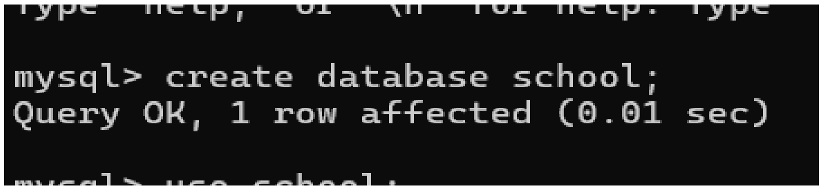
Name: Sakshi

Assignment (20-08-25)

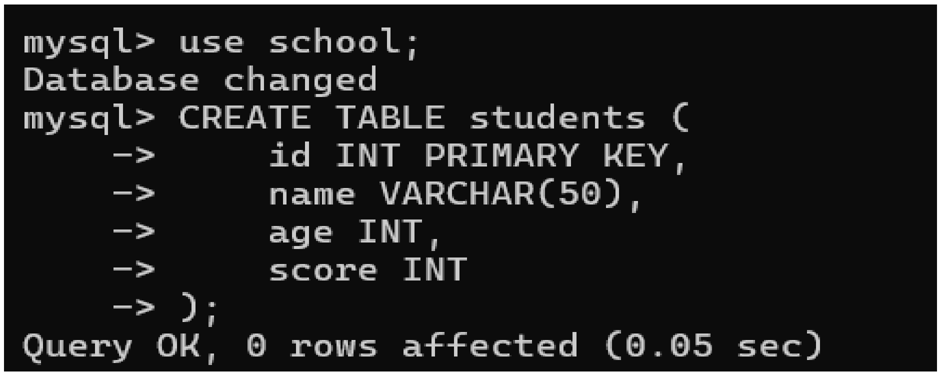
- Create a database called school.



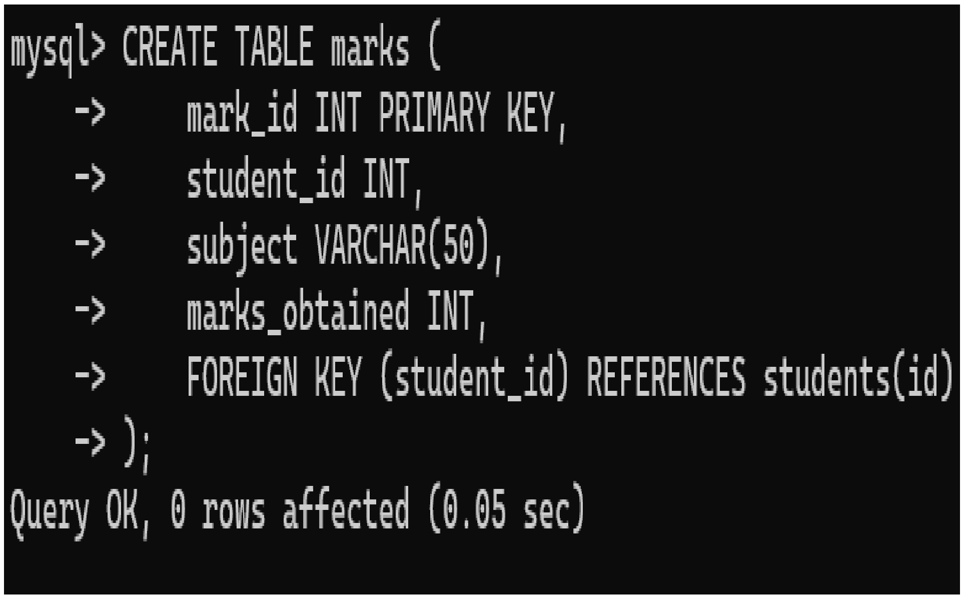
```
mysql> create database school;
Query OK, 1 row affected (0.01 sec)
```

- Use the database school for all exercises:

- Now create 2 tables students, marks with the same structure



```
mysql> use school;
Database changed
mysql> CREATE TABLE students (
    ->     id INT PRIMARY KEY,
    ->     name VARCHAR(50),
    ->     age INT,
    ->     score INT
    -> );
Query OK, 0 rows affected (0.05 sec)
```



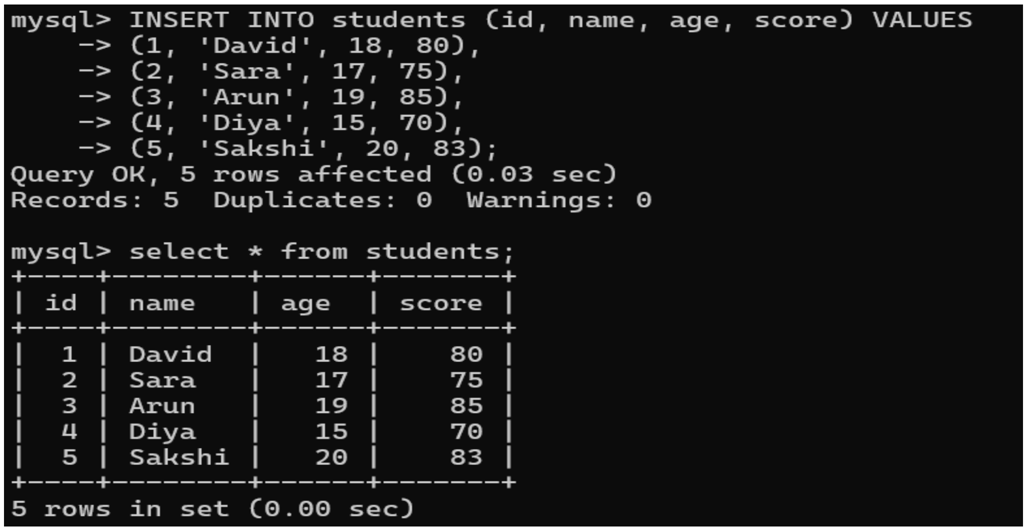
```
mysql> CREATE TABLE marks (
    ->     mark_id INT PRIMARY KEY,
    ->     student_id INT,
    ->     subject VARCHAR(50),
    ->     marks_obtained INT,
    ->     FOREIGN KEY (student_id) REFERENCES students(id)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

Name: Sakshi

Exercises

Part A: Adding Data

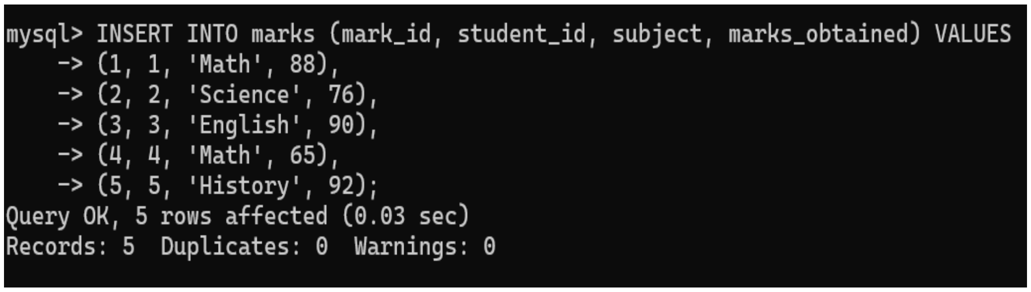
1.Insert at least 5 students into the students table.



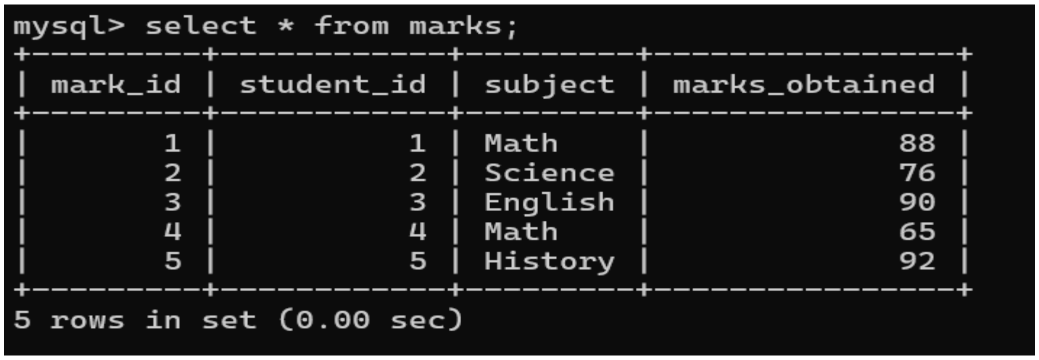
```
mysql> INSERT INTO students (id, name, age, score) VALUES
    -> (1, 'David', 18, 80),
    -> (2, 'Sara', 17, 75),
    -> (3, 'Arun', 19, 85),
    -> (4, 'Diya', 15, 70),
    -> (5, 'Sakshi', 20, 83);
Query OK, 5 rows affected (0.03 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> select * from students;
+----+--------+------+-------+
| id | name   | age  | score |
+----+--------+------+-------+
|  1 | David  |   18 |    80 |
|  2 | Sara   |   17 |    75 |
|  3 | Arun   |   19 |    85 |
|  4 | Diya   |   15 |    70 |
|  5 | Sakshi |   20 |    83 |
+----+--------+------+-------+
5 rows in set (0.00 sec)
```

2.Insert at least 5 marks for different students into the marks table.



```
mysql> INSERT INTO marks (mark_id, student_id, subject, marks_obtained) VALUES
    -> (1, 1, 'Math', 88),
    -> (2, 2, 'Science', 76),
    -> (3, 3, 'English', 90),
    -> (4, 4, 'Math', 65),
    -> (5, 5, 'History', 92);
Query OK, 5 rows affected (0.03 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

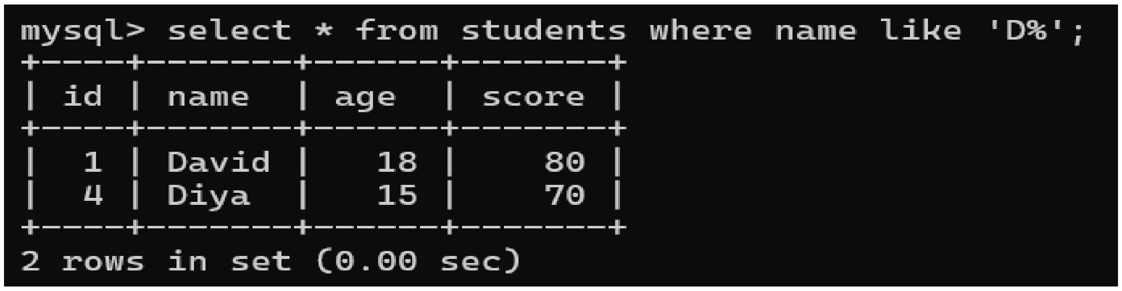


```
mysql> select * from marks;
+---------+------------+---------+----------------+
| mark_id | student_id | subject | marks_obtained |
+---------+------------+---------+----------------+
|       1 |          1 | Math    |             88 |
|       2 |          2 | Science |             76 |
|       3 |          3 | English |             90 |
|       4 |          4 | Math    |             65 |
|       5 |          5 | History |             92 |
+---------+------------+---------+----------------+
5 rows in set (0.00 sec)
```

Name: Sakshi

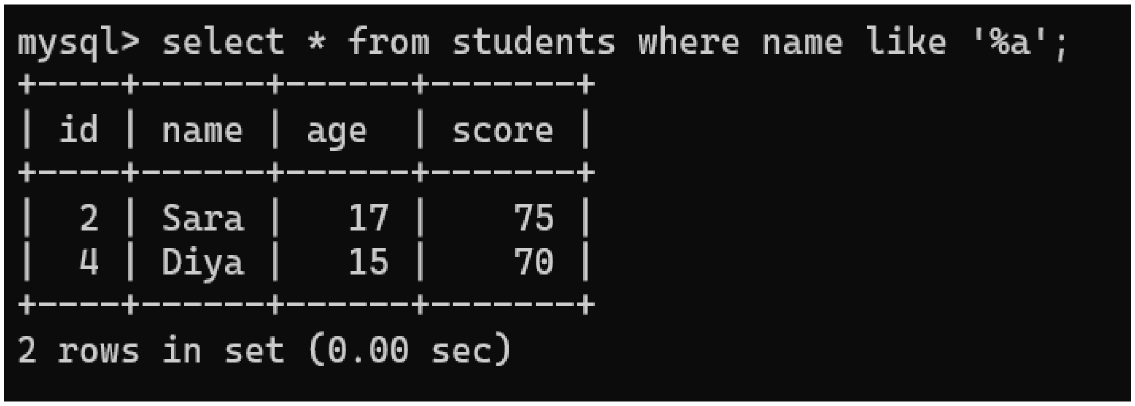
Part B: SELECT & LIKE Queries

3.Retrieve all students whose names start with “D”.



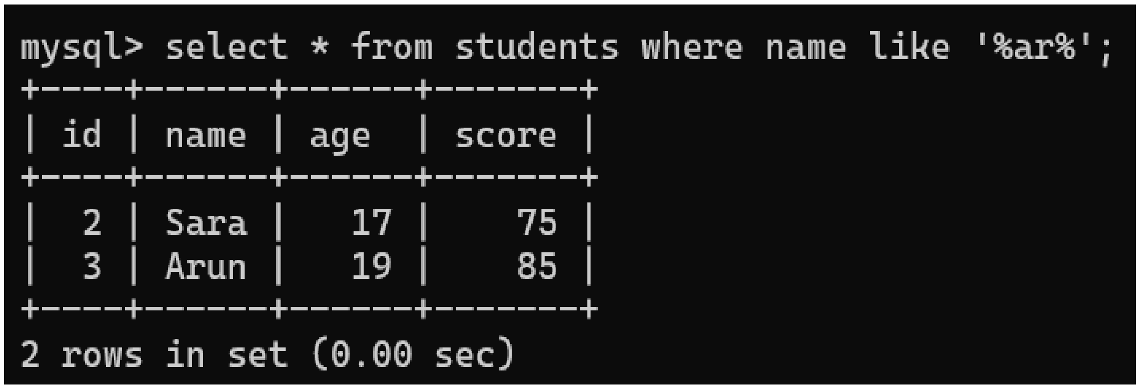
```
mysql> select * from students where name like 'D%';
+----+-------+------+-------+
| id | name  | age  | score |
+----+-------+------+-------+
|  1 | David |   18 |    80 |
|  4 | Diya  |   15 |    70 |
+----+-------+------+-------+
2 rows in set (0.00 sec)
```

4. Retrieve all students whose names end with “a”.



```
mysql> select * from students where name like '%a';
+----+------+------+-------+
| id | name | age  | score |
+----+------+------+-------+
|  2 | Sara |   17 |    75 |
|  4 | Diya |   15 |    70 |
+----+------+------+-------+
2 rows in set (0.00 sec)
```

5. Retrieve all students whose names contain “ar”.

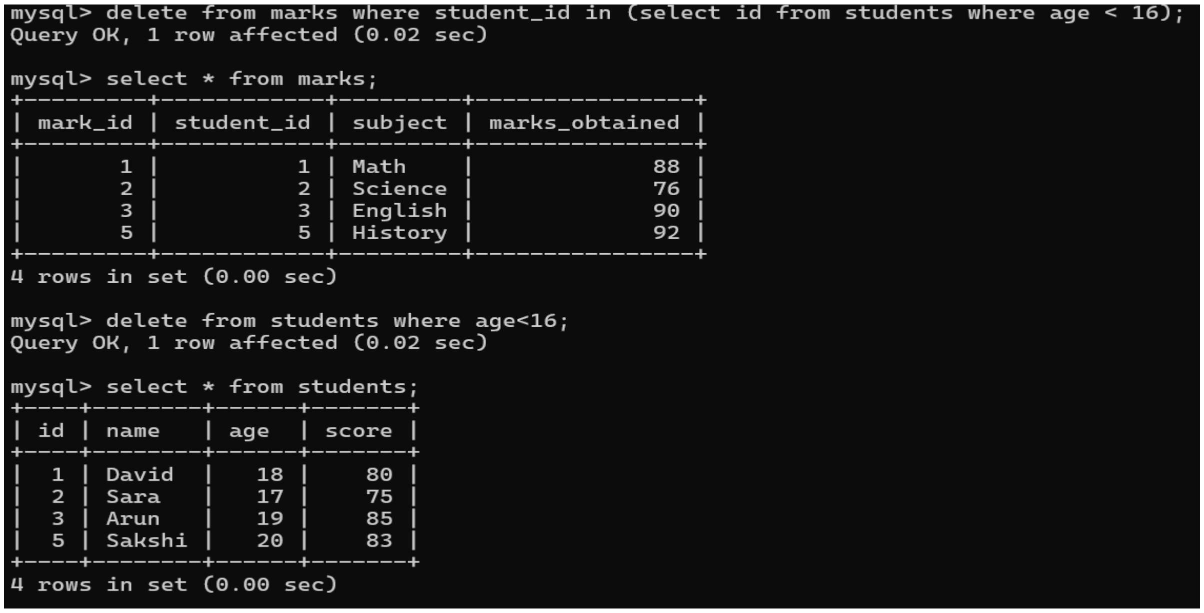


```
mysql> select * from students where name like '%ar%';
+----+------+------+-------+
| id | name | age  | score |
+----+------+------+-------+
|  2 | Sara |   17 |    75 |
|  3 | Arun |   19 |    85 |
+----+------+------+-------+
2 rows in set (0.00 sec)
```

Name: Sakshi

Part C: DELETE & DROP

6. Delete students whose age is less than 16.



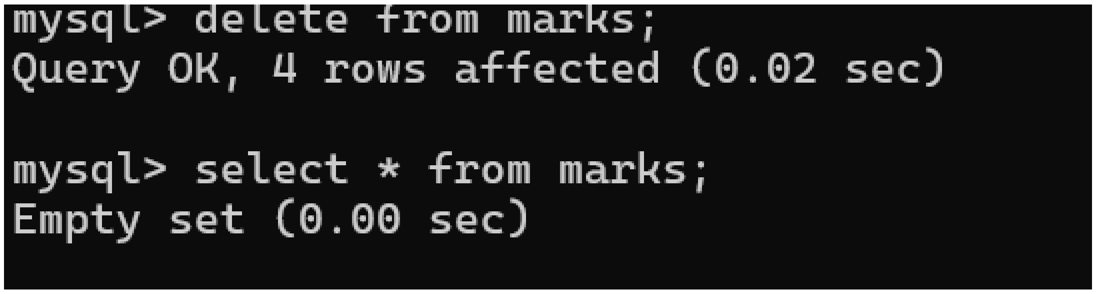
```
mysql> delete from marks where student_id in (select id from students where age < 16);
Query OK, 1 row affected (0.02 sec)

mysql> select * from marks;
+---------+------------+---------+----------------+
| mark_id | student_id | subject | marks_obtained |
+---------+------------+---------+----------------+
|       1 |          1 | Math    |             88 |
|       2 |          2 | Science |             76 |
|       3 |          3 | English |             90 |
|       5 |          5 | History |             92 |
+---------+------------+---------+----------------+
4 rows in set (0.00 sec)

mysql> delete from students where age<16;
Query OK, 1 row affected (0.02 sec)

mysql> select * from students;
+----+--------+------+-------+
| id | name   | age  | score |
+----+--------+------+-------+
|  1 | David  |   18 |    80 |
|  2 | Sara   |   17 |    75 |
|  3 | Arun   |   19 |    85 |
|  5 | Sakshi |   20 |    83 |
+----+--------+------+-------+
4 rows in set (0.00 sec)
```

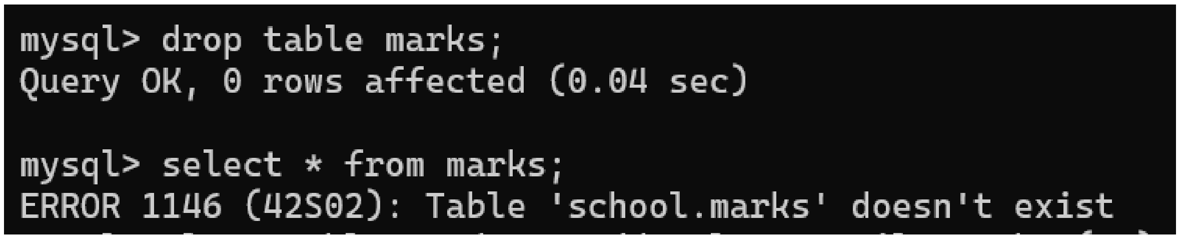
7. Delete all data from the marks table.



```
mysql> delete from marks;
Query OK, 4 rows affected (0.02 sec)

mysql> select * from marks;
Empty set (0.00 sec)
```

8. Drop the marks table completely.



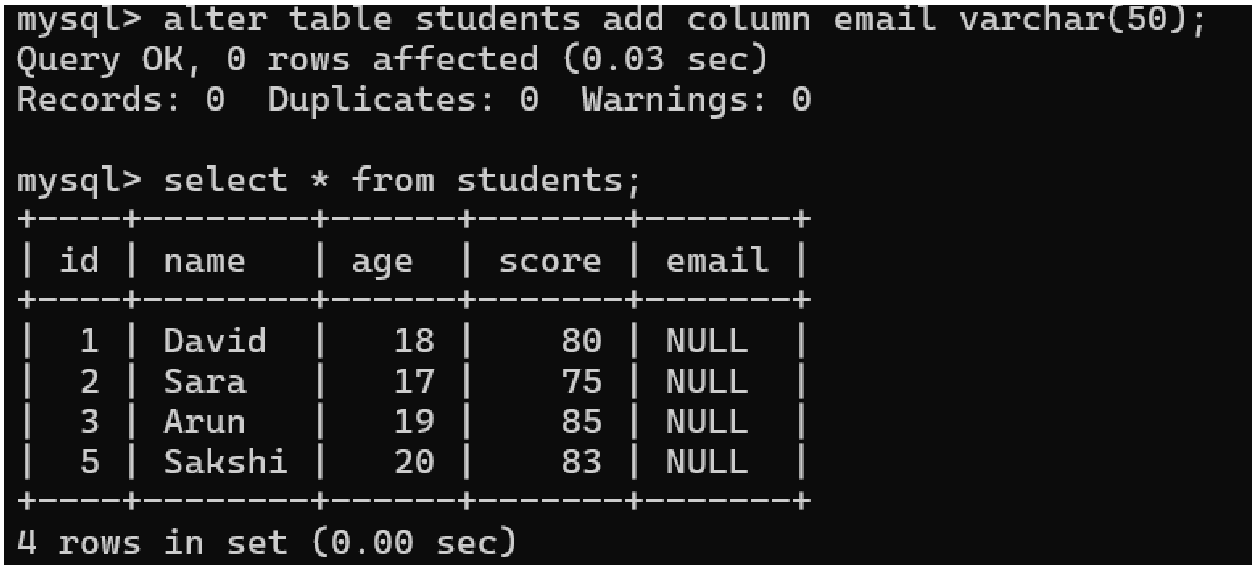
```
mysql> drop table marks;
Query OK, 0 rows affected (0.04 sec)

mysql> select * from marks;
ERROR 1146 (42S02): Table 'school.marks' doesn't exist
```

Name: Sakshi

Part D: ALTER TABLE

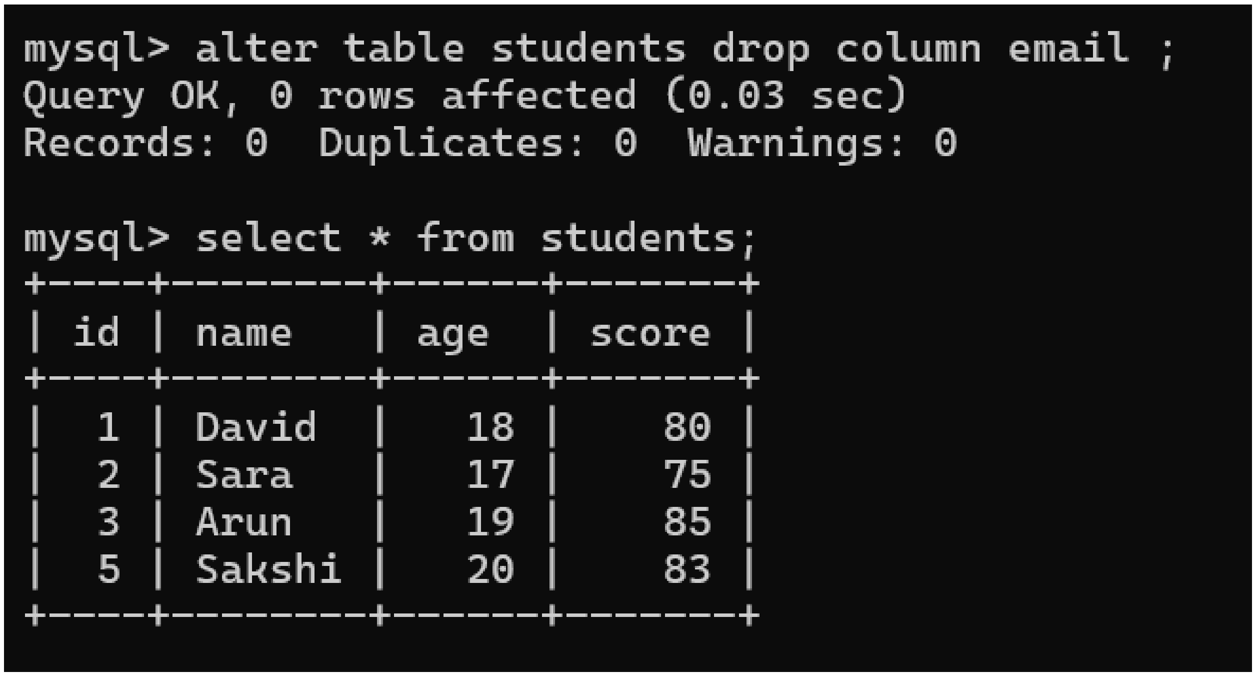
9. Add a new column email to the students table.



```
mysql> alter table students add column email varchar(50);
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from students;
+----+--------+------+-------+-------+
| id | name   | age  | score | email |
+----+--------+------+-------+-------+
|  1 | David  |   18 |    80 | NULL  |
|  2 | Sara   |   17 |    75 | NULL  |
|  3 | Arun   |   19 |    85 | NULL  |
|  5 | Sakshi |   20 |    83 | NULL  |
+----+--------+------+-------+-------+
4 rows in set (0.00 sec)
```

10. Remove the email column from the students table.



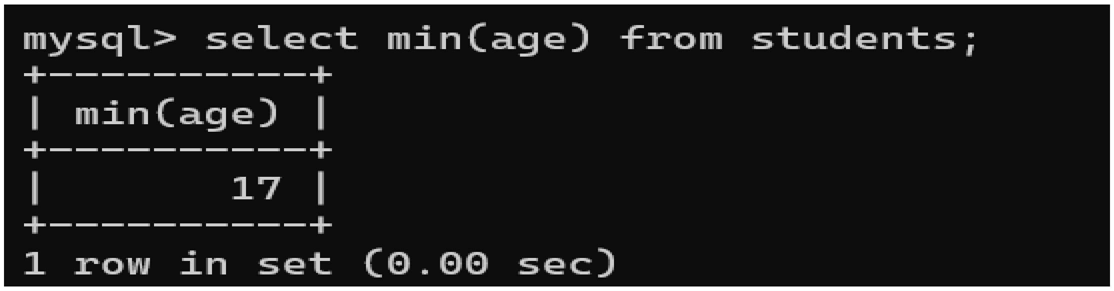
```
mysql> alter table students drop column email ;
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from students;
+----+--------+------+-------+
| id | name   | age  | score |
+----+--------+------+-------+
|  1 | David  |   18 |    80 |
|  2 | Sara   |   17 |    75 |
|  3 | Arun   |   19 |    85 |
|  5 | Sakshi |   20 |    83 |
+----+--------+------+-------+
```

Name: Sakshi

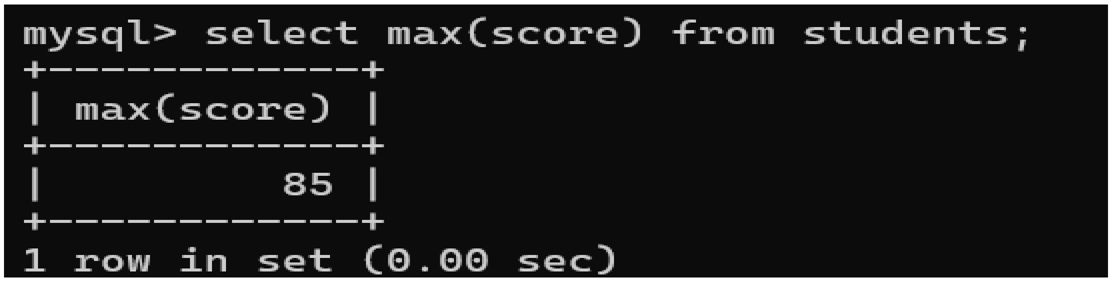
Part E: Aggregates

11. Find the youngest student’s age using MIN().



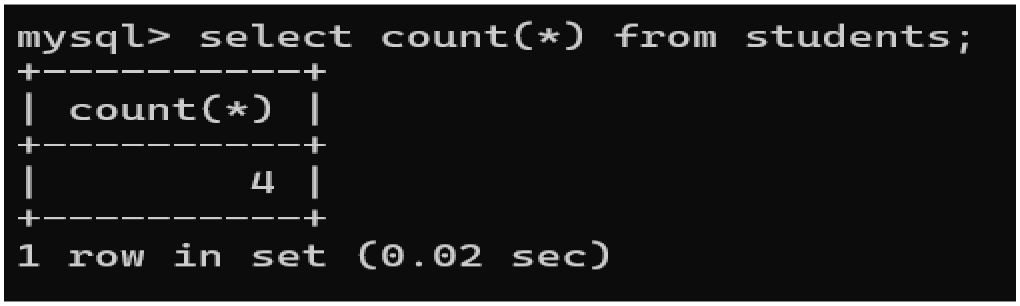
```
mysql> select min(age) from students;
+----------+
| min(age) |
+----------+
|       17 |
+----------+
1 row in set (0.00 sec)
```

12. Find the highest score using MAX().



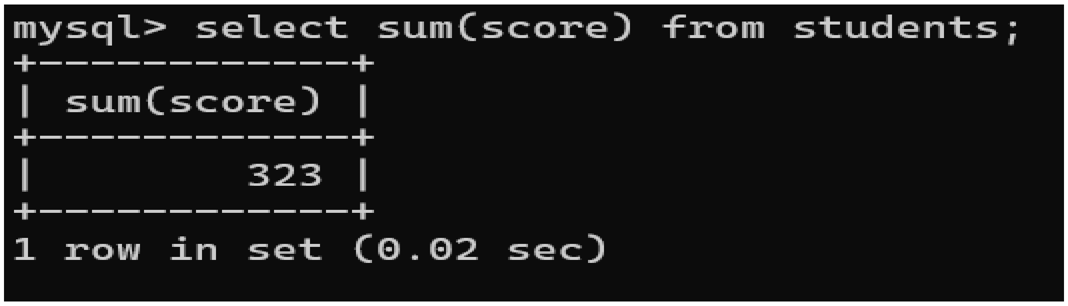
```
mysql> select max(score) from students;
+------------+
| max(score) |
+------------+
|         85 |
+------------+
1 row in set (0.00 sec)
```

13. Count the total number of students using COUNT().



```
mysql> select count(*) from students;
+----------+
| count(*) |
+----------+
|        4 |
+----------+
1 row in set (0.02 sec)
```

14. Find the total score of all students using SUM().

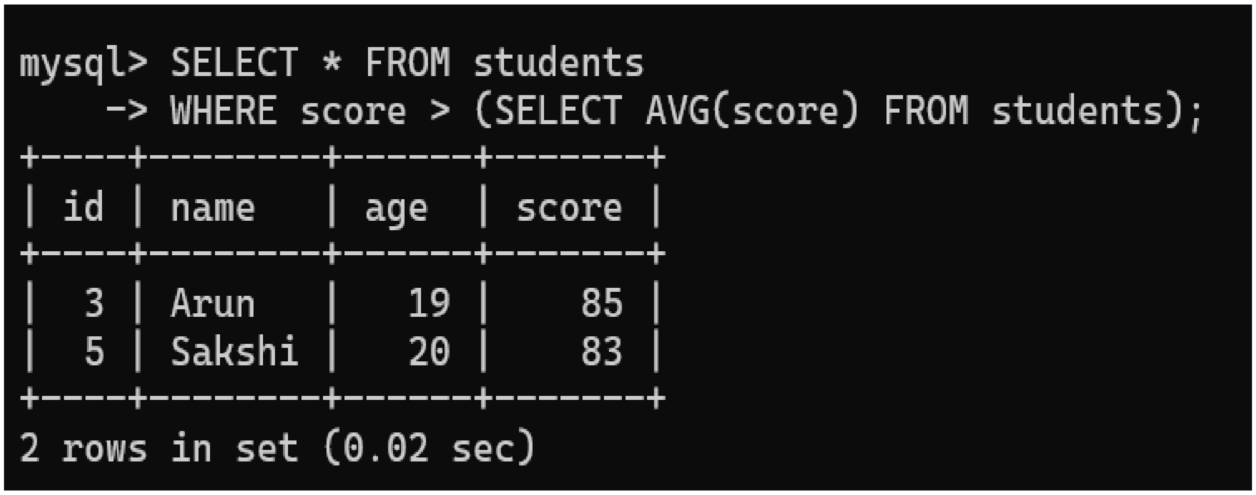


```
mysql> select sum(score) from students;
+------------+
| sum(score) |
+------------+
|        323 |
+------------+
1 row in set (0.02 sec)
```

Name: Sakshi

Part F: Nested Queries

15. Retrieve all students whose score is greater than the average score.



```
mysql> SELECT * FROM students
    -> WHERE score > (SELECT AVG(score) FROM students);
+----+--------+------+-------+
| id | name   | age  | score |
+----+--------+------+-------+
|  3 | Arun   |   19 |    85 |
|  5 | Sakshi |   20 |    83 |
+----+--------+------+-------+
2 rows in set (0.02 sec)
```