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

MUM-DBMS

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:

- Aggregate Function Usage: Aggregate functions such as SUM, AVG, MIN, MAX, COUNT, etc., can be applied within the SELECT statement. However, they necessitate data grouping using the GROUP

BY clause. Additionally, filtering can be applied to the aggregated results using the HAVING clause.

- Handling of Null Values: With the exception of COUNT(*), each aggregate function disregards null values and operates solely on the non-null values remaining after elimination.

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: The order is: WHERE, GROUP BY, HAVING.

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

ANS:

The GROUP BY clause: Divides the result set into groups based on specified column(s); Aggregates data within each group using aggregate functions like COUNT, SUM, AVG, etc. Used to perform calculations on groups of rows rather than individual rows; Operates after the WHERE clause and before the HAVING clause.

WHERE	HAVING
Filters individual rows from the original data based on specified conditions	Filters groups of rows after they have been grouped using the GROUP BY clause
Applies conditions to individual rows before any grouping or aggregation takes place	Applies conditions to aggregated data, specifically to groups of rows
Cannot be used with aggregated results because it filters rows before they are grouped	Operates after the GROUP BY clause and can only be used with aggregated results
	Used to further filter the grouped results based on specified conditions

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

ANS:

No, the ANY and ALL operators cannot be directly used on the DATE data type in SQL. These operators are typically used in comparison operations with numerical or string values, not with dates.

```
SELECT *
FROM Employee
WHERE
    depart_name = “Sale” AND
    hire_date > ALL (SELECT hire_date FROM Employee WHERE depart_name = “Development”);
```

11) 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 will be a problem if there is more than one branch at '163 Main St' because the subquery will return multiple branch numbers, causing the main query to return an error.

To correct this, use the IN operator instead of "=" in the main query's WHERE clause. This allows for multiple values to be compared.

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

MUM-DBMS

12) What is Referential integrity constraint?

ANS: The referential integrity constraint is a rule that ensures the validity of relationships between tables within a database. This constraint is implemented through foreign keys, which reference the primary keys of other tables.

13) What is the difference between primary key and unique key?

ANS:

Primary key:

- uniquely identifies each record in a table.
- must be unique and not null.
- there can be only one primary key in a table.

Unique key:

- ensures that all values in a column or a set of columns are unique.
- allows null values (except in MySQL where it behaves like a primary key).
- multiple unique keys can exist in a table.

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

ANS:

```
CREATE TABLE hotel (
   hotelNo int NOT NULL,
   city varchar(128),
   hotelName varchar(128),
   PRIMARY KEY (hotelNo)
);
```

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

ANS:

```
CREATE TABLE BookingArchive (
   hotelNo INT NOT NULL,
   guestNo INT NOT NULL,
   roomNo VARCHAR(16),
   dateFrom DATE NOT NULL,
   dateTo DATE,
   PRIMARY KEY (hotelNo, guestNo, dateFrom),
   CONSTRAINT fk_guestNo FOREIGN KEY (guestNo) REFERENCES Guest (guestNo),
   CONSTRAINT fk_hotelNo FOREIGN KEY (hotelNo) REFERENCES Hotel (hotelNo)
);

INSERT INTO BookingArchive
SELECT *
FROM Booking
WHERE dateFrom < '2013-01-01';

DELETE FROM Booking
WHERE dateFrom < '2013-01-01';
```

MUM-DBMS