Worksheet, Q3:

1. grade >80 , Lakemeyer course
not related

2. grade >80, Lakemeyer course
in this course

3. grade >80, Lakemeyer course
taken together

```
SELECT sid, dept||cnum AS course, grade
FROM Took NATURAL JOIN Offering
WHERE
   grade >= 80 AND
   (cnum, dept) IN (
      SELECT cnum, dept
      FROM Took NATURAL JOIN Offering
                NATURAL JOIN Student
      WHERE surname = 'Lakemeyer');
```

all courses taken by Lakemeyer

csc343, csc148

Worksheet, Q4:

R	
a	1
b	2

S	
1	A
2	B
1	α

Suppose we have tables R(a, b) and S(b, c).

1. What does this query do?

Result
a
b

```
SELECT a
FROM R
WHERE b IN (SELECT b FROM S);
```

b
1
2
1

b

```
FROM R
WHERE b IN (SELECT b FROM S);
```

2
1

2. Can we express this query without using IN?

Select a
From R,S
where R.b= S.b;

R join S

a	1	1	A
a	1	1	∝
b	2	2	B

The Operator EXISTS

Select
From
where EXISTS
( );

- Syntax:
  EXISTS («subquery»)
- Semantics:
  Its value is true iff the subquery has at least one tuple.

- Read it as "exists a row in the subquery result"

NOT EXISTS
(reverse)

( )
1 or more — T
empty — F

Example: EXISTS

```
SELECT surname, cgpa
FROM Student
WHERE EXISTS (
   SELECT *
   FROM Took
   WHERE Student.sid = Took.sid and
         grade > 85 );
```

correlated subquery

→ s1

Worksheet, Q5:

```
SELECT instructor
FROM Offering Off1
WHERE NOT EXISTS (
  SELECT *
  FROM Offering
  WHERE
      oid <> Off1.oid AND
      instructor = Off1.instructor );
```

correlated

not equal

instructors such that they taught only 1 offering.

Worksheet, Q6:

H.W

```
SELECT DISTINCT oid
FROM Took
WHERE EXISTS (
   SELECT *
   FROM Took t, Offering o
   WHERE
      t.oid = o.oid AND
      t.oid <> Took.oid AND
      o.dept = 'CSC' AND
      took.sid = t.sid );
```

x *«comparison»* ALL (*«subquery»*)

$\forall$ y $\in$ *«subquery results»* | x *«comparison»* y

x *«comparison»* SOME (*«subquery»*)

$\exists$ y $\in$ *«subquery results»* | x *«comparison»* y

x IN (*«subquery»*)

Same as x = SOME (*«subquery»*)

x NOT IN (*«subquery»*)

Same as x <> ALL (*«subquery»*)

just for convenience

EXISTS (*«subquery»*)

$\exists$ y $\in$ *«subquery results»*

Scope

- If a name might refer to more than one thing, use the most closely nested one.
- If a subquery refers only to names defined inside it, it can be evaluated once and used repeatedly in the outer query.
- If it refers to any name defined outside of itself, it must be evaluated once for each tuple in the outer query.
  These are called correlated subqueries.

Renaming can make scope explicit

```
 SELECT instructor
FROM Offering Off1
WHERE NOT EXISTS (
  SELECT *
  FROM Offering Off2
  WHERE
    Off2.oid <> Off1.oid AND
    Off2.instructor = Off1.instructor );
```

Summary: where subqueries can go

- As a relation in a FROM clause.
- As a value in a WHERE clause.
- With ANY, ALL, IN or EXISTS in a WHERE clause.
- As operands to UNION, INTERSECT or EXCEPT.
- Reference: textbook, section 6.3.