

NAME : LIKITHA B

USN : 1BM19CS079

DATE : 22/06/21

DBMS LAB PROGRAM – 8

8. STUDENT-FACULTY DATABASE

QUESTION :

CONSIDER THE FOLLOWING DATABASE FOR STUDENT ENROLMENT FOR COURSE:

STUDENT (SNUM: INTEGER, SNAME: STRING, MAJOR: STRING, LEVEL: STRING, AGE: INTEGER)

CLASS (NAME: STRING, MEETS AT: TIME, ROOM: STRING, FID: INTEGER)

ENROLLED (SNUM: INTEGER, CNAME: STRING)

FACULTY (FID: INTEGER, FNAME: STRING, DEPTID: INTEGER)

THE MEANING OF THESE RELATIONS IS STRAIGHTFORWARD; FOR EXAMPLE, ENROLLED HAS ONE RECORD PER STUDENT-CLASS PAIR SUCH THAT THE STUDENT IS ENROLLED IN THE CLASS. LEVEL IS A TWO CHARACTER CODE WITH 4 DIFFERENT VALUES (EXAMPLE:

JUNIOR: JR ETC)

WRITE THE FOLLOWING QUERIES IN SQL. NO DUPLICATES SHOULD BE PRINTED IN ANY OF THE ANSWERS.

I. FIND THE NAMES OF ALL JUNIORS (LEVEL = JR) WHO ARE ENROLLED IN A CLASS TAUGHT BY

II. FIND THE NAMES OF ALL CLASSES THAT EITHER MEET IN ROOM R128 OR HAVE FIVE OR MORE STUDENTS ENROLLED.

III. FIND THE NAMES OF ALL STUDENTS WHO ARE ENROLLED IN TWO CLASSES THAT MEET AT THE SAME TIME.

IV. FIND THE NAMES OF FACULTY MEMBERS WHO TEACH IN EVERY ROOM IN WHICH SOME CLASS IS TAUGHT.

V. FIND THE NAMES OF FACULTY MEMBERS FOR WHOM THE COMBINED ENROLMENT OF THE COURSES THAT THEY TEACH IS LESS THAN FIVE.

VI. FIND THE NAMES OF STUDENTS WHO ARE NOT ENROLLED IN ANY CLASS.

VII. FOR EACH AGE VALUE THAT APPEARS IN STUDENTS, FIND THE LEVEL VALUE THAT APPEARS MOST OFTEN. FOR EXAMPLE, IF

THERE ARE MORE FR LEVEL STUDENTS AGED 18 THAN SR, JR, OR SO STUDENTS AGED 18, YOU SHOULD PRINT THE PAIR (18,

FR).

PROGRAM CODE :

```
CREATE DATABASE STUDENT_FACULTY;
USE STUDENT_FACULTY;
CREATE TABLE STUDENT(
SNUM INT,
SNAME VARCHAR(10),
MAJOR VARCHAR(2),
LVL VARCHAR(2),
AGE INT, PRIMARY KEY(SNUM));

CREATE TABLE FACULTY(
FID INT,FNAME VARCHAR(20),
DEPTID INT,
PRIMARY KEY(FID));

CREATE TABLE CLASS(
CNAME VARCHAR(20),
METTS_AT TIMESTAMP,
ROOM VARCHAR(10),
FID INT,
PRIMARY KEY(CNAME),
FOREIGN KEY(FID) REFERENCES FACULTY(FID));

CREATE TABLE ENROLLED(
SNUM INT,
CNAME VARCHAR(20),
PRIMARY KEY(SNUM,CNAME),
FOREIGN KEY(SNUM) REFERENCES STUDENT(SNUM),
FOREIGN KEY(CNAME) REFERENCES CLASS(CNAME));

INSERT INTO STUDENT VALUES(1, 'JHON', 'CS', 'SR', 19);
INSERT INTO STUDENT VALUES(2, 'SMITH', 'CS', 'JR', 20);
INSERT INTO STUDENT VALUES(3 , 'JACOB', 'CV', 'SR', 20);
INSERT INTO STUDENT VALUES(4, 'TOM ', 'CS', 'JR', 20);
INSERT INTO STUDENT VALUES(5, 'RAHUL', 'CS', 'JR', 20);
INSERT INTO STUDENT VALUES(6, 'RITA', 'CS', 'SR', 21);
SELECT * FROM STUDENT;
INSERT INTO FACULTY VALUES(11, 'HARISH', 1000);
INSERT INTO FACULTY VALUES(12, 'MV', 1000);
INSERT INTO FACULTY VALUES(13 , 'MIRA', 1001);
INSERT INTO FACULTY VALUES(14, 'SHIVA', 1002);
INSERT INTO FACULTY VALUES(15, 'NUPUR', 1000);
SELECT * FROM FACULTY;
INSERT INTO CLASS VALUES('CLASS1', '12/11/15 10:15:16', 'R1', 14);
INSERT INTO CLASS VALUES('CLASS10', '12/11/15 10:15:16', 'R128', 14);
INSERT INTO CLASS VALUES('CLASS2', '12/11/15 10:15:20', 'R2', 12);
```

```

INSERT INTO CLASS VALUES('CLASS3', '12/11/15 10:15:25', 'R3', 11);
INSERT INTO CLASS VALUES('CLASS4', '12/11/15 20:15:20', 'R4', 14);
INSERT INTO CLASS VALUES('CLASS5', '12/11/15 20:15:20', 'R3', 15);
INSERT INTO CLASS VALUES('CLASS6', '12/11/15 13:20:20', 'R2', 14);
INSERT INTO CLASS VALUES('CLASS7', '12/11/15 10:10:10', 'R3', 14);
SELECT * FROM CLASS;
INSERT INTO ENROLLED VALUES(1, 'CLASS1');
INSERT INTO ENROLLED VALUES(2, 'CLASS1');
INSERT INTO ENROLLED VALUES(3, 'CLASS3');
INSERT INTO ENROLLED VALUES(4, 'CLASS3');
INSERT INTO ENROLLED VALUES(5, 'CLASS4');
INSERT INTO ENROLLED VALUES(1, 'CLASS5');
INSERT INTO ENROLLED VALUES(2, 'CLASS5');
INSERT INTO ENROLLED VALUES(3, 'CLASS5');
INSERT INTO ENROLLED VALUES(4, 'CLASS5');
INSERT INTO ENROLLED VALUES(5, 'CLASS5');
SELECT * FROM ENROLLED;
-- QUERY 1
SELECT DISTINCT S.SNAME
FROM STUDENT S, CLASS C, ENROLLED E, FACULTY F
WHERE S.SNUM = E.SNUM AND E.CNAME = C.CNAME AND C.FID = F.FID AND
F.FNAME = 'HARISH' AND S.LVL = 'JR';
-- QUERY 2
SELECT DISTINCT CNAME
FROM CLASS
WHERE ROOM='ROOM128'
OR
CNAME IN (SELECT E.CNAME FROM ENROLLED E GROUP BY E.CNAME HAVING COUNT(*)>=5);
-- QUERY 3
SELECT DISTINCT S.SNAME
FROM STUDENT S
WHERE S.SNUM IN (SELECT E1.SNUM
FROM ENROLLED E1, ENROLLED E2, CLASS C1, CLASS C2
WHERE E1.SNUM = E2.SNUM AND E1.CNAME <> E2.CNAME
AND E1.CNAME = C1.CNAME
AND E2.CNAME = C2.CNAME AND C1.METTS_AT = C2.METTS_AT);
-- QUERY 4
SELECT F.FNAME, F.FID
FROM FACULTY F
WHERE F.FID IN ( SELECT FID FROM CLASS
GROUP BY FID HAVING COUNT(*)=(SELECT COUNT(DISTINCT ROOM) FROM CLASS) );
-- QUERY 5
SELECT DISTINCT F.FNAME
FROM FACULTY F
WHERE 5 > (SELECT COUNT(E.SNUM)
FROM CLASS C, ENROLLED E
WHERE C.CNAME = E.CNAME
AND C.FID = F.FID);

```

```

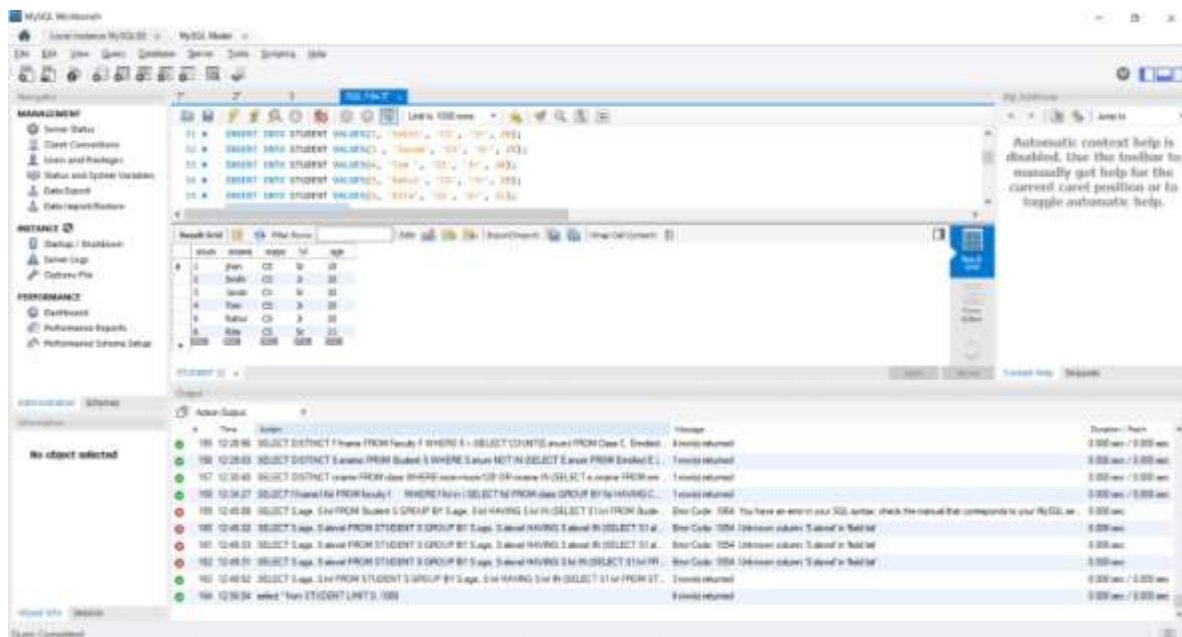
-- QUERY 6
SELECT DISTINCT S.SNAME
FROM STUDENT S
WHERE S.SNUM NOT IN (SELECT E.SNUM
FROM ENROLLED E );

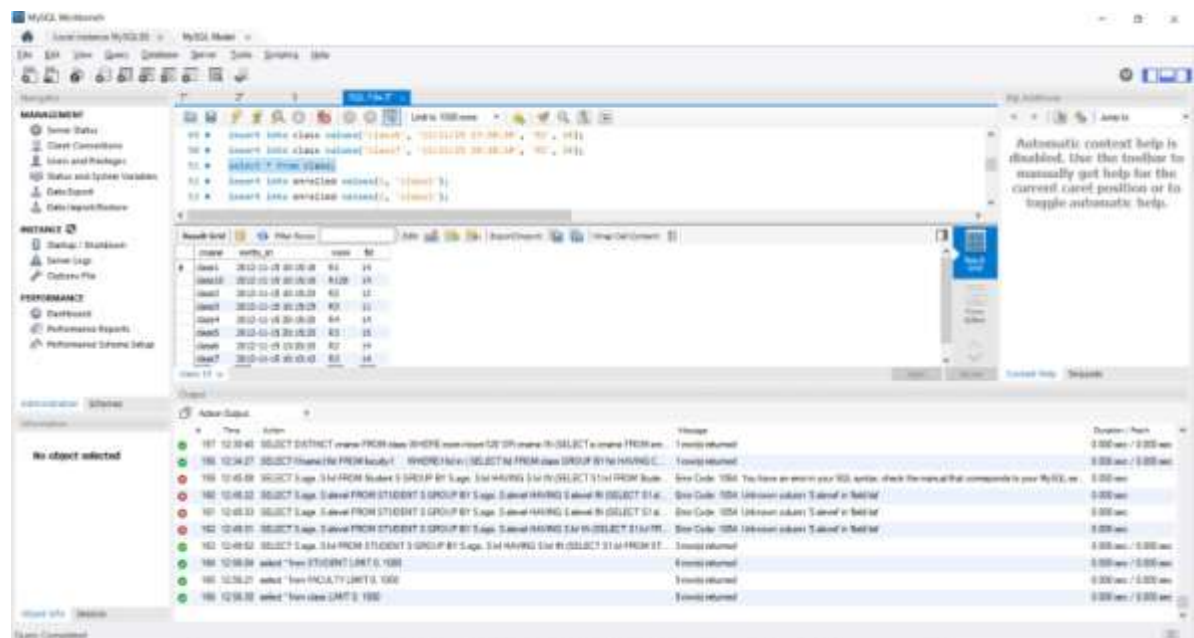
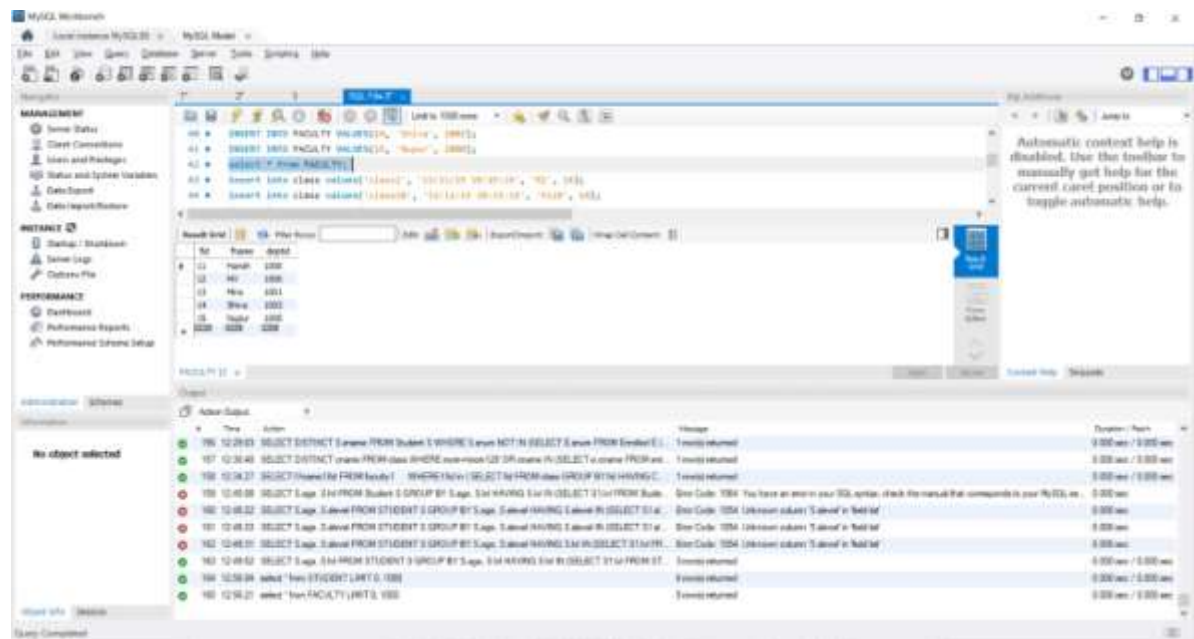
-- QUERY 7
SELECT S.AGE, S.LVL
FROM STUDENT S
GROUP BY S.AGE, S.LVL
HAVING S.LVL IN (SELECT S1.LVL
FROM STUDENT S1
WHERE S1.AGE=S.AGE
GROUP BY S1.AGE, S1.LVL
HAVING COUNT(*) >= ALL (SELECT COUNT(*)
FROM STUDENT S2
WHERE S1.AGE=S2.AGE
GROUP BY S2.LVL, S2.AGE))
ORDER BY S.AGE;

```

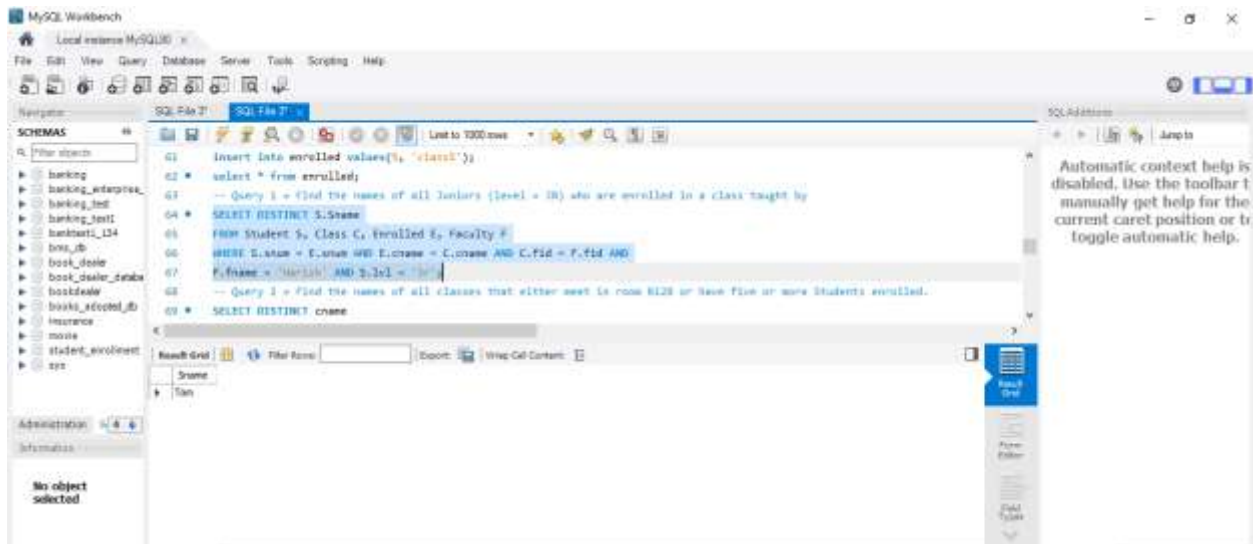
SCREENSHOTS OF THE PROGRAM OUTPUT :

➔ CREATION AND INSERTION OF VALUES

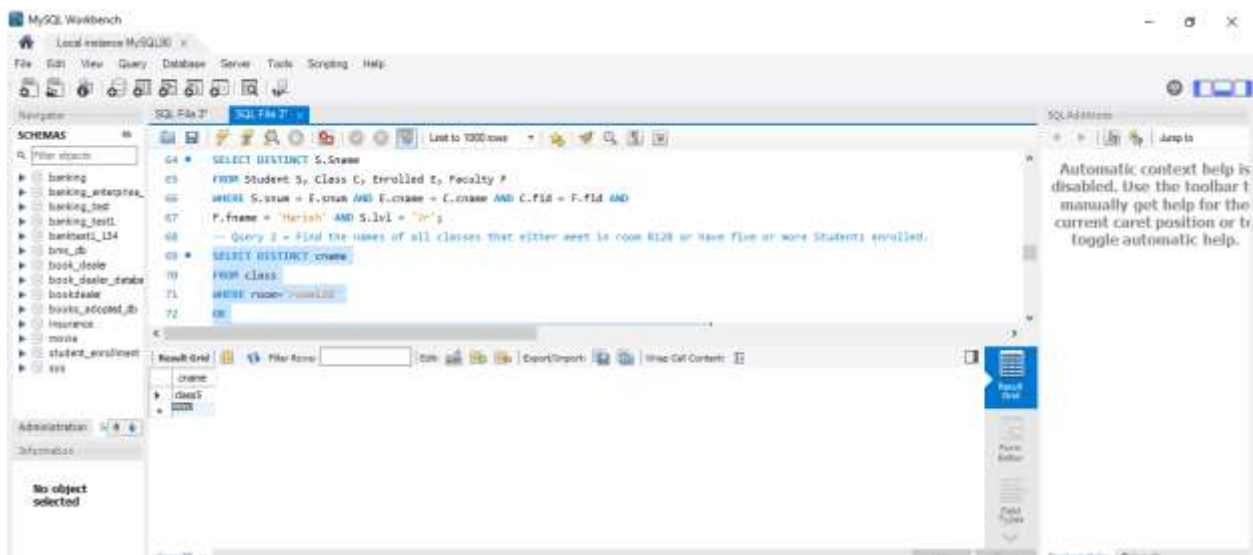




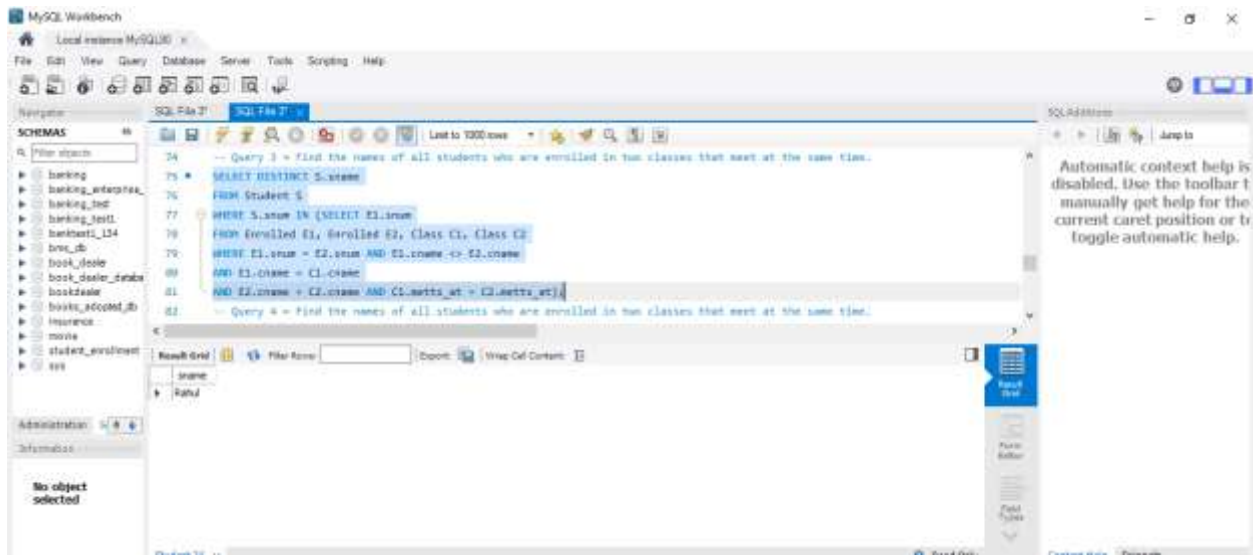
QUERY 1 = FIND THE NAMES OF ALL JUNIORS (LEVEL = JR) WHO ARE ENROLLED IN A CLASS TAUGHT BY



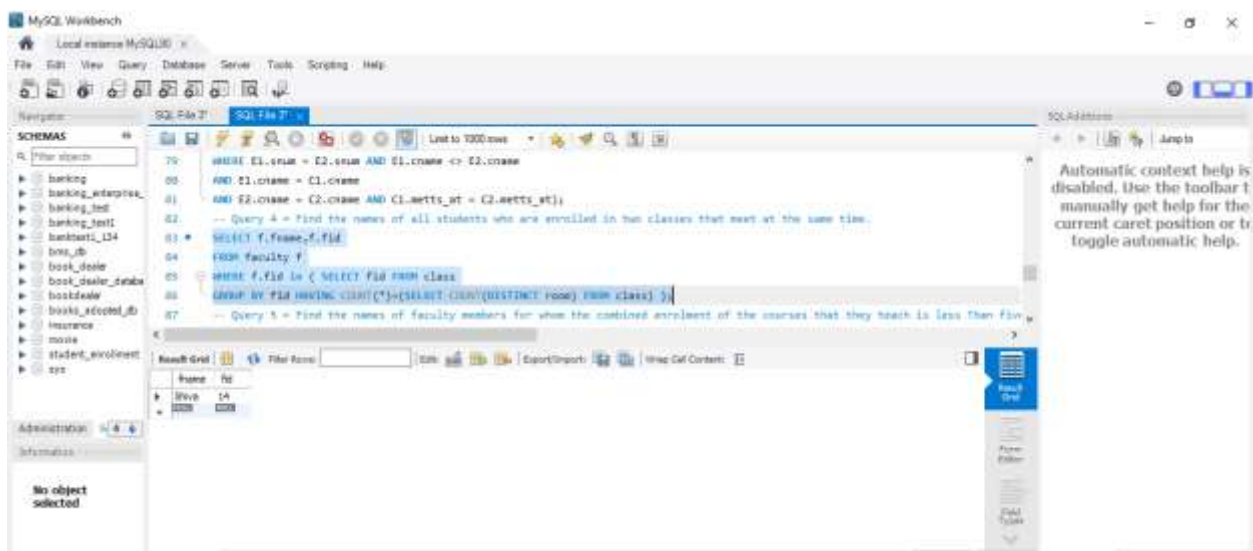
QUERY 2 = FIND THE NAMES OF ALL CLASSES THAT EITHER MEET IN ROOM R128 OR HAVE FIVE OR MORE STUDENTS ENROLLED.



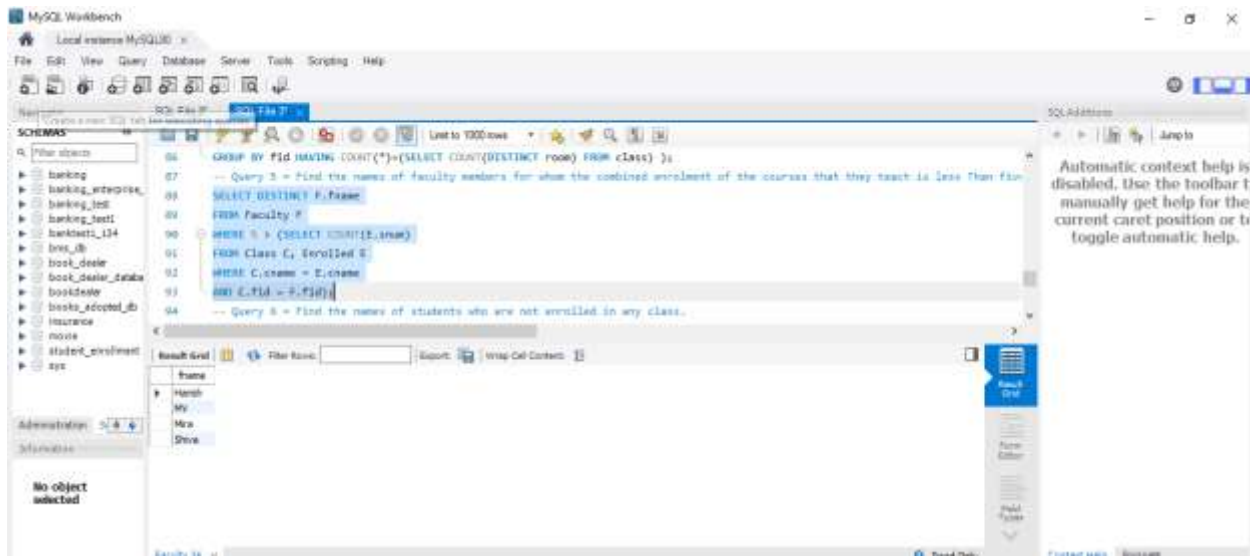
QUERY 3 = FIND THE NAMES OF ALL STUDENTS WHO ARE ENROLLED IN TWO CLASSES THAT MEET AT THE SAME TIME.



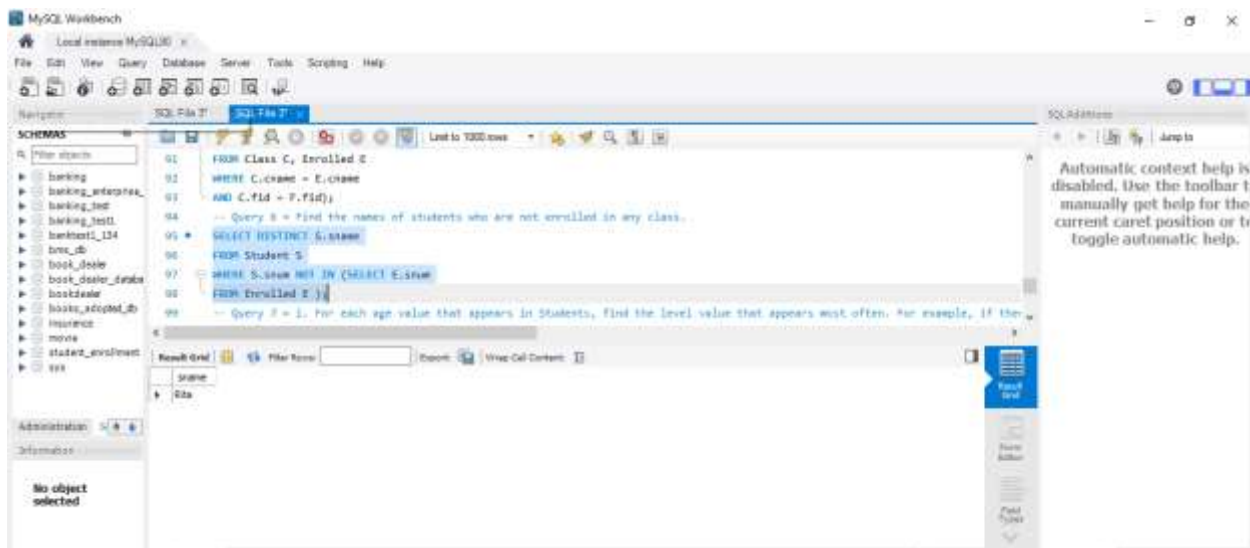
QUERY 4 = FIND THE NAMES OF ALL STUDENTS WHO ARE ENROLLED IN TWO CLASSES THAT MEET AT THE SAME TIME.



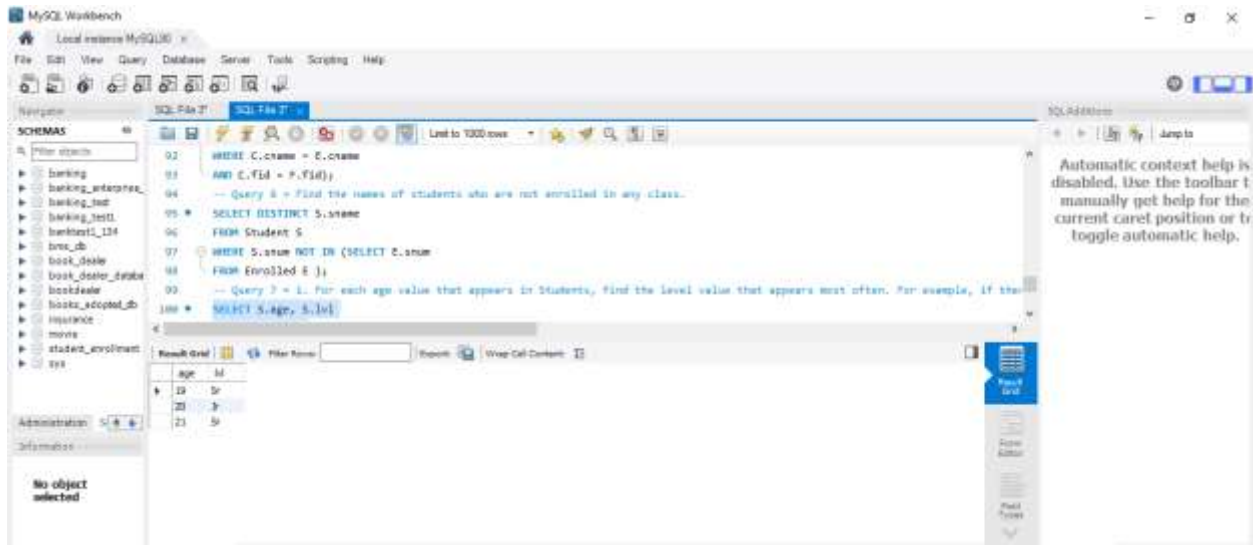
QUERY 5 = FIND THE NAMES OF FACULTY MEMBERS FOR WHOM THE COMBINED ENROLMENT OF THE COURSES THAT THEY TEACH IS LESS THAN FIVE.



QUERY 6 = FIND THE NAMES OF STUDENTS WHO ARE NOT ENROLLED IN ANY CLASS.



QUERY 7 = I. FOR EACH AGE VALUE THAT APPEARS IN STUDENTS, FIND THE LEVEL VALUE THAT APPEARS MOST OFTEN. FOR EXAMPLE, IF THERE ARE MORE FR LEVEL STUDENTS AGED 18 THAN SR, JR, OR SO STUDENTS AGED 18, YOU SHOULD PRINT THE PAIR (18,FR)



**** END OF WEEK 8 PROGRAM ****