

nulls

Let A and B be integer-valued attributes that may be NULL in some tuples. For each of the following conditions (as may appear in a WHERE clause), describe exactly the set of (A, B) tuples that satisfy the condition, including the case where A and/or B is NULL.

1. $A = 10 \text{ OR } B = 20$
2. $A = 10 \text{ AND } B = 20$
3. $A \leq 10 \text{ OR } A > 10$
4. $A = B$
5. $A \leq B$

Solutions

1. Any tuple with A equal to 10 or B equal to 20
2. Any tuple with A equal to 10 and B equal to 20
3. Any tuple with A non-NULL
4. Any tuple with A and B both non-NULL and equal
5. Any tuple with A and B both non-NULL and A less than or equal to B

Exercise

Write part 3 of the previous exercise in a clearer form

Solution

A IS NOT NULL

NULLS and aggregation

The value NULL is ignored in any aggregation. It does not contribute to a sum, average, or count, nor can it be the minimum or maximum in its column. For example, COUNT(*) is always a count of the number of tuples in a relation, but COUNT(A) is the number of tuples with non-NULL values for attribute A

On the other hand, NULL is treated as an ordinary value in a grouped attribute. For example, SELECT A, AVG(B) FROM R GROUP BY A will produce a tuple with NULL for the value of A and the average value of B for the tuples where A is NULL, if there is at least one tuple in the relation with a component NULL.

Exercise

Consider the following relation R

A	B
1	NULL
NULL	2
3	4
3	6
NULL	NULL

What do the following queries produce?

1. `SELECT SUM(B) FROM R`
2. `SELECT COUNT(*) FROM R`
3. `SELECT COUNT(A),COUNT(B) FROM R`
4. `SELECT A,AVG(B) FROM R GROUP BY A`

Solution

1. 12

2. 5

3. (3, 3)

4. (1, NULL), (3, 5), (NULL, 2)

Exercise

Consider the following relational schema

Emp(eid: integer, cname: string, age: integer, salary: real)

Works(eid: integer, did: integer, pct-time: integer)

Dept(did: integer, budget: real, managerid: integer)

1. Define a table constraint on Emp that will ensure that every employee makes at least 10.000
2. Define a table constraint on Dept that will ensure that all managers have age greater than 30
3. Define an assertion on Dept that will ensure that all managers have age greater than 30. Compare with the previous question, and explain which is better

1. Solution

```
CREATE TABLE Emp (eid INTEGER,  
    ename CHAR(10),  
    age INTEGER,  
    salary REAL,  
    PRIMARY KEY (eid),  
    CHECK ( salary >= 10000 ))
```

2. Solution

```
CREATE TABLE Dept (did INTEGER,  
                    buget REAL,  
                    managerid INTEGER,  
                    PRIMARY KEY (did),  
                    FOREIGN KEY (managerid) REFERENCES Emp  
CHECK ( ( SELECT E.age FROM Emp E  
          WHERE E.eid = managerid ) > 30 )
```

3. Solution

```
CREATE ASSERTION managerAge
CHECK ( 30 < ALL (SELECT E.age
                  FROM Emp E, Dept D
                  WHERE E.eid = D.managerid) )
```

Since the constraint involves two relations, it is better to define it as an assertion