Database Management System – cs422 DE

Assignment 3 – Week 3 & 4

This assignment is based on lecture 3 & 4 (chapter 6 & 7).

- Submit your *own work* on time. No credit will be given if the assignment is submitted after the due date.
- Note that the completed assignment should be submitted in .doc, .docx, .rtf or .pdf format only.
- In MCQs, if you think that your answer needs explanation to get credit then please write it down.
- You are encouraged to discuss these questions in the Sakai forum.

---

1) The database schema is written in

(A) HLL (B) DML (C) DDL (D) DCL

ANS: C

2) The language used in application programs to request data from the DBMS is referred to as

(A) DML (B) DDL (C) VDL (D) SDL

ANS: A

3) Count function in SQL returns the number of

(A) values (B) distinct values (C) groups (D) columns

ANS: A

4) 'AS' clause is used in SQL for

(A) Selection (B) Rename (C) Join (D) Projection

ANS: B

5) Which is not a DDL statement?

(A) Create (B) Alter (C) Delete (D) Drop

ANS: C

6) The statement in SQL which allows to change the definition of a table is

(A) Alter (B) Update (C) Create (D) Select

ANS: A

7) What restrictions apply to the use of the aggregate functions within the SELECT statement? How do nulls affect the aggregate functions?

ANS: Within the select statement for using aggregate function can be used only in the SELECT list and in the HAVING clause. Apart from COUNT(*), each function eliminates nulls first.

8) List the order in which the WHERE, GROUP BY, and HAVING clauses are executed by the database in the following SQL statement.

```
SELECT section_id, COUNT(*), final_grade
FROM enrollment
WHERE TRUNC(enroll_date) > TO_DATE('2/16/2003', 'MM/DD/YYYY')
GROUP BY section_id, final_grade HAVING COUNT(*) > 5
```

ANS: 1. WHERE 2. GROUP BY 3. HAVING

9) Explain how the GROUP BY clause works. What is the difference between WHERE and HAVING clauses?

ANS: According to standard GROUP BY requires the SELECT clause. All columns in the SELECT list must appear in the GROUP BY clause unless the name only used in an aggregate function.

10) Can the ANY and ALL operators be used on the DATE data type? Write a simple query to prove your answer.

ANS: **ANY and ALL operators can be used on DATE data type.**

a. Below statement only select those projects if any one users joining date is less that project start date.

```
SELECT id, name FROM projects WHERE startsOn > any( select joiningAt from users);
```

b. Below query only selects those projects, if all users joining date is less that project start date.

```
SELECT id, name FROM projects WHERE startsOn > all( select joiningAt from users);
```


c. The following SQL lists staffs who work in branch at '163 Main St'.

```
SELECT staffNo, fName, lName, position
FROM Staff
WHERE branchNo =
      (SELECT branchNo
      FROM Branch
      WHERE street = '163 Main St');
```

Will there be any problem with this query if there is more than one branch at '163 Main St'? If yes, then explain the problem and right down the correct query.

ANS: yes, there is a problem in this query. If there is more than one branch at given address, then it selects list and SQL generates subquery returns more than 1 row error. To solve this we have to use ANY or ALL operators.

```
SELECT staffNo, fName, lName, position
FROM Staff
WHERE branchNo =
      all(SELECT branchNo
      FROM Branch
      WHERE street = '163 Main St');
```

d. **What is Referential integrity constraint?**

ANS: During FOREIGH KEY implementation we have to implement relation between parent and child table that relation called referential integrity constraint and the value stored on child table should be lies on parent table.

e. What is the difference between primary key and unique key?

ANS: PARIMARY KEY is specified column of every table of database by default this is NOT NULL is assumed. Any operation like CREATE, UPDATE, DELETE will be done with the basis of this key. UNIQUE KEY is also the column of DB table which makes the unique rows.

f. Solve the question 7.10 from the course text book (5th edition).

ANS:
```
CREATE table hotel(
        id int PRIMARY_KEY,
        name varchar,
        address varchar,
        rating varchar
)
```

g. Solve the question 7.12 from the course text book (5th edition).

ANS:
```
CREATE table booking_history_tbl(
        bookingId INT PRIMARY KEY,
        roomId INT,
        guestId INT,
        price DECIMAL(6,2),
        type VARCHAR(10)
        dateFrom DATE,
        dateTo DATE
)

INSERT INTO booking_history_tbl
SELECT * FROM Booking
WHERE dateTo < '01/01/2007';

-- Delete the old bookings from the Booking table
DELETE FROM Booking
WHERE dateTo < '01/01/2007';
```