20MCA134 ADVANCED DBMS LAB
LAB CYCLE 1
Experiment No: 1
Familiarization of DDL Commands

AIM: 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.

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

1. Create a database:

ans: CREATE DATABASE college;

2. Select the current database

ans: USE college;

3. Create the following tables:

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

ans: create table student(roll_no int,name varchar(20),dob date,address text(255),ph_novarchar(10),blood_grp varchar(50));

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

ans: create table course(course_id int,course_name varchar(20),course_duration int);

```
mysql> desc course;
+-----------------+-------------+------+-----+---------+-------+
| Field           | Type        | Null | Key | Default | Extra |
+-----------------+-------------+------+-----+---------+-------+
| course_id       | int         | YES  |     | NULL    |       |
| course_name     | varchar(20) | YES  |     | NULL    |       |
| course_duration | int         | YES  |     | NULL    |       |
+-----------------+-------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

4. List all tables in the current database.

Ans: show databases;

```
mysql> show tables;
+--------------------+
| Tables_in_24mca44 |
+--------------------+
| course             |
| student            |
+--------------------+
2 rows in set (0.00 sec)
```

5. Display the structure of the Student table.

Ans: DESC student;

```
mysql> desc student;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| roll_no   | int         | YES  |     | NULL    |       |
| name      | varchar(20) | YES  |     | NULL    |       |
| dob       | date        | YES  |     | NULL    |       |
| address   | text        | YES  |     | NULL    |       |
| ph_no     | varchar(10) | YES  |     | NULL    |       |
| blood_grp | varchar(50) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

6. Drop the column blood_grp from Student table.

Ans: ALTER TABLE student DROP blood_grp;

```
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| roll_no | int         | YES  |     | NULL    |       |
| name    | varchar(20) | YES  |     | NULL    |       |
| dob     | date        | YES  |     | NULL    |       |
| address | text        | YES  |     | NULL    |       |
| ph_no   | varchar(10) | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

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

ans: ALTER TABLE student ADD Adhaar_no int(12);

```
mysql> desc student;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| roll_no   | int         | YES  |     | NULL    |       |
| name      | varchar(20) | YES  |     | NULL    |       |
| dob       | date        | YES  |     | NULL    |       |
| address   | text        | YES  |     | NULL    |       |
| ph_no     | varchar(10) | YES  |     | NULL    |       |
| Adhaar_no | int         | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
6 rows in set (0.00 sec)
```

8. Change the datatype of phone_no from varchar to int

ans: ALTER TABLE student MODIFY ph_no int(10);

```
mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| roll_no    | int         | YES  |     | NULL    |       |
| name       | varchar(20) | YES  |     | NULL    |       |
| dob        | date        | YES  |     | NULL    |       |
| address    | text        | YES  |     | NULL    |       |
| ph_no      | int         | YES  |     | NULL    |       |
| Aadhaar_no | int         | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

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

1. Create a database
   ans: CREATE DATABASE college;
2. Select the current database
   ans: USE college;
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)

      ans: create table employee(emp_no varchar(10),emp_name varchar(20),dob date,address text(255),ph_no varchar(10),dept_no varchar(50),salary int(20));

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

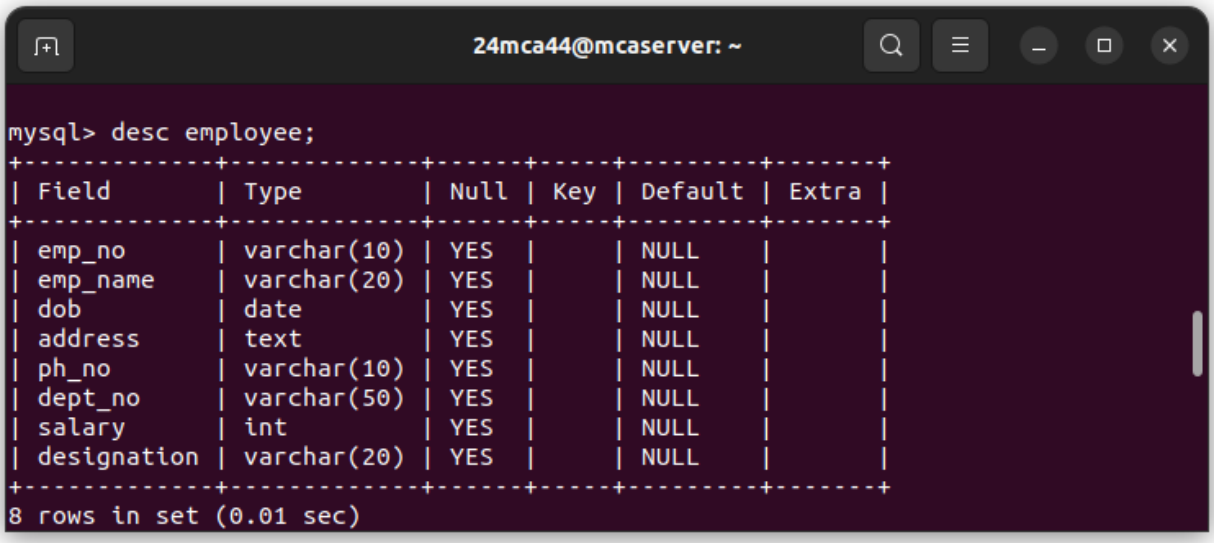
ans: create table department (dept_no varchar(10), dept_name varchar(20), location varchar(20));

4. List all tables in the current database.

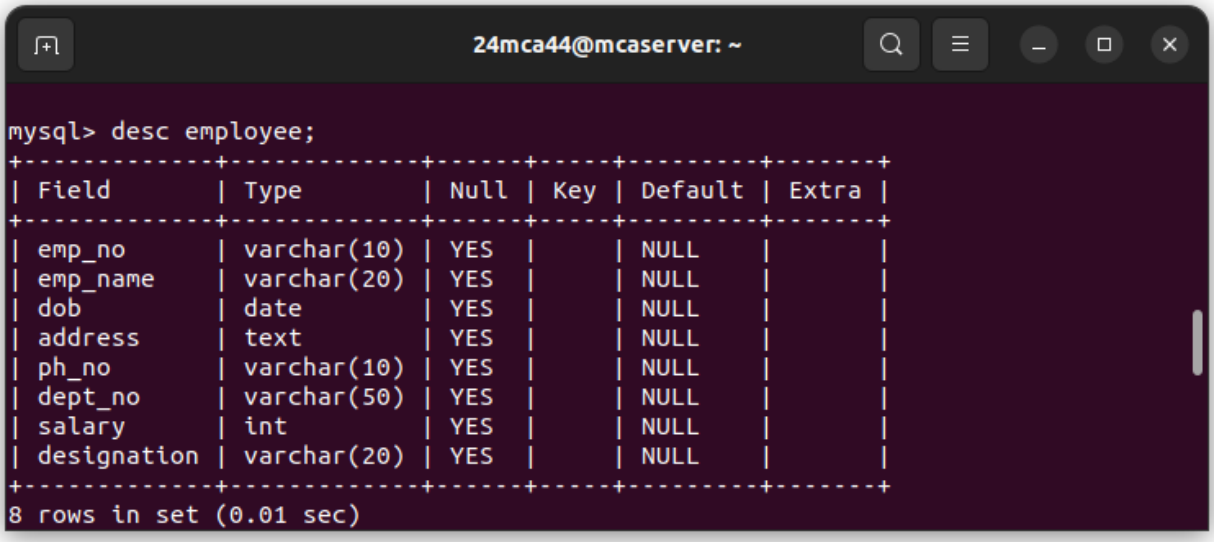
ans: show tables;

```
mysql> show tables;
+--------------------+
| Tables_in_24mca44 |
+--------------------+
| course             |
| department         |
| employee           |
| student            |
+--------------------+
4 rows in set (0.00 sec)
```

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



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



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

ANS: ALTER TABLE department DROP location;

```
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| dept_no   | varchar(10) | YES  |     | NULL    |       |
| dept_name | varchar(20) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

Experiment No: 2
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).

ANS: CREATE TABLE persons(pers_id int PRIMARY KEY,name varchar(20) NOT NULL,Aadhar INT(200) NOT NULL UNIQUE,age int CHECK(age>18));

```
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| pers_id | int         | NO   | PRI | NULL    |       |
| name    | varchar(20) | NO   |     | NULL    |       |
| Aadhar  | int         | NO   | UNI | NULL    |       |
| age     | int         | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
```

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)

ANS: CREATE TABLE orders(order_id int PRIMARY KEY,order_no INT(10) NOT NULL,pers_id INT NOT NULL,FOREIGN KEY(pers_id) REFERENCES persons(pers_id));

```
+----------+------+------+-----+---------+-------+
| Field    | Type | Null | Key | Default | Extra |
+----------+------+------+-----+---------+-------+
| order_id | int  | NO   | PRI | NULL    |       |
| order_no | int  | NO   |     | NULL    |       |
| pers_id  | int  | NO   | MUL | NULL    |       |
+----------+------+------+-----+---------+-------+
```

3. Display the structure of Persons tables.

ANS: DESC persons;

```
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| pers_id | int         | NO   | PRI | NULL    |       |
| name    | varchar(20) | NO   |     | NULL    |       |
| Aadhar  | int         | NO   | UNI | NULL    |       |
| age     | int         | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
```

4. Display the structure of Orders tables.

ANS: DESC orders;

```
+----------+------+------+-----+---------+-------+
| Field    | Type | Null | Key | Default | Extra |
+----------+------+------+-----+---------+-------+
| order_id | int  | NO   | PRI | NULL    |       |
| order_no | int  | NO   |     | NULL    |       |
| pers_id  | int  | NO   | MUL | NULL    |       |
+----------+------+------+-----+---------+-------+
```

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

ANS: ALTER TABLE employee MODIFY emp_no varchar(10) PRIMARY KEY;

```
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| emp_no      | varchar(10) | NO   | PRI | NULL    |       |
| emp_name    | varchar(20) | YES  |     | NULL    |       |
| dob         | date        | YES  |     | NULL    |       |
| address     | text        | YES  |     | NULL    |       |
| ph_no       | varchar(10) | YES  |     | NULL    |       |
| dept_no     | varchar(50) | YES  |     | NULL    |       |
| salary      | int         | YES  |     | NULL    |       |
| designation | varchar(20) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
```

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

ANS: ALTER TABLE department MODIFY dept_no varchar(10) PRIMARY KEY;
Query OK, 0 rows affected (0.69 sec)

```
mysql> desc department;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| dept_no   | varchar(10) | NO   | PRI | NULL    |       |
| dept_name | varchar(20) | 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.

ANS: ALTER TABLE employee ADD CONSTRAINT FK_dept_no FOREIGN KEY(dept_no) REFERENCES department(dept_no);

Field	Type	Null	Key	Default	Extra
emp_no	varchar(10)	NO	PRI	NULL	
emp_name	varchar(20)	YES		NULL	
dob	date	YES		NULL	
address	text	YES		NULL	
ph_no	varchar(10)	YES		NULL	
dept_no	varchar(50)	YES	MUL	NULL	
salary	int	YES		NULL	
designation	varchar(20)	YES		NULL	

8. Drop the primary key of the table Orders.

ANS: ALTER TABLE orders DROP PRIMARY KEY;

Field	Type	Null	Key	Default	Extra
order_id	int	NO		NULL	
order_no	int	NO		NULL	
pers_id	int	NO	MUL	NULL	

Experiment No: 3
Familiarization of DML Commands

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

ANS: INSERT INTO department(dept_no,dept_name) VALUES(1,MCA);

```
mysql> SELECT*FROM department
+---------+-----------+
| dept_no | dept_name |
+---------+-----------+
| 1       | MCA       |
| 10      | BIO TECH  |
| 2       | BTech     |
| 3       | BArch     |
| 4       | MTech     |
| 5       | BCA       |
| 6       | B.Com     |
| 7       | M.Com     |
| 8       | BBA       |
| 9       | MBA       |
+---------+-----------+
10 rows in set (0.00 sec)
```

ANS: INSERT INTO employee(emp_no,emp_name,dob,address,dept_no,salary)
VALUES('emp1','Alex','Asst.Proffessor','1')

```
mysql> SELECT*FROM employee;
+--------+----------+------------+--------------------+------------+---------+--------+-----------------+
| emp_no | emp_name | dob        | address            | ph_no      | dept_no | salary | designation     |
+--------+----------+------------+--------------------+------------+---------+--------+-----------------+
| emp1   | Alex     | 1990-01-12 | Dale 1, 2nd Street | 8547123695 | 1       |  75000 | Asst.Proffessor |
| emp10  | Rose     | 1999-10-20 | Dale 10, 2nd Street| 7366123656 | 10      |  80000 | Asst.Proffessor |
| emp2   | Alan     | 1991-02-11 | Dale 2, 2nd Street | 7545123695 | 2       |  70000 | Asst.Proffessor |
| emp3   | Ann      | 1992-03-13 | Dale 3, 2nd Street | 6565123693 | 3       |  65000 | Asst.Proffessor |
| emp4   | Sara     | 1993-04-14 | Dale 4, 2nd Street | 9465123693 | 4       |  60000 | Asst.Proffessor |
| emp5   | Sam      | 1994-05-15 | Dale 5, 2nd Street | 8466123699 | 5       |  60000 | Asst.Proffessor |
| emp6   | Samuel   | 1995-06-16 | Dale 6, 2nd Street | 6266123697 | 6       |  65000 | Asst.Proffessor |
| emp7   | John     | 1996-07-17 | Dale 7, 2nd Street | 9866123656 | 7       |  65000 | Asst.Proffessor |
| emp8   | Joseph   | 1997-08-18 | Dale 8, 2nd Street | 7066123656 | 8       |  70000 | Asst.Proffessor |
| emp9   | Jose     | 1998-09-19 | Dale 9, 2nd Street | 8566123656 | 9       |  70000 | Asst.Proffessor |
+--------+----------+------------+--------------------+------------+---------+--------+-----------------+
10 rows in set (0.00 sec)
```

2. Display all the records from the above tables

ANS: SELECT*FROM department;

```
mysql> SELECT*FROM department
+---------+-----------+
| dept_no | dept_name |
+---------+-----------+
| 1       | MCA       |
| 10      | BIO TECH  |
| 2       | BTech     |
| 3       | BArch     |
| 4       | MTech     |
| 5       | BCA       |
| 6       | B.Com     |
| 7       | M.Com     |
| 8       | BBA       |
| 9       | MBA       |
+---------+-----------+
10 rows in set (0.00 sec)
```

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

ANS: SELECT emp_no,emp_name FROM emplyee WHERE dept_no=’2’;

```
mysql> SELECT emp_no,emp_name FROM employee WHERE dept_no='2';
+--------+----------+
| emp_no | emp_name |
+--------+----------+
| emp2   | Alan     |
+--------+----------+
1 row in set (0.00 sec)
```

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

ANS: SELECT emp_no, emp_name FROM employee order by salary DESC;

```
mysql> SELECT emp_no,emp_name,designation,dept_no FROM employee order by salary DESC;
+--------+----------+-----------------+---------+
| emp_no | emp_name | designation     | dept_no |
+--------+----------+-----------------+---------+
| emp10  | Rose     | Asst.Proffessor | 10      |
| emp1   | Alex     | Asst.Proffessor | 1       |
| emp2   | Alan     | Asst.Proffessor | 2       |
| emp8   | Joseph   | Asst.Proffessor | 8       |
| emp9   | Jose     | Asst.Proffessor | 9       |
| emp3   | Ann      | Asst.Proffessor | 3       |
| emp6   | Samuel   | Asst.Proffessor | 6       |
| emp7   | John     | Asst.Proffessor | 7       |
| emp4   | Sara     | Asst.Proffessor | 4       |
| emp5   | Sam      | Asst.Proffessor | 5       |
+--------+----------+-----------------+---------+
10 rows in set (0.00 sec)
```

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

ANS: select emp_no, emp_name from employee where salary between 2000 and 5000;

```
mysql> select emp_no, emp_name from employee where salary between 2000 and 5000;
+--------+----------+
| emp_no | emp_name |
+--------+----------+
| emp10  | Rose     |
| emp2   | Alan     |
| emp5   | Sam      |
| emp6   | Samuel   |
+--------+----------+
4 rows in set (0.00 sec)
```

6. Display the designations without duplicate values

ANS: SELECT DISTINCT designation FROM employee;

```
mysql> SELECT DISTINCT designation FROM employee;
+-----------------+
| designation     |
+-----------------+
| MANAGER         |
| clerk           |
| Asst.Proffessor |
+-----------------+
3 rows in set (0.00 sec)
```

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

ANS: UPDATE TABLE employee set salary=’45000’ where designation=’Manager’;

```
mysql> select*from employee;
+--------+----------+------------+----------------------+------------+---------+--------+-----------------+
| emp_no | emp_name | dob        | address              | ph_no      | dept_no | salary | designation     |
+--------+----------+------------+----------------------+------------+---------+--------+-----------------+
| emp1   | Alex     | 1990-01-12 | Dale 1, 2nd Street   | 8547123695 | 1       |  45000 | MANAGER         |
| emp10  | Rose     | 1999-10-20 | Dale 10, 2nd Street  | 7366123656 | 10      |   4000 | clerk           |
| emp2   | Alan     | 1991-02-11 | Dale 2, 2nd Street   | 7545123695 | 2       |   3000 | clerk           |
| emp3   | Ann      | 1992-03-13 | Dale 3, 2nd Street   | 6565123693 | 3       |  65000 | Asst.Proffessor |
| emp4   | Sara     | 1993-04-14 | Dale 4, 2nd Street   | 9465123693 | 4       |  60000 | Asst.Proffessor |
| emp5   | Sam      | 1994-05-15 | Dale 5, 2nd Street   | 8466123699 | 5       |   3000 | clerk           |
| emp6   | Samuel   | 1995-06-16 | Dale 6, 2nd Street   | 6266123697 | 6       |   3000 | clerk           |
| emp7   | John     | 1996-07-17 | Dale 7, 2nd Street   | 9995445625 | 7       |  65000 | Asst.Proffessor |
| emp8   | Joseph   | 1997-08-18 | Dale 8, 2nd Street   | 7066123656 | 8       |   3000 | Asst.Proffessor |
+--------+----------+------------+----------------------+------------+---------+--------+-----------------+
9 rows in set (0.00 sec)
```

8. Change the mobile number of employees named John

ANS: UPDATE TABLE employee set ph_no=’9995445625’ where emp_name=’John’;

```
mysql> select*from employee;
+--------+----------+------------+----------------------+------------+---------+--------+-----------------+
| emp_no | emp_name | dob        | address              | ph_no      | dept_no | salary | designation     |
+--------+----------+------------+----------------------+------------+---------+--------+-----------------+
| emp1   | Alex     | 1990-01-12 | Dale 1, 2nd Street   | 8547123695 | 1       |  45000 | MANAGER         |
| emp10  | Rose     | 1999-10-20 | Dale 10, 2nd Street  | 7366123656 | 10      |   4000 | clerk           |
| emp2   | Alan     | 1991-02-11 | Dale 2, 2nd Street   | 7545123695 | 2       |   3000 | clerk           |
| emp3   | Ann      | 1992-03-13 | Dale 3, 2nd Street   | 6565123693 | 3       |  65000 | Asst.Proffessor |
| emp4   | Sara     | 1993-04-14 | Dale 4, 2nd Street   | 9465123693 | 4       |  60000 | Asst.Proffessor |
| emp5   | Sam      | 1994-05-15 | Dale 5, 2nd Street   | 8466123699 | 5       |   3000 | clerk           |
| emp6   | Samuel   | 1995-06-16 | Dale 6, 2nd Street   | 6266123697 | 6       |   3000 | clerk           |
| emp7   | John     | 1996-07-17 | Dale 7, 2nd Street   | 9995445625 | 7       |  65000 | Asst.Proffessor |
| emp8   | Joseph   | 1997-08-18 | Dale 8, 2nd Street   | 7066123656 | 8       |   3000 | Asst.Proffessor |
+--------+----------+------------+----------------------+------------+---------+--------+-----------------+
9 rows in set (0.00 sec)
```

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

ANS:DELETE FROM employee WHERE salary = ‘7000’;

```
+--------+----------+------------+----------------------+------------+---------+--------+-----------------+
| emp_no | emp_name | dob        | address              | ph_no      | dept_no | salary | designation     |
+--------+----------+------------+----------------------+------------+---------+--------+-----------------+
| emp1   | Alex     | 1990-01-12 | Dale 1, 2nd Street   | 8547123695 | 1       |  45000 | MANAGER         |
| emp10  | Rose     | 1999-10-20 | Dale 10, 2nd Street  | 7366123656 | 10      |   4000 | clerk           |
| emp2   | Alan     | 1991-02-11 | Dale 2, 2nd Street   | 7545123695 | 2       |   3000 | clerk           |
| emp3   | Ann      | 1992-03-13 | Dale 3, 2nd Street   | 6565123693 | 3       |  65000 | Asst.Proffessor |
| emp4   | Sara     | 1993-04-14 | Dale 4, 2nd Street   | 9465123693 | 4       |  60000 | Asst.Proffessor |
| emp5   | Sam      | 1994-05-15 | Dale 5, 2nd Street   | 8466123699 | 5       |   3000 | clerk           |
| emp6   | Samuel   | 1995-06-16 | Dale 6, 2nd Street   | 6266123697 | 6       |   3000 | clerk           |
| emp7   | John     | 1996-07-17 | Dale 7, 2nd Street   | 9995445625 | 7       |  65000 | Asst.Proffessor |
| emp8   | Joseph   | 1997-08-18 | Dale 8, 2nd Street   | 7066123656 | 8       |   3000 | Asst.Proffessor |
+--------+----------+------------+----------------------+------------+---------+--------+-----------------+
```

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

ANS: select emp_name,ph_no from employee where emp_name like ‘A%’;

```
mysql> select emp_name, ph_no from employee where emp_name like 'A%';
+----------+------------+
| emp_name | ph_no      |
+----------+------------+
| Alex     | 8547123695 |
| Alan     | 7545123695 |
| Ann      | 6565123693 |
+----------+------------+
3 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.

ANS: select * from employee where salary>20000 and emp_name like ‘___%’;

```
mysql> select * from employee where salary>20000 and emp_name like '___%';
+--------+----------+------------+--------------------+------------+---------+--------+-----------------+
| emp_no | emp_name | dob        | address            | ph_no      | dept_no | salary | designation     |
+--------+----------+------------+--------------------+------------+---------+--------+-----------------+
| emp1   | Alex     | 1990-01-12 | Dale 1, 2nd Street | 8547123695 | 1       |  45000 | MANAGER         |
| emp3   | Ann      | 1992-03-13 | Dale 3, 2nd Street | 6565123693 | 3       |  65000 | Asst.Proffessor |
| emp4   | Sara     | 1993-04-14 | Dale 4, 2nd Street | 9465123693 | 4       |  60000 | Asst.Proffessor |
| emp7   | John     | 1996-07-17 | Dale 7, 2nd Street | 9995445625 | 7       |  65000 | Asst.Proffessor |
+--------+----------+------------+--------------------+------------+---------+--------+-----------------+
4 rows in set (0.00 sec)
```

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

ANS: select * from employee where emp_no in (‘emp1’,’emp2’) ;

```
mysql> select * from employee where emp_no in ('emp1','emp2');
+--------+----------+------------+--------------------+------------+---------+--------+-------------+
| emp_no | emp_name | dob        | address            | ph_no      | dept_no | salary | designation |
+--------+----------+------------+--------------------+------------+---------+--------+-------------+
| emp1   | Alex     | 1990-01-12 | Dale 1, 2nd Street | 8547123695 | 1       |  45000 | MANAGER     |
| emp2   | Alan     | 1991-02-11 | Dale 2, 2nd Street | 7545123695 | 2       |   3000 | clerk       |
+--------+----------+------------+--------------------+------------+---------+--------+-------------+
2 rows in set (0.00 sec)
```

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

ANS: select emp_name,emp_no from employee where salary between 120000 and 300000;

```
mysql> select emp_name, emp_no from employee where salary between 120000 and 300000;
+----------+--------+
| emp_name | emp_no |
+----------+--------+
| Ann      | emp3   |
| John     | emp7   |
| Joseph   | emp8   |
+----------+--------+
3 rows in set (0.00 sec)
```

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

ANS: select * from employee where designation in (‘computer assistant’,’manager’);

```
mysql> select * from employee where designation in ('computer assistant','manager');
+--------+----------+------------+--------------------+------------+---------+--------+--------------------+
| emp_no | emp_name | dob        | address            | ph_no      | dept_no | salary | designation        |
+--------+----------+------------+--------------------+------------+---------+--------+--------------------+
| emp1   | Alex     | 1990-01-12 | Dale 1, 2nd Street | 8547123695 | 1       |  45000 | MANAGER            |
| emp4   | Sara     | 1993-04-14 | Dale 4, 2nd Street | 9465123693 | 4       |  13000 | computer assistant |
+--------+----------+------------+--------------------+------------+---------+--------+--------------------+
2 rows in set (0.00 sec)
```

15. Displays how many employees work for each department.

ANS: select cout(*), dept_name from department group by dept_name;

```
mysql> select count(*), dept_name from department group by dept_name;
+----------+-----------+
| count(*) | dept_name |
+----------+-----------+
|        1 | MCA       |
|        1 | BIO TECH  |
|        1 | BTech     |
|        1 | BArch     |
|        1 | MTech     |
|        1 | BCA       |
|        1 | B.Com     |
|        1 | M.Com     |
|        1 | BBA       |
|        1 | MBA       |
+----------+-----------+
10 rows in set (0.01 sec)
```

16. Displays average salary of employees in each department.

ANS: select avg(salary), dept_no from employee group by dept_no;

```
mysql> select avg(salary), dept_no from employee group by dept_no;
+-------------+---------+
| avg(salary) | dept_no |
+-------------+---------+
|  45000.0000 | 1       |
|   4000.0000 | 10      |
|   3000.0000 | 2       |
| 130000.0000 | 3       |
|  13000.0000 | 4       |
|   3000.0000 | 5       |
|   3000.0000 | 6       |
| 130000.0000 | 7       |
| 130000.0000 | 8       |
+-------------+---------+
9 rows in set (0.00 sec)
```

17. Displays total salary of employees in each department

ANS: select sum(salary), dept_no from employee group by dept_no;

```
mysql> select sum(salary), dept_no from employee group by dept_no;
+-------------+---------+
| sum(salary) | dept_no |
+-------------+---------+
|       45000 | 1       |
|        4000 | 10      |
|        3000 | 2       |
|      130000 | 3       |
|       13000 | 4       |
|        3000 | 5       |
|        3000 | 6       |
|      130000 | 7       |
|      130000 | 8       |
+-------------+---------+
9 rows in set (0.00 sec)
```

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

ANS: select max(salary),min(salary),dept_no from employee group by dept_no;

```
mysql> select max(salary),min(salary),dept_no from employee group by dept_no;
+-------------+-------------+---------+
| max(salary) | min(salary) | dept_no |
+-------------+-------------+---------+
|       45000 |       45000 | 1       |
|        4000 |        4000 | 10      |
|        3000 |        3000 | 2       |
|      130000 |      130000 | 3       |
|       13000 |       13000 | 4       |
|        3000 |        3000 | 5       |
|        3000 |        3000 | 6       |
|      130000 |      130000 | 7       |
|      130000 |      130000 | 8       |
+-------------+-------------+---------+
9 rows in set (0.00 sec)
```

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

ANS: select avg(salary), dept_no from employee where dept_no <> ‘5’ group by dept_no;

```
mysql> select avg(salary), dept_no from employee where dept_no <> '5' group by dept_no;
+-------------+---------+
| avg(salary) | dept_no |
+-------------+---------+
|  45000.0000 | 1       |
|   4000.0000 | 10      |
|   3000.0000 | 2       |
| 130000.0000 | 3       |
|  13000.0000 | 4       |
|   3000.0000 | 6       |
| 130000.0000 | 7       |
| 130000.0000 | 8       |
+-------------+---------+
8 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.

ANS: select avg(salary), dept_no from employee group by dept_no having dept_no!=’5’ and avg(salary)>20000 order by avg(salary) ASC;

```
mysql> select avg(salary), dept_no from employee group by dept_no having dept_no!='5' and avg(salary)>20000 order by avg(salary) ASC;
+-------------+---------+
| avg(salary) | dept_no |
+-------------+---------+
|  45000.0000 | 1       |
| 130000.0000 | 3       |
| 130000.0000 | 7       |
| 130000.0000 | 8       |
+-------------+---------+
4 rows in set (0.01 sec)
```