp05  | sani     | 2003-04-14 | delhi   | 994633790  | D01    |  45000 | manager     |
| emp06  | anju     | 2003-04-10 | kerala  | 994643790  | D02    |  45000 | manager     |
+--------+----------+------------+---------+------------+--------+--------+-------------+
4 rows in set, 1 warning (0.00 sec)
```

15. Displays how many employees work for each department.

SQL: SELECT dep_no,count(emp_no) As employee_count
FROM employee
GROUP BY dep_no;

```
mysql> select dep_no,count(emp_no)  As employee_count from employee group by dep_no;
+--------+----------------+
| dep_no | employee_count |
+--------+----------------+
| D01    |              3 |
| D02    |              2 |
| D03    |              1 |
| D09    |              1 |
| D2     |              1 |
| D5     |              1 |
+--------+----------------+
6 rows in set (0.00 sec)
```

16. Displays average salary of employees in each department.

SQL: SELECT dep_no,AVG(salary) As average_salary
FROM employee
GROUP BY dep_no;

OUTPUT:

```
mysql> select dep_no,AVG(salary) As average_salary from employee group by dep_no;
+--------+----------------+
| dep_no | average_salary |
+--------+----------------+
| D01    |     36666.6667 |
| D02    |     45000.0000 |
| D03    |      3000.0000 |
| D09    |    150000.0000 |
| D2     |     45000.0000 |
| D5     |     45000.0000 |
+--------+----------------+
6 rows in set (0.00 sec)
```

17. Displays total salary of employees in each department.

SQL: SELECT dep_no,sum(salary)
FROM employee
GROUP BY dep_no;

OUTPUT:

```
mysql> select dep_no,sum(salary) from employee group by dep_no;
+--------+-------------+
| dep_no | sum(salary) |
+--------+-------------+
| D01    |      110000 |
| D02    |       90000 |
| D03    |        3000 |
| D09    |      150000 |
| D2     |       45000 |
| D5     |       45000 |
+--------+-------------+
6 rows in set (0.00 sec)
```

18. Displays top and lower salary of employees in each department.

SQL: SELECT dep_no,Max(salary) As highest_salary,Min(salary) As Lowest_salary
FROM employee
GROUP BY dep_no;

OUTPUT:

```
mysql> SELECT dep_no,Max(salary) As highest_salary,Min(salary) As Lowest_salary
    ->           FROM employee
    ->           GROUP BY dep_no;
+--------+----------------+---------------+
| dep_no | highest_salary | Lowest_salary |
+--------+----------------+---------------+
| D01    |          45000 |         20000 |
| D02    |          45000 |         45000 |
| D03    |           3000 |          3000 |
| D09    |         150000 |        150000 |
| D2     |          45000 |         45000 |
| D5     |          45000 |         45000 |
+--------+----------------+---------------+
6 rows in set (0.00 sec)
```

19. Displays average salary of employees in all departments except department with epartment number ‘D05’.

SQL: SELECT dep_no,AVG(salary) As average_salary
FROM employee
WHERE dep_no!= 'D05'
GROUP BY dep_no;

OUTPUT:

```
ERROR 1054 (42S22): Unknown column 'D05' in 'where clause'
mysql> select dep_no,AVG(salary) As average_salary from employee where dep_no!='D05' group by dep_no;
+--------+----------------+
| dep_no | average_salary |
+--------+----------------+
| D01    |     36666.6667 |
| D02    |     45000.0000 |
| D03    |      3000.0000 |
| D09    |    150000.0000 |
| D2     |     45000.0000 |
| D5     |     45000.0000 |
+--------+----------------+
6 rows in set (0.00 sec)
```

20. Displays average salary of employees in all departments except department with department number ‘D01’ and average salary greater than 20000 in the ascending order of average salary.

SQL: SELECT dep_no,AVG(salary) As average_salary
FROM employee
WHERE dep_no!= 'D01'
GROUP BY dep_no
HAVING AVG(salary)>20000
ORDER BY AVG (salary) ASC;

OUTPUT:

```
mysql> select dep_no,Avg(salary) As average_salary from employee where dep_no!='D01' group by dep_no having Avg(salary)>20000 order by Avg(salary) Asc;
+--------+--------------+
| dep_no | average_salary |
+--------+--------------+
| D02    |    45000.0000 |
| D2     |    45000.0000 |
| D5     |    45000.0000 |
| D09    |   150000.0000 |
+--------+--------------+
4 rows in set (0.00 sec)
```

```
+---------+-----------------+
| dep_no | average_salary |
+---------+-----------------+
| D02     |      45000.0000 |
| D2      |      45000.0000 |
| D5      |      45000.0000 |
| D09     |     150000.0000 |
+---------+-----------------+
4 rows in set (0.00 sec)

mysql>
```