***Database Management Systems Lab***

***ETCS-256***

Faculty Name: Ms. Prerna Student name: Ria Monga

Roll no.: 13514802719

Semester: 4th



Maharaja Agrasen Institute of Technology, PSP area,

Sector – 22, Rohini, New Delhi – 110085

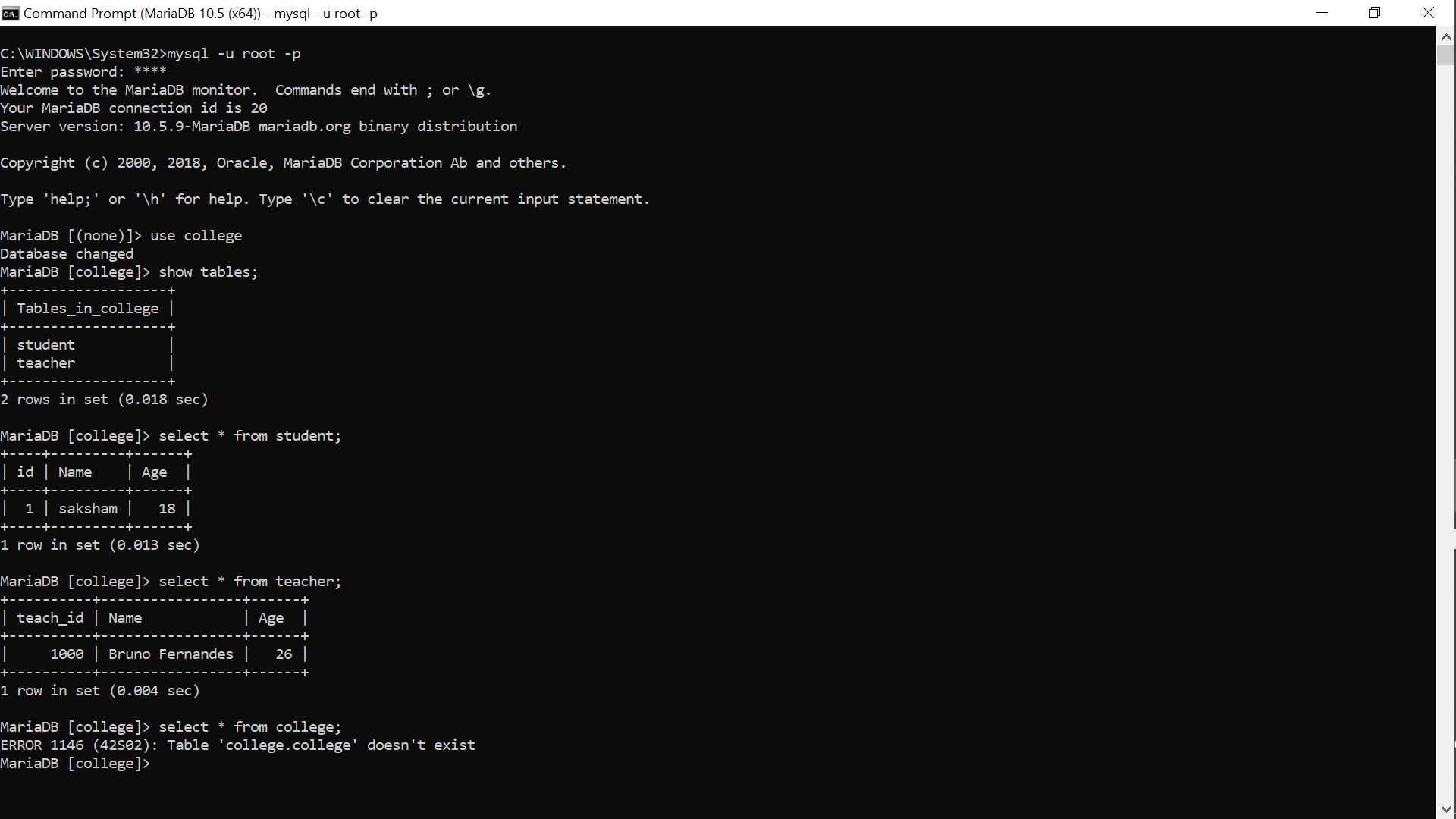
(Affiliated to Guru Gobind Singh Indraprastha University, New Delhi)

EXPERIMENT 1

**Aim:** Creating Database tables and performing the operation of table creations, insert data and fetch data.

1)cmd: CREATE DATABASE COLLEGE;

2)cmd: USE COLLEGE;



3)cmd: CREATE TABLE Students (stud\_id int,LastName varchar(255),FirstName varchar(255),Address varchar(255),City varchar(255));

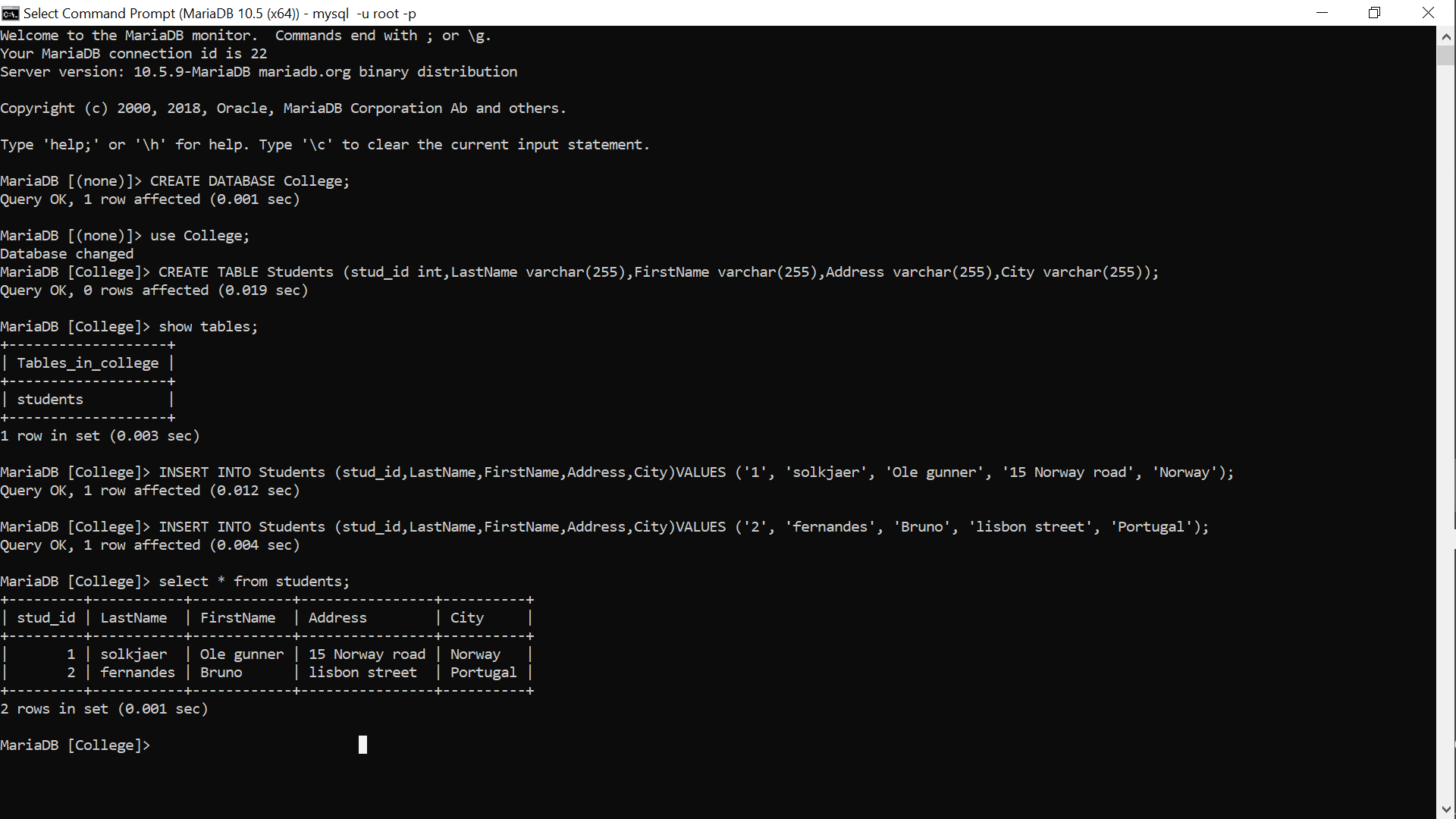
4)cmd: SHOW TABLES;



5)cmd: INSERT INTO Students (stud\_id,LastName,FirstName,Address,City)VALUES ('1', 'solkjaer', 'Ole gunner', '15 Norway road', 'Norway');

6)cmd: INSERT INTO Students (stud\_id,LastName,FirstName,Address,City)VALUES ('2', 'fernandes', 'Bruno', 'lisbon street', 'Portugal');

7)cmd: SELECT \* FROM Students;



EXPERIMENT 2

**Aim:** Understanding PRIMARY KEY FOREIGN KEY AND NOT NULL

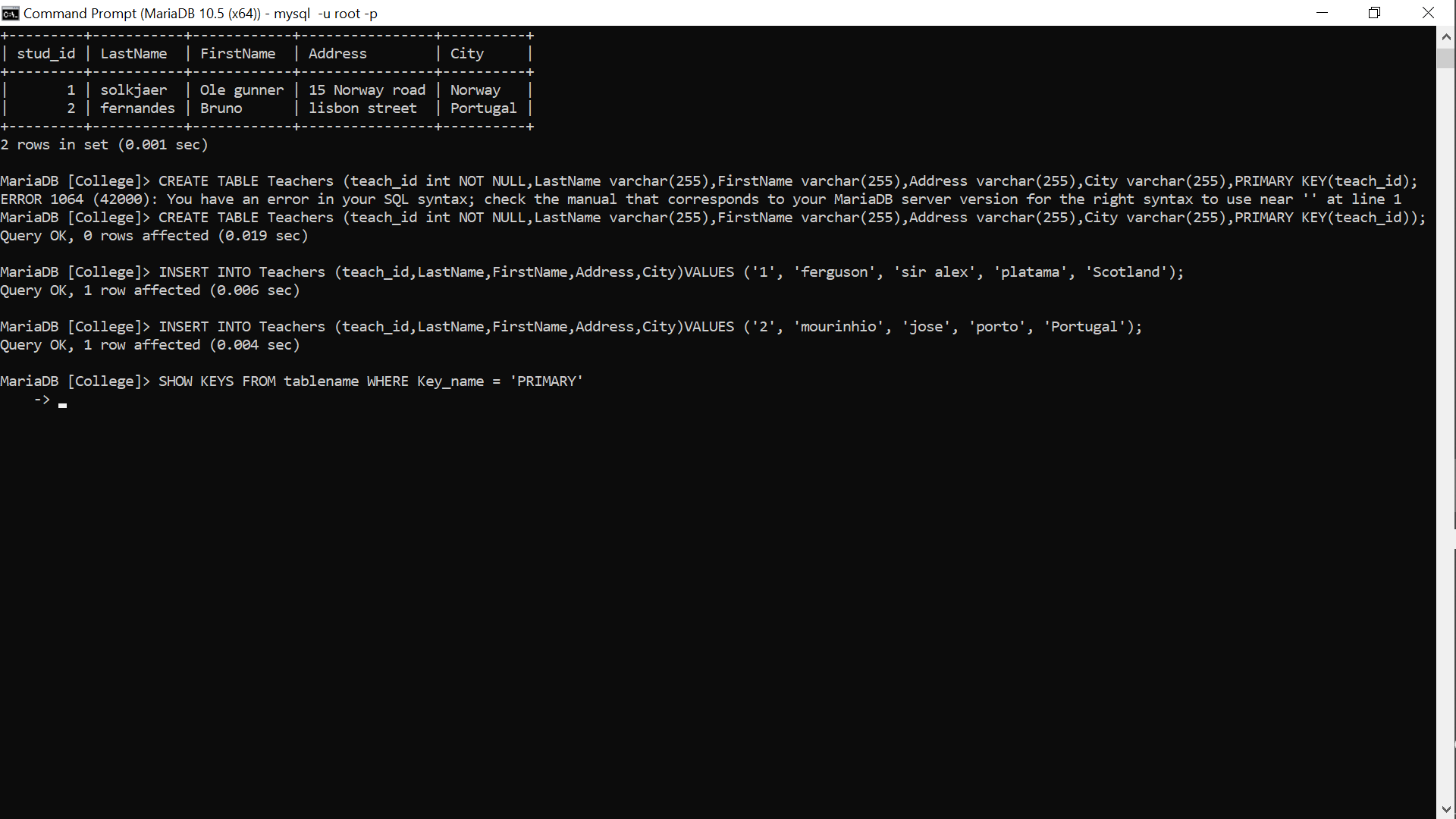
Database already created named college;

1)cmd): use College;

2)cmd: CREATE TABLE Teachers (teach\_id int NOT NULL,LastName varchar(255),FirstName varchar(255),Address varchar(255),City varchar(255),PRIMARY KEY(teach\_id);

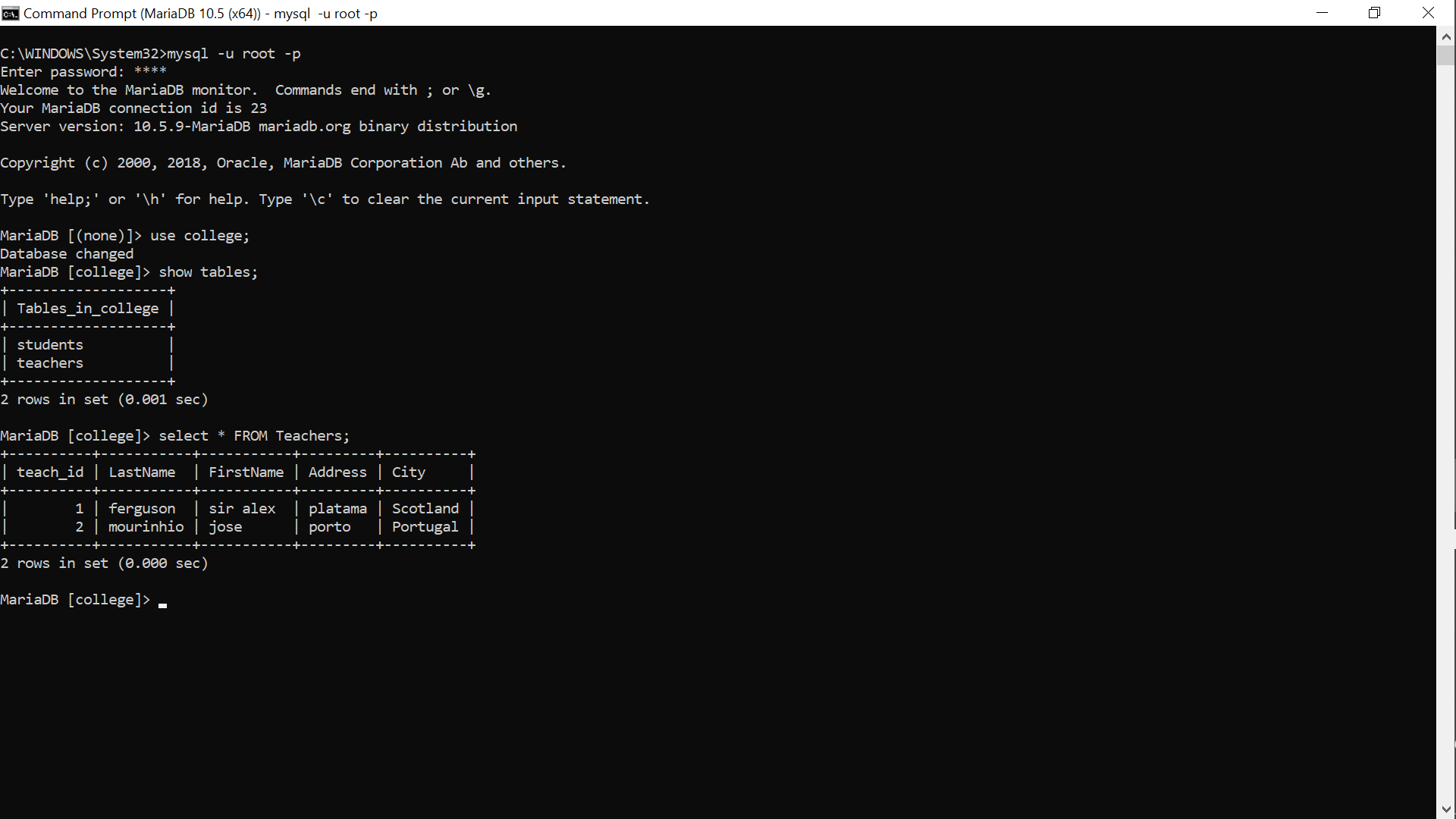
3)cmd: INSERT INTO Teachers (teach\_id,LastName,FirstName,Address,City)VALUES ('1', 'ferguson', 'sir alex', 'platama', 'Scotland');

4)cmd: INSERT INTO Teachers (teach\_id,LastName,FirstName,Address,City)VALUES ('2', 'mourinhio', 'jose', 'porto', 'Portugal');

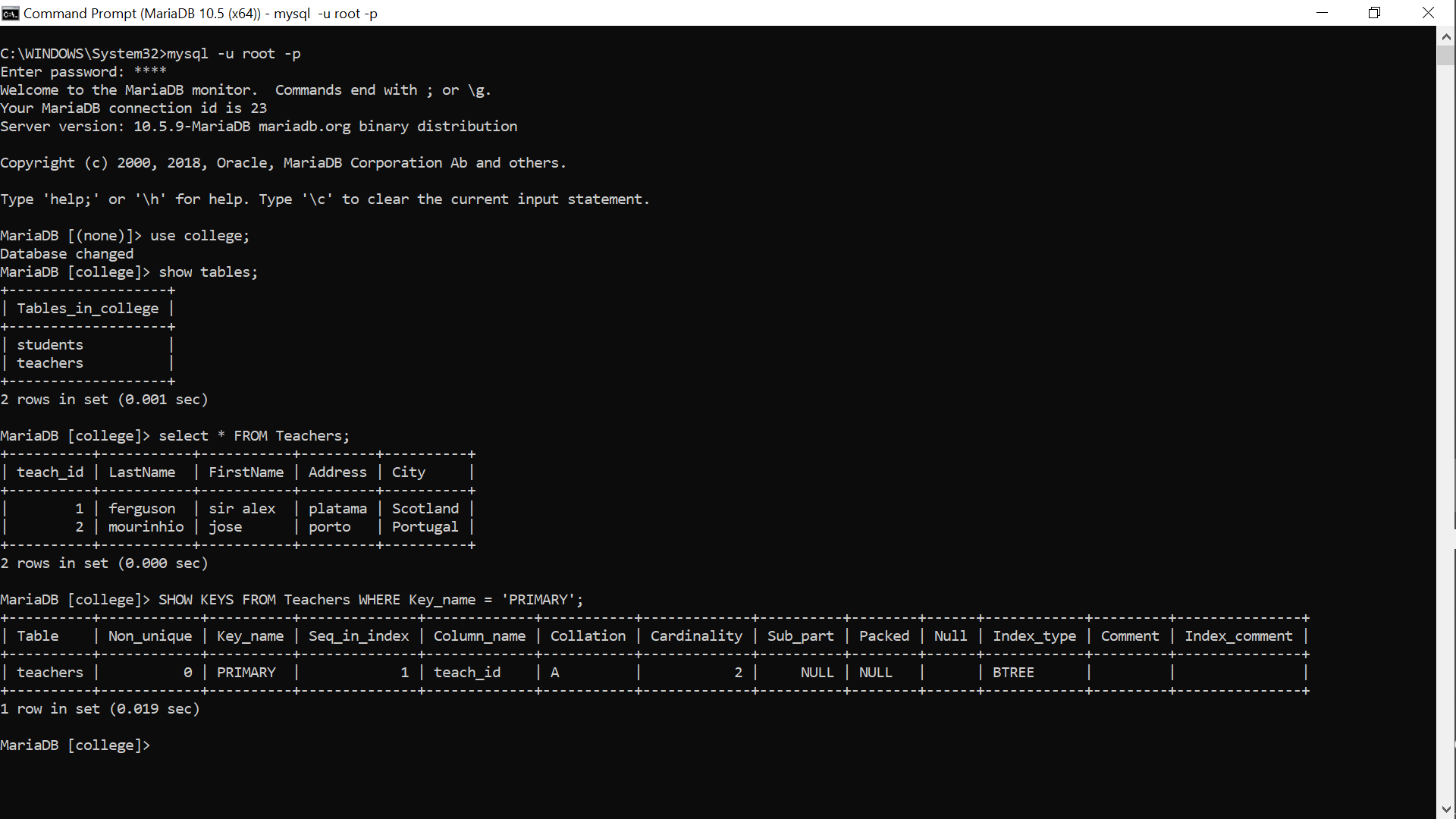


5)cmd: show tables;

6): SELECT \* FROM TABLES;



7): SHOW KEYS FROM Teachers WHERE Key\_name = 'PRIMARY';



Here we can see the implementation of PRIMARY KEY AND NOT NULL CONSTRAINS

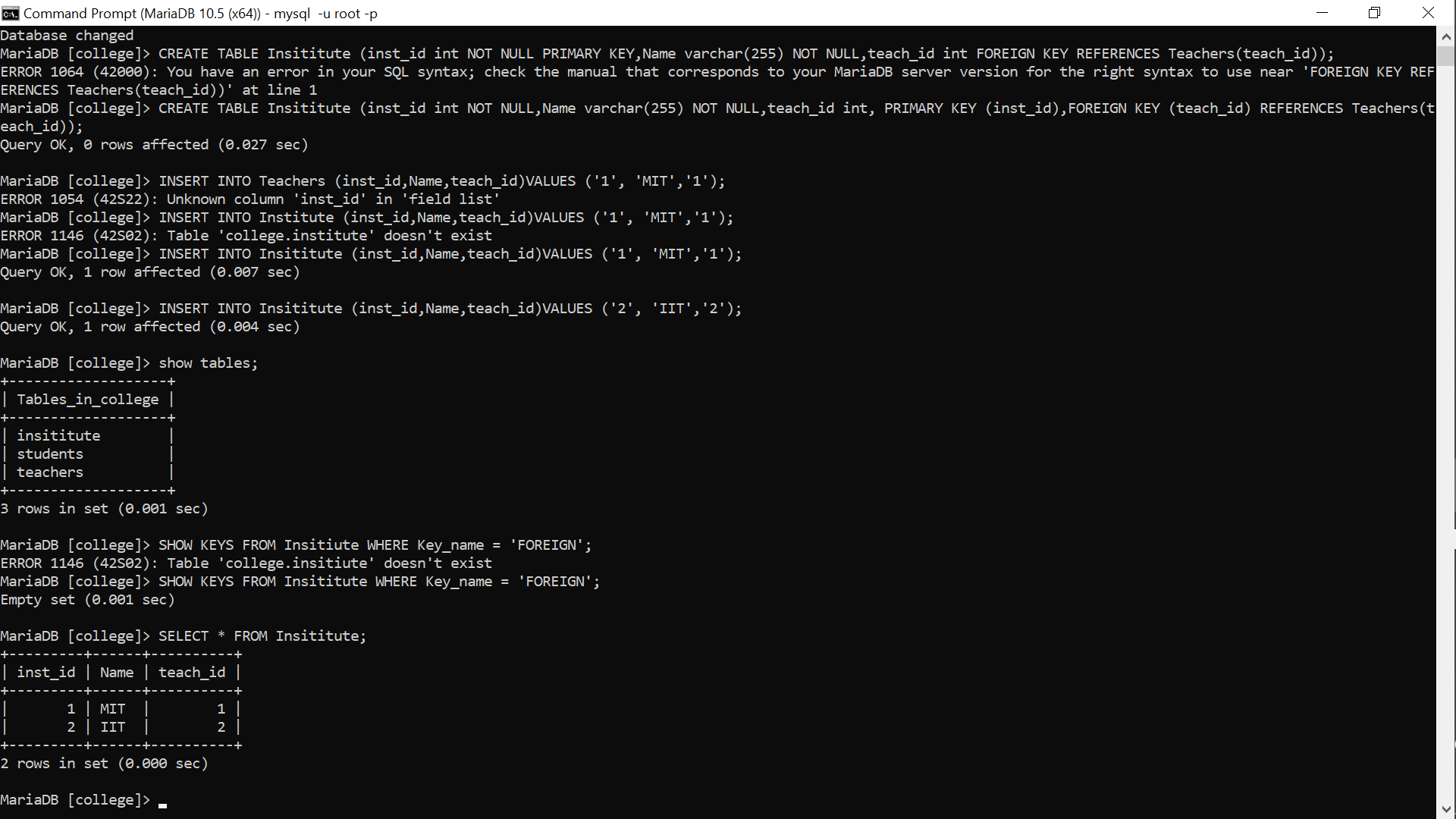
For FOREIGN KEY

8)cmd: CREATE TABLE Insititute (inst\_id int NOT NULL,Name varchar(255) NOT NULL,teach\_id int, PRIMARY KEY (inst\_id),FOREIGN KEY (teach\_id) REFERENCES Teachers(teach\_id));

9)cmd: INSERT INTO Teachers (inst\_id,Name,teach\_id)VALUES ('1', 'MIT','1');

10)cmd: INSERT INTO Insititute (inst\_id,Name,teach\_id)VALUES ('2', 'IIT','2');

11)cmd: SELECT \* FROM Insititute;



EXPERIMENT 3

**Aim:** SQL Statements for Implementing ALTER UPDATE and DELETE;

1)cmd: CREATE DATABASE Office;

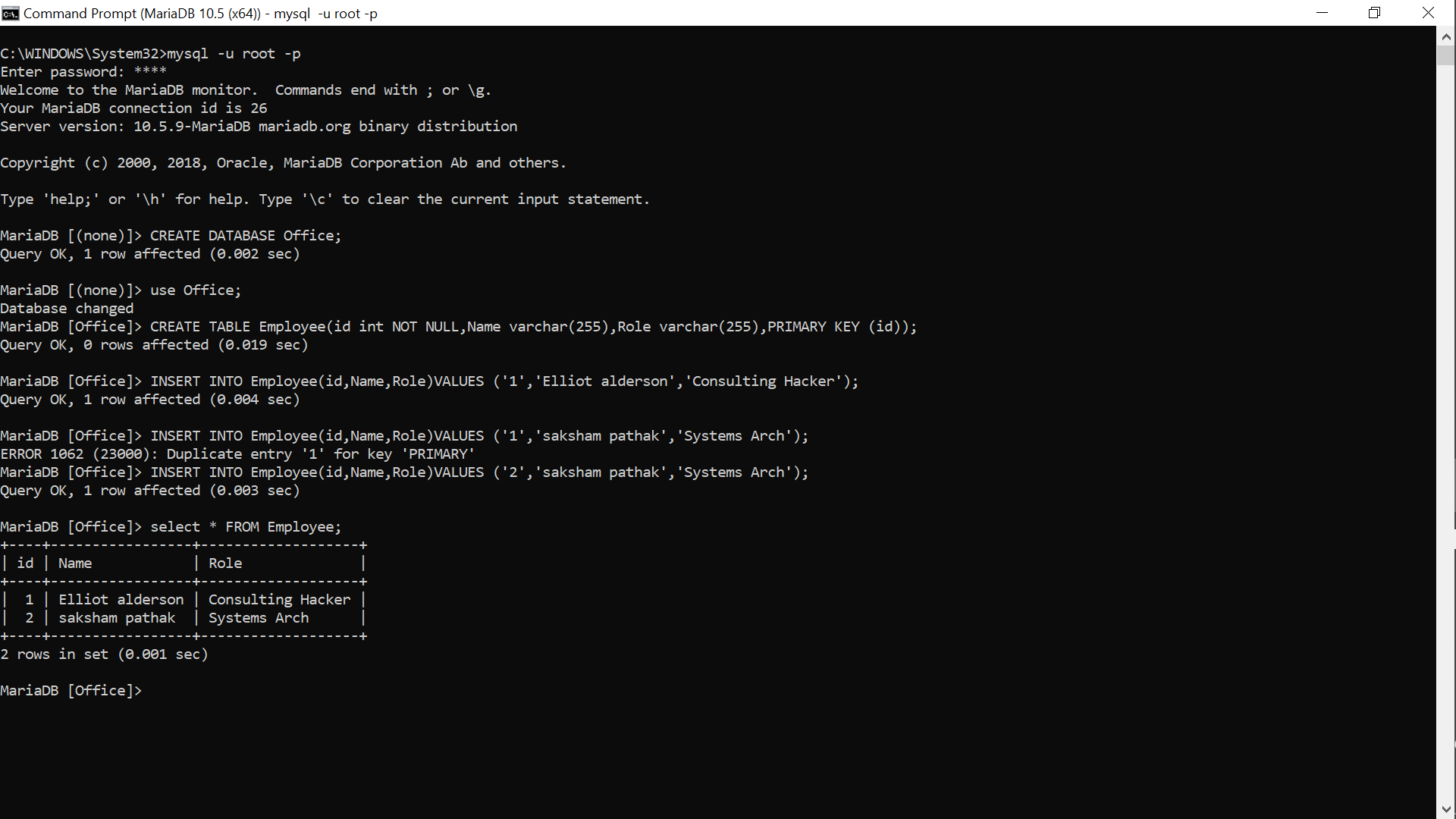
2)cmd: Use Office;

3)cmd: CREATE TABLE Employee (id int NOT NULL,Name varchar(255),Role varchar(255),PRIMARY KEY (id));

4)cmd: INSERT INTO Employee(id,Name,Role)VALUES ('1','Elliot alderson','Consulting Hacker');

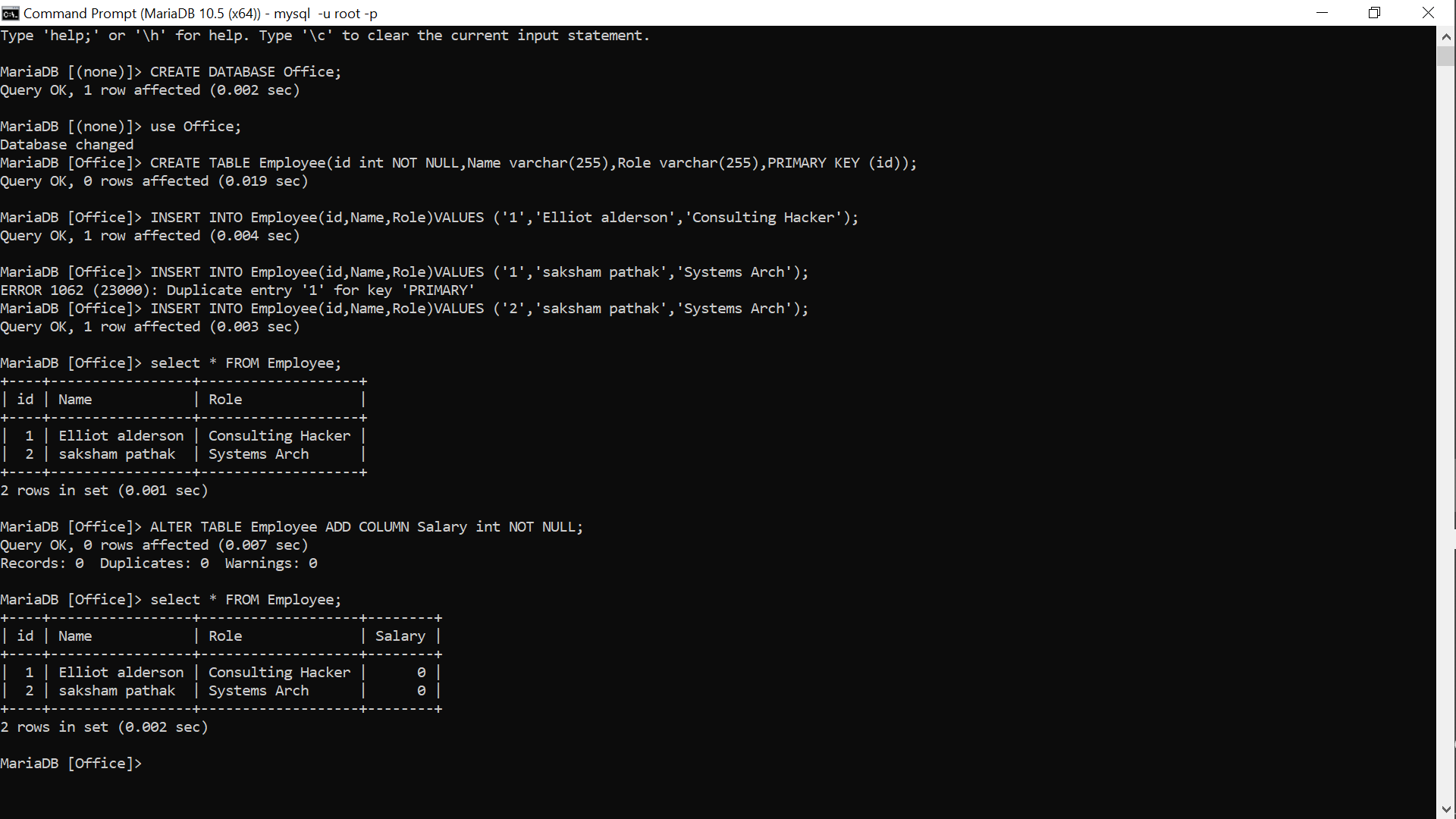
5)cmd: INSERT INTO Employee(id,Name,Role)VALUES ('2','saksham pathak','Systems Arch');

6)cmd: Select \* FROM Employee;



7)cmd: ALTER TABLE Employee ADD COLUMN Salary int NOT NULL;

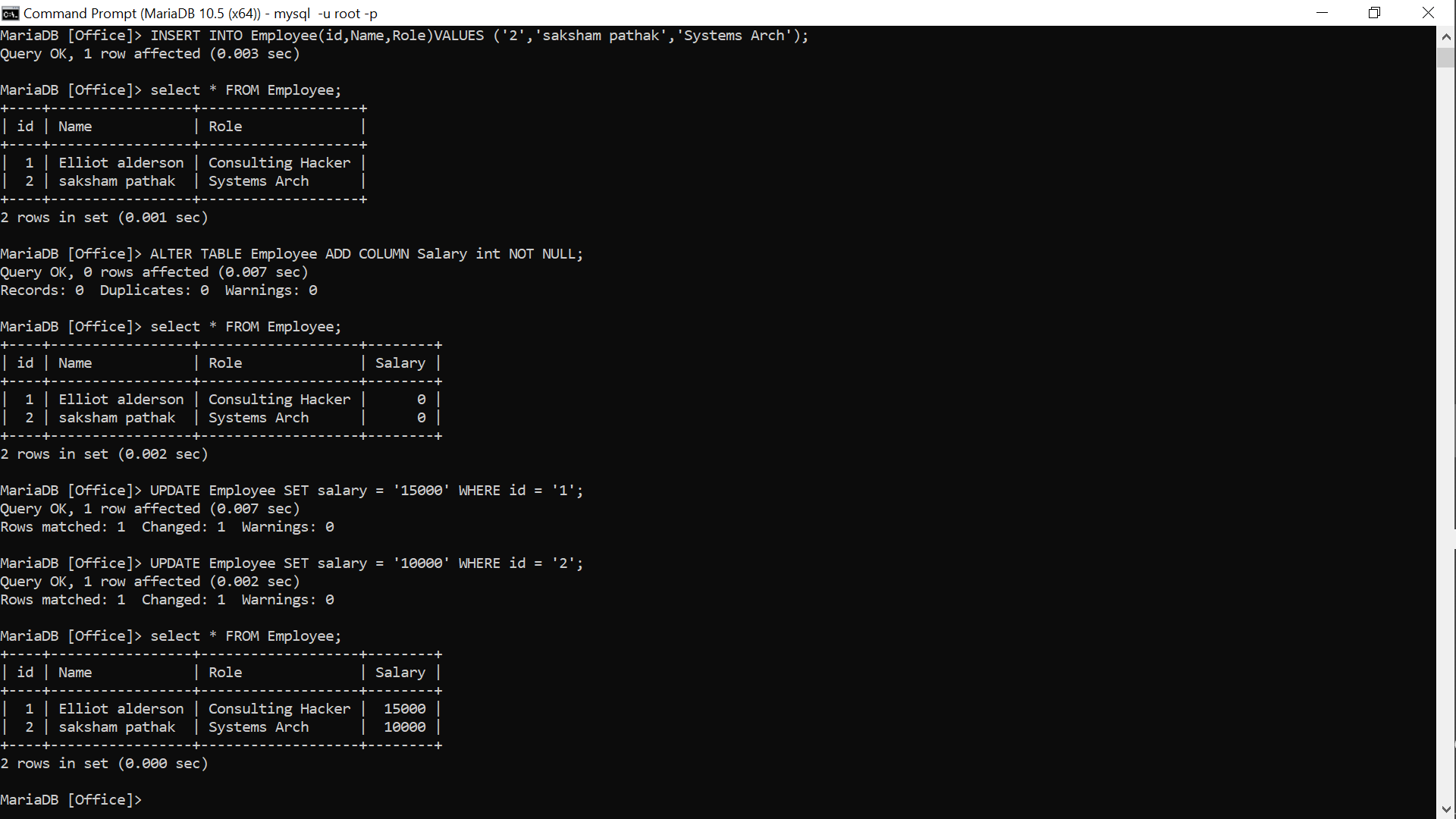
8)cmd: Select \* FROM Employee;



9)cmd: UPDATE Employee SET salary = '15000' WHERE id = '1';

10)cmd: UPDATE Employee SET salary = '10000' WHERE id = '2';

11)cmd: Select \* FROM Employee;

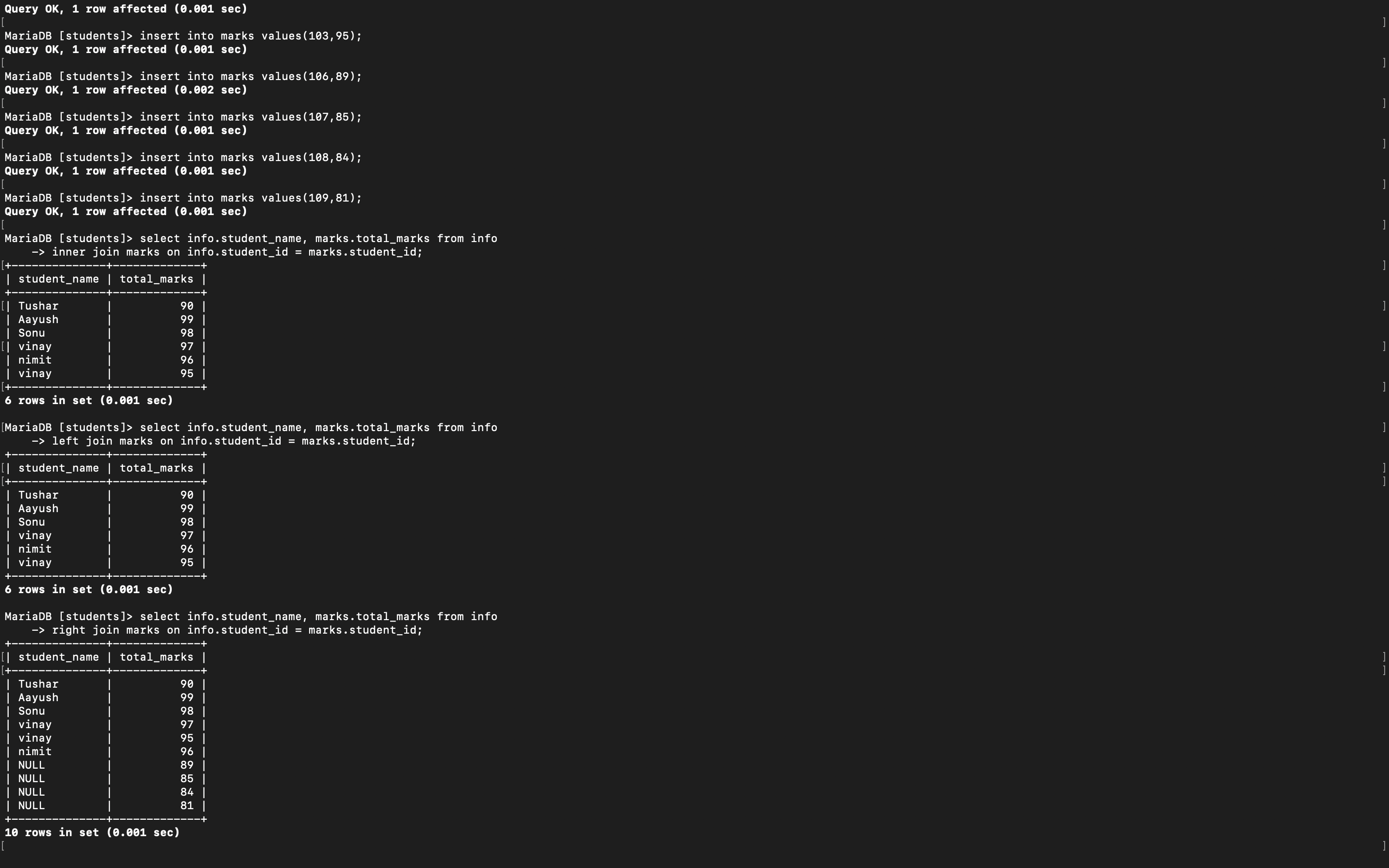
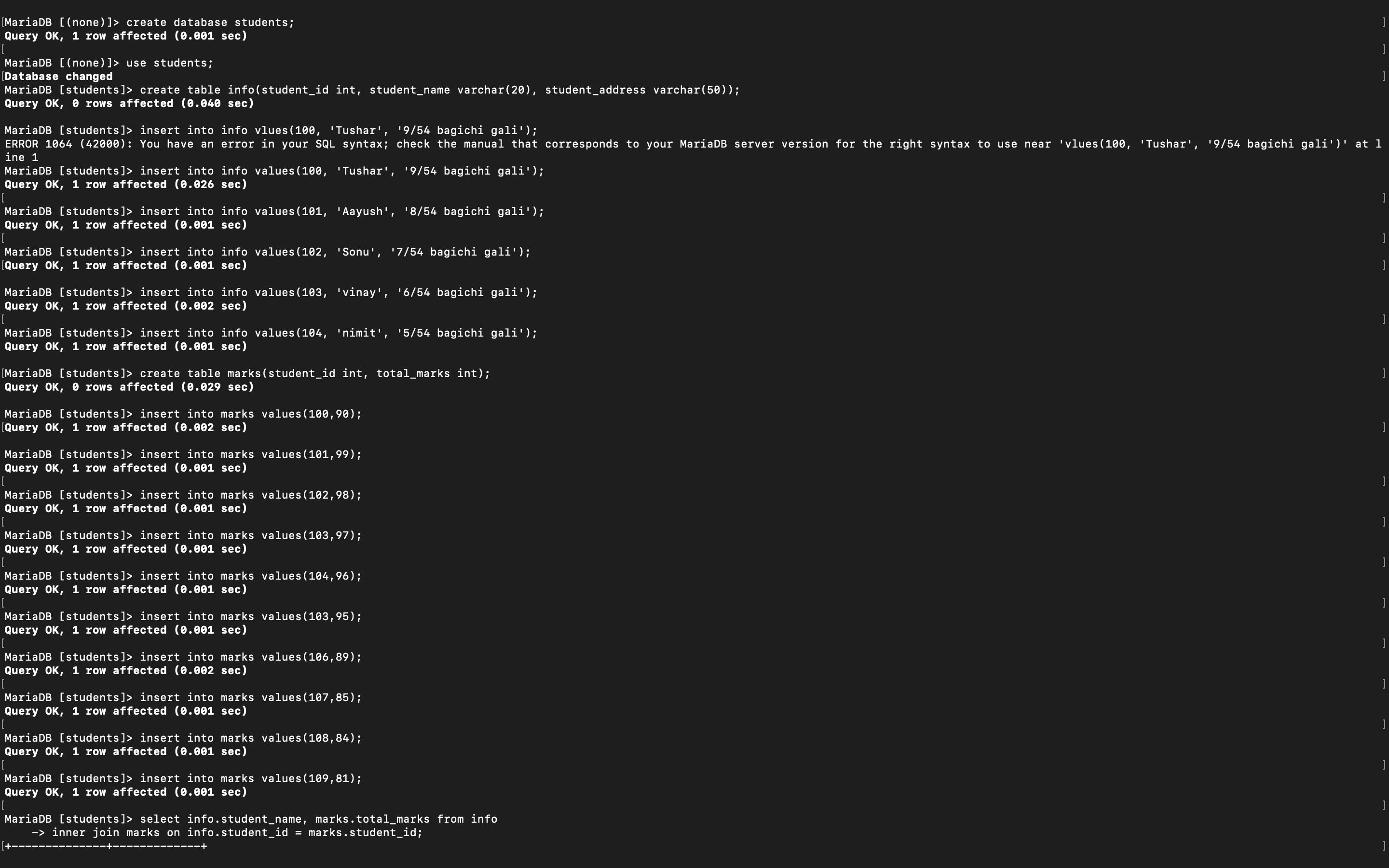


12)cmd: DELETE FROM Employee WHERE Name = 'saksham pathak';

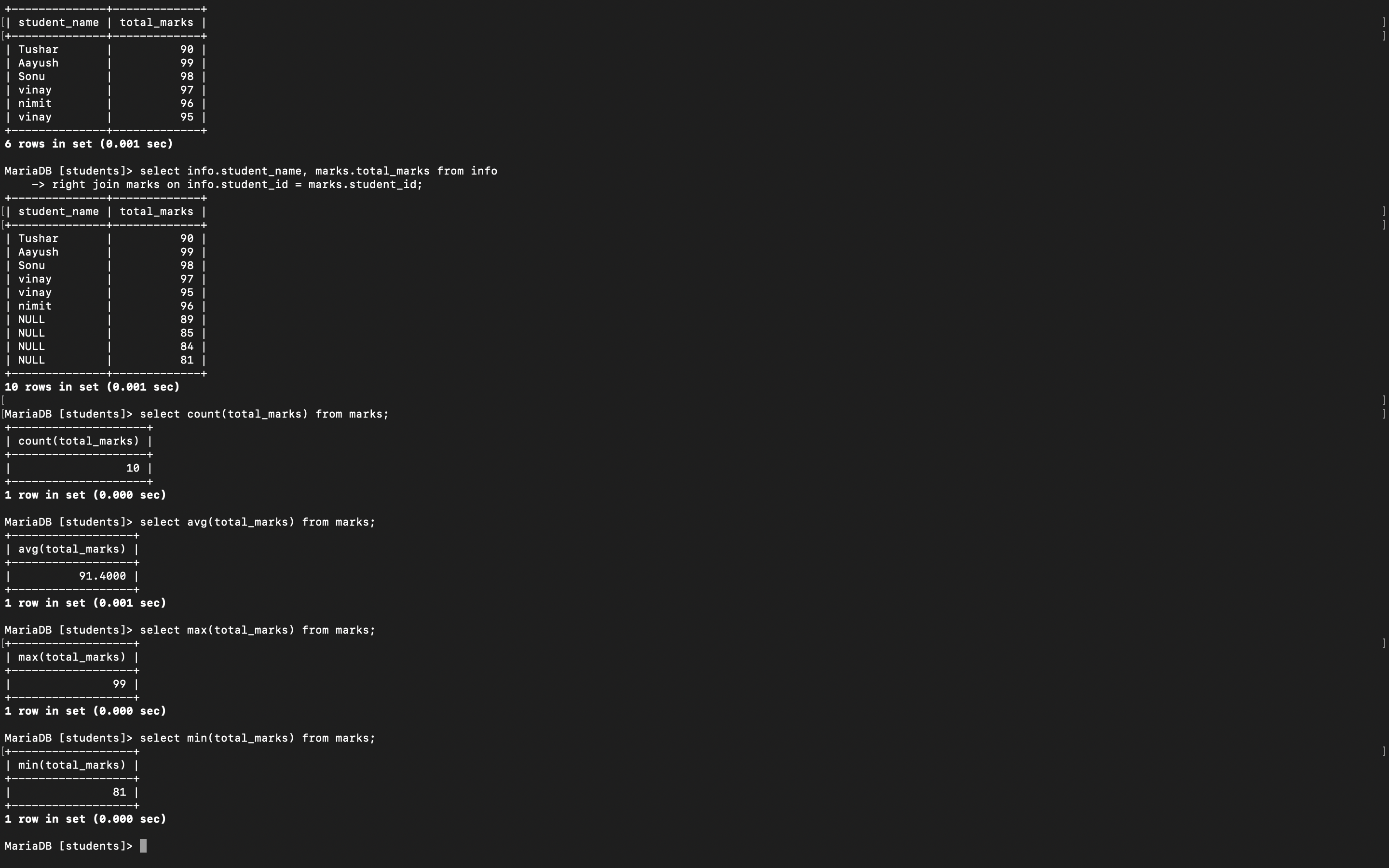
13)cmd: Select \* FROM Employee;



EXPERIMENT 4

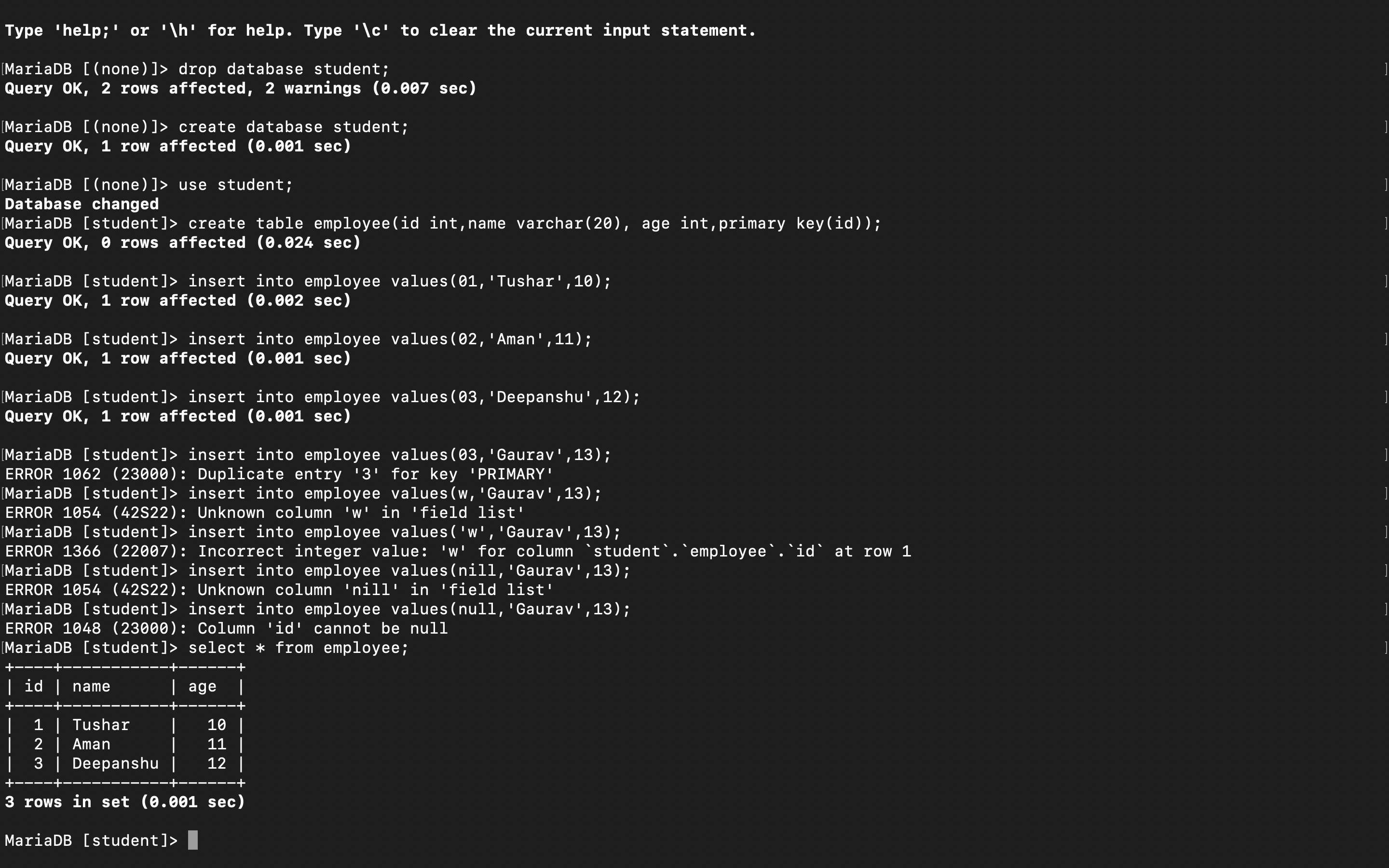
******Aim:** Program to write queries to implement joins.

EXPERIMENT 5

WRITE THE QUERIES FOR IMPLEMENTING THE FOLLOWING FUNCTIONS: MAX() , MIN() , AVG(), COUNT().

Experiment 6

**Aim:** Write the Queries to implement the concept of integrity constraints



Experiment 7

**Aim:** Write the queries to create the views

