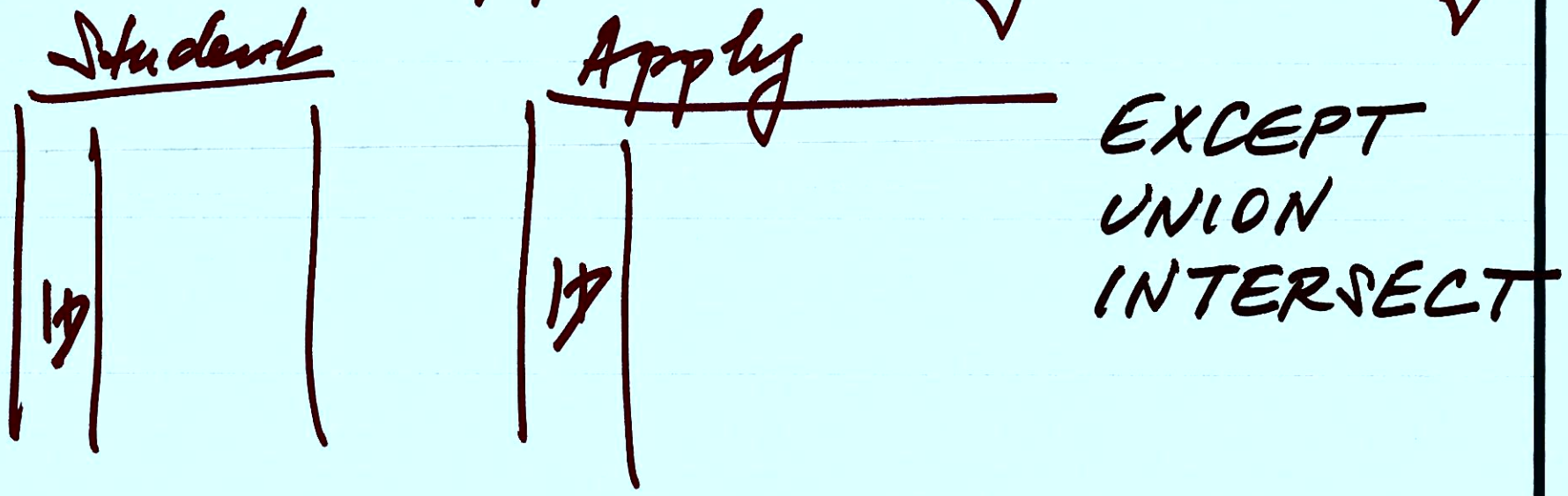
```
SELECT *
FROM Student
WHERE   GPA < 3.2
     OR   GPA >= 3.2
```

ID of all students who have not applied to any university




(SELECT ID FROM STUDENT)

EXCEPT

(SELECT ID FROM APPLY)

Find all names of Students whose GPA is the same as the GPA of the Student with ID 1

```
SELECT name
FROM Student
WHERE GPA =
    (SELECT GPA
     FROM Student
     WHERE ID = 1);
```

Student (outer) Student (inner)

⇒ produces an answer relation once

uncorrelated subquery

# Names and IDs of students applying to CS

SELECT name, ID
FROM Student
WHERE ID IN
(SELECT ID
FROM Apply
WHERE major
LIKE 'CS%')

"set" membership

uncorrelated subquery
⇓
can execute once

"NOT IN" - for the opposite question

Names and IDs of students not applying anywhere

```
SELECT name, ID
FROM Students
WHERE ID NOT IN
      (SELECT ID
       FROM Apply)
```

```
SELECT name, id
FROM Student
WHERE NOT EXISTS
    (SELECT *
     FROM Apply
     WHERE
         Apply.ID = Student.ID)
```

correlated subquery

Find ID of students who apply to >1 campus

```
SELECT ID FROM Apply A
WHERE EXISTS
    (SELECT *
     FROM Apply
     WHERE ID = A.ID
        AND location
        <> A.location)
```

Apply A

A: ID loc

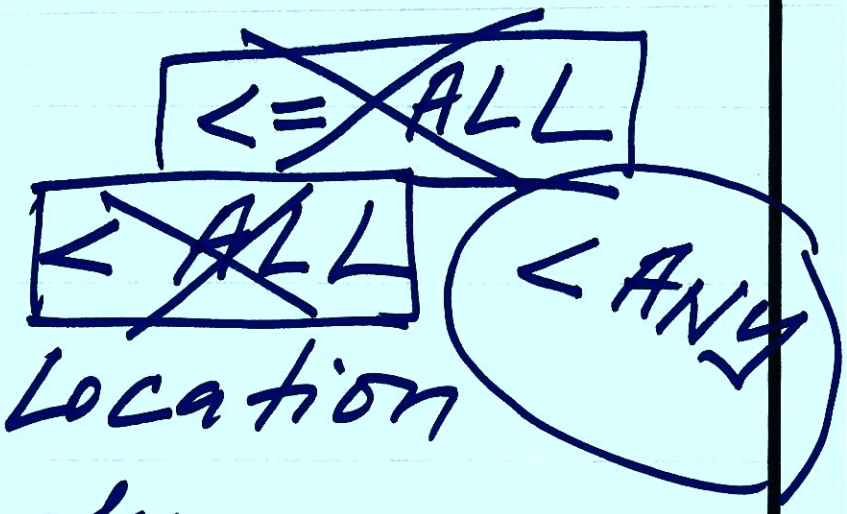
| | | | |
|---|---|---|---|
| ✗ | 1 | 'NCSU' | ⇒ to answ |
| ✗ | 1 | 'UNC-CH' | ⇒ to answ |
| ✗ | 1 | 'Duke' | ⇒ to answ |
| ✗ | 2 | 'UNC-CH' | ✗ |
| → | 3 | 'Duke' | ⇒ to answ |
| → | 3 | 'NCSU' | ⇒ to answ |

Apply

| ID | loc |
|---|---|
| 1 | 'NCSU' |
| 1 | 'UNC-CH' |
| 1 | 'Duke' |
| 2 | 'UNC-CH' |
| 3 | 'Duke' |
| 3 | 'NCSU' |

Find IDs of students who have applied to >1 university

```
SELECT ID
FROM Apply A
WHERE Location
    (SELECT Location
     FROM Apply
     WHERE ID = A.ID)
```




So far
- subqueries
- set operators
- IN, NOT IN
- EXISTS, NOT EXISTS
- ALL, ANY

Return the names of all students and staff, as many times as there are people involved*

```
(SELECT name
 FROM Student)
UNION ALL
(SELECT name
 FROM Staff)
```

⇐ no duplicate elimination

SELECT *
FROM R, S;

Returns relation

SELECT count(*)
FROM R, S;

Returns a number

---

COUNT(distinct GPA)

---

MAX, MIN, AVG, SUM

Find total # of applicants

```
select count
       (distinct ID)
from Apply;
```