# Quiz

1. Let v = null, x = 10, y = 20, and z = 30. What is the result of each expression below?
   1. x + y = z TRUE
   2. v + x = z UNKNOWN
   3. x + z <> y TRUE
   4. x < y TRUE
   5. v < x AND y < z UNKNOWN
   6. v < x OR y < z TRUE
   7. IS NULL v TRUE
   8. IS NULL (v + x + y + z) TRUE
   9. v + y = y + v UNKNOWN
   10. v + y <> y + v UNKNOWN
   11. v + y IS DISTINCT FROM y + v FALSE
2. Consider the following table creation CREATE TABLE emptiness (); can you insert any value into the table?

[**ANS:** This is where *theory* might differ from *practice*. In theory, there is a value that can be inserted and that is the *empty tuple*. Since this is 0-dimension (i.e., no attribute) the empty tuple corresponds to a point (not even a line). Every point in 0-dimension is equal to any other point in 0-dimension. Hence, *in theory*, there could be exactly one element in the table. In practice, INSERT INTO statement cannot create such empty tuple.]

1. Consider the following relation instance X as well as the table Y created as below:

|  |  |  |  |
| --- | --- | --- | --- |
| **X** | |  | CREATE TABLE Y (  c integer,  d integer,  FOREIGN KEY (c,d) REFERENCES X(a,b)  ); |
| **a** | **b** |  |
| 1 | 2 |  |
| 1 | 3 |  |
| 2 | 1 |  |
| 3 | 2 |  |

Which of the following operations will fail?

1. Inserting (2, 3) into Y FAIL: (2,3) does not exist in X
2. Inserting (1, 2) into Y OK
3. Inserting (1, 3) into Y OK
4. Inserting (3, 1) into Y FAIL: (3,1) does not exist in X
5. Inserting (null, 4) into Y OK: one attribute is NULL
6. Inserting (4, null) into Y OK: one attribute is NULL
7. What is the content of the table Y after all the operations above have been run in sequence, ignoring any failed operations?

|  |  |
| --- | --- |
| **Y** | |
| **c** | **d** |
| 1 | 2 |
| 1 | 3 |
| null | 4 |
| 4 | null |

1. Consider the two tables T1 and T2 below:

|  |  |  |
| --- | --- | --- |
| CREATE TABLE T1 (  a integer PRIMARY KEY,  b integer  ); |  | CREATE TABLE T2 (  a integer REFERENCES T1,  b integer PRIMARY KEY  ); |
|  |
|  |
|  |

Write a sequence of operations such that at the end of execution of your operations, the content of the two tables will be as shown below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***T1*** | |  | ***T2*** | |
| **a** | **b** |  | **a** | **b** |
| 1 | 2 |  | 1 | 2 |

[**ANS:** inserting (1,2) into T1 first; inserting (1,2) into T2 next]

1. Consider Question 5 above, but table T1 is created differently as follows (*assume that it can, since you have not yet taught* ALTER TABLE):

CREATE TABLE T1 (

a integer PRIMARY KEY,

b integer REFERENCES T2

);

(Aaron’s side question: can you create these two tables now?

Not possible as to create either T1 or T2 first, the other referenced table must exist first. What you can do is create T1 with no reference to T2 first, then create T2 as written, then alter the table of T1 for b to a foreign key to b in T2)

Most probably not tested, but FYI

Now note the circular references. Can you still solve Question 5 given this additional constraint?

[**ANS:** inserting (1,NULL) into T1 first; inserting (1,2) into T2 next; updating (1,NULL) to (1,2) in T1 last]