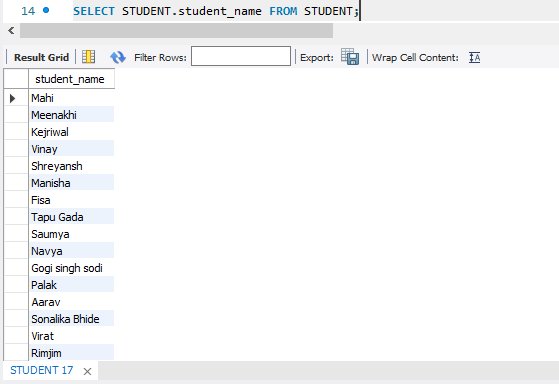
**(PRACTISE SET OF SQL QURERIES) *D.KAVITA***

Q1.Retrieve names of the students enrolled in any course.

SELECT STUDENT.student\_name FROM STUDENT;



Q2. Retrieve names of the students enrolled in at least one part time course.

SELECT STUDENT.student\_name from STUDENT

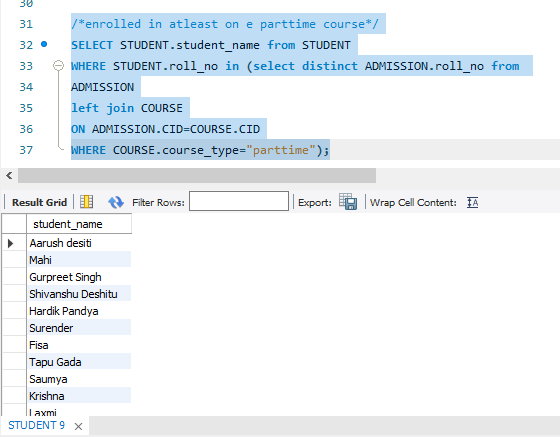
WHERE STUDENT.roll\_no in (select distinct ADMISSION.roll\_no from

ADMISSION

left join COURSE

ON ADMISSION.CID=COURSE.CID

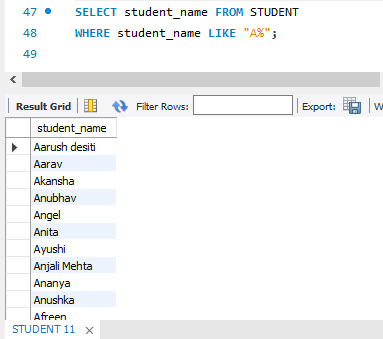
WHERE COURSE.course\_type="parttime");



Q3.Retrieve student’s names starting with letter “A”.

SELECT student\_name FROM STUDENT

WHERE student\_name LIKE "A%";



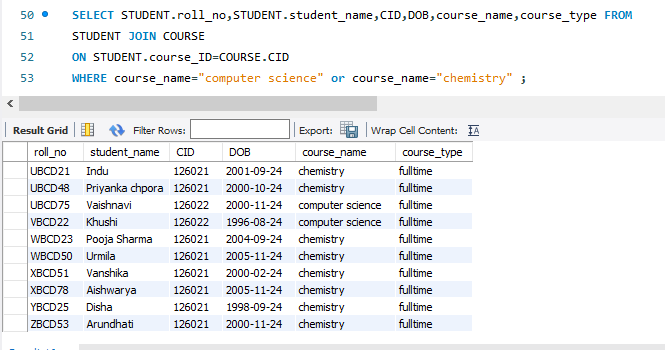
Q4. Retrieve student’details studying in course “coumputer science” or “chemistry”.

SELECT STUDENT.roll\_no,STUDENT.student\_name,CID,DOB,course\_name,course\_type FROM

STUDENT JOIN COURSE

ON STUDENT.course\_ID=COURSE.CID

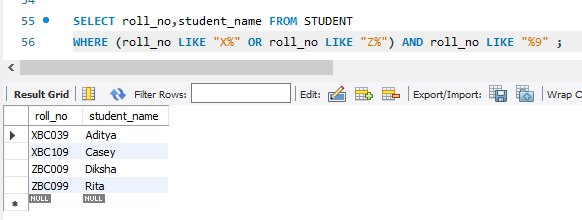
WHERE course\_name="computer science" or course\_name="chemistry" ;



Q5.Retrieve student’s names whose roll no either starts with “X” or “Z” and ends with “9”.

SELECT roll\_no,student\_name FROM STUDENT

WHERE (roll\_no LIKE "X%" OR roll\_no LIKE "Z%") AND roll\_no LIKE "%9" ;

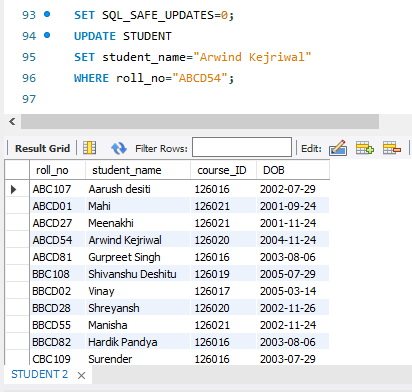


Q6.Update student table for modifying a student name.

UPDATE STUDENT

SET student\_name="Arwind Kejriwal"

WHERE roll\_no="ABCD54";



Q7.Find course name in which more than five sudents have enrolled.

SELECT COURSE.\*, D.CID\_Count FROM COURSE

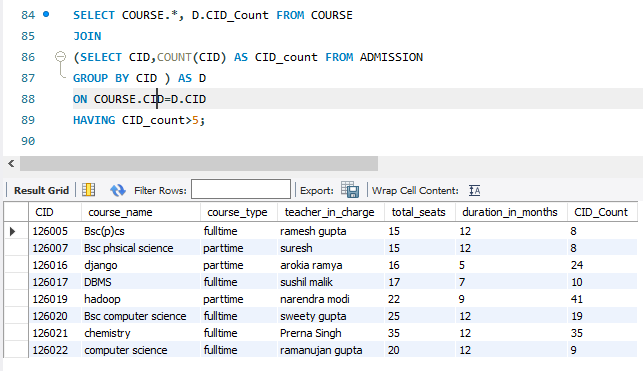
JOIN

(SELECT CID,COUNT(CID) AS CID\_count FROM ADMISSION

GROUP BY CID ) AS D

ON COURSE.CID=D.CID

HAVING CID\_count>5;



Q8.Find the name of the youngest student enrolled in course “Bsc(P)CS”.

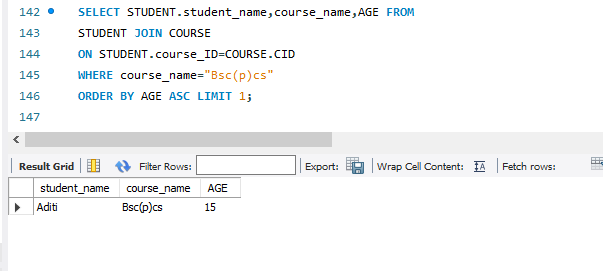
SELECT STUDENT.student\_name,course\_name,AGE FROM

STUDENT JOIN COURSE

ON STUDENT.course\_ID=COURSE.CID

WHERE course\_name="Bsc(p)cs"

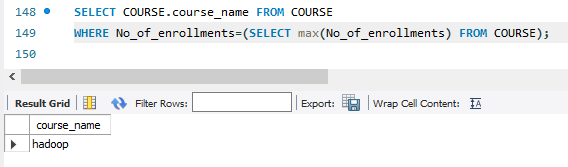
ORDER BY AGE ASC LIMIT 1;



Q9.Find the name of the most popular society (on the basis of enrolled students).

SELECT COURSE.course\_name FROM COURSE

WHERE No\_of\_enrollments=(SELECT max(No\_of\_enrollments) FROM COURSE);

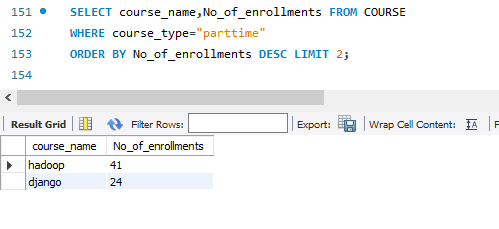


Q10.Find the name of two popular parttime course (on the basis of enrolled students).

SELECT course\_name,No\_of\_enrollments FROM COURSE

WHERE course\_type="parttime"

ORDER BY No\_of\_enrollments DESC LIMIT 2;



Q11.Find the student names who are admitted to full time course only.

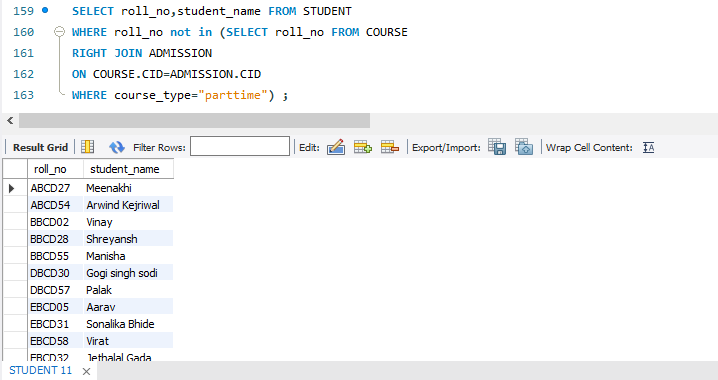
SELECT roll\_no,student\_name FROM STUDENT

WHERE roll\_no not in (SELECT roll\_no FROM COURSE

RIGHT JOIN ADMISSION

ON COURSE.CID=ADMISSION.CID

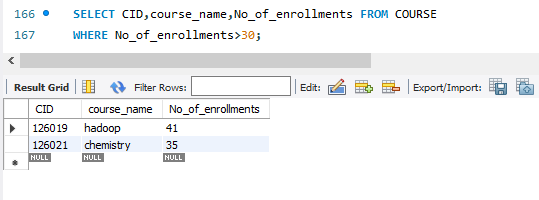
WHERE course\_type="parttime") ;



Q12.Find course names in which more than 30 students took admission.

SELECT CID,course\_name,No\_of\_enrollments FROM COURSE

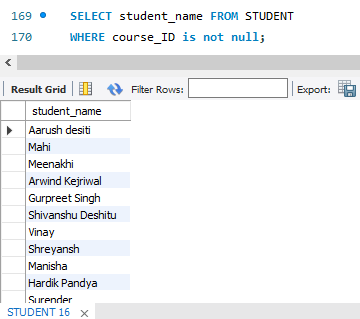
WHERE No\_of\_enrollments>30;



Q13.Find the name of all students who took admision to any course and course names in which at least one student has enrolled.

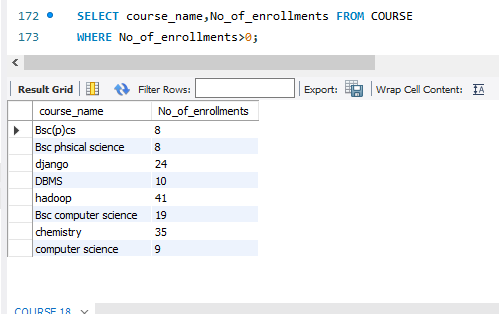
SELECT student\_name FROM STUDENT

WHERE course\_ID is not null;



SELECT course\_name,No\_of\_enrollments FROM COURSE

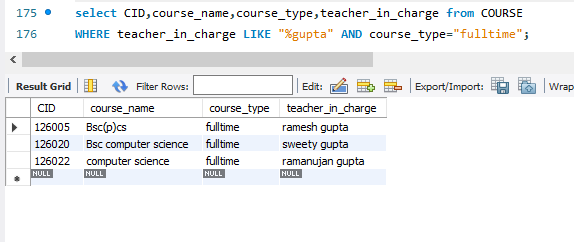
WHERE No\_of\_enrollments>0;



Q14.Find course names where teacher\_in\_charge has “gupta” as surname and the course is full time.

select CID,course\_name,course\_type,teacher\_in\_charge from COURSE

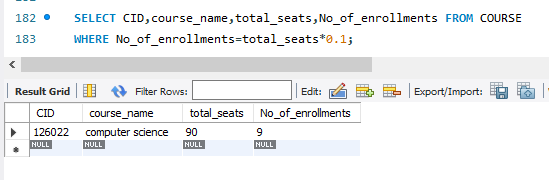
WHERE teacher\_in\_charge LIKE "%gupta" AND course\_type="fulltime";



Q15.Find course names in which the number of enrolled students is only 10% of its total seats.

SELECT CID,course\_name,total\_seats,No\_of\_enrollments FROM COURSE

WHERE No\_of\_enrollments=total\_seats\*0.1;

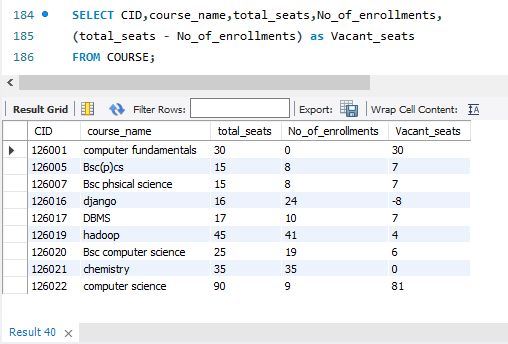


Q16.Display the vacant seats for each course.

SELECT CID,course\_name,total\_seats,No\_of\_enrollments,

(total\_seats - No\_of\_enrollments) as Vacant\_seats

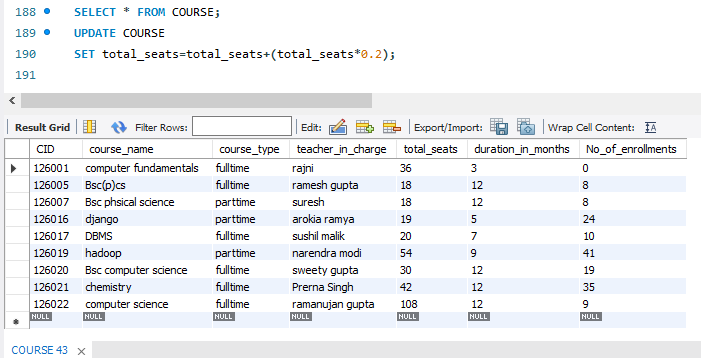
FROM COURSE;



Q17.Increment total seats of each course by 20%.

UPDATE COURSE

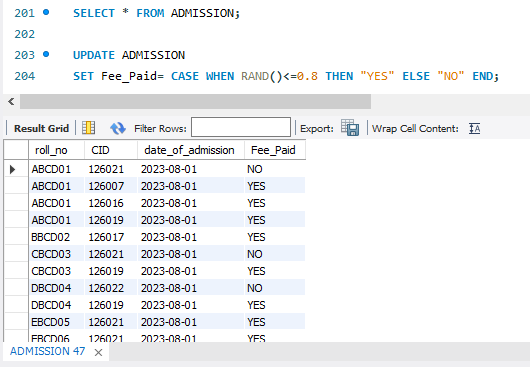
SET total\_seats=total\_seats+(total\_seats\*0.2);



Q18.Add enrollment fees paid (“yes”/”no”) field in the enrollment table.

UPDATE ADMISSION

SET Fee\_Paid= CASE WHEN RAND()<=0.8 THEN "YES" ELSE "NO" END;

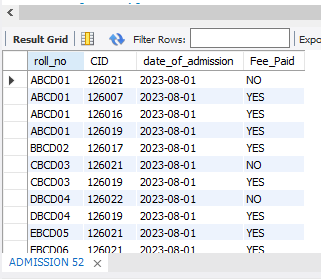


Q19.Update the date of admission for all courses by 1 year.

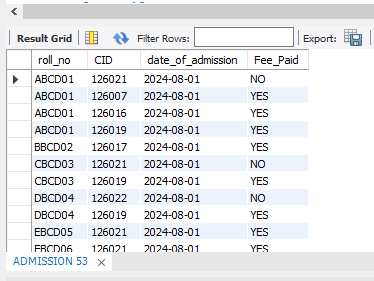
UPDATE ADMISSION

SET date\_of\_admission = DATE\_ADD(date\_of\_admission,INTERVAL 1 YEAR) ;

BEFORE



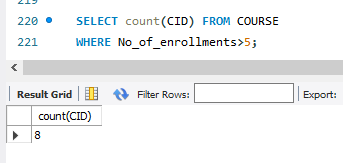
AFTER



Q20.Count the number of courses with more than 5 students enrolled for each type of course.

SELECT count(CID) FROM COURSE

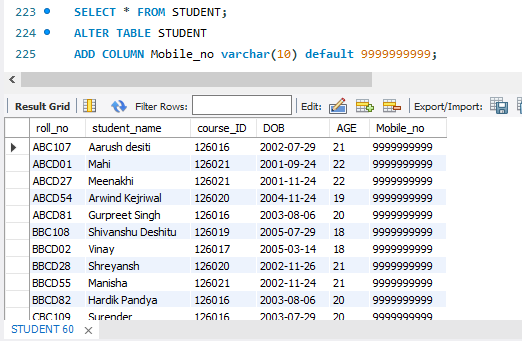
WHERE No\_of\_enrollments>5;



Q21.Add column mobile number in student table with default value “9999999999”.

ALTER TABLE STUDENT

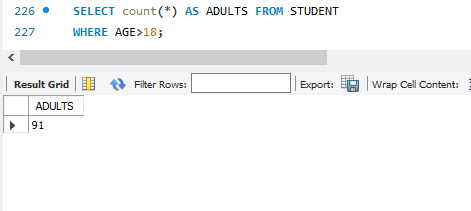
ADD COLUMN Mobile\_no varchar(10) default 9999999999;



Q22.Find the total number of students whose age is>18.

SELECT count(\*) AS ADULTS FROM STUDENT

WHERE AGE>18;



Q23.Find the names of students who are born in 2001 and are admitted to at least one part time course.

SELECT DISTINCT STUDENT.roll\_no,student\_name,DOB,course\_type FROM STUDENT

RIGHT JOIN

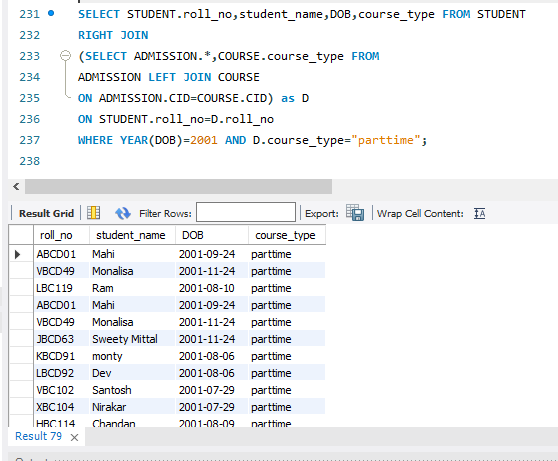
(SELECT ADMISSION.\*,COURSE.course\_type FROM

ADMISSION LEFT JOIN COURSE

ON ADMISSION.CID=COURSE.CID) as D

ON STUDENT.roll\_no=D.roll\_no

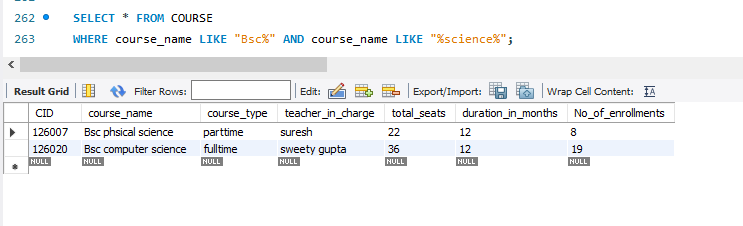
WHERE YEAR(DOB)=2001 AND D.course\_type="parttime";



Q24.Count all courses having “science” in the name and staring with the word “B.sc”.

SELECT \* FROM COURSE

WHERE course\_name LIKE "Bsc%" AND course\_name LIKE "%science%";



­­