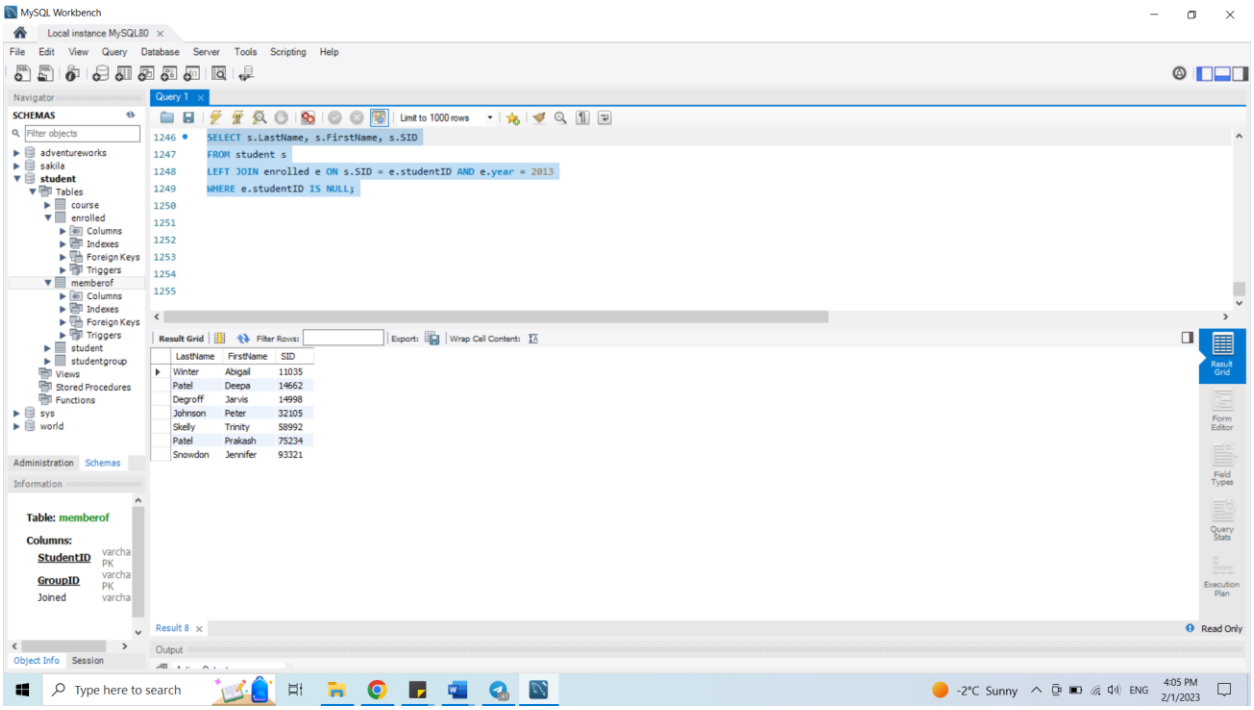
Query 1: Students who did not enroll in 2013

```
SELECT LastName, FirstName, SID
FROM student
WHERE sid NOT IN (SELECT studentID
FROM enrolled
WHERE year = 2013);
```

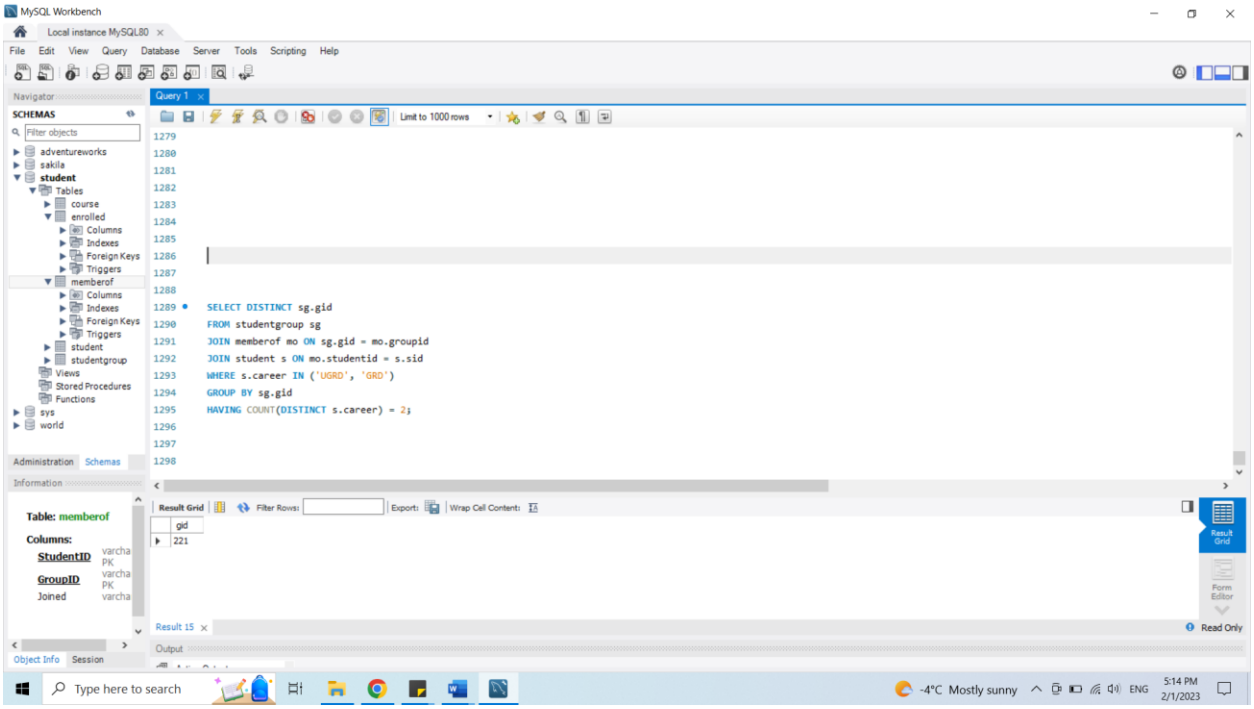
Join:

```
SELECT s.LastName, s.FirstName, s.SID
FROM student s
LEFT JOIN enrolled e ON s.SID = e.studentID AND e.year = 2013
WHERE e.studentID IS NULL;
```



Query 2: With Join:

```
SELECT DISTINCT sg.gid
FROM studentgroup sg
JOIN memberof mo ON sg.gid = mo.groupid
JOIN student s ON mo.studentid = s.sid
WHERE s.career IN ('UGRD', 'GRD')
GROUP BY sg.gid
HAVING COUNT(DISTINCT s.career) = 2;
```

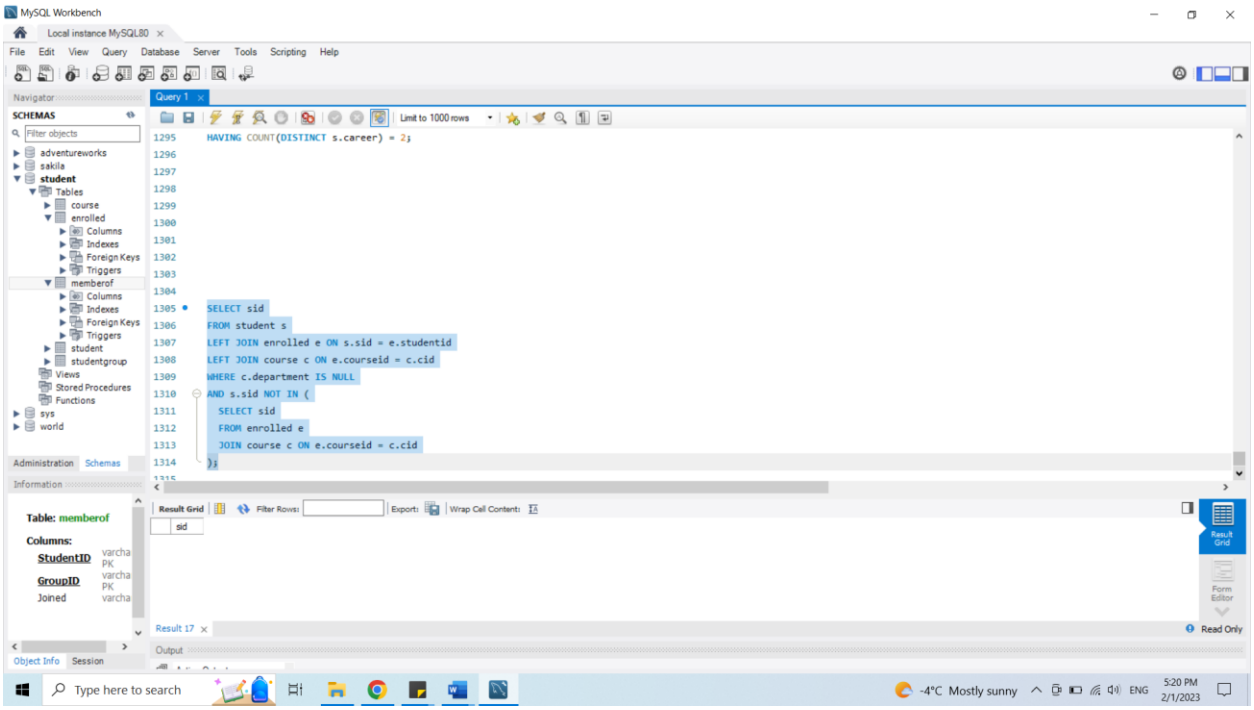


This query performs a join between the "studentgroup", "memberof", and "student" tables and filters the results to only include the group IDs (gid) where the career of the members is either "UGRD" or "GRD". The query then uses the "GROUP BY" clause to group the results by group ID, and the "HAVING" clause to only include the groups where there are 2 distinct values of the "career" column, indicating that the group has both graduate and undergraduate members. The "DISTINCT" keyword ensures that the same group ID is not counted multiple times.

Query 3: I provided two different queries for this part.

First Query:

```
SELECT sid
FROM student s
LEFT JOIN enrolled e ON s.sid = e.studentid
LEFT JOIN course c ON e.courseid = c.cid
WHERE c.department IS NULL
AND s.sid NOT IN (
  SELECT sid
  FROM enrolled e
  JOIN course c ON e.courseid = c.cid
);
```



Second query

```
SELECT DISTINCT student.sid
FROM student
JOIN enrolled ON student.sid = enrolled.studentid
JOIN course ON enrolled.courseid = course.cid
GROUP BY student.sid
HAVING COUNT(DISTINCT course.department) =
   (SELECT COUNT(DISTINCT department) FROM course)
;
```