CS2102 Database Systems

Semester 1 2019/2020

Tutorial 02 (*Selected Answers*)

Quiz

1. Let `v = null`, `x = 10`, `y = 20`, and `z = 30`. What is the result of each expression below?

```
a) x + y = z
b) v + x = z
c) x + z <> y
d) x < y
e) v < x AND y < z
f) v < x OR  y < z
g) IS NULL v
h) IS NULL (v + x + y + z)
i) v + y =  y + v
j) v + y <> y + v
k) v + y IS DISTINCT FROM y + v
```

2. Consider the following table creation `CREATE TABLE emptiness ();` can you insert any value into the table?
3. Consider the following relation instance X as well as the table Y created as below:

X	
a	b
1	2
1	3
2	1
3	2

```
CREATE TABLE Y (
  c  integer,
  d  integer,
  FOREIGN KEY (c,d) REFERENCES X(a,b)
);
```

Which of the following operations will fail?

a) Inserting (2, 3) into Y
b) Inserting (1, 2) into Y
c) Inserting (1, 3) into Y
d) Inserting (3, 1) into Y
e) Inserting (null, 4) into Y
f) Inserting (4, null) into Y

4. What is the content of the table Y after all the operations above have been run in sequence, ignoring any failed operations?
5. 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

a	b
1	2

T2

a	b
1	2

6. 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
);
```

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

Tutorial Questions *[Discussion: 7, 8]*

7. Consider a relational database consisting of the following two tables:
   - `Offices (officeID, building, level, roomNumber, area)`
   - `Employees (empID, name, officeID, managerID)`

   The database satisfies the following constraints:
   - `officeID` is the primary key of Offices
   - `{building, level, roomNumber}` is a candidate key of Offices
   - `empID` is the primary key of Employees
   - The name of each employee must be a non-null value
   - Each employee must be assigned to exactly one office identified by `officeID`
   - Each employee may be managed by at most one manager
   - If an employee is managed by someone, the `empID` of the manager is recorded in `managerID`

   Write SQL statements, to create the database schema with appropriate attribute domains and constraints.

   Solution:

```
CREATE TABLE Offices (
  officeID    integer,
  building    varchar(100) NOT NULL, -- These three are
  level       integer      NOT NULL, -- candidate key
  roomNumber  integer      NOT NULL, -- NOT NULL and UNIQUE
  area        varchar(100),
  PRIMARY KEY (officeID),              -- primary key
  UNIQUE (building, level, roomNumber) -- candidate key
);

CREATE TABLE Employees (
  empID      integer,
  name       varchar(100) NOT NULL, -- NOT NULL constraint
  officeID   integer      NOT NULL, -- exactly one officeID
  managerID  integer,               -- at most one manager (may be null)
  PRIMARY KEY (empID),              -- primary key
  FOREIGN KEY (officeID)  REFERENCES Offices (officeID),
  FOREIGN KEY (managerID) REFERENCES Employees (empID)
);
```

8. Consider the following relational database, where the primary key of each table is shown underlined:
   - `Students (sid, mentee, grade)`

   A tuple `(1, 2, 4.5)` indicates that the student with ID 1 is mentoring another student with ID 2 and the student with ID 1 has a grade of 4.5. A tuple `(1, null, 3.0)` indicates the student with ID 1 is not mentoring other students. The database has to satisfy the following additional constraints:
   - `Student.mentee` is a foreign key that references `Student(sid)`
   - A student can only be mentoring another student if his/her grade is greater than or equal to 4.0
   - A given student may not mentor himself/herself

Write SQL statements, to create the database schema with appropriate attribute domains and constraints.

Solution:

```
CREATE TABLE Students (
  sid     integer,
  mentee  integer,
  grade   numeric,
  PRIMARY KEY (sid),
  FOREIGN KEY (mentee) REFERENCES Students (sid),
  CHECK (sid <> mentee), -- do we need IS DISTINCT FROM?
  CHECK (mentee IS NULL OR grade >= 4.0)
);
```

9. Consider the following relational database, with one table, created as follows:

```
CREATE TABLE Cars (
  plateNum  integer      PRIMARY KEY,
  company   varchar(100) NOT NULL,
  type      integer      NOT NULL,
  model     varchar(5)   NOT NULL,
  capacity  integer,
  UNIQUE (company, type),
  UNIQUE (type, model, capacity),
  UNIQUE (plateNum, model)
);
```

Given the table created as above, list down all the candidate keys of the table.

Solution:

`(plateNum)`	from `PRIMARY KEY` constraint
`(company, type)`	from `NOT NULL` and `UNIQUE(company,type)`

Some wrong answers

`(plateNum, model)`	`plateNum` is a subset and it is a superkey
`(type, model, capacity)`	`capacity` can be `NULL`