SQL Quiz Time

Poll Question

`Student(sid, name, dept, age)`

What is this query asking for?

```
SELECT S.name, MIN(S.age) AS dept_age
  FROM Student S
 GROUP BY S.dept
```

1. Get the student names and the minimum age in their department
2. Get the student names and the minimum age across all departments
3. Get the name and age of the student with the minimum age in their department
4. This is not a legal SQL query

Poll Question

`Student(sid, name, dept, age)`

What is this query asking for?

```
SELECT S.name, MIN(S.age) AS dept_age
  FROM Student S
 GROUP BY S.dept
```

1. Get the student names and the minimum age in their department
2. Get the student names and the minimum age across all departments
3. Get the name and age of the student with the minimum age in their department
4. This is not a legal SQL query 

Poll Question

`Student(sid, name, dept, age)`

Find the student with the highest age

A

```
SELECT MAX(S.age)
  FROM Student S
```

B

```
SELECT S.*, MAX(S.age)
  FROM Student S
```

C

```
SELECT S.*
  FROM Student S
 ORDER BY S.age
 LIMIT 1
```

Poll Question

`Student(sid, name, dept, age)`

Find the student with the highest age

A

```
SELECT MAX(S.age)
  FROM Student S
```

The highest age only

B

```
SELECT S.*, MAX(S.age)
  FROM Student S
```

Invalid query

C

```
SELECT S.*
  FROM Student S
 ORDER BY S.age
 LIMIT 1
```

The student with the lowest age

Poll Question

Student(sid, name, dept, age)

Enrolled(sid, cid, grade)

Find the CS students who have taken at least one course

A

```
SELECT DISTINCT S.*
  FROM Student S, Enrolled E
 WHERE S.dept = 'CS'
```

B

```
SELECT DISTINCT S.*
  FROM Student S
 WHERE S.dept = 'CS' AND ( SELECT COUNT(E.cid)
                           FROM Enrolled E ) >= 1
```

C

```
SELECT DISTINCT S.*
  FROM Student S NATURAL JOIN Enrolled E
 WHERE S.dept = 'CS'
```

Poll Question

Student(sid, name, dept, age)

Enrolled(sid, cid, grade)

Find the CS students who have taken at least one course

A

```
SELECT DISTINCT S.*
  FROM Student S, Enrolled E
 WHERE S.dept = 'CS'
```

✗ Missing join condition S.sid = E.sid

B

```
SELECT DISTINCT S.*
  FROM Student S
 WHERE S.dept = 'CS' AND ( SELECT COUNT(E.cid)
                           FROM Enrolled E ) >= 1
```

✗ Missing correlation S.sid = E.sid in the subquery

C

```
SELECT DISTINCT S.*
  FROM Student S NATURAL JOIN Enrolled E
 WHERE S.dept = 'CS'
```

✓ Correct

Poll Question

Student(sid, name, dept, age)

Enrolled(sid, cid, grade)

Find the students who have **not** taken INF-11199

A

```
SELECT S.* FROM Student S
 WHERE S.sid NOT IN (SELECT E.* FROM Enrolled E
                     WHERE E.cid = 'INF-11199')
```

B

```
SELECT S.* FROM Student S
 WHERE NOT EXISTS (SELECT E.* FROM Enrolled E
                   WHERE E.cid = 'INF-11199')
```

C

```
SELECT S.* FROM Student S
 WHERE S.sid != ALL (SELECT E.sid FROM Enrolled E
                     WHERE E.cid = 'INF-11199')
```

Poll Question

```
Student(sid, name, dept, age)
Enrolled(sid, cid, grade)
```

Find the students who have **not** taken INF-11199

A

```
SELECT S.* FROM Student S
 WHERE S.sid NOT IN (SELECT E.* FROM Enrolled E
                     WHERE E.cid = 'INF-11199')
```

Invalid query. The inner query must return a bag of E.sid values

B

```
SELECT S.* FROM Student S
 WHERE NOT EXISTS (SELECT E.* FROM Enrolled E
                   WHERE E.cid = 'INF-11199')
```

Missing correlation S.sid = E.sid in the subquery

C

```
SELECT S.* FROM Student S
 WHERE S.sid != ALL (SELECT E.sid FROM Enrolled E
                     WHERE E.cid = 'INF-11199')
```

Correct

Poll Question

`Student(sid, name, dept, age)`

Find all the students with the highest age

A

```
SELECT S.*
  FROM Student S
 WHERE S.age > ALL
   (SELECT S2.age
      FROM Student S2
   )
```

B

```
SELECT S.*
  FROM Student S
 WHERE S.age > ANY
   (SELECT S2.age
      FROM Student S2
   )
```

C

```
SELECT S.*
  FROM Student S
 WHERE NOT EXISTS
  (SELECT S2.age
     FROM Student S2
    WHERE S2.age > S.age
  )
```

Poll Question

`Student(sid, name, dept, age)`

Find all the students with the highest age

A

```
SELECT S.*
  FROM Student S
 WHERE S.age > ALL
   (SELECT S2.age
      FROM Student S2
   )
```

Always empty.
`S.age >= ALL (…)` would be correct

B

```
SELECT S.*
  FROM Student S
 WHERE S.age > ANY
   (SELECT S2.age
      FROM Student S2
   )
```

All except the lowest-age students

C

```
SELECT S.*
  FROM Student S
 WHERE NOT EXISTS
  (SELECT S2.age
     FROM Student S2
    WHERE S2.age > S.age
  )
```

No other student has a higher age

Poll Question

Student(sid, name, dept, age)

A

```
SELECT S.*
  FROM Student S
 WHERE S.age =
   (SELECT MAX(S2.age)
      FROM Student S2)
```

B

```
SELECT S.*
  FROM Student S
 ORDER BY S.age DESC
 LIMIT 1
```

C

```
SELECT S.*
  FROM Student S
 WHERE S.age >= ALL
   (SELECT S2.age
      FROM Student S2)
```

Which queries always return the same students (in any order)?

1) None
2) A and B
3) B and C
4) A and C
5) All

Poll Question

`Student(sid, name, dept, age)`

A

```
SELECT S.*
  FROM Student S
 WHERE S.age =
   (SELECT MAX(S2.age)
      FROM Student S2)
```

B

```
SELECT S.*
  FROM Student S
 ORDER BY S.age DESC
 LIMIT 1
```

C

```
SELECT S.*
  FROM Student S
 WHERE S.age >= ALL
   (SELECT S2.age
      FROM Student S2)
```

Which queries always return the same students (in any order)?

1) None
2) A and B
3) B and C
4) A and C ✓
5) All

A and **C** return *all* students with the highest age.

B returns *one* student with the highest age.

Pool Question

Student(sid, name, dept, age)

Enrolled(sid, cid, grade)

Course(cid, name, year)

```
SELECT S.name FROM Student S
WHERE NOT EXISTS ( SELECT C.cid FROM Course C
                   WHERE NOT EXISTS ( SELECT 42 FROM Enrolled E
                                      WHERE E.cid = C.cid
                                        AND E.sid = S.sid ) )
```

What is this query asking for?

1) Names of students that have taken at least one course
2) Names of students that have taken no course
3) Names of students that have taken all courses
4) Names of students that have not taken all courses
5) Nobody really knows

Pool Question

Student(sid, name, dept, age)

Enrolled(sid, cid, grade)

Course(cid, name, year)

```
SELECT S.name FROM Student S
WHERE NOT EXISTS ( SELECT C.cid FROM Course C
                   WHERE NOT EXISTS ( SELECT 42 FROM Enrolled E
                                      WHERE E.cid = C.cid
                                        AND E.sid = S.sid ) )
```

What is this query asking for?

1) Names of students that have taken at least one course
2) Names of students that have taken no course
3) Names of students that have taken all courses 
4) Names of students that have not taken all courses
5) Nobody really knows

Congratulations!