Course Outcome 1

Design and build a simple relational database system and demonstrate competence with the fundamentals tasks involved with modelling, designing and implementing a database.

PROGRAM 1

AIM: Creation a database, S2MCA and Two tables, Department and Employee using DDL commands and perform insertion , selection using where condition , logical operation using DML commands.

1.Creates a new database named S1MCA.

```
MariaDB [(none)]> CREATE DATABASE S2MCA;
Query OK, 1 row affected (0.002 sec)
MariaDB [S2MCA]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| phpmyadmin         |
| s1mca              |
| s2mca              |
| s3mca              |
| student            |
| test               |
+--------------------+
```

2.Selects a specific database to perform operations within.

```
MariaDB [s1mca]> use S2MCA;
Database changed
```

3.Create a table to store employee information with primary key and foreign key constraints.

```
MariaDB [S2MCA]> create table Employee(employee_id int primary key,first_name
varchar(50),last_name varchar(50),department_id int,salary decimal(10,2),foreign key (department_id)
references Department(department_id));
Query OK, 0 rows affected (0.049 sec)
```

4. Create a table to store department information with primary key .

```
MariaDB [S2MCA]> create table Department(department_id int primary
key,department_namevarchar(50));
Query OK, 0 rows affected (0.013 sec)
```

5. Insert data into the department table.

```
MariaDB [S2MCA]> insert into Department
values(1,'HR'),(2,'Finance'),(3,'IT'),(4,'Marketing'),(5,'Operations'),(6,'Sales'),(7,'Research'),(8,'Engineer
ing'),(9,'Customer Service'),(10,'Administration'),(11,'Logistics'),(12,'Quality
Control'),(13,'Production'),(14,'Distribution'),(15,'Legal'),(16,'Purchasing'),(17,'Public
Relations'),(18,'Advertising'),(19,'Human Resources'),(20,'Information Technology');
Query OK, 20 rows affected (0.006 sec)
Records: 20 Duplicates: 0 Warnings: 0
```

6. Insert data into the employee table.

```
MariaDB [S2MCA]> insert into Employee
values(1,'John','Doe',1,50000.00),(2,'Jane','Smith',2,60000.00),(3,'Alice','Johnson',3,70000.00),(4,'Bob','
Williams',1,55000.00),(5,'Sarah','Lee',4,62000.00),(6,'Michael','Brown',3,72000.00),(7,'Lisa','Taylor',2,
65000.00),(8,'Kevin','Clark',1,58000.00),(9,'Amanda','Martinez',4,60000.00),(10,'Eric','Anderson,3,750
00.00),(11,'Emily','Wilson',2,58000.00),(12,'Ryan','Garcia',3,67000.00),(13,'Samantha','Martinez',1,560
00.00),(14,'David','Lee',4,64000.00),(15,'Jessica','Brown',3,69000.00),(16,'Andrew','Johnson',2,62000.0
0),(17,'Lauren','White',1,57000.00),(18,'Christopher','Lopez',4,61000.00),(19,'Kimberly','Young',3,7300
0.00),(20,'Mathew','Hall',2,64000.00);
Query OK, 20 rows affected (0.008 sec)
Records: 20 Duplicates: 0 Warnings: 0
```

7. Use DISTINCT to select unique department names.

```
MariaDB [S2MCA]> select distinct department_name from Department;
+------------------------+
| department_name        |
+------------------------+
| HR                     |
| Finance                |
| IT                     |
| Marketing              |
| Operations             |
| Sales                  |
| Research               |
| Engineering            |
| Customer Service       |
| Administration         |
| Logistics              |
| Quality                |
|  Control               |
| Production             |
| Distribution           |
| Legal                  |
| Purchasing             |
| Public                 |
| Relations              |
| Advertising            |
| Human Resources        |
| Information Technology |
+------------------------+
```

8. Select all columns from the employee table.

```
MariaDB [S2MCA]> select * from Employee;
+-------------+-------------+-----------+---------------+----------+
| employee_id | first_name  | last_name | department_id | salary   |
+-------------+-------------+-----------+---------------+----------+
|           1 | John        | Doe       |             1 | 50000.00 |
|           2 | Jane        | Smith     |             2 | 60000.00 |
|           3 | Alice       | Johnson   |             3 | 70000.00 |
|           4 | Bob         | Williams  |             1 | 55000.00 |
|           5 | Sarah       | Lee       |             4 | 62000.00 |
|           6 | Michael     | Brown     |             3 | 72000.00 |
|           7 | Lisa        | Taylor    |             2 | 65000.00 |
|           8 | Kevin       | Clark     |             1 | 58000.00 |
|           9 | Amanda      | Martinez  |             4 | 60000.00 |
|          10 | Eric        | Anderson  |             3 | 75000.00 |
|          11 | Emily       | Wilson    |             2 | 58000.00 |
|          12 | Ryan        | Garcia    |             3 | 67000.00 |
|          13 | Samantha    | Martinez  |             1 | 56000.00 |
|          14 | David       | Lee       |             4 | 64000.00 |
|          15 | Jessica     | Brown     |             3 | 69000.00 |
|          16 | Andrew      | Johnson   |             2 | 62000.00 |
|          17 | Lauren      | White     |             1 | 57000.00 |
|          18 | Christopher | Lopez     |             4 | 61000.00 |
|          19 | Kimberly    | Young     |             3 | 73000.00 |
|          20 | Mathew      | Hall      |             2 | 64000.00 |
+-------------+-------------+-----------+---------------+----------+
```

9. Select specific columns (first_name, last_name) from the employee table.

```
MariaDB [S2MCA]> select first_name,last_name from Employee;
+-------------+-----------+
| first_name  | last_name |
+-------------+-----------+
| John        | Doe       |
| Jane        | Smith     |
| Alice       | Johnson   |
| Bob         | Williams  |
| Sarah       | Lee       |
| Michael     | Brown     |
| Lisa        | Taylor    |
| Kevin       | Clark     |
| Amanda      | Martinez  |
| Eric        | Anderson  |
| Emily       | Wilson    |
| Ryan        | Garcia    |
| Samantha    | Martinez  |
| David       | Lee       |
| Jessica     | Brown     |
| Andrew      | Johnson   |
| Lauren      | White     |
| Christopher | Lopez     |
| Kimberly    | Young     |
| Mathew      | Hall      |
+-------------+-----------+
```

10. Select employees earning more than $60,000.

```
MariaDB [S2MCA]> select * from Employee where salary>60000;
+-------------+-------------+-----------+---------------+----------+
| employee_id | first_name  | last_name | department_id | salary   |
+-------------+-------------+-----------+---------------+----------+
|           3 | Alice       | Johnson   |             3 | 70000.00 |
|           5 | Sarah       | Lee       |             4 | 62000.00 |
|           6 | Michael     | Brown     |             3 | 72000.00 |
|           7 | Lisa        | Taylor    |             2 | 65000.00 |
|          10 | Eric        | Anderson  |             3 | 75000.00 |
|          12 | Ryan        | Garcia    |             3 | 67000.00 |
|          14 | David       | Lee       |             4 | 64000.00 |
|          15 | Jessica     | Brown     |             3 | 69000.00 |
|          16 | Andrew      | Johnson   |             2 | 62000.00 |
|          18 | Christopher | Lopez     |             4 | 61000.00 |
|          19 | Kimberly    | Young     |             3 | 73000.00 |
|          20 | Mathew      | Hall      |             2 | 64000.00 |
+-------------+-------------+-----------+---------------+----------+
```

11. Select employees in the HR department(dept_id =1)

```
+-------------+------------+-----------+---------------+----------+
| employee_id | first_name | last_name | department_id | salary   |
+-------------+------------+-----------+---------------+----------+
|           1 | John       | Doe       |             1 | 50000.00 |
|           4 | Bob        | Williams  |             1 | 55000.00 |
|           8 | Kevin      | Clark     |             1 | 58000.00 |
|          13 | Samantha   | Martinez  |             1 | 56000.00 |
|          17 | Lauren     | White     |             1 | 57000.00 |
+-------------+------------+-----------+---------------+----------+
```

12. Add a new column (hired) to the employee table and insert values.

```
MariaDB [S2MCA]> alter table Employee add hired Date;
Query OK, 0 rows affected (0.073 sec)
Records: 0 Duplicates: 0 Warnings: 0
MariaDB [S2MCA]> update Employee set hired='2023-05-12' where department_id=1;
Query OK, 5 rows affected (0.003 sec)
Rows matched: 5 Changed: 5 Warnings: 0
MariaDB [S2MCA]> update Employee set hired='2012-02-10' where department_id=2;
Query OK, 5 rows affected (0.003 sec)
Rows matched: 5 Changed: 5 Warnings: 0
MariaDB [S2MCA]> update Employee set hired='2002-01-18' where department_id=3;
Query OK, 6 rows affected (0.004 sec)
Rows matched: 6 Changed: 6 Warnings: 0
MariaDB [S2MCA]> update Employee set hired='2023-05-12' where department_id=4;
Query OK, 4 rows affected (0.005 sec)
Rows matched: 4 Changed: 4 Warnings: 0
```

```
MariaDB [S2MCA]> select * from Employee;
+-------------+-------------+-----------+---------------+----------+------------+
| employee_id | first_name  | last_name | department_id | salary   | hired      |
+-------------+-------------+-----------+---------------+----------+------------+
|           1 | John        | Doe       |             1 | 50000.00 | 2023-05-12 |
|           2 | Jane        | Smith     |             2 | 60000.00 | 2012-02-10 |
|           3 | Alice       | Johnson   |             3 | 70000.00 | 2002-01-18 |
|           4 | Bob         | Williams  |             1 | 55000.00 | 2023-05-12 |
|           5 | Sarah       | Lee       |             4 | 62000.00 | 2023-05-12 |
|           6 | Michael     | Brown     |             3 | 72000.00 | 2002-01-18 |
|           7 | Lisa        | Taylor    |             2 | 65000.00 | 2012-02-10 |
|           8 | Kevin       | Clark     |             1 | 58000.00 | 2023-05-12 |
|           9 | Amanda      | Martinez  |             4 | 60000.00 | 2023-05-12 |
|          10 | Eric        | Anderson  |             3 | 75000.00 | 2002-01-18 |
|          11 | Emily       | Wilson    |             2 | 58000.00 | 2012-02-10 |
|          12 | Ryan        | Garcia    |             3 | 67000.00 | 2002-01-18 |
|          13 | Samantha    | Martinez  |             1 | 56000.00 | 2023-05-12 |
|          14 | David       | Lee       |             4 | 64000.00 | 2023-05-12 |
|          15 | Jessica     | Brown     |             3 | 69000.00 | 2002-01-18 |
|          16 | Andrew      | Johnson   |             2 | 62000.00 | 2012-02-10 |
|          17 | Lauren      | White     |             1 | 57000.00 | 2023-05-12 |
|          18 | Christopher | Lopez     |             4 | 61000.00 | 2023-05-12 |
|          19 | Kimberly    | Young     |             3 | 73000.00 | 2002-01-18 |
|          20 | Mathew      | Hall      |             2 | 64000.00 | 2012-02-10 |
+-------------+-------------+-----------+---------------+----------+------------+
```

13. Select employees hired after January 1, 2023.

```
MariaDB [S2MCA]> select * from Employee where hired>"2023-01-01";
+-------------+-------------+-----------+---------------+----------+------------+
| employee_id | first_name  | last_name | department_id | salary   | hired      |
+-------------+-------------+-----------+---------------+----------+------------+
|           1 | John        | Doe       |             1 | 50000.00 | 2023-05-12 |
|           4 | Bob         | Williams  |             1 | 55000.00 | 2023-05-12 |
|           5 | Sarah       | Lee       |             4 | 62000.00 | 2023-05-12 |
|           8 | Kevin       | Clark     |             1 | 58000.00 | 2023-05-12 |
|           9 | Amanda      | Martinez  |             4 | 60000.00 | 2023-05-12 |
|          13 | Samantha    | Martinez  |             1 | 56000.00 | 2023-05-12 |
|          14 | David       | Lee       |             4 | 64000.00 | 2023-05-12 |
|          17 | Lauren      | White     |             1 | 57000.00 | 2023-05-12 |
|          18 | Christopher | Lopez     |             4 | 61000.00 | 2023-05-12 |
+-------------+-------------+-----------+---------------+----------+------------+
```

14. Write a query to retrieve all employees with a salary greater than $50,000.

```
MariaDB [S2MCA]> select * from Employee where salary>50000;
+-------------+-------------+-----------+---------------+----------+------------+
| employee_id | first_name  | last_name | department_id | salary   | hired      |
+-------------+-------------+-----------+---------------+----------+------------+
|           2 | Jane        | Smith     |             2 | 60000.00 | 2012-02-10 |
|           3 | Alice       | Johnson   |             3 | 70000.00 | 2002-01-18 |
|           4 | Bob         | Williams  |             1 | 55000.00 | 2023-05-12 |
|           5 | Sarah       | Lee       |             4 | 62000.00 | 2023-05-12 |
|           6 | Michael     | Brown     |             3 | 72000.00 | 2002-01-18 |
|           7 | Lisa        | Taylor    |             2 | 65000.00 | 2012-02-10 |
|           8 | Kevin       | Clark     |             1 | 58000.00 | 2023-05-12 |
|           9 | Amanda      | Martinez  |             4 | 60000.00 | 2023-05-12 |
|          10 | Eric        | Anderson  |             3 | 75000.00 | 2002-01-18 |
|          11 | Emily       | Wilson    |             2 | 58000.00 | 2012-02-10 |
|          12 | Ryan        | Garcia    |             3 | 67000.00 | 2002-01-18 |
|          13 | Samantha    | Martinez  |             1 | 56000.00 | 2023-05-12 |
|          14 | David       | Lee       |             4 | 64000.00 | 2023-05-12 |
|          15 | Jessica     | Brown     |             3 | 69000.00 | 2002-01-18 |
|          16 | Andrew      | Johnson   |             2 | 62000.00 | 2012-02-10 |
|          17 | Lauren      | White     |             1 | 57000.00 | 2023-05-12 |
|          18 | Christopher | Lopez     |             4 | 61000.00 | 2023-05-12 |
|          19 | Kimberly    | Young     |             3 | 73000.00 | 2002-01-18 |
|          20 | Mathew      | Hall      |             2 | 64000.00 | 2012-02-10 |
+-------------+-------------+-----------+---------------+----------+------------+
```

15.Find all employees whose department ID is not equal to 3.

```
MariaDB [S2MCA]> select * from Employee where department_id!=3;
+-------------+-------------+-----------+---------------+----------+------------+
| employee_id | first_name  | last_name | department_id | salary   | hired      |
+-------------+-------------+-----------+---------------+----------+------------+
|           1 | John        | Doe       |             1 | 50000.00 | 2023-05-12 |
|           2 | Jane        | Smith     |             2 | 60000.00 | 2012-02-10 |
|           4 | Bob         | Williams  |             1 | 55000.00 | 2023-05-12 |
|           5 | Sarah       | Lee       |             4 | 62000.00 | 2023-05-12 |
|           7 | Lisa        | Taylor    |             2 | 65000.00 | 2012-02-10 |
|           8 | Kevin       | Clark     |             1 | 58000.00 | 2023-05-12 |
|           9 | Amanda      | Martinez  |             4 | 60000.00 | 2023-05-12 |
|          11 | Emily       | Wilson    |             2 | 58000.00 | 2012-02-10 |
|          13 | Samantha    | Martinez  |             1 | 56000.00 | 2023-05-12 |
|          14 | David       | Lee       |             4 | 64000.00 | 2023-05-12 |
|          16 | Andrew      | Johnson   |             2 | 62000.00 | 2012-02-10 |
|          17 | Lauren      | White     |             1 | 57000.00 | 2023-05-12 |
|          18 | Christopher | Lopez     |             4 | 61000.00 | 2023-05-12 |
|          20 | Mathew      | Hall      |             2 | 64000.00 | 2012-02-10 |
+-------------+-------------+-----------+---------------+----------+------------+
```

16.Retrieve employees with a salary greater than $50,000 and who belong to department ID 2.

```
MariaDB [S2MCA]> select * from Employee where salary>50000 and department_id=2;
+-------------+------------+-----------+---------------+----------+------------+
| employee_id | first_name | last_name | department_id | salary   | hired      |
+-------------+------------+-----------+---------------+----------+------------+
|           2 | Jane       | Smith     |             2 | 60000.00 | 2012-02-10 |
|           7 | Lisa       | Taylor    |             2 | 65000.00 | 2012-02-10 |
|          11 | Emily      | Wilson    |             2 | 58000.00 | 2012-02-10 |
|          16 | Andrew     | Johnson   |             2 | 62000.00 | 2012-02-10 |
|          20 | Mathew     | Hall      |             2 | 64000.00 | 2012-02-10 |
+-------------+------------+-----------+---------------+----------+------------+
```

17. Select all employees who belong to department ID 1 or 2.

```
MariaDB [S2MCA]> select * from Employee where department_id=1 or department_id=2;
+-------------+------------+-----------+---------------+----------+------------+
| employee_id | first_name | last_name | department_id | salary   | hired      |
+-------------+------------+-----------+---------------+----------+------------+
|           1 | John       | Doe       |             1 | 50000.00 | 2023-05-12 |
|           2 | Jane       | Smith     |             2 | 60000.00 | 2012-02-10 |
|           4 | Bob        | Williams  |             1 | 55000.00 | 2023-05-12 |
|           7 | Lisa       | Taylor    |             2 | 65000.00 | 2012-02-10 |
|           8 | Kevin      | Clark     |             1 | 58000.00 | 2023-05-12 |
|          11 | Emily      | Wilson    |             2 | 58000.00 | 2012-02-10 |
|          13 | Samantha   | Martinez  |             1 | 56000.00 | 2023-05-12 |
|          16 | Andrew     | Johnson   |             2 | 62000.00 | 2012-02-10 |
|          17 | Lauren     | White     |             1 | 57000.00 | 2023-05-12 |
|          20 | Mathew     | Hall      |             2 | 64000.00 | 2012-02-10 |
+-------------+------------+-----------+---------------+----------+------------+
```

18. Write a query to fetch the first name, last name, and department name of employees from the "employees" and "departments" tables, joining them based on the department ID.

```
MariaDB [S2MCA]> select first_name,last_name,department_name from Employee,Department
where Employee.department_id=Department.department_id;
+-------------+-----------+-----------------+
| first_name  | last_name | department_name |
+-------------+-----------+-----------------+
| John        | Doe       | HR              |
| Jane        | Smith     | Finance         |
| Alice       | Johnson   | IT              |
| Bob         | Williams  | HR              |
| Sarah       | Lee       | Marketing       |
| Michael     | Brown     | IT              |
| Lisa        | Taylor    | Finance         |
| Kevin       | Clark     | HR              |
| Amanda      | Martinez  | Marketing       |
| Eric        | Anderson  | IT              |
| Emily       | Wilson    | Finance         |
| Ryan        | Garcia    | IT              |
| Samantha    | Martinez  | HR              |
| David       | Lee       | Marketing       |
| Jessica     | Brown     | IT              |
| Andrew      | Johnson   | Finance         |
| Lauren      | White     | HR              |
| Christopher | Lopez     | Marketing       |
| Kimberly    | Young     | IT              |
| Mathew      | Hall      | Finance         |
+-------------+-----------+-----------------+
```

19.Write a query to retrieve the first name, last name, and department name of all employees who belong to the "IT" department

```
MariaDB [S2MCA]> select first_name,last_name,department_name from Employee,Department
where Employee.department_id=Department.department_id and Department.department_name="IT";
```

```
+------------+-----------+-----------------+
| first_name | last_name | department_name |
+------------+-----------+-----------------+
| Alice      | Johnson   | IT              |
| Michael    | Brown     | IT              |
| Eric       | Anderson  | IT              |
| Ryan       | Garcia    | IT              |
| Jessica    | Brown     | IT              |
| Kimberly   | Young     | IT              |
+------------+-----------+-----------------+
```

20.Find employees who have the same salary.

MariaDB [S2MCA]> select * from Employee where salary in(select salary from Employee e where Employee.employee_id<>e.employee_id);

```
+-------------+------------+-----------+---------------+----------+------------+
| employee_id | first_name | last_name | department_id | salary   | hired      |
+-------------+------------+-----------+---------------+----------+------------+
|           9 | Amanda     | Martinez  |             4 | 60000.00 | 2023-05-12 |
|          16 | Andrew     | Johnson   |             2 | 62000.00 | 2012-02-10 |
|          11 | Emily      | Wilson    |             2 | 58000.00 | 2012-02-10 |
|           2 | Jane       | Smith     |             2 | 60000.00 | 2012-02-10 |
|           8 | Kevin      | Clark     |             1 | 58000.00 | 2023-05-12 |
|          20 | Mathew     | Hall      |             2 | 64000.00 | 2012-02-10 |
|           5 | Sarah      | Lee       |             4 | 62000.00 | 2023-05-12 |
|          14 | David      | Lee       |             4 | 64000.00 | 2023-05-12 |
+-------------+------------+-----------+---------------+----------+------------+
```

21. Drop the salary column from the employee table.

MariaDB [S2MCA]>alter table Employee drop salary;
Query OK, 0 rows affected (0.008 sec)
Records: 0 Duplicates: 0 Warnings: 0
MariaDB [S2MCA]> select * from Employee;

```
+-------------+-------------+-----------+---------------+------------+
| employee_id | first_name  | last_name | department_id | hired      |
+-------------+-------------+-----------+---------------+------------+
|           1 | John        | Doe       |             1 | 2023-05-12 |
|           2 | Jane        | Smith     |             2 | 2012-02-10 |
|           3 | Alice       | Johnson   |             3 | 2002-01-18 |
|           4 | Bob         | Williams  |             1 | 2023-05-12 |
|           5 | Sarah       | Lee       |             4 | 2023-05-12 |
|           6 | Michael     | Brown     |             3 | 2002-01-18 |
|           7 | Lisa        | Taylor    |             2 | 2012-02-10 |
|           8 | Kevin       | Clark     |             1 | 2023-05-12 |
|           9 | Amanda      | Martinez  |             4 | 2023-05-12 |
|          10 | Eric        | Anderson  |             3 | 2002-01-18 |
|          11 | Emily       | Wilson    |             2 | 2012-02-10 |
|          12 | Ryan        | Garcia    |             3 | 2002-01-18 |
|          13 | Samantha    | Martinez  |             1 | 2023-05-12 |
|          14 | David       | Lee       |             4 | 2023-05-12 |
|          15 | Jessica     | Brown     |             3 | 2002-01-18 |
|          16 | Andrew      | Johnson   |             2 | 2012-02-10 |
|          17 | Lauren      | White     |             1 | 2023-05-12 |
|          18 | Christopher | Lopez     |             4 | 2023-05-12 |
|          19 | Kimberly    | Young     |             3 | 2002-01-18 |
|          20 | Mathew      | Hall      |             2 | 2012-02-10 |
+-------------+-------------+-----------+---------------+------------+
```

22. Delete a row from employee where department-id =4.

```
MariaDB [S2MCA]> delete from Employee where department_id=4;
Query OK, 0 rows affected (0.001 sec)

MariaDB [S2MCA]> select * from Employee;
+-------------+------------+-----------+---------------+------------+
| employee_id | first_name | last_name | department_id | hired      |
+-------------+------------+-----------+---------------+------------+
|           1 | John       | Doe       |             1 | 2023-05-12 |
|           2 | Jane       | Smith     |             2 | 2012-02-10 |
|           3 | Alice      | Johnson   |             3 | 2002-01-18 |
|           4 | Bob        | Williams  |             1 | 2023-05-12 |
|           6 | Michael    | Brown     |             3 | 2002-01-18 |
|           7 | Lisa       | Taylor    |             2 | 2012-02-10 |
|           8 | Kevin      | Clark     |             1 | 2023-05-12 |
|          10 | Eric       | Anderson  |             3 | 2002-01-18 |
|          11 | Emily      | Wilson    |             2 | 2012-02-10 |
|          12 | Ryan       | Garcia    |             3 | 2002-01-18 |
|          13 | Samantha   | Martinez  |             1 | 2023-05-12 |
|          15 | Jessica    | Brown     |             3 | 2002-01-18 |
|          16 | Andrew     | Johnson   |             2 | 2012-02-10 |
|          17 | Lauren     | White     |             1 | 2023-05-12 |
|          19 | Kimberly   | Young     |             3 | 2002-01-18 |
|          20 | Mathew     | Hall      |             2 | 2012-02-10 |
+-------------+------------+-----------+---------------+------------+
```

23. Delete employee table from the database

```
MariaDB [S2MCA]> drop table Employee;
Query OK, 0 rows affected (0.011 sec)
MariaDB [S2MCA]> select * from Employee;
ERROR 1146 (42S02): Table 's2mca.employee' doesn't exist
```

24. Deletes an existing database (S2MCA).

```
MariaDB [S2MCA]> drop database S2MCA;
Query OK, 1 row affected (0.034 sec)
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| phpmyadmin         |
| s1mca              |
| s2mca              |
| s3mca              |
| student            |
| test               |
+--------------------+
```

PROGRAM 2

AIM: Perform Alter commands, Join operations, Update commands, order by clause, Like operator using DML commands

1. Create tables named Employee (with Emp_ID as primary key and Dept_ID as foreign key) and Department (with Dept_ID as primary key).

```
MariaDB [(none)]> CREATE DATABASE S2MCA;
Query OK, 1 row affected (0.001 sec)
MariaDB [(none)]> use S2MCA;
Database changed

MariaDB [S2MCA]> create table Department(department_id int primary key,department_name
varchar(50));
Query OK, 0 rows affected (0.011 sec)

MariaDB [S2MCA]> create table Employee(employee_id int primary key,first_name
varchar(50),last_name varchar(50),department_id int,email varchar(50),foreign key (department_id)
references Department(department_id));

MariaDB [S2MCA]> insert into Department
values(1,'HR'),(2,'Finance'),(3,'IT'),(4,'Marketing'),(5,'Sales'),(6,'Operations'),(7,'Research'),(8,'Develop
ment'),(9,'Customer Service'),(10,'Administration');
Query OK, 10 rows affected (0.003 sec)
Records: 10 Duplicates: 0 Warnings: 0

MariaDB [S2MCA]> insert into Employee (employee_id, first_name, last_name, email, department_id)
values (101, 'John', 'Doe', 'john.doe@email.com', 1), (102, 'Jane', 'Smith',
'jane.smith@email.com', 2), (103, 'Robert', 'Johnson', 'robert.johnson@email.com', 1), (104,
'Mary','Jones', 'mary.jones@email.com', 3), (105, 'Michael', 'Brown', 'michael.brown@email.com',
4),(106,'Jennifer', 'Davis', 'jennifer.davis@email.com', 5),(107, 'David',
'Martinez','david.martinez@email.com', 6), (108, 'Lisa', 'Rodriguez', 'lisa.rodriguez@email.com', 7),
(109,'William', 'Taylor', 'william.taylor@email.com', 8),(110, 'Sarah', 'Thomas',
'sarah.thomas@email.com', 9);
Query OK, 10 rows affected (0.005 sec)
Records: 10 Duplicates: 0 Warnings: 0
```

2. Add a new column named "Salary" to the Employee table with the data type DECIMAL (10,2).

```
MariaDB [S2MCA]> alter table Employee add salary decimal(10,2);
Query OK, 0 rows affected (0.006 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

3.Alter the Department table to rename the column "DepartmentName" to "DeptName".

```
MariaDB [S2MCA]> ALTER TABLE Department CHANGE COLUMN department_name DeptName
VARCHAR(50);
Query OK, 0 rows affected (0.006 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

4.Update the email of the employee with EmployeeID 102 to "jane.smith@example.com".

```
MariaDB [S2MCA]> UPDATE Employee SET Email = 'jane.smith@example.com'WHERE
employee_id = 102;
Query OK, 1 row affected (0.003 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

5. Provide an example of an INNER JOIN between the Employee table and the Department table on the common column "DepartmentID".

```
MariaDB [S2MCA]> select e.employee_id, e.first_name, e.last_name, e.email, d.DeptName from
Employee e inner join Department d on e.department_id = d.department_id;
+-------------+------------+-----------+--------------------------+------------------+
| employee_id | first_name | last_name | email                    | DeptName         |
+-------------+------------+-----------+--------------------------+------------------+
|         101 | John       | Doe       | john.doe@email.com       | HR               |
|         102 | Jane       | Smith     | jane.smith@example.com   | Finance          |
|         103 | Robert     | Johnson   | robert.johnson@email.com | HR               |
|         104 | Mary       | Jones     | mary.jones@email.com     | IT               |
|         105 | Michael    | Brown     | michael.brown@email.com  | Marketing        |
|         106 | Jennifer   | Davis     | jennifer.davis@email.com | Sales            |
|         107 | David      | Martinez  | david.martinez@email.com | Operations       |
|         108 | Lisa       | Rodriguez | lisa.rodriguez@email.com | Research         |
|         109 | William    | Taylor    | william.taylor@email.com | Development      |
|         110 | Sarah      | Thomas    | sarah.thomas@email.com   | Customer Service |
+-------------+------------+-----------+--------------------------+------------------+
```

6. Perform a LEFT JOIN between the Employee table and the Department table to display all employees regardless of whether they are assigned to a department or not.

```
select e.employee_id, first_name, last_name, e.email, d.DeptName from Employee e left join
Department d on e.Department_id = d.department_id;
+------------+------------+-----------+---------------+-------------------------+----------+------------+
| employee_id| first_name | last_name | department_id | email                   | salary   | DeptName   |
+------------+------------+-----------+---------------+-------------------------+----------+------------+
| 101        | John       | Doe       | 1             | john.doe@email.com      | 50000.00 | HR         |
| 102        | Jane       | Smith     | 2             | jane.smith@example.com  | 65000.00 | Finance    |
| 103        | Robert     | Johnson   | 1             | robert.johnson@email.com| 50000.00 | HR         |
| 104        | Mary       | Jones     | 3             | mary.jones@email.com    | 54000.00 | IT         |
| 105        | Michael    | Brown     | 4             | michael.brown@email.com | 70000.00 | Marketing  |
| 106        | Jennifer   | Davis     | 5             | jennifer.davis@email.com| 90000.00 | Sales      |
| 107        | David      | Martinez  | 6             | david.martinez@email.com| 75000.00 | Operations |
| 108        | Lisa       | Rodriguez | 7             | lisa.rodriguez@email.com| 66000.00 | Research   |
| 109        | William    | Taylor    | 8             | william.taylor@email.com| 88000.00 | Development|
| 110        | Sarah      | Thomas    | 9             | sarah.thomas@email.com  | 80000.00 | Customer   |
|            |            |           |               |                         |          | Service    |
+------------+------------+-----------+---------------+-------------------------+----------+------------+
```

7.Write a SQL query to calculate the total number of employees in each department.

MariaDB [S2MCA]> select d.department_id, d.DeptName, COUNT(e.employee_id) as TotalEmployees from Department d left join Employee e on d.department_id = e.department_id group by d.department_id, d.DeptName;

```
+---------------+------------------+----------------+
| department_id | DeptName         | TotalEmployees |
+---------------+------------------+----------------+
|             1 | HR               |              2 |
|             2 | Finance          |              1 |
|             3 | IT               |              1 |
|             4 | Marketing        |              1 |
|             5 | Sales            |              1 |
|             6 | Operations       |              1 |
|             7 | Research         |              1 |
|             8 | Development      |              1 |
|             9 | Customer Service |              1 |
|            10 | Administration   |              0 |
+---------------+------------------+----------------+
```

8. Retrieve the concatenation of the FirstName and LastName columns for all employees in the Employee table.

MariaDB [S2MCA]> select concat(first_name, ' ', last_name) as FullName from Employee;

```
+-----------------+
| FullName        |
+-----------------+
| John Doe        |
| Jane Smith      |
| Robert Johnson |
| Mary Jones      |
| Michael Brown   |
| Jennifer Davis |
| David Martinez |
| Lisa Rodriguez |
| William Taylor |
| Sarah Thomas    |
+-----------------+
```

9. Write a query that uses the WHERE clause to select all employees whose FirstName is "Jennifer".

MariaDB [S2MCA]> SELECT * FROM Employee WHERE first_name = 'Jennifer';

```
+-------------+------------+-----------+---------------+--------------------------+--------+
| employee_id | first_name | last_name | department_id | email                    | salary |
+-------------+------------+-----------+---------------+--------------------------+--------+
|         106 | Jennifer   | Davis     |             5 | jennifer.davis@email.com |   90000|
+-------------+------------+-----------+---------------+--------------------------+--------+
```

10. Use the ORDER BY clause to sort the records in the Employee table based on the LastName column in ascending order.

MariaDB [S2MCA]> select * from Employee order by last_name asc;

```
+-------------+------------+-----------+---------------+------------------------+----------+
| employee_id | first_name | last_name | department_id | email                  | salary   |
+-------------+------------+-----------+---------------+------------------------+----------+
| 105         | Michael    | Brown     | 4             | michael.brown@email.com | 70000.00 |
| 106         | Jennifer   | Davis     | 5             | jennifer.davis@email.com| 90000.00 |
| 101         | John       | Doe       | 1             | john.doe@email.com      | 50000.00 |
| 103         | Robert     | Johnson   | 1             | robert.johnson@email.com| 50000.00 |
| 104         | Mary       | Jones     | 3             | mary.jones@email.com    | 54000.00 |
| 107         | David      | Martinez  | 6             | david.martinez@email.com| 75000.00 |
| 108         | Lisa       | Rodriguez | 7             | lisa.rodriguez@email.com| 66000.00 |
| 102         | Jane       | Smith     | 2             | jane.smith@example.com  | 65000.00 |
| 109         | William    | Taylor    | 8             | william.taylor@email.com| 88000.00 |
| 110         | Sarah      | Thomas    | 9             | sarah.thomas@email.com  | 80000.00 |
+-------------+------------+-----------+---------------+------------------------+----------+
```

11. Provide an example of using the LIKE operator to select all employees whose last names start with the letter "S".

MariaDB [S2MCA]> SELECT * FROM Employee WHERE last_name LIKE 'S%';

```
 +-------------+------------+-----------+---------------+------------------------+----------+
| employee_id | first_name | last_name | department_id | email                  | salary   |
+-------------+------------+-----------+---------------+------------------------+----------+
| 102         | Jane       | Smith     | 2             | jane.smith@example.com | 65000.00 |
+-------------+------------+-----------+---------------+------------------------+----------+
```

12. Write a query using the IN operator to select all employees whose DepartmentID is either 7, 8, or 9.

MariaDB [S2MCA]> SELECT * FROM Employee WHERE department_id IN (7, 8, 9);

```
+-------------+------------+-----------+---------------+------------------------+----------+
| employee_id | first_name | last_name | department_id | email                  | salary   |
+-------------+------------+-----------+---------------+------------------------+----------+
| 108         | Lisa       | Rodriguez | 7             | lisa.rodriguez@email.com| 66000.00 |
| 109         | William    | Taylor    | 8             | william.taylor@email.com| 88000.00 |
| 110         | Sarah      | Thomas    | 9             | sarah.thomas@email.com  | 80000.00 |
+-------------+------------+-----------+---------------+------------------------+----------+
```

PROGRAM 3

AIM: Creation of a database COMPANY, and tables using DDL commands including integrity constraints. Populate the tables with DML commands.

DDL COMMANDS

1. The employees table stores the data of employees.

```
MariaDB [S2MCA]>create table employees(employee_id int,first_name varchar(30),last_name
varchar(30),email varchar(30),phone int,hire_date date,job_id int,salary int,manager_id
int,department_id int,primary key(employee_id),foreign key(job_id) references jobs(job_id),foreign
key(department_id) references departments(department_id));
```

```
MariaDB [S2MCA]>INSERT INTO employees (employee_id, first_name, last_name, email, phone,
hire_date, job_id, salary, manager_id, department_id) VALUES (1, 'John', 'Doe',
'john.doe@example.com', '1555555655', '1996-01-15', '1', 70000, 1, 1),(2, 'Jane', 'Smith',
'jane.smith@example.com', 1855555655', '1995-03-20', '2', 80000, 1, 2),(3, 'Bob', 'Johnson',
'bob.johnson@example.com', '1678564907', '1997-05-10', '3', 60000, 1, 3),(4, 'Alice', 'Fay',
'alice.fay@example.com', "1678564967', '1995-07-25', '4', 65000, 2, 4),(5, 'David', 'Grant',
'david.grant@example.com', '1045625781', '1996-09-05', '5', 75000, 2, 5),(6, 'Eva', 'Whalen',
'eva.whalen@example.com', ‘9845674314', '1997-11-15', '6', 90000, 3, 6),(7, 'Michael', 'Taylor',
'michael.tayLor@example.com', '9812998744', '1996-12-30', '7', 100000, 1, 7);
```

2.The jobs table stores the job data including job title and salary range.

```
MariaDB [S2MCA]>create table jobs(job_id int,job_title varchar(50),min_salary int,max_salary
int,primary key(job_id));
```

```
MariaDB [S2MCA]>INSERT INTO jobs (job_id, job_title, min_salary, max_salary) VALUES (1,
'Software Engineer', 60000, 100000),(2, 'Database Administrator', 65000, 110000),(3, 'Marketing
Specialist', 50000, 90000),(4, 'Financial Analyst', 55000, 95000),(5, 'HR Manager', 4500, 5000),(6,
'Sales Representative', 48000, 85000),(7, 'Shipping Clerk', 45000, 55000);
```

3.The departments table stores department data.

```
 MariaDB [S2MCA]>create table departments(department_id int,department_name
varchar(30),location_id int,primary key(department_id),foreign key(location_id) references
locations(location_id));
```

```
MariaDB [S2MCA]>INSERT INTO departments (department_id, department_name, location_id)
VALUES (1, 'IT Department', 1),(2, 'Finance Department', 2),(3, 'Marketing Department', 3),(4, 'HR
Department', 4),(5, 'Sales Department', 5),(6, 'Operations Department', 6),(7, 'R&D Department', 7),(8,
'Shipping', 4),(11, 'Administration', 7);
```

4.The dependents table stores the employee's dependents.

```
MariaDB [S2MCA]>create table dependents(dependent_id int,first_name varchar(30),last_name
varchar(30),relationship varchar(30),employee_id int,primary key(dependent_id),foreign
key(employee_id)references employees(employee_id));
```

```
MariaDB [S2MCA]>INSERT INTO dependents (dependent_id, first_name, last_name, relationship,
employee_id) VALUES(1, 'Emma', 'Doe', 'Child', 1),(2, 'Alex', 'Smith', 'Spouse', 2),(3, 'Olivia',
'Johnson', 'Child', 3),(4, 'Liam', 'Fay', 'Child', 4),(5, 'Sophia', 'Grant', 'Spouse', 5),(6, 'Noah', 'Whalen',
'Child', 6),(7, 'Ava', 'Taylor', 'Child', 7);
```

DML COMMANDS

1. The locations table stores the location of the departments of the company.

```
MariaDB [S2MCA]>create table locations(location_id int,street_address varchar(50),postal_code
int,city varchar(50),state varchar(30),country_id int,primary key(location_id),foreign key(country_id)
references countries(country_id));
```

```
MariaDB [S2MCA]>INSERT INTO locations (location_id, street_address, postal_code, city, state,
country_id) VALUES (1, '123 Main St', '12345', 'New York', 'NY', 1),(2, '456 Oak St', '67890', 'Los
Angeles', 'CA', 2),(3, '789 Pine St', '98765', 'Chicago', 'IL', 3),(4, '321 Elm St', '54321', 'Houston', 'TX',
1),(5, '654 Maple St', '13579', 'San Francisco', 'CA', 2),(6, '987 Birch St', '24680', 'Miami', 'FL', 3),(7,
'555 Pineapple Ln', '54321', 'Honolulu', 'HI', 4);
```

2.The countries table stores the data of countries where the company is doing business.

```
MariaDB [S2MCA]>create table countries(country_id int,country_name varchar(30),region_id
int,primary key(country_id),foreign key(region_id)references regions(region_id));
```

```
 MariaDB [S2MCA]>INSERT INTO countries (country_id, country_name, region_id) VALUES (1,
'America', 1),(2, 'England', 3),(3, 'Australia', 2),(4, 'UK', 3),(5, 'India', 5),(6, 'Canada', 2),(7,
'Switzerland', 3);
```

3.The regions table stores the data of regions such as Asia, Europe, America, and the Middle East and Africa. The countries are grouped into regions.

```
 MariaDB [S2MCA]>create table regions(region_id int,region_name varchar(30),primary
key(region_id));
```

```
MariaDB [S2MCA]>insert into regions values(1,"Asia"),(2,"Europe"),(3,"America"),(4,"Middle
east"),(5,"Africa");
```

Question 4

1.Write a query to display all the countries.

MariaDB [S2MCA]> SELECT * FROM countries;

```
+------------+--------------+-----------+
| country_id | country_name | region_id |
+------------+--------------+-----------+
|          1 | America      |         1 |
|          2 | England      |         3 |
|          3 | Australia    |         2 |
|          4 | UK           |         3 |
|          5 | India        |         5 |
|          6 | Canada       |         2 |
|          7 | Switzerland  |         3 |
+------------+--------------+-----------+
```

2.Write a query to display specific columns like email and phone number for all the employees.

MariaDB [S2MCA]> select email, phone from employees;

```
+-----------------------------+-------+
| email                       | phone |
+-----------------------------+-------+
| john.doe@example.com        |   123 |
| jane.smith@example.com      |   234 |
| bob.johnson@example.com     |   345 |
| alice.fay@example.com       |   456 |
| david.grant@example.com     |   567 |
| eva.whalen@example.com      |   678 |
| michael.tayLor@example.com  |   789 |
+-----------------------------+-------+
```

3.Write a query to display the data of employee whose last name is "Fay".

MariaDB [S2MCA]> SELECT * FROM employees WHERE last_name = 'Fay';

```
+-----------+----------+---------+---------------------+----------+----------+------+------+----------+-------_-----+
|employee_id|first_name|last_name|email                |phone     |hire_date |job_id|salary|manager_id|department_id|
|
+-----------+----------+---------+---------------------+----------+----------+------+------+----------+-------------+
|          4| Alice    |Fay      |alice.fay@example.com|1678564967|1995-07-25|    4 | 65000|         2|            4 |
+-----------+----------+---------+---------------------+----------+----------+------+------+----------+--------------
```

4.Write a query to find the hire date for employees whose last name is "Grant" or "Whalen".

MariaDB [S2MCA]> SELECT last_name,hire_date FROM employees WHERE last_name = 'Grant' OR last_name = 'Whalen';

```
+-----------+------------+
| last_name | hire_date  |
+-----------+------------+
| Grant     | 1996-09-05 |
| Whalen    | 1997-11-15 |
+-----------+------------+
```

5.Write a query to display name of the employee who is shipping clerk.

MariaDB [S2MCA]> select first_name ,last_name from employees where job_id='7';

```
+------------+-----------+
| first_name | last_name |
+------------+-----------+
| Michael    | Kayle     |
+------------+-----------+
```

6.Write a query to get all the employees who work for department 8.

```
+-----------+---------+--------+---------------------+----------+----------+------+-----+----------+------------+
|employee_id|first_name|last_name|email                | phone    |hire_date |job_id |salary|manager_id |department_id|
+-----------+---------+--------+---------------------+----------+----------+------+-----+----------+------------+
|          2| Jane     | Smith   |jane.smith@example.com|1855555655|1995-03-20|      2|80000 |          1 |            8 |
+-----------+---------+--------+---------------------+----------+----------+------+-----+----------+------------+
```

7.Write a query to display the departments in the descending order.

MariaDB [S2MCA]> SELECT * FROM departments ORDER BY department_id DESC;

```
+---------------+-----------------------+-------------+
| department_id | department_name       | location_id |
+---------------+-----------------------+-------------+
|            11 | Administration        |           7 |
|             8 | Shipping              |           4 |
|             7 | R&D Department        |           7 |
|             6 | Operations Department |           6 |
|             5 | Sales Department      |           5 |
|             4 | HR Department         |           4 |
|             3 | Marketing Department  |           3 |
|             2 | Finance Department    |           2 |
|             1 | IT Department         |           1 |
+---------------+-----------------------+-------------+
```

8.Write a query to display all the employees whose last name starts with "K".

MariaDB [S2MCA]>SELECT * FROM employees WHERE last_name LIKE 'K%';

```
+-----------+----------+---------+------------------------+-----------+-----------+-------+-------+-----------+--------------+
| employee_id| first_name| last_name| email                  | phone      | hire_date  | job_id | salary |manager_id  | department_id |
+-----------+----------+---------+------------------------+-----------+-----------+-------+-------+-----------+--------------+
|          2| Jane      | Khanna   | jane.smith@example.com | 2147483647 | 2023-02-01 |      2 |  60000 |          1 |             2 |
|         10| Junaih    | Khan     | jack.white@example.com | 1234567890 | 2023-10-01 |     10 |  40000 |          5 |            10 |
+-----------+----------+---------+------------------------+-----------+-----------+-------+-------+-----------+--------------+
```

9.Display name of the employees whose hire dates are between 1995 and 1997.

MariaDB [S2MCA]>SELECT first_name, last_name FROM employees WHERE hire_date BETWEEN '1995-01-01' AND '1997-12-31';

```
+------------+-----------+
| first_name | last_name |
+------------+-----------+
| John       | Doe       |
| Jane       | Smith     |
| Bob        | Johnson   |
| Alice      | Fay       |
| David      | Grant     |
| Eva        | Whalen    |
| Michael    | Kayle     |
+------------+-----------+
```

10.Write a query to display jobs where the maximum salary is less than 5000.

MariaDB [S2MCA]> SELECT * FROM jobs WHERE max_salary < 5000;

```
+--------+------------+------------+------------+
| job_id | job_title  | min_salary | max_salary |
+--------+------------+------------+------------+
|      5 | HR Manager |       4500 |       4900 |
+--------+------------+------------+------------+
```

11.Write a query to display email address in lower case.

```
MariaDB [S2MCA]> SELECT LOWER(email) AS lowercase_email FROM employees;
+----------------------------+
| lowercase_email            |
+----------------------------+
| john.doe@example.com       |
| jane.smith@example.com     |
| bob.johnson@example.com    |
| alice.fay@example.com      |
| david.grant@example.com    |
| eva.whalen@example.com     |
| michael.taylor@example.com |
+----------------------------+
```

12.Write a query to display name of the employees who were hired in 1995.

```
MariaDB [S2MCA]>  SELECT first_name, last_name FROM employees WHERE YEAR(hire_date) =
1995;
+------------+-----------+
| first_name | last_name |
+------------+-----------+
| Jane       | Smith     |
| Alice      | Fay       |
+------------+-----------+
```

13.Write a query to insert an employee "Paul Newton" in department 11.

```
MariaDB [S2MCA]>  INSERT INTO employees (employee_id, first_name, last_name, email, phone,
hire_date, job_id,salary, manager_id, department_id)VALUES (8, 'Paul', 'Newton',
'paul.newton@example.com', '1278965736', '2023-06-11', 1, 60000, 5, 11);
Query OK, 1 row affected (0.004 sec)

MariaDB [S2MCA]>  select * from employees;
+-----------+----------+----------+-------------------------+-----------+-----------+-------+-------+----------+------------+
|employee_id | first_name| last_name | email                       | phone      | hire_date  | job_id | salary | manager_id|department_id|
+-----------+----------+----------+-------------------------+-----------+-----------+-------+-------+----------+------------+
|          1| John      | Doe       | john.doe@example.com        | 1555555655 | 1996-01-15 |      1 |  70000 |          1|           1  |
|          2| Jane      | Smith     | jane.smith@example.com      | 1855555655 | 1995-03-20 |      2 |  80000 |          1|           8  |
|          3| Bob       | Johnson   | bob.johnson@example.com     | 1678564907 | 1997-05-10 |      3 |  60000 |          1|           3  |
|          4| Alice     | Fay       | alice.fay@example.com       | 1678564967 | 1995-07-25 |      4 |  65000 |          2|           4  |
|          5| David     | Grant     | david.grant@example.com     | 1045625781 | 1996-09-05 |      5 |  75000 |          2|           5  |
|          6| Eva       | Whalen    | eva.whalen@example.com      | 2147483647 | 1997-11-15 |      6 |  90000 |          3|           6  |
|          7| Michael   | Kayle     | michael.tayLor@example.com| 2147483647 | 1996-12-30 |      7 | 100000 |          1|           7  |
|          8| Paul      | Newton    | paul.newton@example.com     | 1278965736 | 2023-06-11 |      1 |  60000 |          5|          11  |
+-----------+----------+----------+-------------------------+-----------+-----------+-------+-------+----------+------------+
```

14. Write a query to delete the shipping department.

```
MariaDB [S2MCA]>  DELETE FROM departments WHERE department_name = 'Shipping';
Query OK, 0 rows affected (0.01 sec)

MariaDB [details]> select * from departments;
```

```
+---------------+-----------------------+-------------+
| department_id | department_name       | location_id |
+---------------+-----------------------+-------------+
|             1 | IT Department         |           1 |
|             2 | Finance Department    |           2 |
|             3 | Marketing Department  |           3 |
|             4 | HR Department         |           4 |
|             5 | Sales Department      |           5 |
|             6 | Operations Department |           6 |
|             7 | R&D Department        |           7 |
|             8 | Executives            |           4 |
|            11 | Administration        |           7 |
+---------------+-----------------------+-------------+
```

PROGRAM 4

AIM:Apply DCL and TCL commands to impose restrictions on database.

DDL

1.Create an employee table with the attributes specified above (set E_ID as the primary key).

create table Employee(E_id int primary key,E_name varchar(50),E_gender varchar(50),E_salary int,E_branch varchar(50));

2. Insert the given values into the attributes.

insert into Employee values('E1','A','M',12500,'B1'),('E2','B','F',15000,'B4'),('E3','C','M',36574,'B3'),('E4,'D','F',35674,'B2'),('E5','E','F',46572,'B3'),('E6','F','M',43564,'B4'),('E7','G','M',65431,'B2');

```
MariaDB [s3mca]> select * from Employee;
+------+--------+----------+----------+----------+
| E_id | E_name | E_gender | E_salary | E_branch |
+------+--------+----------+----------+----------+
| E1   | A      | M        |    12500 | B1       |
| E2   | B      | F        |    15000 | B4       |
| E3   | C      | M        |    36574 | B3       |
| E4   | D      | F        |    35674 | B2       |
| E5   | E      | F        |    46572 | B3       |
| E6   | F      | M        |    43564 | B4       |
| E7   | G      | M        |    65431 | B2       |
+------+--------+----------+----------+----------+
```

3. Delete all rows from the table and free the space containing the table (TRUNCATE). Redo question 2.

```
MariaDB [s3mca]> truncate table Employee;
Query OK, 0 rows affected (0.014 sec)
MariaDB [s3mca]> select * from Employee;
Empty set (0.000 sec)
MariaDB [s3mca]> insert into Employee
values('E1','A','M',12500,'B1'),('E2','B','F',15000,'B4'),('E3','C','M',36574,'B3'),('E4','D','F',35674,'B2'),('E5','E','F',46572,'B3'),('E6','F','M',43564,'B4'),('E7','G','M',65431,'B2');
Query OK, 7 rows affected (0.003 sec)
Records: 7  Duplicates: 0  Warnings: 0

MariaDB [s3mca]> select * from Employee;
+------+--------+----------+----------+----------+
| E_id | E_name | E_gender | E_salary | E_branch |
+------+--------+----------+----------+----------+
| E1   | A      | M        |    12500 | B1       |
| E2   | B      | F        |    15000 | B4       |
| E3   | C      | M        |    36574 | B3       |
| E4   | D      | F        |    35674 | B2       |
| E5   | E      | F        |    46572 | B3       |
| E6   | F      | M        |    43564 | B4       |
| E7   | G      | M        |    65431 | B2       |
+------+--------+----------+----------+----------+
```

DML

1. Select all attributes of the EMP table.

```
MariaDB [s3mca]> select * from Employee;
+------+--------+----------+----------+----------+
| E_id | E_name | E_gender | E_salary | E_branch |
+------+--------+----------+----------+----------+
| E1   | A      | M        |    12500 | B1       |
| E2   | B      | F        |    15000 | B4       |
| E3   | C      | M        |    36574 | B3       |
| E4   | D      | F        |    35674 | B2       |
| E5   | E      | F        |    46572 | B3       |
| E6   | F      | M        |    43564 | B4       |
| E7   | G      | M        |    65431 | B2       |
+------+--------+----------+----------+----------+
```

2. Retrieve the average salary of the employees.

```
MariaDB [s3mca]> select  avg(E_salary) from Employee;
+---------------+
| avg(E_salary) |
+---------------+
|    36606.8333 |
+---------------+
```

3. Retrieve the name of the employee with the minimum salary.

```
 MariaDB [s3mca]> select E_name,min(E_salary) from Employee;
+--------+---------------+
| E_name | min(E_salary) |
+--------+---------------+
| A      |         12500 |
+--------+---------------+
```

4. Retrieve the name of the employee with the maximum salary.

```
 MariaDB [s3mca]> select E_name,max(E_salary) from Employee;
+--------+---------------+
| E_name | max(E_salary) |
+--------+---------------+
| G      |         65431 |
+--------+---------------+
```

5. Find the total number of employees.

```
MariaDB [s3mca]> select count(*) from Employee;
+----------+
| count(*) |
+----------+
|        7 |
+----------+
```

6.Calculate the total amount of salary.

```
MariaDB [s3mca]> select sum(E_salary) from Employee;
+---------------+
| sum(E_salary) |
+---------------+
|        255315 |
+---------------+
```

7. Find the total number of employees segregated based on branches.

MariaDB [s3mca]> select E_branch,count(*) from employee group by E_branch;
```
+----------+----------+
| E_branch | count(*) |
+----------+----------+
| B1       |        1 |
| B2       |        2 |
| B3       |        2 |
| B4       |        2 |
+----------+----------+
```

8. Find out the name of the employee having a salary > 15,000.

MariaDB [s3mca]> select E_name from Employee where E_salary>15000;
```
+--------+
| E_name |
+--------+
| C      |
| D      |
| E      |
| F      |
| G      |
+--------+
```

9. Display the names of the employees in ascending order.

MariaDB [s3mca]> select E_name from Employee order by E_name asc;
```
+--------+
| E_name |
+--------+
| A      |
| B      |
| C      |
| D      |
| E      |
| F      |
| G      |
+--------+
```

10. Display the names of the employees in descending order.

MariaDB [s3mca]> select E_name from Employee order by E_name desc;
```
+--------+
| E_name |
+--------+
| G      |
| F      |
| E      |
| D      |
| C      |
| B      |
| A      |
+--------+
```

11. Find names of the employees belonging to the same branch. (Hint: GROUP BY, HAVING).

MariaDB [s3mca]> SELECT E_name, E_branch FROM employee WHERE E_branch IN (SELECT E_branch FROM employee GROUP BY E_branch HAVING COUNT(*) > 1)ORDER BY E_branch, E_name;

```
+--------+----------+
| E_name | E_branch |
+--------+----------+
| D      | B2       |
| G      | B2       |
| C      | B3       |
| E      | B3       |
| B      | B4       |
| F      | B4       |
+--------+----------+
```

12. List names and salaries of the employees whose salary is more than the average salary of the employees.

```
MariaDB [s3mca]>  select E_name,E_salary from employee where E_salary>(select avg(E_salary)
from employee);
+--------+----------+
| E_name | E_salary |
+--------+----------+
| C      |    36574 |
| E      |    46572 |
| F      |    43564 |
| G      |    65431 |
+--------+----------+
```

13. Create a view named name of employee whose salary is greater than the average salary.

```
MariaDB [s3mca]> create view name_of_employee as select E_salary from Employee where
E_salary>(select avg(E_salary) from Employee);
Query OK, 0 rows affected (0.009 sec)

MariaDB [s3mca]> select * from name_of_employee;
+----------+
| E_salary |
+----------+
|    36574 |
|    46572 |
|    43564 |
|    65431 |
+----------+
```

14. The name of the employee who is in branch B2 or male.

```
MariaDB [s3mca]> select E_name from Employee where E_branch="B2" or E_gender="M";
+--------+
| E_name |
+--------+
| A      |
| C      |
| D      |
| F      |
| G      |
+--------+
```

15. The name of the employee who is in branch B3 and female.

```
MariaDB [s3mca]> select E_name from Employee where E_branch="B3" and E_gender="F";
```

```
+--------+
| E_name |
+--------+
| E      |
+--------+
```

16. The name of the employee who is in branch B2 but not male.

```
MariaDB [s3mca]> select E_name from Employee where E_branch="B2" and E_gender="F";
+--------+
| E_name |
+--------+
| D      |
+--------+
```

DCL

1. Grant user access privileges to a database. (GRANT).

In SQL, the GRANT command is used to grant specific privileges or permissions to a user or a group of users on a database object such as a table, view, or stored procedure.
The syntax for the GRANT command:
GRANT privilage_name ON object_name TO user_or_role;
Where:
privilege_name is the specific permission being granted, such as SELECT, INSERT, UPDATE, DELETE, or ALL.
object_name is the name of the database object (e.g., table, view, stored procedure) on which the permission is being granted.
user_or_role is the user or role to whom the permission is being granted.

2. Revoke permissions from the user. (REVOKE).

In SQL, the REVOKE command is used to revoke previously granted privileges or permissions from a user or a group of users on a database object. This command essentially removes the specified privileges from the user or group.
The syntax for the REVOKE command:
REVOKE privilege_name ON object_name FROM user_or_role;
Where:
privilege_name is the specific permission being revoked, such as SELECT, INSERT, UPDATE, DELETE, or ALL.
object_name is the name of the database object (e.g., table, view, stored procedure) from which the permission is being revoked.
user_or_role is the user or role from whom the permission is being revoked.

TCL

1. Save all transactions to the database. (COMMIT).

In SQL, the COMMIT command is used to save all transactions made within a transaction block to the database. When you execute a COMMIT command, all changes made within the current transaction are permanently saved to the database, and the transaction is completed. This means that the changes become visible to other users and transactions.

2. Undo transactions that have not already been saved to the database. (Rollback).
In SQL, the ROLLBACK command is used to undo transactions that have not already been saved to the database. When you execute a ROLLBACK command, all changes made within the current transaction are discarded, and the database is reverted to its state before the transaction began.

3. Roll the transaction back to a certain point without rolling back the entire transaction. (Savepoint).

In SQL, a savepoint is a point within a transaction to which you can roll back without rolling back the entire transaction. Savepoints are used to create intermediate points in a transaction, allowing you to undo changes made after a specific savepoint while keeping changes made before that savepoint intact.
The syntax for creating a savepoint is:
SAVEPOINT savepoint_name;

PROGRAM 5

AIM: Application of views and joins for query optimization.

1.Create two tables officers and student. Insert some record into both tables.

```
mysql>create table officers(officer_id int,officer_name varchar(50),address  varchar(50),primary
key(officer_id));
Query OK, 0 rows affected (0.40 sec)

mysql> create table student(student_id int,student_name varchar(50),course
varchar(50),primary key(student_id));
Query OK, 0 rows affected (0.38 sec)

mysql> insert into officers values (1,"Ajeet","Goa"),(2,"Deepika","Lucknow"),
 (3,"Vimal","Delhi"),(4,"Rahul","Mumbai");
Query OK, 4 rows affected (0.18 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> insert into student
values(1,"Aryan","Java"),(2,"Rohini","Hadoop"),(3,"Manu","MongoDB");
Query OK, 3 rows affected (0.14 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

2.Perform join operations over the tables.

```
mysql>  SELECT officers.officer_name, officers.address, student.course FROM officers
   -> INNER JOIN student ON officers.officer_id = student.student_id;
+--------------+---------+---------+
| officer_name | address | course  |
+--------------+---------+---------+
| Ajeet        | Goa     | Java    |
| Deepika      | Lucknow | Hadoop  |
| Vimal        | Delhi   | MongoDB |
+--------------+---------+---------+

mysql> SELECT officers.officer_name, officers.address, student.course,
 student.student_name FROM officers RIGHT JOIN student ON officers.officer_id =
 student.student_id;
+--------------+---------+---------+--------------+
| officer_name | address | course  | student_name |
+--------------+---------+---------+--------------+
| Ajeet        | Goa     | Java    | Aryan        |
| Deepika      | Lucknow | Hadoop  | Rohini       |
| Vimal        | Delhi   | MongoDB | Manu         |
+--------------+---------+---------+--------------+
```

mysql> SELECT * FROM officers INNER JOIN student ON officers.officer_id =
student.student_id;

```
+------------+--------------+---------+------------+--------------+---------+
| officer_id | officer_name | address | student_id | student_name | course  |
+------------+--------------+---------+------------+--------------+---------+
|          1 | Ajeet        | Goa     |          1 | Aryan        | Java    |
|          2 | Deepika      | Lucknow |          2 | Rohini       | Hadoop  |
|          3 | Vimal        | Delhi   |          3 | Manu         | MongoDB |
+------------+--------------+---------+------------+--------------+---------+
```

mysql> SELECT * FROM officers RIGHT JOIN student ON officers.officer_id =
student.student_id;

```
+------------+--------------+---------+------------+--------------+---------+
| officer_id | officer_name | address | student_id | student_name | course  |
+------------+--------------+---------+------------+--------------+---------+
|          1 | Ajeet        | Goa     |          1 | Aryan        | Java    |
|          2 | Deepika      | Lucknow |          2 | Rohini       | Hadoop  |
|          3 | Vimal        | Delhi   |          3 | Manu         | MongoDB |
+------------+--------------+---------+------------+--------------+---------+
```

3.Perform view operation in any table.

mysql> CREATE VIEW officer_view AS SELECT officer_id, officer_name, address FROM officers;
Query OK, 0 rows affected (0.12 sec)

mysql> select *from officer_view;

```
+------------+--------------+---------+
| officer_id | officer_name | address |
+------------+--------------+---------+
|          1 | Ajeet        | Goa     |
|          2 | Deepika      | Lucknow |
|          3 | Vimal        | Delhi   |
|          4 | Rahul        | Mumbai  |
+------------+--------------+---------+
```

Course Outcome 2

Apply PL/SQL for processing databases.

PROGRAM 6

AIM: Basics of PL/SQL

1. Write a program to add two numbers

```
set serveroutput on;
DECLARE
a integer;
b integer;
BEGIN
a:=&a;
b:=&b;
dbms_output.put_line('sum is : ' || (a+b));
END;
/
```

OUTPUT

```
Enter value for a: 3
old 7: a:=&a;
new 7: a:=3;
Enter value for b: 6
old 8: b:=&b;
new 8: b:=6;
sum is : 9
PL/SQL procedure successfully completed.
```

2. Write a program to find largest of 3 numbers

```
set serveroutput on;
DECLARE
a integer;
b integer;
c integer;
BEGIN
a:=&a;
b:=&b;
c:=&c;
IF a>b and a>c
THEN
dbms_output.put_line('largest: ' || a);
ELSIF b>c
THEN
dbms_output.put_line('largest: ' || b);
```

```
ELSE
dbms_output.put_line('largest: ' || c);
END IF;
END;
/
```

OUTPUT

```
Enter value for a: 1
old 6: a:=&a;
new 6: a:=1;
Enter value for b: 2
old 7: b:=&b;
new 7: b:=2;
Enter value for c: 3
old 8: c:=&c;
new 8: c:=3;
largest: 3
PL/SQL procedure successfully completed.
```

3. Write a program to check whether given number is even or odd

```
set serveroutput on;
DECLARE
n integer;
mod integer;
a integer;
BEGIN
n:=&n;
a:=mod(n,2);
if a=0
then
dbms_output.put_line(n || 'is even');
else
dbms_output.put_line(n || 'is odd');
END IF;
END;
/
```

OUTPUT

```
Enter value for n: 3
old 6: n:=&n;
new 6: n:=3;
3 is odd
PL/SQL procedure successfully completed.
```

PROGRAM 7

AIM: Create a function to print annual salary of the employees in HR department.

CODE:

```
create table empl(emp_id number,name varchar(20),dept varchar(20),sal number);
insert into empl values(101,'Bobby','HR',25000);
insert into empl values(102,'George','HR',32000);
insert into empl values(103,'James','Hardware',55000);
insert into empl values(104,'David','Hardware',65000);
insert into empl values(105,'Sona','Marketing',20000);
insert into empl values(106,'Saira','HR',21000);
insert into empl values(107,'Fawaz','Software',50000);

SQL> CREATE OR REPLACE FUNCTION CALC_TOT_SAL RETURN NUMBER IS
      S NUMBER := 0;
   BEGIN
   FOR A IN (SELECT SAL FROM EMPL WHERE DEPT = 'HR') LOOP
     S := S + A.SAL;
    END LOOP;
    DBMS_OUTPUT.PUT_LINE('ANNUAL SALARY: ' || (S * 12));
    RETURN S;
    END;
   /
Function created.
```

OUTPUT

```
SQL> SELECT CALC_TOT_SAL FROM DUAL;

CALC_TOT_SAL
------------
      78000

ANNUAL SALARY: 936000
```

PROGRAM 8

AIM: Create a cursor to print employee name of employees whose salary is greater than 50000.

CODE:

```
SQL> DECLARE
    CURSOR curs_work IS
    SELECT name
     FROM worker
     WHERE sal > 50000;
     v_ename worker.name%TYPE; -- Variable to hold employee name
     BEGIN
    OPEN curs_work;
    LOOP
    FETCH curs_work INTO v_ename;
    EXIT WHEN curs_work%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('Employee Name: ' || v_ename);
   END LOOP;
   CLOSE curs_work;
   END;
   /
```

OUTPUT

```
Employee Name: Rahul
Employee Name: Meenu
PL/SQL procedure successfully completed.
```

PROGRAM 9

AIM: Construct a Trigger code for a table in database

CODE:

```
create table worker as select *from empl;
CREATE OR REPLACE TRIGGER salary_update_trigger
BEFORE UPDATE OF sal ON worker
FOR EACH ROW
DECLARE
    old_salary worker.sal%TYPE;
    new_salary worker.sal%TYPE;
BEGIN
    old_salary := :OLD.sal;
    new_salary := :NEW.sal;

    DBMS_OUTPUT.PUT_LINE('Salary difference: ' || (new_salary - old_salary));
END;
/
```

OUTPUT

```
SQL> update worker set sal=50000 where EMP_ID=102;
Salary difference: 18000

1 row updated.

SQL> SELECT * FROM worker;

    EMP_ID NAME                 DEPT                        SAL
---------- -------------------- -------------------- ----------
       101 Bobby                HR                        25000
       102 George               HR                        50000
       103 James                Hardware                  55000
       104 David                Hardware                  65000
       105 Sona                 Marketing                 20000
       106 Saira                HR                        21000
       107 Fawaz                Software                  50000

7 rows selected.
```

PROGRAM 10

AIM: Write a PL/SQL Procedure to list all even and odd number between 1 and 20

CODE:

```
SQL> create or replace procedure eve_odd is
   i number;
   BEGIN
         FOR i IN 1..20 LOOP
            IF MOD(i, 2) = 0 THEN
               dbms_output.put_line(i||' IS EVEN');
            ELSE
               dbms_output.put_line(i||' IS ODD');
            END IF;
        END LOOP;
  END;
  /
Procedure created.
```

OUTPUT

```
SQL> execute eve_odd;
1 IS ODD
2 IS EVEN
3 IS ODD
4 IS EVEN
5 IS ODD
6 IS EVEN
7 IS ODD
8 IS EVEN
9 IS ODD
10 IS EVEN
11 IS ODD
12 IS EVEN
13 IS ODD
14 IS EVEN
15 IS ODD
16 IS EVEN
17 IS ODD
18 IS EVEN
19 IS ODD
20 IS EVEN
PL/SQL procedure successfully complete.
```

PROGRAM 11

AIM: Write A PL/SQL Procedure to find factorial of a number.

CODE:

```
set serveroutput on;
CREATE OR REPLACE PROCEDURE factorial(num number) is
res NUMBER;
BEGIN
res := 1;
FOR i IN 1..num LOOP
res := res * i;
END LOOP;
dbms_output.put_line(res);
END;
/
Procedure created.
```

OUTPUT

```
SQL> execute factorial(4);
24
PL/SQL procedure successfully completed.
```

Course Outcome 3

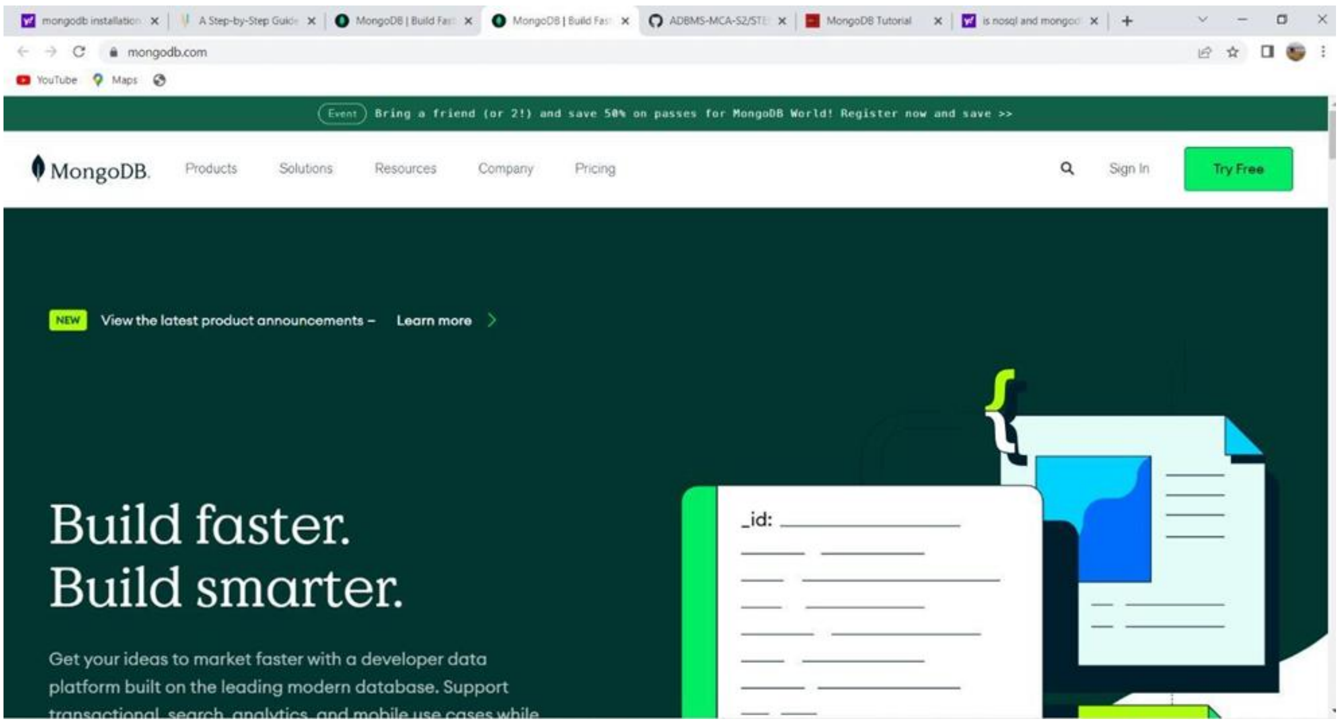
Comparison between relational and non-relational (NoSQL) databases and the configuration of NoSQL Databases. Apply CRUD operations and retrieve data in NoSQL environment.

PROGRAM 12

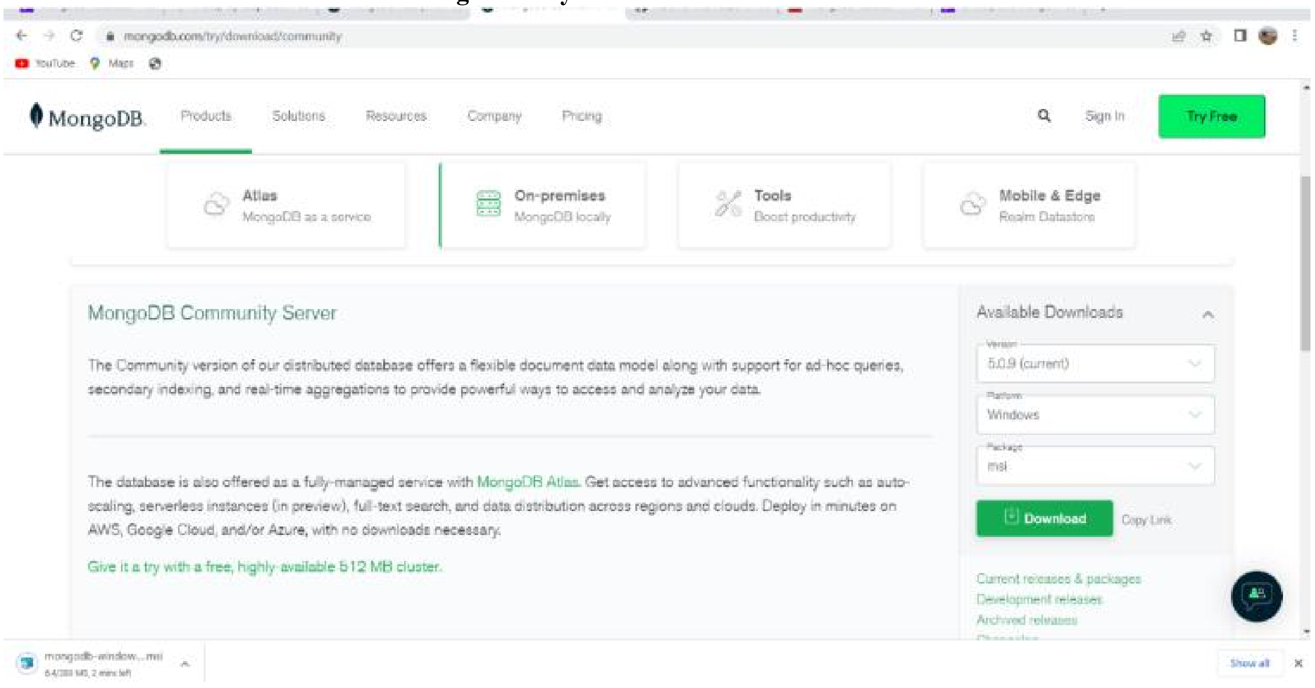
AIM: Installation and configuration of NoSQL database- MongoDB

MongoDB is a cross-platform, document oriented NoSql database that provides, high performance, high availability, and easy scalability. MongoDB works on concept of collection and document.

STEP 1:
Navigate to the official MongoDB website.


mongodb.com
Event Bring a friend (or 2!) and save 50% on passes for MongoDB World! Register now and save >>
MongoDB.
Products
Solutions
Resources
Company
Pricing
Sign In
Try Free
NEW View the latest product announcements – Learn more
Build faster.
Build smarter.
Get your ideas to market faster with a developer data platform built on the leading modern database. Support transactional, search, analytics, and mobile use cases while
_id:

STEP 2:
Under the products section, click on the Community server version. Make sure that the specifications to the right of the screen are correct. At the time of writing, the latest version is 4.4.5. Ensure that the platform is Windows, and the package is MSI. Go ahead and click on download.



MongoDB Community Server

The Community version of our distributed database offers a flexible document data model along with support for ad-hoc queries, secondary indexing, and real-time aggregations to provide powerful ways to access and analyze your data.

The database is also offered as a fully-managed service with MongoDB Atlas. Get access to advanced functionality such as auto-scaling, serverless instances (in preview), full-text search, and data distribution across regions and clouds. Deploy in minutes on AWS, Google Cloud, and/or Azure, with no downloads necessary.

Give it a try with a free, highly-available 512 MB cluster.

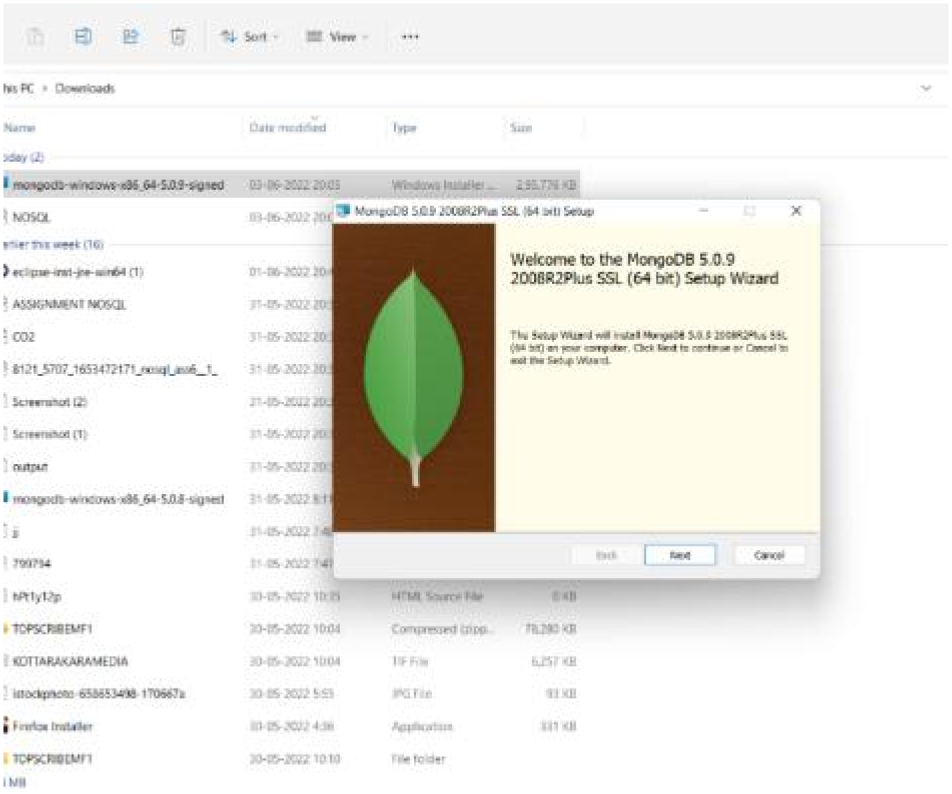
Available Downloads

Version: 5.0.9 (current)

Platform: Windows

Package: msi

STEP 3:
You can find the downloaded file in the downloads directory. Install the software step by step.

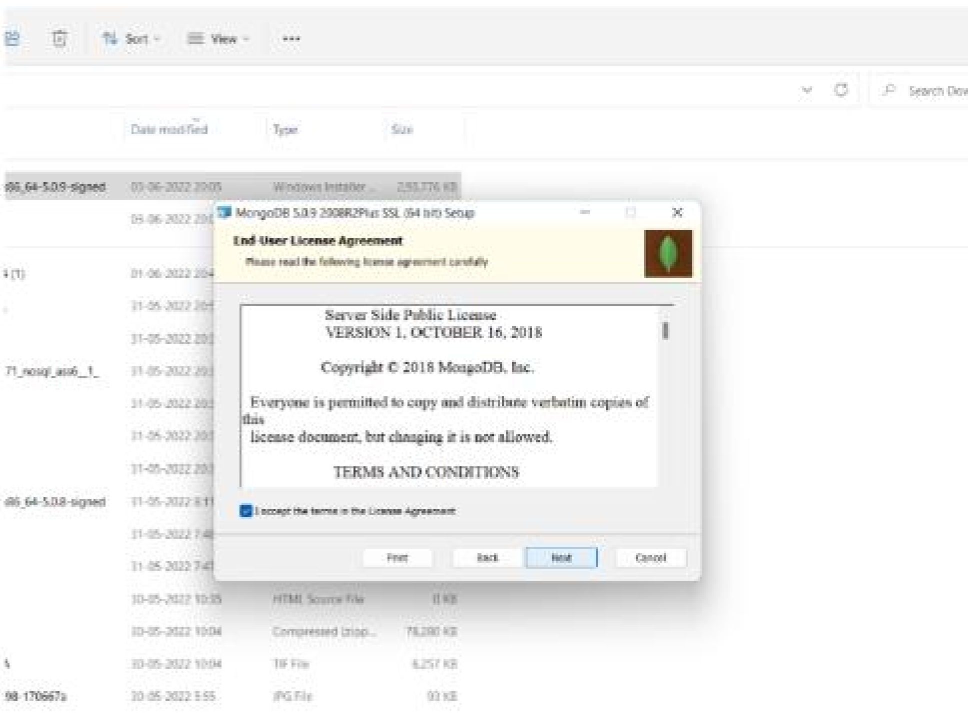


Name	Date modified	Type	Size
mongodb-windows-x86_64-5.0.9-signed	03-06-2022 20:03	Windows Installer ...	2,95,776 KB
NOSQL	03-06-2022 20:		
eclipse-inst-jre-win64 (1)	01-06-2022 20:		
ASSIGNMENT NOSQL	31-05-2022 20:		
CO2	31-05-2022 20:		
8121_5707_1653472171_nosql_ass6__1_	31-05-2022 20:		
Screenshot (2)	31-05-2022 20:		
Screenshot (1)	31-05-2022 20:		
output	31-05-2022 20:		
mongodb-windows-x86_64-5.0.8-signed	31-05-2022 8:1		
ji	31-05-2022 7:4		
799794	31-05-2022 7:4		
hPt1y12p	30-05-2022 10:35	HTML Source File	8 KB
TOPSCRIBEMF1	30-05-2022 10:04	Compressed (zipp...	78,280 KB
KOTTARAKARAMEDIA	30-05-2022 10:04	TIF File	6,257 KB
istockphoto-658653498-170667a	30-05-2022 5:55	JPG File	93 KB
Firefox Installer	30-05-2022 4:36	Application	331 KB
TOPSCRIBEMF1	30-05-2022 10:10	File folder	

MongoDB 5.0.9 2008R2Plus SSL (64 bit) Setup

Welcome to the MongoDB 5.0.9 2008R2Plus SSL (64 bit) Setup Wizard

The Setup Wizard will install MongoDB 5.0.9 2008R2Plus SSL (64 bit) on your computer. Click Next to continue or Cancel to exit the Setup Wizard.



MongoDB 5.0.9 2008R2Plus SSL (64 bit) Setup

End-User License Agreement

Please read the following license agreement carefully

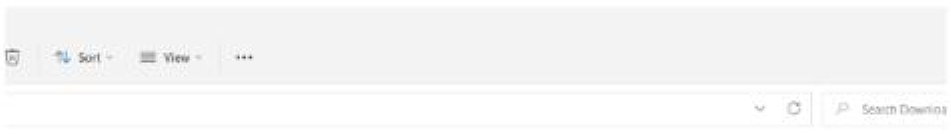
Server Side Public License
VERSION 1, OCTOBER 16, 2018

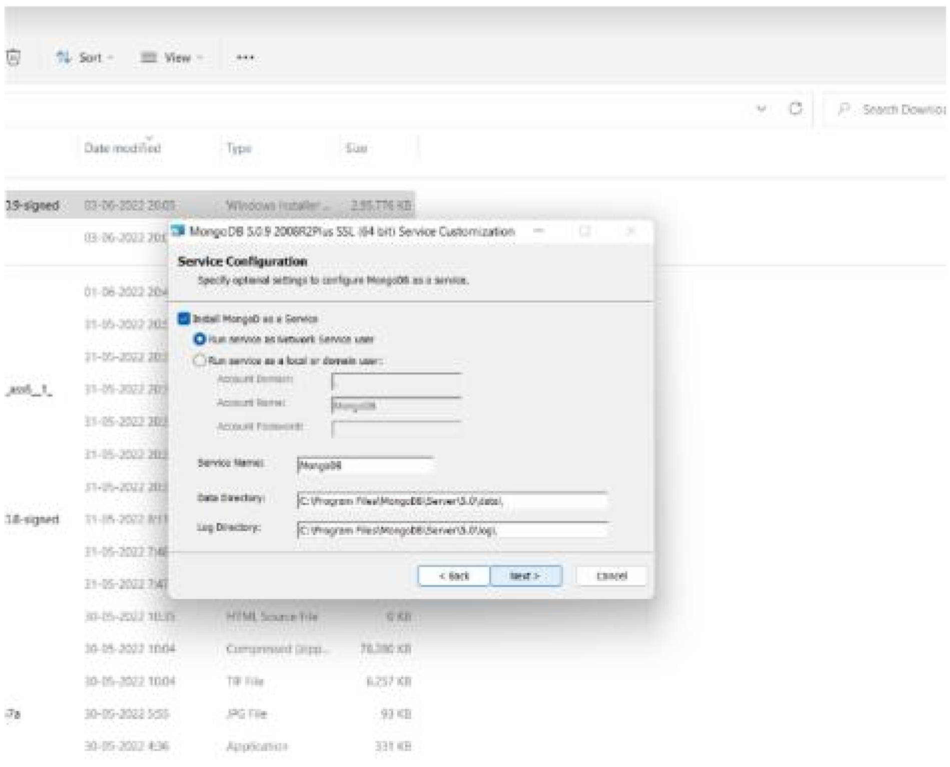
Copyright © 2018 MongoDB, Inc.

Everyone is permitted to copy and distribute verbatim copies of this
license document, but changing it is not allowed.

TERMS AND CONDITIONS

☑ I accept the terms in the License Agreement





MongoDB 5.0.9 2008R2Plus SSL (64 bit) Service Customization

Service Configuration

Specify optional settings to configure MongoDB as a service.

☑ Install MongoD as a Service

● Run service as Network Service user

○ Run service as a local or domain user:

Account Domain: .

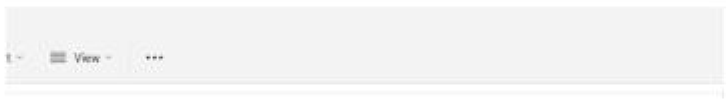
Account Name: MongoDB

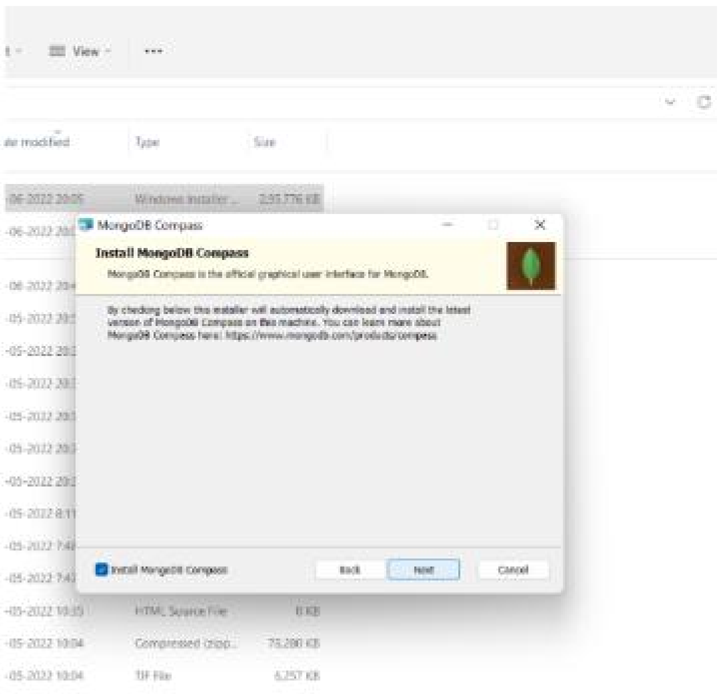
Account Password:

Service Name: MongoDB

Data Directory: C:\Program Files\MongoDB\Server\5.0\data\

Log Directory: C:\Program Files\MongoDB\Server\5.0\log\



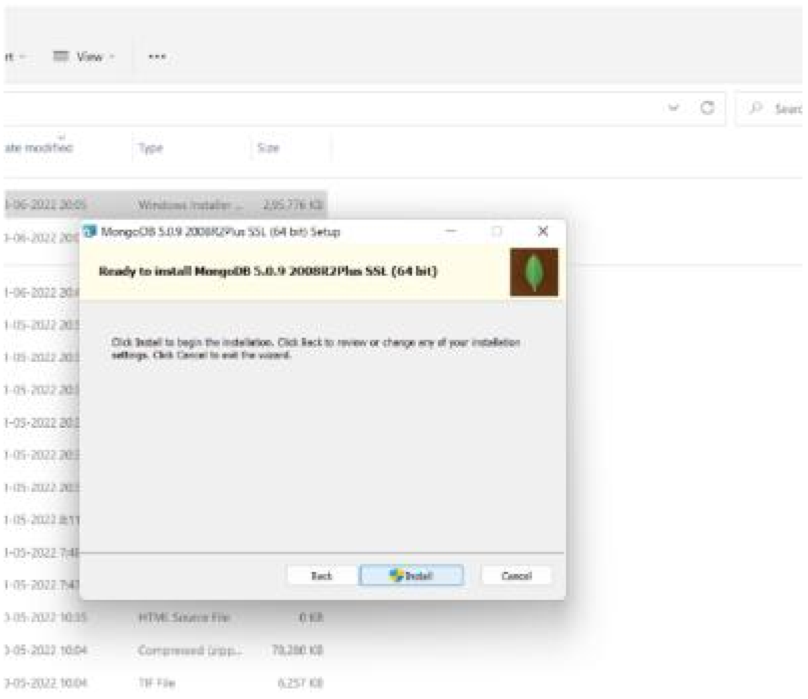

MongoDB Compass

Install MongoDB Compass

MongoDB Compass is the official graphical user interface for MongoDB.

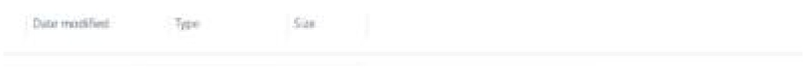
By checking below this installer will automatically download and install the latest version of MongoDB Compass on this machine. You can learn more about MongoDB Compass here: https://www.mongodb.com/products/compass

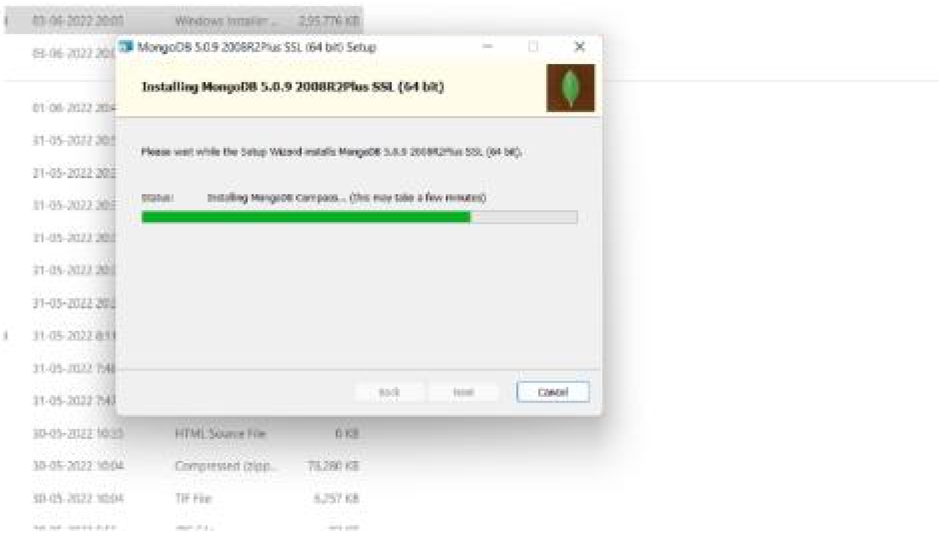
☑ Install MongoDB Compass


MongoDB 5.0.9 2008R2Plus SSL (64 bit) Setup

Ready to install MongoDB 5.0.9 2008R2Plus SSL (64 bit)

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.



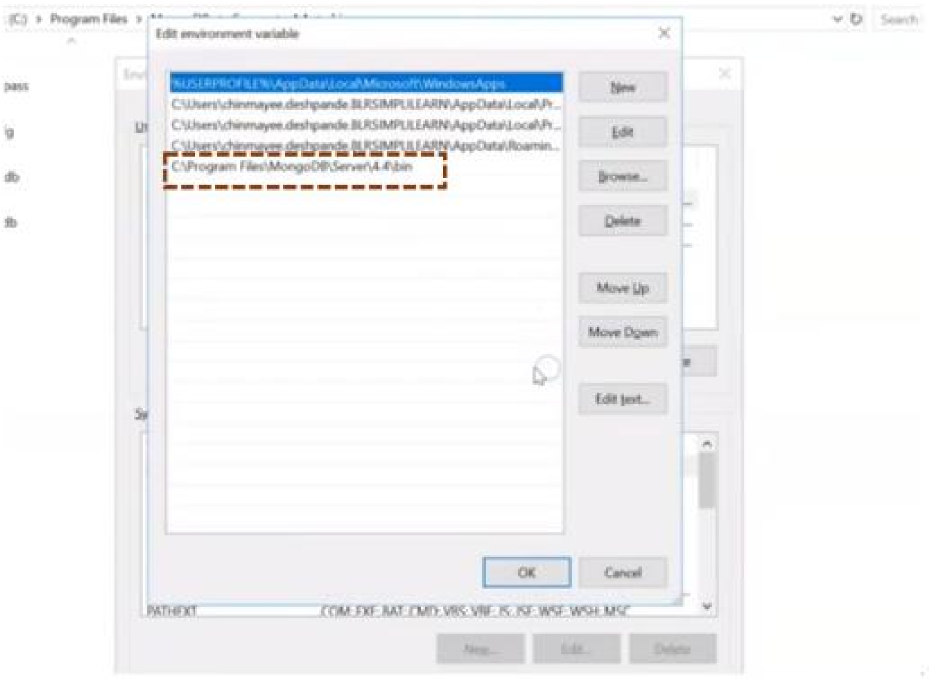


Installing MongoDB 5.0.9 2008R2Plus SSL (64 bit)

Please wait while the Setup Wizard installs MongoDB 5.0.9 2008R2Plus SSL (64 bit).

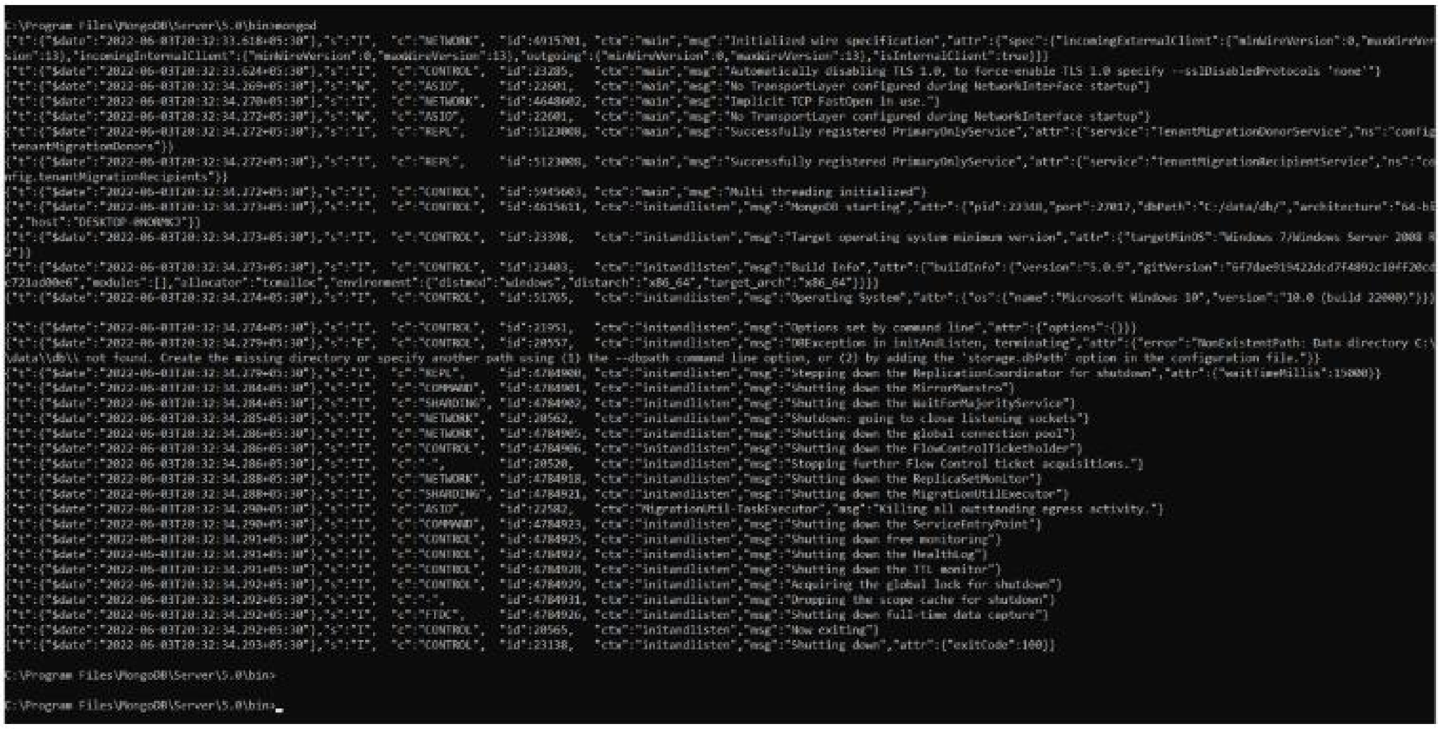
Status: Installing MongoDB compass... (this may take a few minutes)

STEP4:
create an environment variable for the executable file so that we don't have to change the directory structure every time we want to execute the file.



```
%USERPROFILE%\AppData\Local\Microsoft\WindowsApps
C:\Users\chinmayee.deshpande.BLRSIMPLILEARN\AppData\Local\Pr...
C:\Users\chinmayee.deshpande.BLRSIMPLILEARN\AppData\Local\Pr...
C:\Users\chinmayee.deshpande.BLRSIMPLILEARN\AppData\Roamin...
C:\Program Files\MongoDB\Server\4.4\bin
```

STEP 5:
After creating an environment path, download mongosh and install. we can open the command prompt and type mongod. An instance of mongodb server is started. Now take another terminal and type mongosh. This creates a client instance of mongodb in your local system.

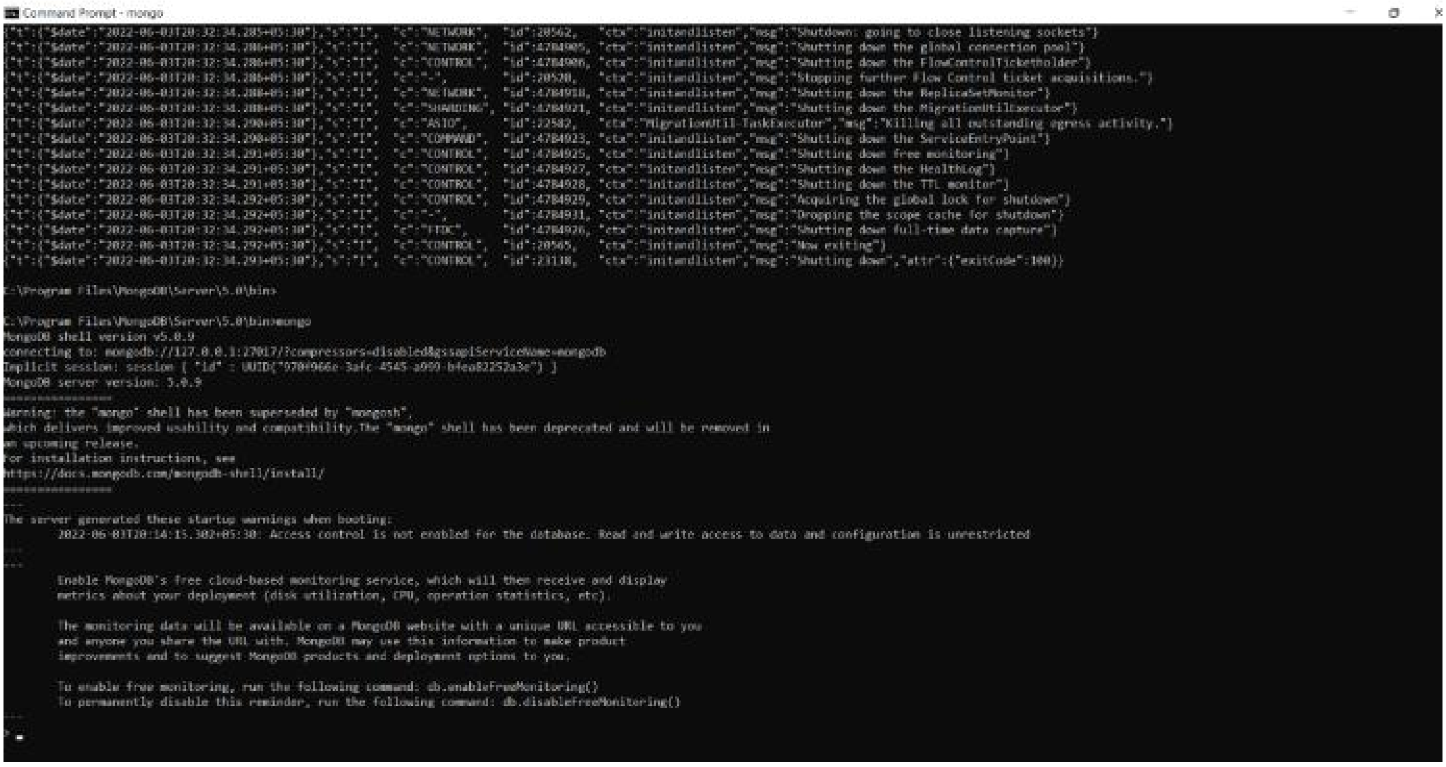


```
C:\Program Files\MongoDB\Server\5.0\bin>mongod
{"t":{"$date":"2022-06-03T20:32:33.618+05:30"},"s":"I",  "c":"NETWORK",  "id":4915701, "ctx":"main","msg":"Initialized wire specification","attr":{"spec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVer
sion":13},"incomingInternalClient":{"minWireVersion":0,"maxWireVersion":13},"outgoing":{"minWireVersion":0,"maxWireVersion":13},"isInternalClient":true}}}
{"t":{"$date":"2022-06-03T20:32:33.624+05:30"},"s":"I",  "c":"CONTROL",  "id":23285,   "ctx":"main","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2022-06-03T20:32:34.269+05:30"},"s":"W",  "c":"ASIO",     "id":22601,   "ctx":"main","msg":"No TransportLayer configured during NetworkInterface startup"}
{"t":{"$date":"2022-06-03T20:32:34.270+05:30"},"s":"I",  "c":"NETWORK",  "id":4648602, "ctx":"main","msg":"Implicit TCP FastOpen in use."}
{"t":{"$date":"2022-06-03T20:32:34.272+05:30"},"s":"W",  "c":"ASIO",     "id":22601,   "ctx":"main","msg":"No TransportLayer configured during NetworkInterface startup"}
{"t":{"$date":"2022-06-03T20:32:34.272+05:30"},"s":"I",  "c":"REPL",     "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"TenantMigrationDonorService","ns":"config
.tenantMigrationDonors"}}
{"t":{"$date":"2022-06-03T20:32:34.272+05:30"},"s":"I",  "c":"REPL",     "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"TenantMigrationRecipientService","ns":"co
nfig.tenantMigrationRecipients"}}
{"t":{"$date":"2022-06-03T20:32:34.272+05:30"},"s":"I",  "c":"CONTROL",  "id":5945603, "ctx":"main","msg":"Multi threading initialized"}
{"t":{"$date":"2022-06-03T20:32:34.273+05:30"},"s":"I",  "c":"CONTROL",  "id":4615611, "ctx":"initandlisten","msg":"MongoDB starting","attr":{"pid":22348,"port":27017,"dbPath":"C:/data/db/","architecture":"64-bi
t","host":"DESKTOP-0MORMKJ"}}
{"t":{"$date":"2022-06-03T20:32:34.273+05:30"},"s":"I",  "c":"CONTROL",  "id":23398,   "ctx":"initandlisten","msg":"Target operating system minimum version","attr":{"targetMinOS":"Windows 7/Windows Server 2008 R
2"}}
{"t":{"$date":"2022-06-03T20:32:34.273+05:30"},"s":"I",  "c":"CONTROL",  "id":23403,   "ctx":"initandlisten","msg":"Build Info","attr":{"buildInfo":{"version":"5.0.9","gitVersion":"6f7dae919422dcd7f4892c10ff20cd
c721ad00e6","modules":[],"allocator":"tcmalloc","environment":{"distmod":"windows","distarch":"x86_64","target_arch":"x86_64"}}}}
{"t":{"$date":"2022-06-03T20:32:34.274+05:30"},"s":"I",  "c":"CONTROL",  "id":51765,   "ctx":"initandlisten","msg":"Operating System","attr":{"os":{"name":"Microsoft Windows 10","version":"10.0 (build 22000)"}}}

{"t":{"$date":"2022-06-03T20:32:34.274+05:30"},"s":"I",  "c":"CONTROL",  "id":21951,   "ctx":"initandlisten","msg":"Options set by command line","attr":{"options":{}}}
{"t":{"$date":"2022-06-03T20:32:34.279+05:30"},"s":"E",  "c":"CONTROL",  "id":20557,   "ctx":"initandlisten","msg":"DBException in initAndListen, terminating","attr":{"error":"NonExistentPath: Data directory C:\
\data\\db\\ not found. Create the missing directory or specify another path using (1) the --dbpath command line option, or (2) by adding the 'storage.dbPath' option in the configuration file."}}
{"t":{"$date":"2022-06-03T20:32:34.279+05:30"},"s":"I",  "c":"REPL",     "id":4784900, "ctx":"initandlisten","msg":"Stepping down the ReplicationCoordinator for shutdown","attr":{"waitTimeMillis":15000}}
{"t":{"$date":"2022-06-03T20:32:34.284+05:30"},"s":"I",  "c":"COMMAND",  "id":4784901, "ctx":"initandlisten","msg":"Shutting down the MirrorMaestro"}
{"t":{"$date":"2022-06-03T20:32:34.284+05:30"},"s":"I",  "c":"SHARDING", "id":4784902, "ctx":"initandlisten","msg":"Shutting down the WaitForMajorityService"}
{"t":{"$date":"2022-06-03T20:32:34.285+05:30"},"s":"I",  "c":"NETWORK",  "id":20562,   "ctx":"initandlisten","msg":"Shutdown: going to close listening sockets"}
{"t":{"$date":"2022-06-03T20:32:34.286+05:30"},"s":"I",  "c":"NETWORK",  "id":4784905, "ctx":"initandlisten","msg":"Shutting down the global connection pool"}
{"t":{"$date":"2022-06-03T20:32:34.286+05:30"},"s":"I",  "c":"CONTROL",  "id":4784906, "ctx":"initandlisten","msg":"Shutting down the FlowControlTicketholder"}
{"t":{"$date":"2022-06-03T20:32:34.286+05:30"},"s":"I",  "c":"-",        "id":20520,   "ctx":"initandlisten","msg":"Stopping further Flow Control ticket acquisitions."}
{"t":{"$date":"2022-06-03T20:32:34.288+05:30"},"s":"I",  "c":"NETWORK",  "id":4784918, "ctx":"initandlisten","msg":"Shutting down the ReplicaSetMonitor"}
{"t":{"$date":"2022-06-03T20:32:34.288+05:30"},"s":"I",  "c":"SHARDING", "id":4784921, "ctx":"initandlisten","msg":"Shutting down the MigrationUtilExecutor"}
{"t":{"$date":"2022-06-03T20:32:34.290+05:30"},"s":"I",  "c":"ASIO",     "id":22582,   "ctx":"MigrationUtil-TaskExecutor","msg":"Killing all outstanding egress activity."}
{"t":{"$date":"2022-06-03T20:32:34.290+05:30"},"s":"I",  "c":"COMMAND",  "id":4784923, "ctx":"initandlisten","msg":"Shutting down the ServiceEntryPoint"}
{"t":{"$date":"2022-06-03T20:32:34.291+05:30"},"s":"I",  "c":"CONTROL",  "id":4784925, "ctx":"initandlisten","msg":"Shutting down free monitoring"}
{"t":{"$date":"2022-06-03T20:32:34.291+05:30"},"s":"I",  "c":"CONTROL",  "id":4784927, "ctx":"initandlisten","msg":"Shutting down the HealthLog"}
{"t":{"$date":"2022-06-03T20:32:34.291+05:30"},"s":"I",  "c":"CONTROL",  "id":4784928, "ctx":"initandlisten","msg":"Shutting down the TTL monitor"}
{"t":{"$date":"2022-06-03T20:32:34.292+05:30"},"s":"I",  "c":"CONTROL",  "id":4784929, "ctx":"initandlisten","msg":"Acquiring the global lock for shutdown"}
{"t":{"$date":"2022-06-03T20:32:34.292+05:30"},"s":"I",  "c":"-",        "id":4784931, "ctx":"initandlisten","msg":"Dropping the scope cache for shutdown"}
{"t":{"$date":"2022-06-03T20:32:34.292+05:30"},"s":"I",  "c":"FTDC",     "id":4784926, "ctx":"initandlisten","msg":"Shutting down full-time data capture"}
{"t":{"$date":"2022-06-03T20:32:34.292+05:30"},"s":"I",  "c":"CONTROL",  "id":20565,   "ctx":"initandlisten","msg":"Now exiting"}
{"t":{"$date":"2022-06-03T20:32:34.293+05:30"},"s":"I",  "c":"CONTROL",  "id":23138,   "ctx":"initandlisten","msg":"Shutting down","attr":{"exitCode":100}}

C:\Program Files\MongoDB\Server\5.0\bin>

C:\Program Files\MongoDB\Server\5.0\bin>_
```



Command Prompt - mongo

```
{"t":{"$date":"2022-06-03T20:32:34.285+05:30"},"s":"I",  "c":"NETWORK",  "id":20562,   "ctx":"initandlisten","msg":"Shutdown: going to close listening sockets"}
{"t":{"$date":"2022-06-03T20:32:34.286+05:30"},"s":"I",  "c":"NETWORK",  "id":4784905, "ctx":"initandlisten","msg":"Shutting down the global connection pool"}
{"t":{"$date":"2022-06-03T20:32:34.286+05:30"},"s":"I",  "c":"CONTROL",  "id":4784906, "ctx":"initandlisten","msg":"Shutting down the FlowControlTicketholder"}
{"t":{"$date":"2022-06-03T20:32:34.286+05:30"},"s":"I",  "c":"-",        "id":20520,   "ctx":"initandlisten","msg":"Stopping further Flow Control ticket acquisitions."}
{"t":{"$date":"2022-06-03T20:32:34.288+05:30"},"s":"I",  "c":"NETWORK",  "id":4784918, "ctx":"initandlisten","msg":"Shutting down the ReplicaSetMonitor"}
{"t":{"$date":"2022-06-03T20:32:34.288+05:30"},"s":"I",  "c":"SHARDING", "id":4784921, "ctx":"initandlisten","msg":"Shutting down the MigrationUtilExecutor"}
{"t":{"$date":"2022-06-03T20:32:34.290+05:30"},"s":"I",  "c":"ASIO",     "id":22582,   "ctx":"MigrationUtil-TaskExecutor","msg":"Killing all outstanding egress activity."}
{"t":{"$date":"2022-06-03T20:32:34.290+05:30"},"s":"I",  "c":"COMMAND",  "id":4784923, "ctx":"initandlisten","msg":"Shutting down the ServiceEntryPoint"}
{"t":{"$date":"2022-06-03T20:32:34.291+05:30"},"s":"I",  "c":"CONTROL",  "id":4784925, "ctx":"initandlisten","msg":"Shutting down free monitoring"}
{"t":{"$date":"2022-06-03T20:32:34.291+05:30"},"s":"I",  "c":"CONTROL",  "id":4784927, "ctx":"initandlisten","msg":"Shutting down the HealthLog"}
{"t":{"$date":"2022-06-03T20:32:34.291+05:30"},"s":"I",  "c":"CONTROL",  "id":4784928, "ctx":"initandlisten","msg":"Shutting down the TTL monitor"}
{"t":{"$date":"2022-06-03T20:32:34.292+05:30"},"s":"I",  "c":"CONTROL",  "id":4784929, "ctx":"initandlisten","msg":"Acquiring the global lock for shutdown"}
{"t":{"$date":"2022-06-03T20:32:34.292+05:30"},"s":"I",  "c":"-",        "id":4784931, "ctx":"initandlisten","msg":"Dropping the scope cache for shutdown"}
{"t":{"$date":"2022-06-03T20:32:34.292+05:30"},"s":"I",  "c":"FTDC",     "id":4784926, "ctx":"initandlisten","msg":"Shutting down full-time data capture"}
{"t":{"$date":"2022-06-03T20:32:34.292+05:30"},"s":"I",  "c":"CONTROL",  "id":20565,   "ctx":"initandlisten","msg":"Now exiting"}
{"t":{"$date":"2022-06-03T20:32:34.293+05:30"},"s":"I",  "c":"CONTROL",  "id":23138,   "ctx":"initandlisten","msg":"Shutting down","attr":{"exitCode":100}}

C:\Program Files\MongoDB\Server\5.0\bin>

C:\Program Files\MongoDB\Server\5.0\bin>mongo
MongoDB shell version v5.0.9
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("970f966e-3afc-4545-a099-b4ea82252a3e") }
MongoDB server version: 5.0.9
================
Warning: the "mongo" shell has been superseded by "mongosh",
which delivers improved usability and compatibility.The "mongo" shell has been deprecated and will be removed in
an upcoming release.
For installation instructions, see
https://docs.mongodb.com/mongodb-shell/install/
================
---
The server generated these startup warnings when booting:
        2022-06-03T20:14:15.302+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
---
---
        Enable MongoDB's free cloud-based monitoring service, which will then receive and display
        metrics about your deployment (disk utilization, CPU, operation statistics, etc).

        The monitoring data will be available on a MongoDB website with a unique URL accessible to you
        and anyone you share the URL with. MongoDB may use this information to make product
        improvements and to suggest MongoDB products and deployment options to you.

        To enable free monitoring, run the following command: db.enableFreeMonitoring()
        To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
> _
```

Step 6:
You can start creating new databases and use them.

PROGRAM 13

AIM: Compare relational and non-relational databases.

1.Create a student table/collection with columns/fields for name, age, and city.

SQL

```
CREATE TABLE students (sname VARCHAR(100),sage INT,scity VARCHAR(100));
```

MongoDB

```
test> use college
switched to db college
college> db.createCollection("students")
 { ok: 1 }
```

2.Insert a new row/document into the students table/collection with the following information: Name: John Doe, Age: 25, City:New York.

SQL

```
INSERT INTO students (sname, sage, scity) VALUES ('Akhila','22','Kochi');
INSERT INTO students (sname, sage, scity) VALUES ('Anu','23','Kottayam');
INSERT INTO students (sname, sage, scity) VALUES ('Binu','22','Ernakulam');
INSERT INTO students (sname, sage, scity) VALUES ('Pooja','21','Alappuzha');
INSERT INTO students (sname, sage, scity) VALUES ('Jency','20','Pattimattam');
INSERT INTO students (sname, sage, scity) VALUES ('John Doe', 25, 'New York');
INSERT INTO students (sname, sage, scity) VALUES ('Vishal','23','Piravom');
```

MongoDB

```
college> db.students.insert({sname:'Akhila',sage:'22',scity:'Kochi'})
{
acknowledged: true,
insertedIds: { '0': ObjectId('6602339b68ed4179038bf202') }
}
 college> db.students.insert({sname:'Anu',sage:'23',scity:'Kottayam'})
{
acknowledged: true,
insertedIds: { '0': ObjectId('6602339b68ed4179038bf203') }
}
 college> db.students.insert({sname:'Binu',sage:'22',scity:'Ernakulam'})
{
 acknowledged: true,
insertedIds: { '0': ObjectId('6602339b68ed4179038bf204') }
}
 college> db.students.insert({sname:'Pooja',sage:'21',scity:'Alappuzha'})
{
 acknowledged: true, insertedIds: { '0': ObjectId('6602339b68ed4179038bf205') }
 }
 college> db.students.insert({sname:'Jency',sage:'20',scity:'Pattimattam'})
```

```
{

 acknowledged: true,
 insertedIds: { '0': ObjectId('6602339b68ed4179038bf206') }
}
 college> db.students.insert({sname:'John Doe',sage:'25',scity:'New York'})
{
 acknowledged: true,
insertedIds: { '0': ObjectId('6602349268ed4179038bf207') }
}
 college> db.students.insert({sname:'Vishal',sage:'23',scity:'Piravom'})
{
acknowledged: true, insertedIds: { '0': ObjectId('6602349268ed4179038bf207') }
}
```

3.Retrieve all data from the students table/collection.

SQL
SELECT * FROM students;

MongoDB

```
college> db.students.find().pretty()
[
  {
  _id: ObjectId('6602339b68ed4179038bf202'),
  sname: 'Akhila',
  sage: '22',
  scity: 'Kochi'
  },
    {
     _id: ObjectId('6602339b68ed4179038bf203'),
    sname: 'Anu',
    sage: '23',
    scity: 'Kottayam'
    },
    {
     _id: ObjectId('6602339b68ed4179038bf204'),
     sname: 'Binu',
    sage: '22',
    scity: 'Ernakulam'
    },
     {
     _id: ObjectId('6602339b68ed4179038bf205'),
    sname: 'Pooja',
    sage: '21',
    scity: 'Alappuzha'
    },
    {
     _id: ObjectId('6602339b68ed4179038bf206'),
     sname: 'Jency',
 sage: '20',
scity: 'Pattimattam'

},
 {
 _id: ObjectId('6602349268ed4179038bf207'),
sname: 'John Doe',
sage: '25',
scity: 'New York'
}
{
 _id: ObjectId('6602429168ed4179038bf208'),
sname: 'Vishal',
sage: '23',
scity: 'Piravom'
}
]
```

4.Update the age of a student named John Doe to 30 in the students table/collection, assuming there's already a record/document for him.

SQL

```
UPDATE students SET age = 30 WHERE name = 'John Doe';
```

MongoDB

```
college> db.students.update({sname:"John Doe"},{$set:{sage:30}})
DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or
 bulkWrite.
   {
   acknowledged: true,
   insertedId: null,
   matchedCount: 1,
   modifiedCount: 1,
   upsertedCount: 0
   }
   college> db.students.find()
   [
     {
     _id: ObjectId('6602339b68ed4179038bf202'),
     sname: 'Akhila',
     sage: '22',
     scity: 'Kochi'
     },
     {
       _id: ObjectId('6602339b68ed4179038bf203'),
       sname: 'Anu',
       sage: '23',
       scity: 'Kottayam'
     },
   {

    _id: ObjectId('6602339b68ed4179038bf204'),
    sname: 'Binu',
   sage: '22',

   scity: 'Ernakulam'
   },
   {
    _id: ObjectId('6602339b68ed4179038bf205'),
   sname: 'Pooja',
   sage: '21',
   scity: 'Alappuzha'
   },
   {
    _id: ObjectId('6602339b68ed4179038bf206'),
   sname: 'Jency',
   sage: '20',
   scity: 'Pattimattam'
   },
   {
    _id: ObjectId('6602349268ed4179038bf207'),
     sname: 'John Doe',
     sage: 30,
      scity: 'New York'
   },
   {
   _id: ObjectId('6602429168ed4179038bf208'),
   sname: 'Vishal',
   sage: '23',
   scity: 'Piravom'
   }
  ]
```

5.Get all details of students whose age is older than 25.

SQL

```
SELECT * FROM students WHERE age > 25;
```

MongoDB

```
college> db.students.find({sage:{$gt:25}})
[
  {
  _id: ObjectId('6602349268ed4179038bf207'),
   sname: 'John Doe',
  sage: 30,
  scity: 'New York'
  ]
```

6.Get the name of students whose names begin with the letter 'V'.

SQL

```
SELECT name FROM students WHERE name LIKE 'V%';
```

MongoDB

```
college> db.students.find({sname:/^V/},{"sname":1})
[
   {
   _id: ObjectId('6602429168ed4179038bf208'),
  sname: 'Vishal'
  }
   ]
```