NAME (PRINT, Last, First):\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_GT ID:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# CS 4400 - Summer 2015 - Q3 Navathe – VERSION A

SCORE: 3\* Correct answers +1

# (1) A B C D E

(2) A B C D E

(3) A B C D E

(4) A B C D E

(5) A B C D E

-----------------------------------------------------------------------------------------------------

(6) A B C D E

(7) A B C D E

(8) A B C D E

(9) A B C D E

(10) A B C D E

-----------------------------------------------------------------------------------------------------

(11) A B C D E

(12) A B C D E

(13) A B C D E

(14) A B C D E

(15) A B C D E

-----------------------------------------------------------------------------------------------------

(16) A B C D E

(17) A B C D E

(18) A B C D E

(19) A B C D E

(20) A B C D E

-----------------------------------------------------------------------------------------------------

(21) A B C D E

(22) A B C D E

(23) A B C D E

(24) A B C D E

(25) A B C D E

-----------------------------------------------------------------------------------------------------

(26) A B C D E

(27) A B C D E

(28) A B C D E

(29) A B C D E

(30) A B C D E

-----------------------------------------------------------------------------------------------------

(31) A B C D E

(32) A B C D E

(33) A B C D E

-----------------------------------------------------------------------------------------------------

NAME (PRINT, Last, First):\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_GT ID:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**CS4400 – Summer 2015**

**QUIZ 3 (Navathe/Omiecinski)**

**PART 1: Functional Dependencies and Relational Design**

**For questions 1 through 13, use the following relational schema, with the primary key underlined,**

PET\_SHELTER\_ADOPT (PetID, Sname, Pname, Type, AdoptID, SCity, Aname, Acity)

Note: each shelter gives its own Pet-id. Hence the key is (PetID,Sname).

Sname: name of a shelter where a Pet is kept

Pname: name of the Pet

Type: refers to type of pet. (e.g., cat, dog etc.)

Adoptid: id of the adopter of the pet

Scity: city where the shelter is located

Aname: name of the adopter of the pet

Acity: city where the adopter lives

**and the set of functional dependencies**

F = { PetID, Sname → Pname

PetID, Sname → Type

PetID, Sname → AdoptID

AdoptID → Acity

AdoptID → Aname

Sname → Scity }

1. What is the meaning of the functional dependency, AdoptID → Aname, with regard to the state of PET\_SHELTER\_ADOPT?

1. Every tuple in PET\_SHELTER\_ADOPT must have a unique AdoptID value.
2. Every tuple in PET\_SHELTER\_ADOPT must have a unique Aname value.
3. If any two tuples in PET\_SHELTER\_ADOPT have the same Aname value then those tuples must have the same AdoptID value.
4. If any two tuples in PET\_SHELTER\_ADOPT have the same AdoptID value then those tuples must have the same Aname value
5. None of the above.

2. Which one of the following functional dependencies can be derived from F using only the augmentation inference rule?

1. PetID, Sname → Aname
2. AdoptID, Sname → Aname, Sname
3. Sname → Sname
4. AdoptID → Acity, Aname

3. Can the functional dependency, *PetID, Sname → Acity* be derived from F using the transitive inference rule?

1. Yes
2. No

4. Is it allowable, according to the functional dependencies F, to have multiple tuples in the PET\_SHELTER\_ADOPT relation having the same pair of values for the <AdoptID, Acity> pair?

1. Yes
2. No

5. To arrive at the functional dependency *PetID, Sname → Aname,* from Fwhich of the Armstrong’s rules must be used:

(a) The reflexive rule

(b) The augmentation rule

(c ) The transitive rule

(d) A combination of rules

6. The highest normal form for the relation PET\_SHELTER\_ADOPT is

1. 1NF
2. 2NF
3. 3NF
4. BCNF

7. If we consider the dependencies that cause violation of 2NF and those that cause violation of 3NF, the dependency that violates 3NF in PET\_SHELTER\_ADOPT is

1. AdoptID → Acity
2. AdoptID → Aname
3. Sname → Scity
4. Both (a) and (b)
5. PetID, Sname → Type

8. Does the following dependency, *AdoptID → Acity* violates 2nd normal form for PET\_SHELTER\_ADOPT?

1. Yes
2. No

9. Suppose we decompose the PET\_SHELTER\_ADOPT schema into 2 schemas:

PSA1 (PetID, Sname, Pname, Type, AdoptID, Scity)

PSA2 (AdoptID, Aname, Acity)

Which of the following statements is true?

1. The highest normal form for PSA1 is 3rd normal form
2. The highest normal form for PSA1 is 2nd normal form
3. The highest normal form for PSA2 is BCNF
4. The highest normal form for PSA2 is 2nd normal form

10. Consider the decomposition of PET\_SHELTER\_ADOPT into two relations: PSA1 and PSA2 as shown in Q9 above. Is this decomposition a non-additive join (also called “lossless join”) decomposition?

1. Yes
2. No

11. Suppose we decompose the PET\_SHELTER\_ADOPT schema into 3 schemas:

PSA1 (PetID, Sname, Pname, Type, AdoptID)

PSA2 (AdoptID, Aname, Acity)

PSA3 (Sname, Scity)

Which of the following statements is true?

1. This is a complete 3rd normal form design (where all relations meet 3NF)
2. This design cannot be called a 3NF design because some relations do not meet 3NF
3. This design does not even meet 2NF requirements completely
4. The design is not acceptable because some functional dependencies are lost

12. What is the closure for the set of attributes, {AdoptID, Sname}, with respect to F for PET\_SHELTER\_ADOPT?

1. {AdoptID, Sname, Acity, Aname, Scity}
2. { AdoptID, Sname Acity, Scity}
3. {Adoptid, Aname, Sname}
4. {AdoptID, Sname}

13. Can there be two tuples in the PET\_SHELTER\_ADOPT relation that have the same values for PetID, Sname, Aname?

1. Yes
2. No

**In problems 14 and 15**, consider the set of attributes {A, B, C, D, E} and the two sets of functional dependencies F={A→B, AB→C, D→AC, D→E} and G={A→BC, D→AE}.

14. What is {D}+ computed with respect to F?

1. {D, A, E, C, B}
2. {D, A, C, E}
3. {D, A, C}
4. {D}

15. From G given above, which of the following is true?

1. A→B is in G+
2. AB→C is in G+
3. A→D is in G+
4. Both a) and b)
5. None of the above

16.For the relational schema R(A,B,C,D,E,G), with the set of FDs:

{ABC → D; B → E; C → G; EG → D; E → A}. This is not minimal. We can reduce the LHS of some of the f.d.s and still be equivalent to the given set.

Which FDs can be reduced by removing an attribute from the LHS?

(a) ABC → D

(b) EG → D

(c) Both (a) and (b)

(d) None of the above. No reduction possible.

17. Which of the following may be considered as general guidelines for relational schema design?

1. Minimize the number of attributes per relation.
2. Increase the number of candidate keys per relation
3. Reduce redundant information in tuples
4. Reduce the number of attributes in the candidate keys

18. Given a relation R(A, B, C, D) with the functional dependencies F={ABC→D, D→B}, what highest normal form is the relation in?

1. 1NF
2. 2NF
3. 3NF
4. BCNF

19. Given a relation R(A, B, C, D) with the functional dependencies F={A→B, A→C, A→D, B→D}, what highest normal form is the relation in?

1. 1NF
2. 2NF
3. 3NF
4. BCNF

20. Given a relation R(A, B, C, D) with functional dependencies F={AB→C, AB→D, CD→AB}, what highest normal form is the relation in?

1. 1NF
2. 2NF
3. 3NF
4. BCNF

**SQL QUESTIONS**

**For questions 21 TO 24, use the following relational schema about projects and employees:**

CREATE TABLE Project(

Pnumber INT PRIMARY KEY,

Loc VARCHAR(10),

Budget DECIMAL (8,2));

CREATE TABLE Employee(

Enumber INT,

Name CHAR (30),

Pnum INT,

PRIMARY KEY (Enumber),

FOREIGN KEY (Pnum) References Project (Pnumber)

On DELETE Set Null On UPDATE CASCADE);

Also, consider the following data:

PROJECT

Pnumber Loc Budget

|  |  |  |
| --- | --- | --- |
| 11 | NYC | 25000.00 |
| 12 | CHI | 40000.00 |
| 13 | MIA | 30000.00 |

EMPLOYEE

Enumber Name Pnum

|  |  |  |
| --- | --- | --- |
| 101 | John Smith | 12 |
| 121 | Bill Hill | 11 |
| 131 | Fred Reed | 11 |

**21. What will be the effect of the following statement?**

INSERT INTO Employee VALUES (141, ‘George Burdell’, 15);

a) The statement will succeed and a new tuple will be inserted into the Employee relation

b) The statement will succeed, and a new tuple will be inserted into the Employee relation, and a tuple will automatically be inserted in Project relation with Pnum =15 and leaving Loc and Budget columns null.

c) The statement produces a referential integrity violation and is hence rejected

d) The statement produces an entity integrity violation and is hence rejected

**22. What will be the effect of the following statement?**

DELETE FROM Employee

WHERE Pnum =11;

1. Two tuples from EMPLOYEE are deleted and one tuple from PROJECT gets deleted
2. Two tuples from EMPLOYEE are deleted
3. Since EMPLOYEE table is not referenced from PROJECT, no additional action is required in PROJECT when employee tuples with Enumbers 121 and 131 are deleted
4. The operation cannot be performed because of foreign key constraint violation
5. Both (b) and (c)

**23. Consider the operation**

UPDATE Project

SET Pnumber =14, Loc = ‘MIA’

WHERE Pnumber =11;

**It will have the following result :**

1. It cannot be processed because of duplication of value in the Loc column.
2. It will only update one tuple in Project with Pnumber=11. No update in EMPLOYEE.
3. It will be rejected since it violates entity integrity
4. It will only update one tuple in Project with Pnumber=11 and two corresponding tuples in Employee will have their Pnum value set to 14.
5. It will update one tuple in Project with Pnumber=11 and raise an error condition;

**24. Consider the operation**

DELETE FROM Project

WHERE Pnumber =12;

**It will result in the following:**

a) It will only delete one tuple in Project with Pnumber=12

b) It will be rejected since its effect on other relations is unclear

c) It will delete one tuple in Project with Pnumber=12 and set the Pnumber to null for ‘John Smith’.

d) It will delete one tuple in Project with Pnumber=12 and delete the corresponding tuple in Employee for ‘John Smith’.

----------

**Consider the following populated relations R and S for questions 25 to 27:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| R |  |  |  | S |  |  |
| A | B | C |  | D | E | F |
| 41 | 21 | 32 |  | 41 | 20 | 4 |
| 42 | 22 | 32 |  | 42 | 22 | 5 |
| 43 | 24 | 32 |  | 43 | 23 | 6 |
| 43 | 21 | 31 |  | 43 | 24 | 6 |
| 45 | 21 | 31 |  |  |  |  |
| 41 | 20 | 31 |  |  |  |  |

**25. What is the result of the SQL query   
SELECT A, C,F**

**FROM R,S**

**WHERE A = D AND B > E;**

(a) A table with columns A, C and F whose 3 rows are (41,32,5), (42,32,6) and (43,31,4)

(b) A table with columns A, C and F whose 2 rows are (41,32,4), and (43,32,6)

(c)A table with columns A, C and F whose 1 row is (41,32,4)

(d ) A table with columns A, C and F whose 3 rows are (41,32,4) , (43,32,6) and

(45,31,null)

**26. What is the result of the SQL query**

**SELECT A,B FROM R**

**MINUS**

**SELECT D,E FROM S;**

(a) A table with two columns whose contents are empty

(b) A table with two columns containing only the row (45,21)

© A table with two columns containing two rows: (45,21) and (41,20)

(d) A table with two columns containing three rows: (41,21), (43,21),and (45,21)

(e) Operation cannot be done due to incompatibility

**27. What is the result of the query:**

**SELECT A,B,C,F**

**FROM R LEFT OUTER JOIN S ON (A=D and B=E)**

**WHERE C=31;**

(a) A four column result with three tuples where two tuples have null values for F

(b) A four column result with four tuples where one tuple has a null value for F

(c ) A four column result with three tuples where one tuple has a null value for F

(d ) A four column result with four tuples where two tuples have a null value for F

**Consider the following database of sailors, boats and reservations for boats in questions 28 to 30.**

**Sid stands for a sailor’s id and Bid stands for a boat’s id**

**Sailors (Sid, Sname, Rating, Age)**

**Boats (Bid, Bname, Color)**

**Reserves (Sid, Bid, Date)**

**28. Which relational algebra expression is equivalent to the following SQL query?**

**SELECT** **DISTINCT** R.sid

**FROM** Boats B, Reserves R

**WHERE** B.bid = R.bid **AND** B.Color = ‘red’

(a) π sid (σ color = ‘red’ (Boats\*Reserves))

(b) π sid (R\* (σ color = ‘red’ Boats))

(c) Both (a) and (b)

(d) π sid (σ color = ‘red’ (Boats∪Reserves))

**29. Consider the following query:**

**SELECT** S.sid, R.bid

**FROM** Sailors S **NATURAL LEFT OUTER JOIN** Reserves R

Evaluate the following statements:

* 1. This query result gives ids of sailors and ids of boats that they have reserved only for those sailors who have some boats reserved
  2. This query result includes all sailors and if they reserved any boats, it includes the boats they reserved
  3. This query result may include some tuples with null values in them

Which of the above are correct?

1. I only
2. II only
3. II and III
4. I and III

**30. Consider the following view definition:**

**CREATE VIEW** Reserved-Red

**SELECT** S.sid, S.rating, S.sname, B.bid

**FROM** Sailors **AS** S, Reserves **AS** R, Boats **AS** B

**WHERE** R.sid = S.sid **AND** R.bid = B.bid **AND** B.color = ‘red’

Now consider the query:

**SELECT X.sname, X.bid**

**FROM Reserved-Red X**

**WHERE X.rating>10**

What does this query do?

a) It lists names of sailors with rating above 10 and ids of all red boats

b) It lists names of red boats that have been reserved by sailors with rating above 10.

c) It lists names of sailors with rating above 10 who reserved red boats, and the ids of boats they reserved

d) The query does not work because Reserved-Red is only a view and not a base table.

---------

**31.** **Given the relational schema consisting of *Faculty (Fnumber, Fname, Homedeptnumber), Course(Cnumber, Cname, Deptnumber) and Teaches(Fnumber, Cnumber),* which SQL query returns the courses *(names and number)* that are taught by more than one faculty.**

1. SELECT C.Cname, C.Cnumber  
   FROM Course C, Teaches T, Teaches S  
   WHERE C.Cnumber = T.Cnumber AND  
    NOT(T.Fnumber = S.Fnumber);
2. SELECT C.Cname, C.Cnumber  
   FROM Course C, Teaches T, Teaches S  
   WHERE C.Cnumber = T.Cnumber AND

T.Cnumber =S.Cnumber AND   
 NOT(T.Fnumber = S.Fnumber);

1. SELECT C.Cname, C.Cnumber  
   FROM Course C  
   WHERE NOT EXISTS

(SELECT \*

FROM Teaches T, Teaches S

WHERE T.Cnumber =C.Cnumber AND   
 NOT(T.Fnumber = S.Fnumber);

1. None of the above.

------------

CONSIDER the following relation schema for the next two questions:

EMPLOYEE (Ssn, Lname, Fname, Salary, City, Age, Deptnumber, Superssn)

32. What is returned by the following query?

SELECT DEPTNUMBER, AVG (SALARY)  
FROM EMPLOYEE E  
WHERE DEPTNUMBER IN

SELECT A.DEPTNUMBER  
 FROM EMPLOYEE A

WHERE A.SALARY > ( SELECT 2\* MIN (SALARY)

FROM EMPLOYEE)

GROUP BY DEPTNUMBER;

* + - * 1. the department number and average salary of employees in all departments where all employees earn more than twice the minimum salary
        2. the department number and average salary for the one department that has the employee with more than two times the minimum salary
        3. query is formulated incorrectly
        4. the department number and average salary for one or more departments that have some (one or more) employees whose salary is greater than twice the minimum salary for all employees

33. What is returned by the following query?

SELECT DEPTNUMBER, MAX (SALARY)  
FROM EMPLOYEE  
WHERE DEPTNUMBER IN (SELECT DEPTNUMBER  
 FROM EMPLOYEE

GROUP BY DEPTNUMBER

HAVING COUNT(\*) >50 )

AND AGE <30

GROUP BY DEPTNUMBER;

1. the maximum salary of all employees under age 30 and over age 50
2. the department number and maximum salary of employees under age 30 from departments that have over 50 employees
3. the department number and maximum salary of employees under age 30 from departments where the under 30 employee population exceeds 50
4. the department number and maximum salary of all employees from departments that have over 50 employees under age 30