/*-- Project 1: Creating Databases and Operations

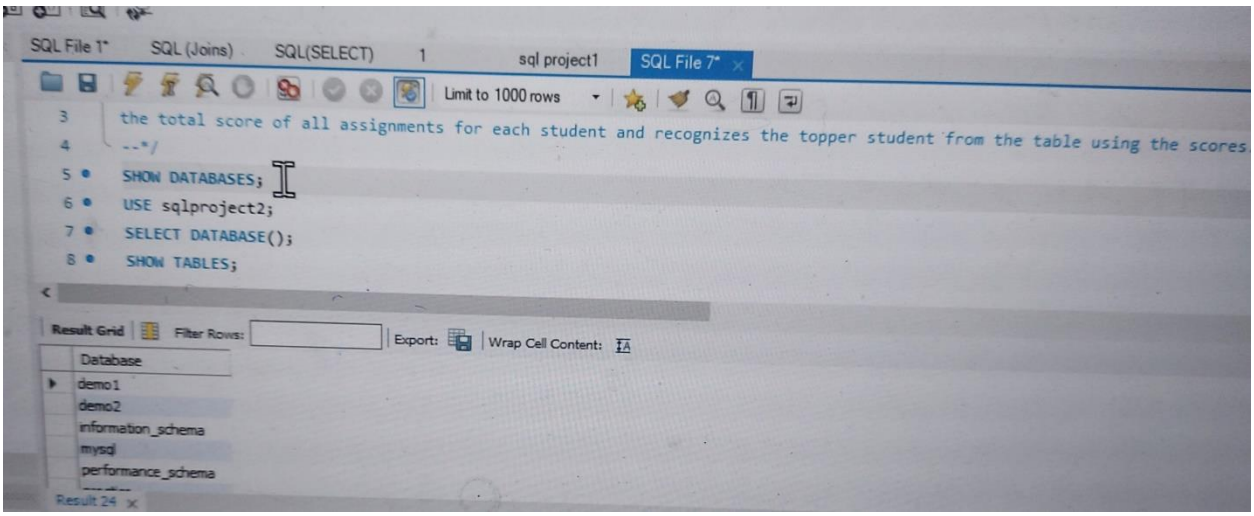
Create a table containing assignment scores of 6 students in class for Python and write a query that calculates

the total score of all assignments for each student and recognizes the topper student from the table using the scores.

--*/

-- To know the available database

SHOW DATABASES;

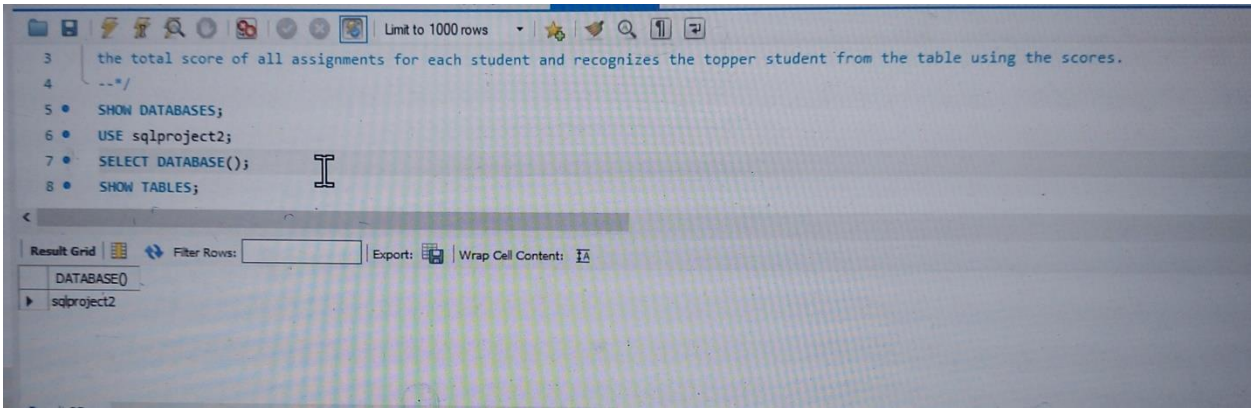


-- to use sqlproject db

USE sqlproject2;

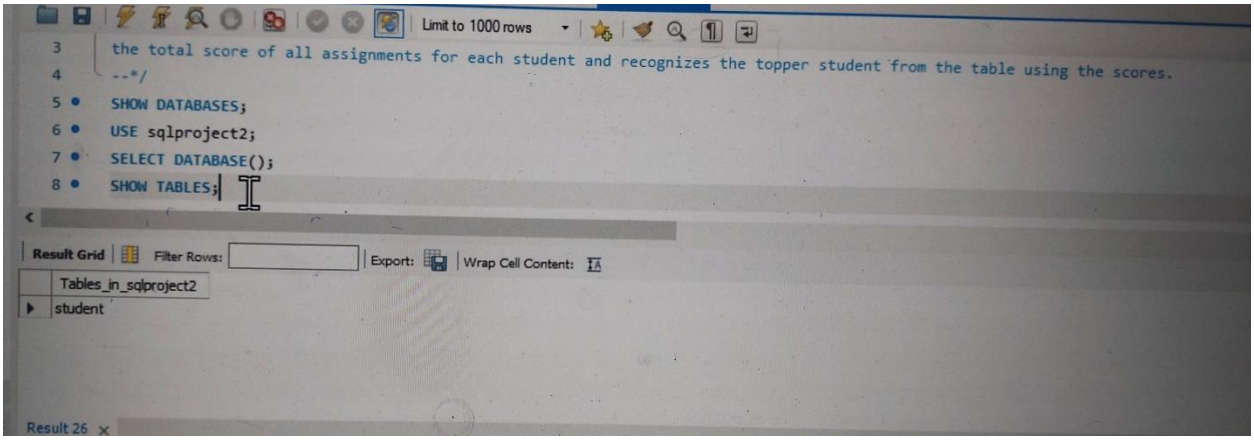
-- To know the current db

SELECT DATABASE();



-- To know the tables

SHOW TABLES;



-- delete if that table exists previously

DROP TABLE IF EXISTS student;

-- Create table student with valid constraint

CREATE TABLE student (

   student_id INT PRIMARY KEY,

   first_name VARCHAR(25) NOT NULL,

   last_name VARCHAR(20) NOT NULL,

   department VARCHAR(10) NOT NULL,

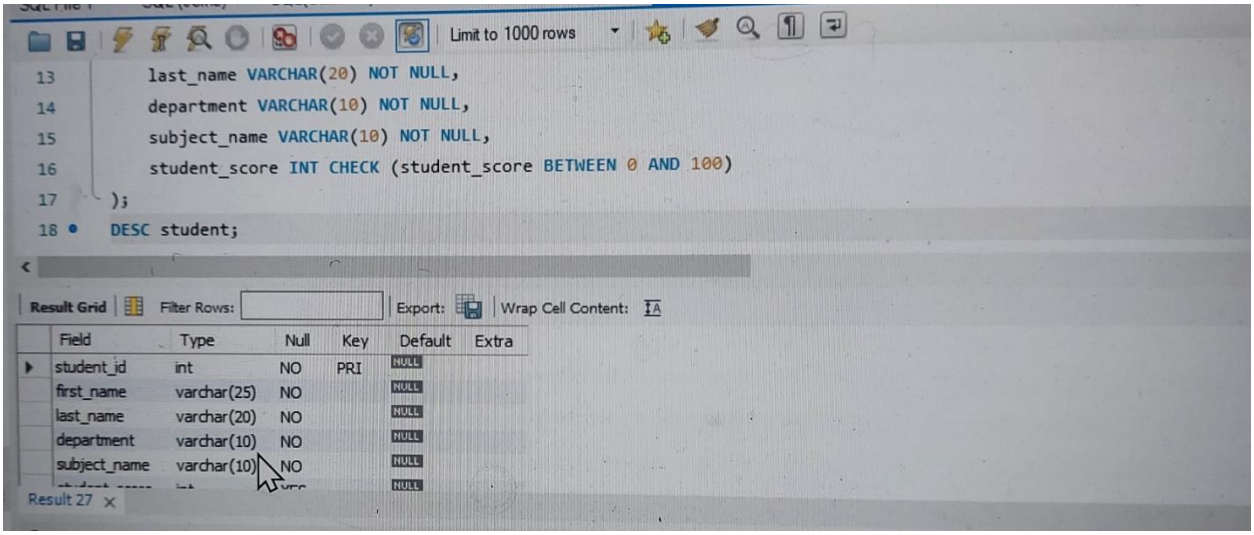
   subject_name VARCHAR(10) NOT NULL,

   student_score INT CHECK (student_score BETWEEN 0 AND 100)

);

-- To describe the table

DESC student;



-- To insert the values in to the table one by one

INSERT INTO student(student_id, first_name, last_name, department, subject_name, student_score)

VALUES (101, 'Kavitha', 'Kumar', 'CSE', 'Python', 98);

-- To insert more than one value

INSERT INTO student(student_id, first_name, last_name, department, subject_name, student_score)

VALUES (102, 'Priya', 'Dayalan', 'IT', 'Python', 100),

(103, 'Kanimozhi', 'Elumalai', 'ECE', 'Python', 91),

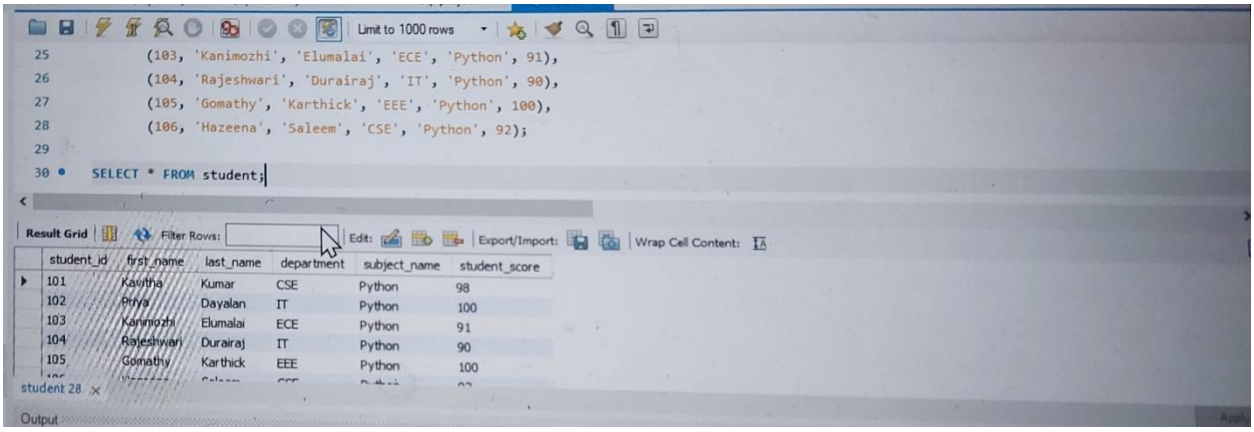
(104, 'Rajeshwari', 'Durairaj', 'IT', 'Python', 90),

(105, 'Gomathy', 'Karthick', 'EEE', 'Python', 100),

(106, 'Hazeena', 'Saleem', 'CSE', 'Python', 92);

-- To retrieve all data from the record

SELECT * FROM student;

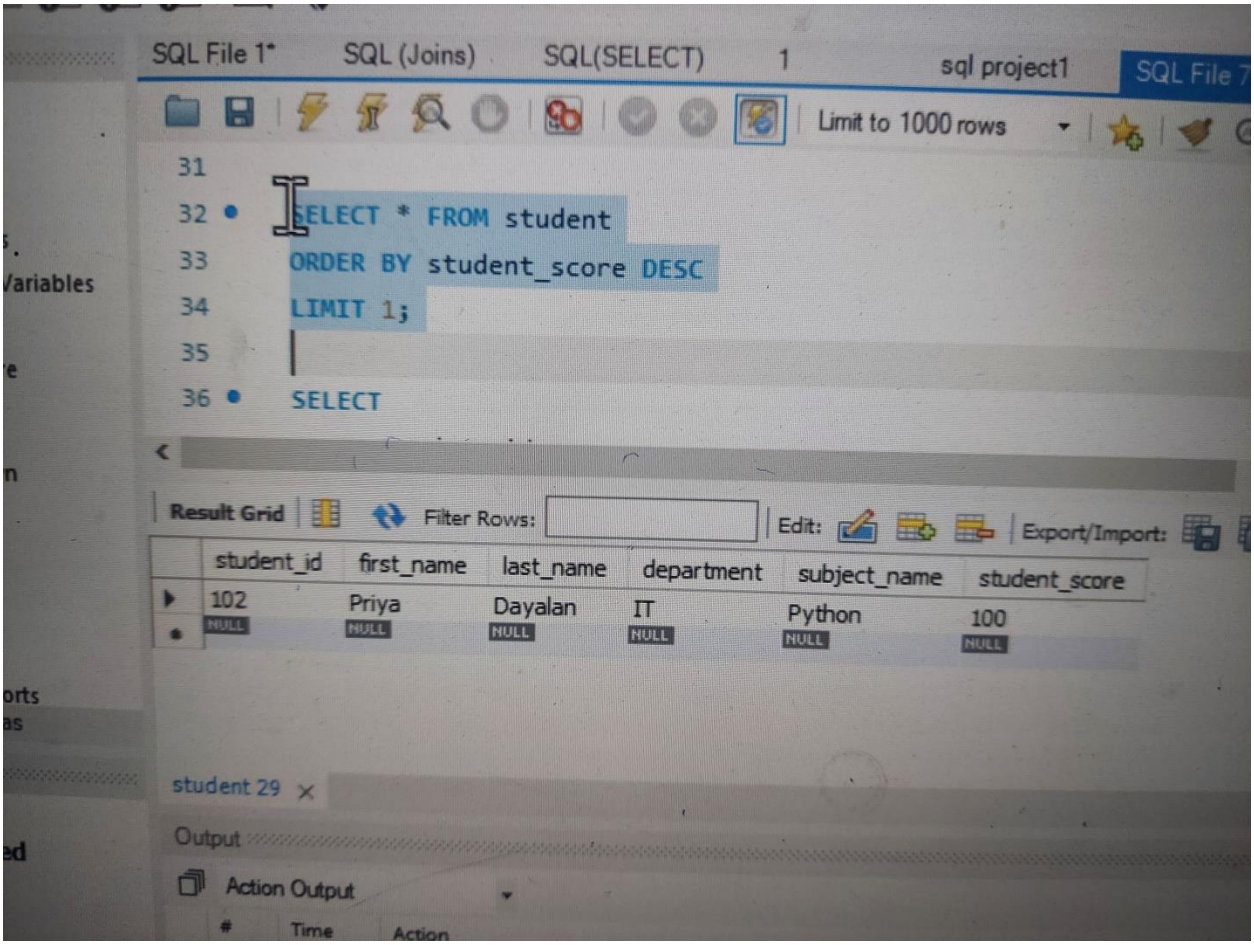


-- To get the topper score restriction it will show only one record

SELECT * FROM student

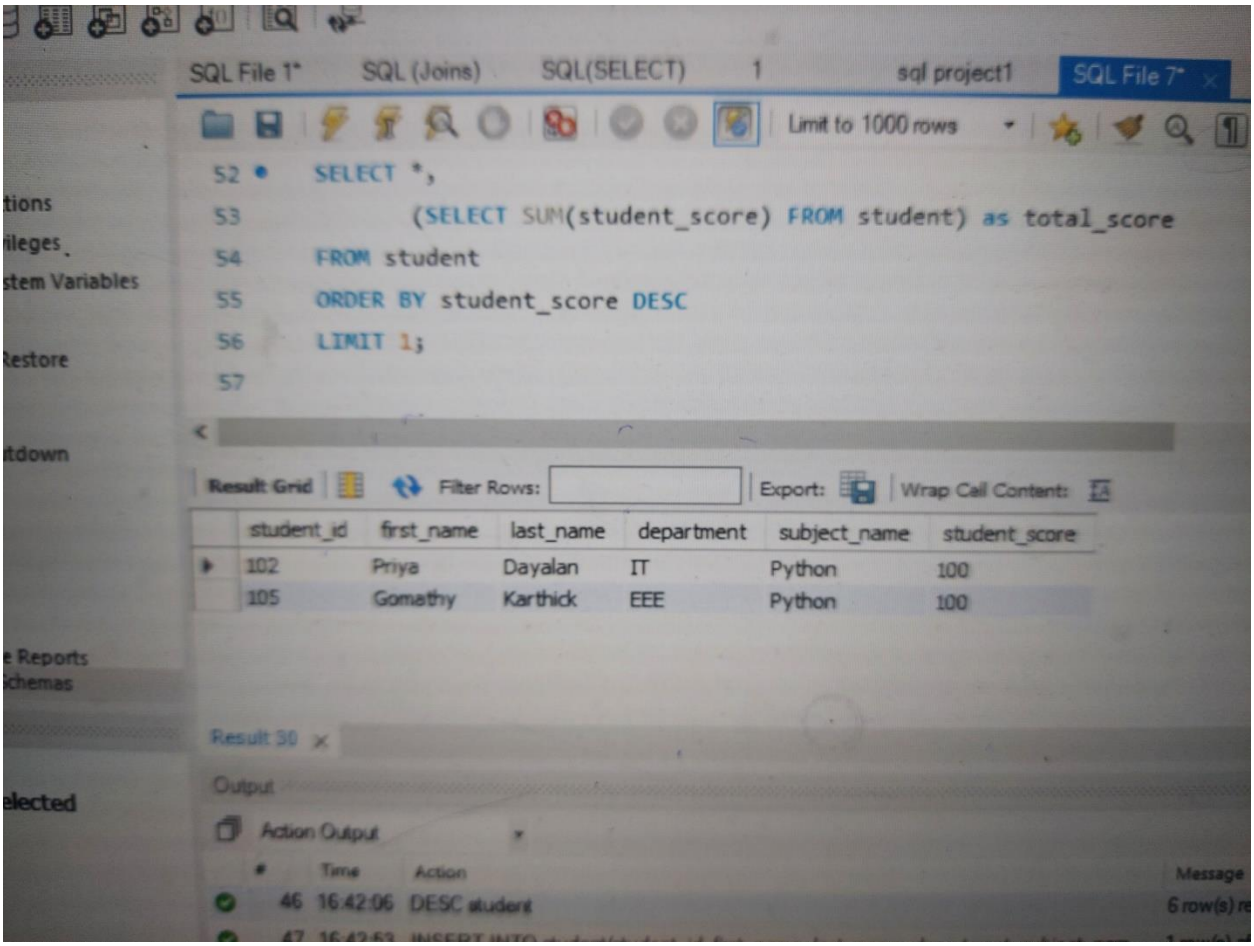
ORDER BY student_score DESC

LIMIT 1;

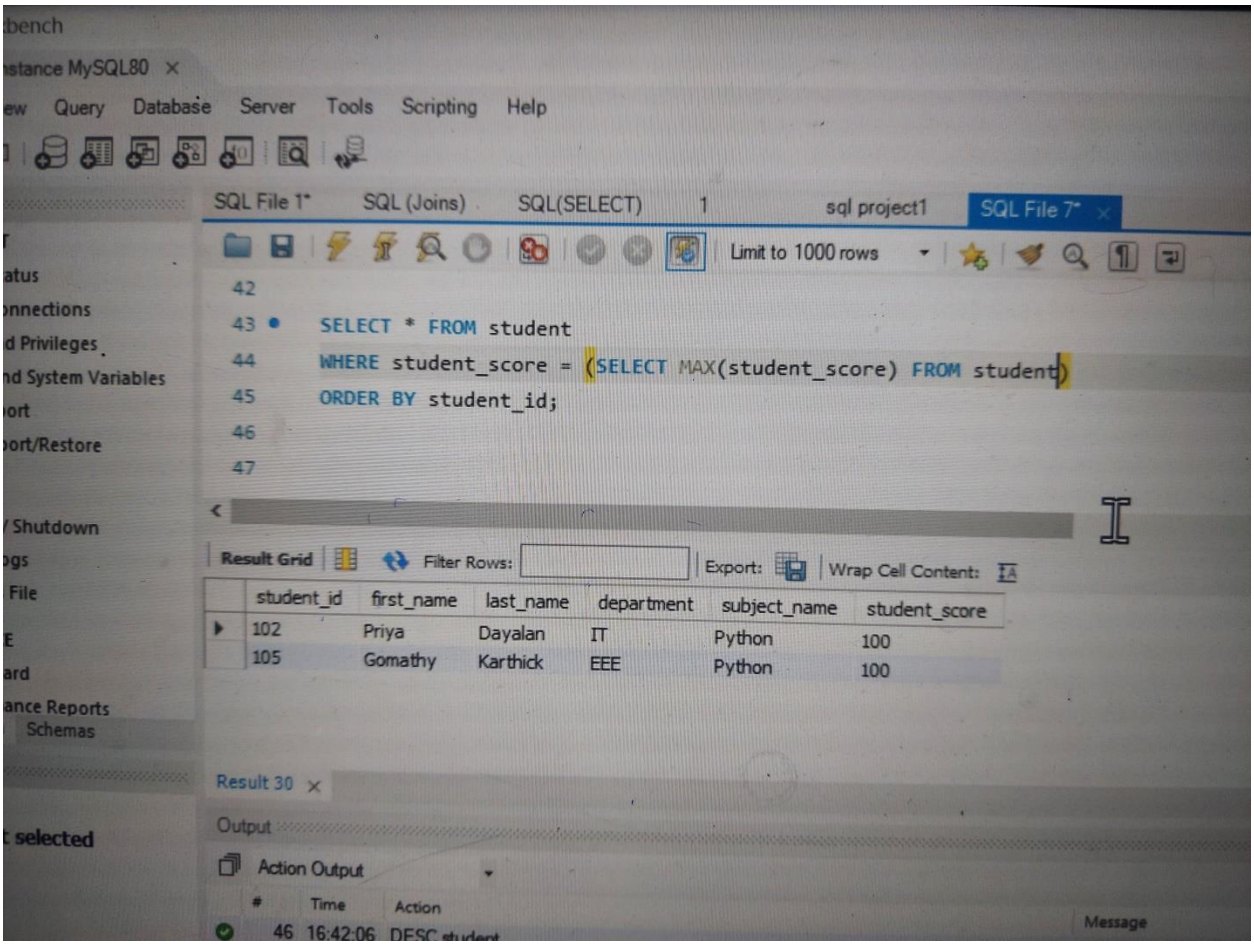


-- to select topper by using aggregate fn and subqueries which satisfies all condition

```
SELECT *
FROM student
WHERE student_score = (
   SELECT MAX(student_score) FROM student
)
ORDER BY student_id;
```



```
-- To get the topper score even more than one student satisfies the condition using self join
SELECT
   s.student_id,
   s.first_name,
   s.last_name,
   s.department,
   s.subject_name,
   s.student_score
FROM
   student s
JOIN (
   SELECT
      MAX(student_score) AS max_score
   FROM
      student
) t ON s.student_score = t.max_score;
```


bench
stance MySQL80
ew Query Database Server Tools Scripting Help
SQL File 1* SQL (Joins) SQL(SELECT) 1 sql project1 SQL File 7*
Limit to 1000 rows
42
43 SELECT * FROM student
44 WHERE student_score = (SELECT MAX(student_score) FROM student)
45 ORDER BY student_id;
46
47
Result Grid Filter Rows: Export: Wrap Cell Content:
student_id first_name last_name department subject_name student_score
102 Priya Dayalan IT Python 100
105 Gomathy Karthick EEE Python 100
Result 30
Output
Action Output
Time Action
46 16:42:06 DESC student
Message
atus
nnections
d Privileges
nd System Variables
ort
ort/Restore
Shutdown
gs
File
E
ard
ance Reports
Schemas
selected