

OVERALL ANALYSIS

Solution Report

[All](#)
[Correct Answers](#)
[Wrong Answers](#)
[Not Attempted Questions](#)

Q.1)

Max Marks: 1

Suppose we have a relation R(U, V, W, X, Y) and functional dependencies U → XY, X → V, and Y → W. If we project R onto schema UVW, what is true about the key(s) for UVW?

 A Only XY is a key

 B Each one of U, V and W are keys.

 C Only UVW is a key

 D Only U is a key

Correct Option

Solution: (D)

Solution: (iv)

Note that, U → UVWX. Thus when we project the FD's on UVW, U is certainly a key. However, there cannot be any other keys because neither V nor W have anything else in their closures as they are not on the left side of any FD's. Thus U is the only key.

Q.2)

Max Marks: 1

Consider the following table with attribute x is primary key and y is the foreign key referencing x with on delete cascade.

X	Y
1	21
4	9
5	12
6	5
12	8
10	16
2	6

What is the number of tuples deleted additionally in order to preserve referential integrity, when the tuple (12,8) is deleted _____



Correct Answer

Solution: (3)

Solution: 3 tuples

On deleting the tuple (12,8) all the rows referring to value of X = 12 will get deleted. Since there is only one tuple referring to X = 12, therefore first tuple which is deleted is (5, 12).

Now all the tuples referring to X = 5 will be deleted. Since there is only one tuple which is referring to X = 5, therefore (6, 5) is the second tuple which got deleted.

Now due to deletion of (6, 5), all those tuple which refer to X = 6 will also get deleted. Hence, third tuple which got deleted is (2, 6).

Since there is no more tuple which refer to the X = 2. Hence, total number of tuples got deleted in order to preserve the referential integrity is 3.

Q.3)

Max Marks: 1

Given the following structure of STUDENT relation:

```
Student_id INTEGER PRIMARY KEY,
```

```
First_name VARCHAR (25),
```

```
Last_name VARCHAR (25);
```

On assuming that all the following four statements are executed independently, which of the following statements is not a valid insertion of a row in STUDENT relation?



```
INSERT INTO STUDENT VALUES ('25', 'Pallavi', NULL);
```



```
INSERT INTO STUDENT(Student_id, First_name, Last_name) VALUES (25, 'Pallavi', '');
```



```
INSERT INTO STUDENT (First_name, Student_id, Last_name) VALUES (25, 'Pallavi', 'Gupta')
```

Correct Option

Solution: (C)

Solution: (iii)

I. It is valid insertion. We can insert a row without mentioning the attribute name along with the relation until and unless we input the values in the given sequence of attribute names. And since Last_name is not the primary key, therefore, it could be NULL.

II. It is also a valid insertion of tuple. We can set any attribute empty until and unless it is not the prime attribute.

III. It is not a valid insertion as the sequence of attribute name mentioned is different from the sequence of values inserted. Therefore, there will be an error due to type mismatch. Hence it is not valid.

IV. It is a valid insertion. We can insert values by mentioning a single attribute name. The SQL server will consider the other attribute values to be NULL implicitly.
Hence, the correct option is (iii).

D INSERT INTO STUDENT(Student_id) VALUES (25);

Q.4)

Which of the following will return the result as $R \cap P$ where R and P are given as $R(a, b, c)$, $P(a, b, c)$ respectively.

Max Marks: 1

A SELECT * FROM R WHERE EXISTS (SELECT * FROM P WHERE R.a = P.a AND R.b = P.b AND R.c = P.c)

B SELECT * FROM R WHERE (a, b, c) NOT IN(SELECT * FROM R WHERE (a, b, c) NOT IN(SELECT * FROM P))

C Both (i) and (ii)

Correct Option

Solution: (c)

Solution: (iii)

Both (i) and (ii) will result in $R \cap P$. In (i) it is easily visible that it is intersection and in (ii) nested query 2 will return all p tuples (SELECT * FROM P) and nested query 1 will return tuples of r which is not similar to tuples return by nested query 2 (SELECT * FROM R WHERE(a, b, c) NOT IN (nested query 2)) and finally main query will return all those tuples which are common in both the relations.

D Neither (i) nor (ii)

Q.5)

Consider the following instance R.

Max Marks: 1

A	1	2	3	4
B	NULL	1	2	2

What is the number of tuples returned by the following SQL query?

SELECT * from R as R1

WHERE NOT EXISTS (SELECT* from R
WHERE B = R1.A)

Correct Answer

Solution: (2)

Solution: 2

$R=\{(1,NULL),(2,1),(3,2),(4,2)\}$

Now, a tuple is output if where condition is TRUE. Here the where condition is NOT EXISTS which returns true if and only if we give empty set to it. So, the condition for output here is that the inner query should not return any tuple.

Based on query, the inner query returns a tuple if the 'A' value being considered exist in any of the tuple as a 'B' value. This happens for A values, 1 and 2 and hence NOT EXISTS is FALSE for these two as the inner query returns $\{(2,1)\}$ and $\{(3,2),(4,2)\}$ respectively. For the tuples (3,2) and (4,2), the inner query returns {} and hence they are output. So, the output is $\{(3,2),(4,2)\}$ and answer is 2.

Q.6)

Consider the following relation customer:

Max Marks: 1

Relation customer:

Cust_id	Name	City
1	Tarun	Kolkata
2	Shivam	Lucknow
3	Rahul	Kolkata
4	Venkatesh	NULL
5	Ravi	Kolkata
6	Kumar	NULL

7	Reddy	NULL
---	-------	------

Given the SQL query as: "SELECT DISTINCT(city) FROM customer;". What is the number of rows returned by the given following SQL query is _____

Correct Answer

Solution: (3)

Solution: 3 tuples

In SQL, the DISTINCT clause doesn't ignore NULL values. So when using the DISTINCT clause in any SQL statement, the result set will include NULL as a distinct value. But the DISTINCT operator treats NULL values to be duplicate of each other, therefore if the SELECT statement returns NULL values, only one NULL value is returned when we use the DISTINCT operator. Therefore, the query will return 3 tuples, as it have three distinct values (lucknow, Kolkata, (null)). Hence, the answer is 3.

Q.7)

Consider the following SQL queries given below:

Max Marks: 1

I.

SELECT name FROM people WHERE age > 25;

II.

SELECT name FROM people WHERE height > 180;

III.

SELECT name FROM people WHERE (age > 25) OR (height > 180);

If the number of rows returned by the SQL query I and II are 10 and 7 respectively. Then what is the possible number of tuples returned by the query III from the given options?

- A 3
- B 7
- C 21
- D 15

Correct Option

Solution: (D)

15 is one of the possible values because,

Case 1: Minimum number of tuples returned by SELECT name FROM people WHERE (Age > 25) or (Height > 180); will return 10 rows. Since there is OR between the condition. (Since people having age > 25 are 10 in number).

Case 2: Maximum number of tuples will be 17 as

There might be 7 people whose height > 180 but their age < 25 (select based on height) and 10 people whose age > 25 but their height < 180 (select based on age). So total rows returned by the query III will be $10 + 7 = 17$.

Hence, the number of output tuples will range from 10 – 17 and therefore, the valid option is (iv) with value 15.

Q.8)

Consider the relation R (ABCDE) with the following FD's

Max Marks: 1

CD \rightarrow E

E \rightarrow A

B \rightarrow D

A \rightarrow BC

If the relation R is decomposed into two relation R1(ABC) and R2 (ADE), then which of the following statements is correct about its decomposition.

- A Lossless join and dependency preserving decomposition
- B Lossy join and dependency preserving decomposition
- C Lossless join and not dependency preserving decomposition
- D Lossy join and not dependency preserving decomposition.

Correct Option

Solution: (C)

Solution: (iii)

In both the relations R1 and R2, A is the common attribute and is the primary key of relation R1 (A \rightarrow BC). Hence, it is a lossless join. But the dependencies are not preserved.

D Lossy join and not dependency preserving decomposition.

Q.9)

Max Marks: 1

Consider the following statements:

- I. SQL does not distinguish between the different meaning of NULL
- II. SQL allows a relation to have two or more tuples that are identical in all this attributes values.

Which of the above statement is true?

 A Only I B Only II C Both I and II

Correct Option

Solution: (c)

Solution: (iii)

- I. In SQL, duplicate rows are allowed, if no primary key is set.
- II. All the NULL values are considered as NULL and we cannot differentiate between the different NULL values obtained

Hence, both the statements are true.

 D Neither I nor II

Q.10)

Max Marks: 1

A relational table Employee(ENO, EName, Dept) has 88 tuples. What will be the result of the following SQL statement?

SELECT COUNT (ENO) FROM Employee WHERE ENO NOT IN (NULL);

 A 0

Correct Option

Solution: (A)

Solution: (i)

Here, We are asked to COUNT all ENo's not in NULL – this predicate evaluates to unknown because comparison with NULL can never result in either True or False, it always result in a third logical result, Unknown. Hence, no rows will be printed.

 B 44 C 88 D 87

Q.11)

Max Marks: 2

Which of the following statement is correct?

- I. In natural join, say $R \bowtie S$ if the two tuples, $t_r \in R$ and $t_s \in S$, both have a NULL value in a common attribute, then the tuples do not match.
- II. Using UPDATE in SQL we can change the primary key data type.

 A Only I

Correct Option

Solution: (A)

Solution: (i)

Only statement I is correct.

I. Joins in RA can be expresses as a Cartesian product followed by selection. Thus, definition of how selection handles NULL's also defines how join operation handles NULL's. As we know selection operation evaluates predicate P on each tuple t in relation E. If the predicate returns the value true, t is added to the result. Otherwise, if the predicate returns unknown (NULL) or false, t is not added to the result. Hence, joins will also not consider the unknown(NULL) values and therefore a tuple t having NULL value in a common attribute will not be added in the result set.

II. In order to change the data type of primary key ALTER command is used. If the table already exists and we wish to add a primary key or change its data type later, we can use the ALTER TABLE statement to do that.

 B Only II C Both I and II D Neither I nor II

Q.12)

Max Marks: 2

Let relation $S(S, T, U, V, W, X, Y, Z)$ satisfy the following FD's: $S \rightarrow T$ $UZ \rightarrow S$ $T \rightarrow W$ $TV \rightarrow U$ $WY \rightarrow Z$ $VW \rightarrow X$

Which of the following FD is also guaranteed satisfied in S?

A TXY → SW

B UWY → ST

Correct Option

Solution: (B)

Solution: (ii)

To check whether a functional dependency is satisfied by S we need to take its left-hand side, calculate the closure with respect to the original set of the functional dependencies and then check whether the right-hand side is contained within the closure. So in the question only UWY → ST is satisfied because closure of UWY={U, W, Y, S, T, Z} and ST is in the closure.

C UYZ → TX

D SVW → UZ

Q.13)

Max Marks: 2

For a relation R (PQRST) and the FD set F = {PQ → R, R → S, S → T, T → P}. Which of the following is true for the relation R?

- I. PQ → R is preserved after the decomposition.
- II. FD set G = {T → P, S → T, R → S} covers FD set F.

A Only I

Correct Option

Solution: (A)

Solution: (ii)

I. When we decompose the relation R (PQRST), the candidate key of R will be

$$PQ^+ = P, Q, R, S, T$$

$$QR^+ = P, Q, R, S, T$$

$$QS^+ = P, Q, R, S, T$$

$$TQ^+ = P, Q, R, S, T$$

Hence, PQ, QR, QS, TQ are the candidate keys of R. The decomposition of R will be:

$$R1(PQR) = PQ \rightarrow R$$

$$R2(RST) = S \rightarrow T \text{ and } R \rightarrow S$$

Hence, PQ → R is preserved.

Therefore, it is correct

II. Since the decomposition is given as above therefore final FD set will be [PQ → R, S → T, R → S]. Since it is not same as G. Therefore, this statement is not correct.

B Only II

C Both I and II

D Neither I nor II

Q.14)

Max Marks: 2

Consider the following relation:

Relation Employee:

Emp_id	Dept_id	Salary
1	1	1000
2	1	1100
3	2	950
4	3	700
5	2	1200
6	4	800

Relation Department:

id	Dept_name
1	A
2	B
3	C
4	D

What is the number of tuples returned by each query?

Query 1: Select D.Dept_name From Department D

Where (Select COUNT(*) From Employee E
Where E.Dept_id = D.id and E.salary > 1000) >=2;

Query 2: Select MIN(D.Dept_name) as Dname

From Department D inner join Employee E on D.id = E. Dept_id Where
E.salary > 1000 Group by (E.Emp_id) Having Count(E.Emp_id)
>2;

A

2, 2

B

0, 0

Correct Option

Solution: (B)

Solution: (ii)

Query 1: The inner query will return the employees having salary > 1000 for number of employees greater than 2. Since none of the department is having employees ≥ 2 with salary greater than 1000. Therefore, the output of the relation will be 0.

Query 2: Here the final output is not a relation set. MIN(D.Dept_name) will return a value as an output. Now if we execute the query, we can see that it is grouping based on Emp_id which is unique for each employee hence the count of Emp_id will be 1 for each Emp_id. Therefore no tuple will have value > 2 . Hence output is 0

Hence, the correct option is (ii).

C

3, 2

D

3, 0

close