

SQL Activity – Monday October 17

Listed below are 29 forms of SQL queries.

- **Rlist1** and **Rlist2** are disjoint lists of relations.
- All attribute names are unique across the entire database.
- **cond2** may include references to attributes from relations in **Rlist1**, i.e., correlated references to relations outside of the subquery.

Your goal is to find a minimal set of query types that is sufficient to express queries equivalent to all 29 query forms.

A set of query types is minimal if taking any type out of the set would produce a strict loss in expressive power, i.e., not all 29 query forms would be expressible. Include query type 1 in your minimal set.

If you finish with extra time, repeat the problem removing **distinct** from all 29 query forms.

(1) plain: **select distinct Alist from Rlist where cond**
(**cond** contains no subqueries)

(2) in: **select distinct Alist from Rlist1**
where cond and A1 in (select A2 from Rlist2 where cond2)

(3) not in: **select distinct Alist from Rlist1**
where cond1 and A1 not in (select A2 from Rlist2 where cond2)

(4) exists: **select distinct Alist from Rlist1**
where cond1 and exists (select * from Rlist2 where cond2)

(5) not exists: **select distinct Alist from Rlist1**
where cond1 and not exists (select * from Rlist2 where cond2)

(6) = all: **select distinct Alist from Rlist1**
where cond1 and A1 = all (select A2 from Rlist2 where cond2)

(7) not = all: **select distinct Alist from Rlist1**
where cond1 and not A1 = all (select A2 from Rlist2 where cond2)

(8) <> all: **select distinct Alist from Rlist1**
where cond1 and A1 <> all (select A2 from Rlist2 where cond2)

(9) not <> all: **select distinct Alist from Rlist1**
where cond1 and not A1 <> all (select A2 from Rlist2 where cond2)

(10) < all: **select distinct Alist from Rlist1**
where cond1 and A1 < all (select A2 from Rlist2 where cond2)

- (11) not < all: `select distinct Alist from Rlist1
where cond1 and not A1 < all (select A2 from Rlist2 where cond2)`
- (12) <= all: `select distinct Alist from Rlist1
where cond1 and A1 <= all (select A2 from Rlist2 where cond2)`
- (13) not <= all: `select distinct Alist from Rlist1
where cond1 and not A1 <= all (select A2 from Rlist2 where cond2)`
- (14) > all: `select distinct Alist from Rlist1
where cond1 and A1 > all (select A2 from Rlist2 where cond2)`
- (15) not > all: `select distinct Alist from Rlist1
where cond1 and not A1 > all (select A2 from Rlist2 where cond2)`
- (16) >= all: `select distinct Alist from Rlist1
where cond1 and A1 >= all (select A2 from Rlist2 where cond2)`
- (17) not >= all: `select distinct Alist from Rlist1
where cond1 and not A1 >= all (select A2 from Rlist2 where cond2)`
- (18) = any: `select distinct Alist from Rlist1
where cond1 and A1 = any (select A2 from Rlist2 where cond2)`
- (19) not = any: `select distinct Alist from Rlist1
where cond1 and not A1 = any (select A2 from Rlist2 where cond2)`
- (20) <> any: `select distinct Alist from Rlist1
where cond1 and A1 <> any (select A2 from Rlist2 where cond2)`
- (21) not <> any: `select distinct Alist from Rlist1
where cond1 and not A1 <> any (select A2 from Rlist2 where cond2)`
- (22) < any: `select distinct Alist from Rlist1
where cond1 and A1 < any (select A2 from Rlist2 where cond2)`
- (23) not < any: `select distinct Alist from Rlist1
where cond1 and not A1 < any (select A2 from Rlist2 where cond2)`
- (24) <= any: `select distinct Alist from Rlist1
where cond1 and A1 <= any (select A2 from Rlist2 where cond2)`
- (25) not <= any: `select distinct Alist from Rlist1
where cond1 and not A1 <= any (select A2 from Rlist2 where cond2)`

(26) > any: `select distinct Alist from Rlist1
where cond1 and A1 > any (select A2 from Rlist2 where cond2)`

(27) not > any: `select distinct Alist from Rlist1
where cond1 and not A1 > any (select A2 from Rlist2 where cond2)`

(28) >= any: `select distinct Alist from Rlist1
where cond1 and A1 >= any (select A2 from Rlist2 where cond2)`

(29) not >= any: `select distinct Alist from Rlist1
where cond1 and not A1 >= any (select A2 from Rlist2 where cond2)`