

QUERIES WITH QUANTIFIERS USING COUNT FUNCTION



QUERIES WITH QUANTIFIERS

- Find the sid of each student who takes **some** CS courses
- Find the sid of each student who takes **no** CS courses
- Find the sid of each student who takes **not only** CS courses
- Find the sid of each student who takes **only** CS courses
- Find the sid of each student who takes **not all** CS courses
- Find the sid of each student who takes **all** CS courses
- Find the sid of each student who takes **at least half of** CS courses

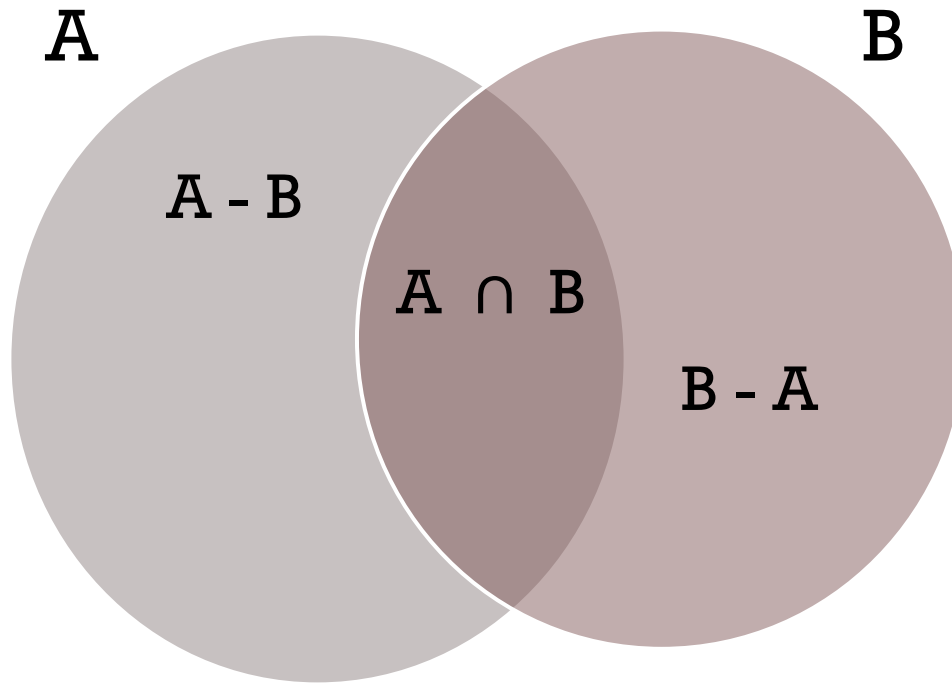


VENN DIAGRAM AND SQL TEMPLATE

- There is a **Venn diagram with conditions** to express a query with a quantifier
- These conditions can be expressed as **counting conditions**
- There is a corresponding SQL statement to express this Venn diagram with condition using the **COUNT function**



VENN DIAGRAM OF 2 SETS



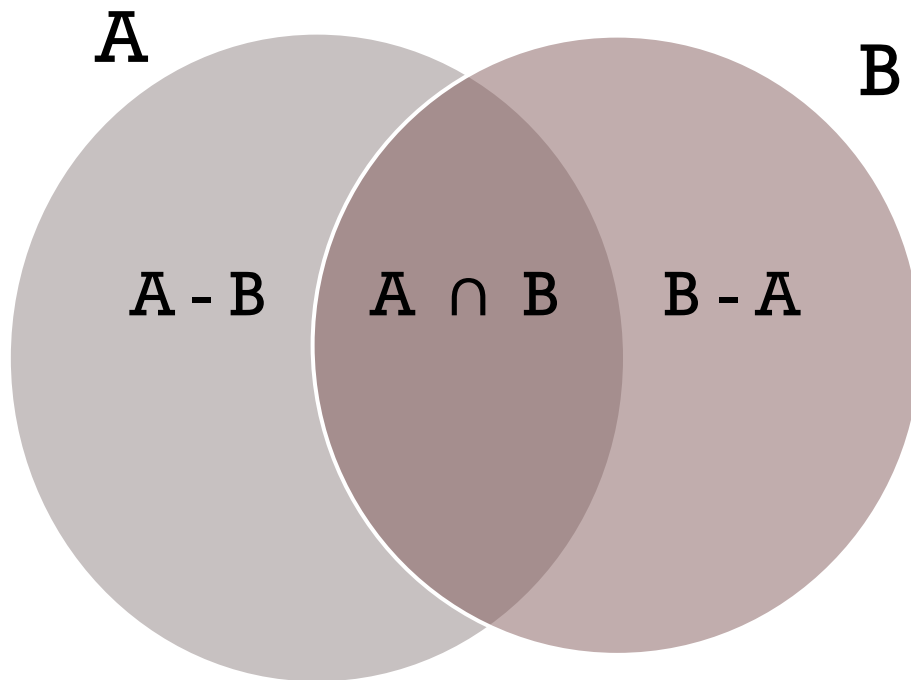
A-B Left Ear

B-A Right Ear

A \cap B Lens



VENN DIAGRAM OF 2 SETS WITH CONDITIONS



Condition
$A \cap B \neq \emptyset \Leftrightarrow A \cap B \geq 1$
$A \cap B = \emptyset \Leftrightarrow A \cap B = 0$
$A - B \neq \emptyset \Leftrightarrow A - B \geq 1$
$A - B = \emptyset \Leftrightarrow A - B = 0$
$B - A \neq \emptyset \Leftrightarrow B - A \geq 1$
$B - A = \emptyset \Leftrightarrow B - A = 0$
$ A - B = 0$ and $ B - A = 0$
$ A \cap B \geq 2$
...



VENN DIAGRAM FOR OUR QUERIES

- For a student with key `sid`, `CoursesEnrolledIn(sid)` denotes the set of courses taken by this student
- `CS_courses` denotes the set of courses offered by the 'CS' department
- So in our previous Venn diagram

$A = \text{CoursesEnrolledIn}(\text{sid})$

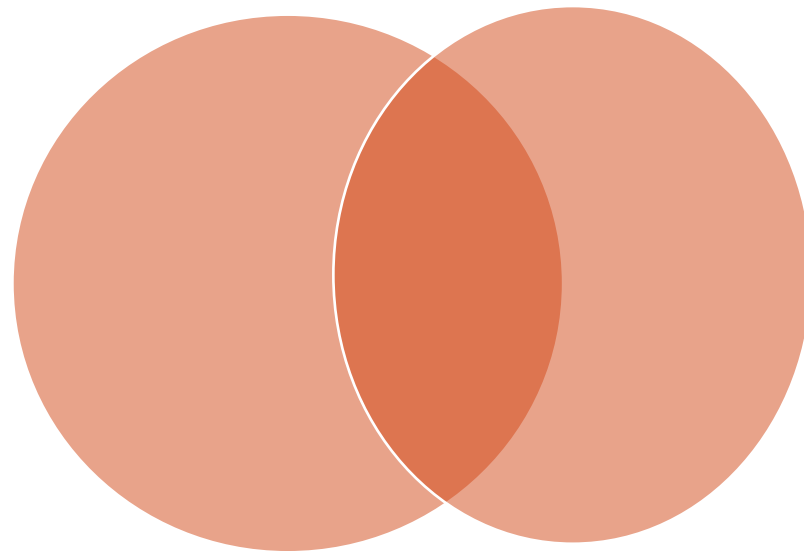
$B = \text{CS_Courses}$

- Note that for different values of `sid`, `CoursesEnrolledIn(sid)` denote different sets



VENN DIAGRAM FOR OUR QUERIES

CoursesEnrolledIn(sid)



CS_Courses

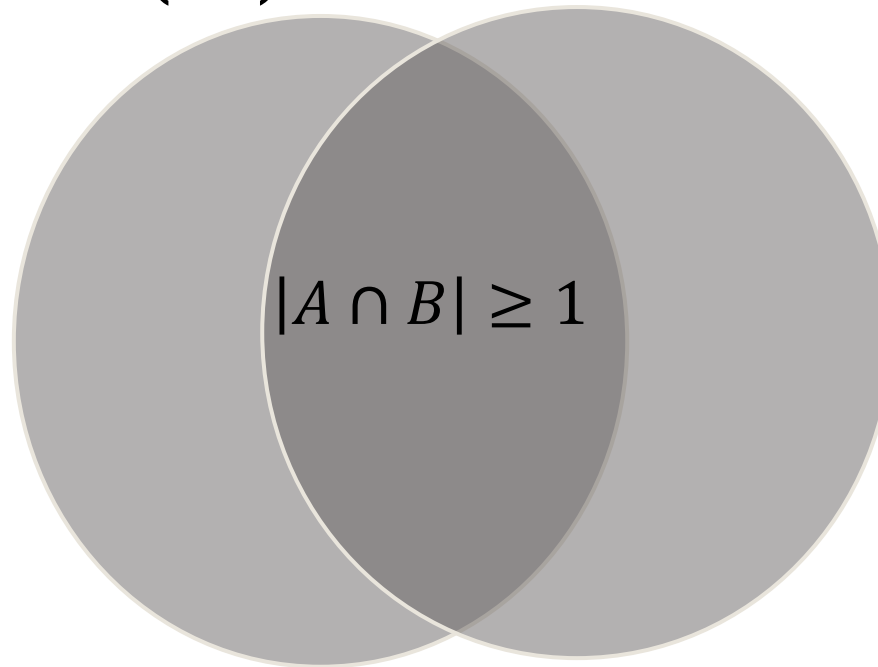


SOME

Find **sid** of each student who takes **some** CS courses

CoursesEnrolledIn(**sid**)

CS_courses



SOME

```
SELECT sid
FROM Student S
WHERE (SELECT COUNT(1)
      FROM (SELECT cno
            FROM CoursesEnrolledIn(sid)
            INTERSECT
            SELECT cno
            FROM CS_Courses) q) ≥ 1;
```

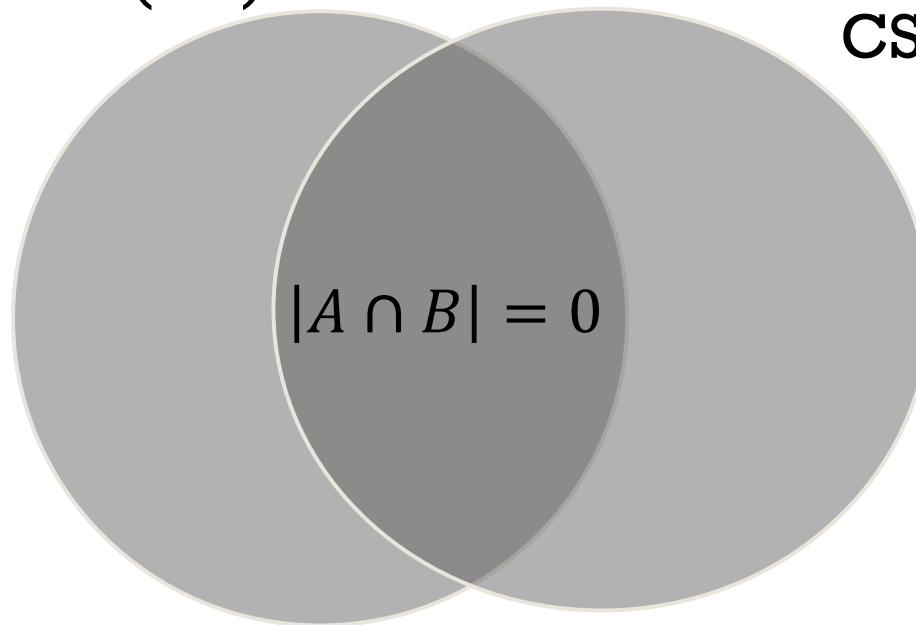


NO

Find sid of each student who takes **no** CS courses

CoursesEnrolledIn(sid)

CS courses



NO

```
SELECT sid
FROM Student S
WHERE (SELECT COUNT(1)
      FROM (SELECT cno
            FROM CoursesEnrolledIn(sid)
            INTERSECT
            SELECT cno
            FROM CS_Courses) q) = 0;
```



NOT ONLY

Find sid of each student who takes **not only** CS courses

CoursesEnrolledIn(sid)

CS courses


$$|A - B| \geq 1$$



NOT ONLY

```
SELECT sid
FROM Student S
WHERE (SELECT COUNT(1)
      FROM (SELECT cno
            FROM CoursesEnrolledIn(sid)
            EXCEPT
            SELECT cno
            FROM CS_Courses) q) ≥ 1;
```



ONLY

Find sid of each student who takes **only** CS courses

CoursesEnrolledIn(sid)

CS courses


$$|A - B| = 0$$



ONLY

```
SELECT sid
FROM Student S
WHERE (SELECT COUNT(1)
      FROM (SELECT cno
            FROM CoursesEnrolledIn(sid)
            EXCEPT
            SELECT cno
            FROM CS_Courses) q) = 0;
```

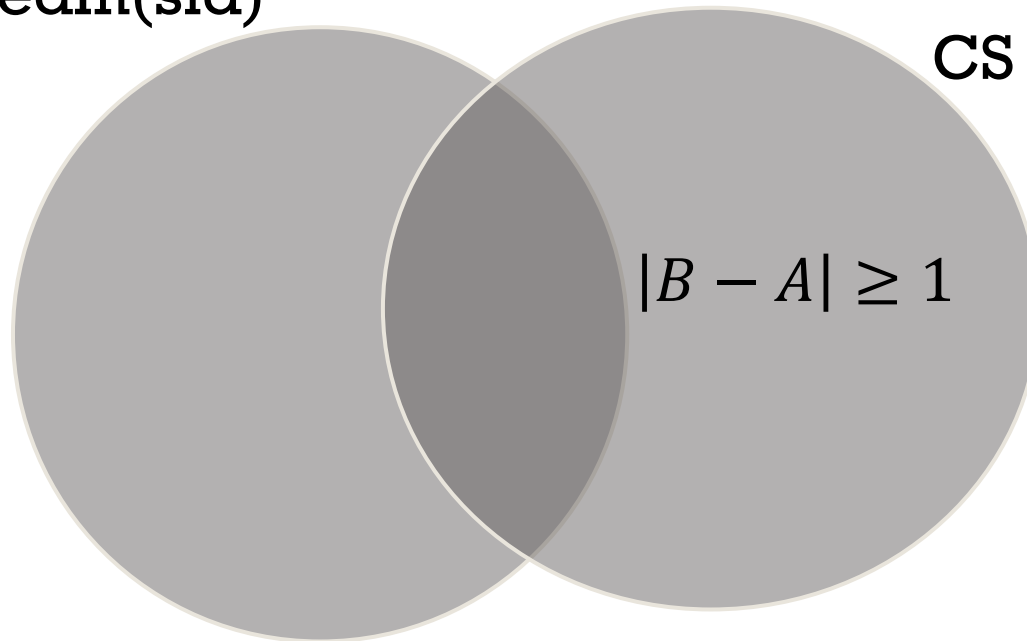


NOT ALL

Find sid of each student who takes **not all** CS courses

CoursesEnrolledIn(sid)

CS courses



NOT ALL

```
SELECT sid
FROM Student S
WHERE (SELECT COUNT(1)
      FROM (SELECT cno
            FROM CS_courses
            EXCEPT
            SELECT cno
            FROM CoursesEnrolledIn(sid)) q) ≥ 1;
```

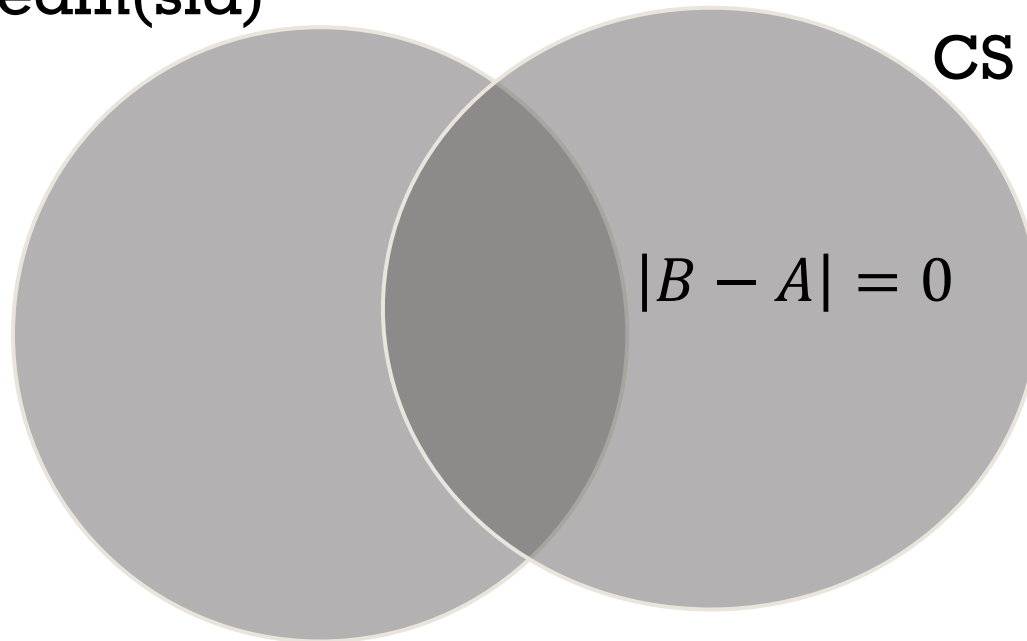


ALL

Find sid of each student who takes **all** CS courses

CoursesEnrolledIn(sid)

CS courses



ALL

```
SELECT sid
FROM Student S
WHERE (SELECT COUNT(1)
      FROM (SELECT cno
            FROM CS_courses
            EXCEPT
            SELECT cno
            FROM CoursesEnrolledIn(sid)) q) = 0;
```



ALL AND ONLY

We have multiple quantifiers: **ALL** and **ONLY**

These must both be specified in the WHERE clause using the **ALL** and **ONLY** quantifier templates:

```
SELECT S.Sid  
FROM   Student S  
WHERE  ALL template  AND  
       ONLY template
```

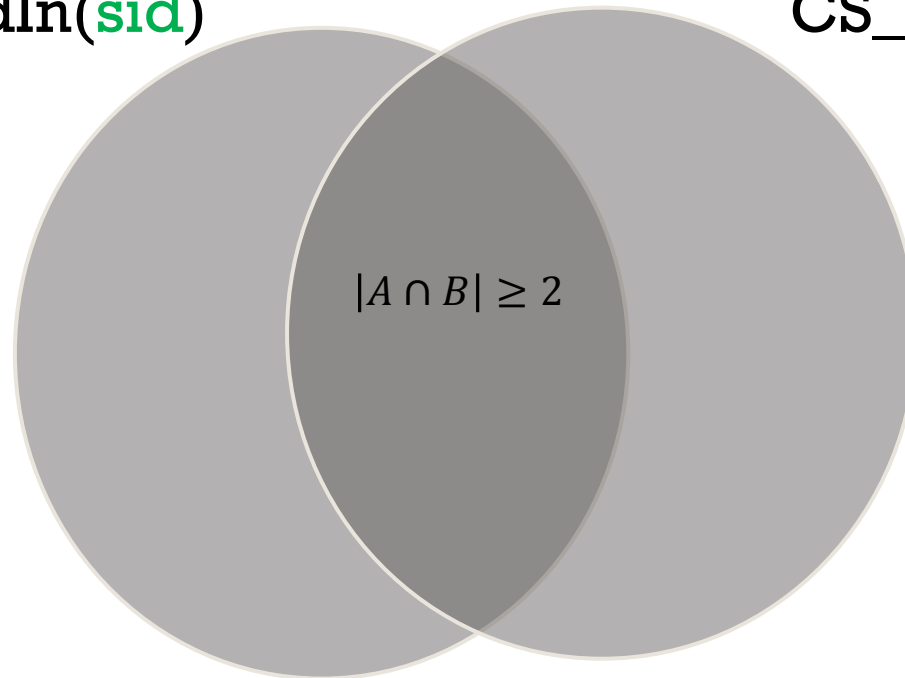


AT LEAST TWO

Find **sid** of each student who takes **at least two**
CS courses

CoursesEnrolledIn(**sid**)

CS_courses



AT LEAST 2

```
SELECT sid
FROM Student S
WHERE (SELECT COUNT(1)
      FROM (SELECT cno
            FROM CoursesEnrolledIn(sid)
            INTERSECT
            SELECT cno
            FROM CS_Courses) q) ≥ 2;
```

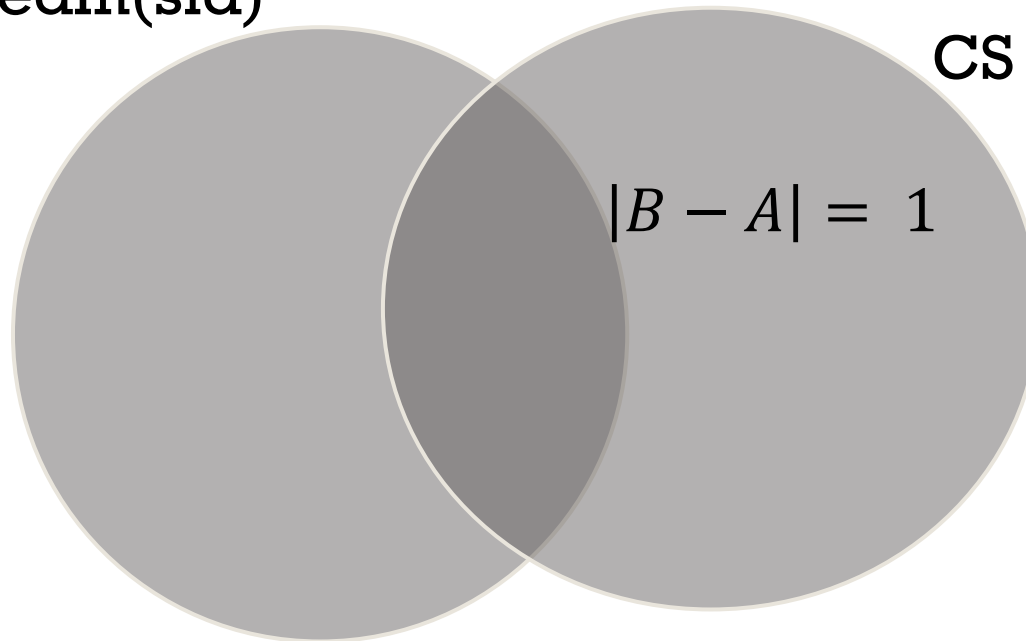


ALL BUT ONE

Find sid of each student who takes **all but one** CS course

CoursesEnrolledIn(sid)

CS courses



ALL BUT ONE

```
SELECT sid
FROM Student S
WHERE (SELECT COUNT(1)
      FROM (SELECT cno
            FROM CS_courses
            EXCEPT
            SELECT cno
            FROM CoursesEnrolledIn(sid)) q) = 1;
```

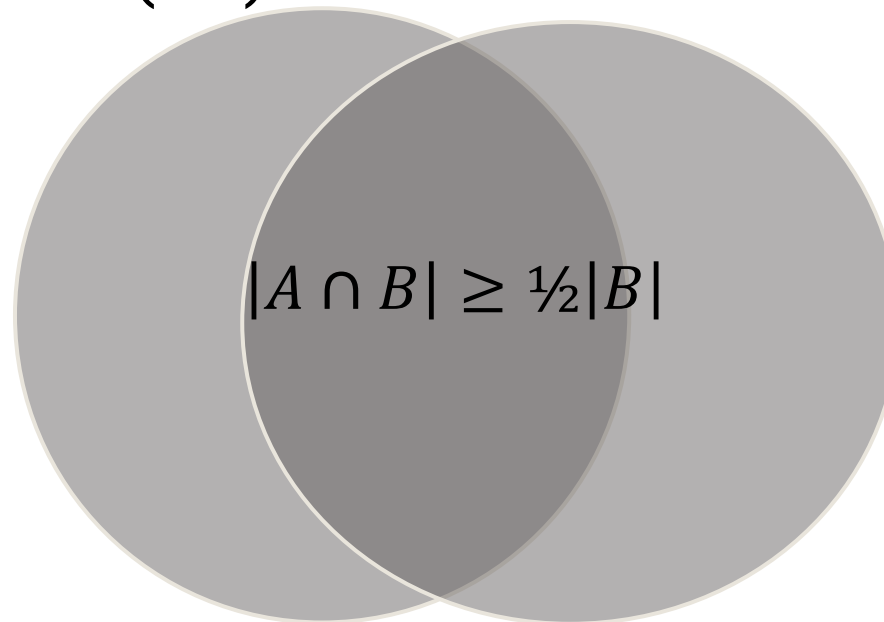


AT LEAST HALF OF

Find the sid of each student who takes **at least half of** the CS courses

CoursesEnrolledIn(sid)

CS courses



AT LEAST HALF OF

```
SELECT sid
FROM Student S
WHERE 2* (SELECT COUNT(1)
          FROM (SELECT cno
                FROM CoursesEnrolledIn(sid)
                INTERSECT
                SELECT cno
                FROM CS_Courses) q) ≥ (SELECT
                                         COUNT (1)
                                         FROM CS_courses)
```

$$|A \cap B| \geq \frac{1}{2}|B|$$
$$2 |A \cap B| \geq |B|$$

