Assignment Day 5–SQL: Comprehensive practice

1. What is an object in SQL?

An object in SQL is any defined object which is used to store or reference data. SQL Server has many database objects such as table, view, stored procedure, function, constraints, rule, Synonym, triggers.

2. What is Index? What are the advantages and disadvantages of using Indexes?

An index is an on-disk structure associated with a table or view which is used to speed up the data retrieval. An index contains keys built from one or more columns in the table or view. These keys are stored in a structure (B-tree) that enables SQL Server to find the row or rows associated with the key values quickly and efficiently.

The advantages include, records from the table can be fetched quickly, help the sorting and grouping records.

The disadvantages are additional disk space is required for nonclustered index, since it is stored separately from tables. And the DML statement such as Insert, Update and Delete modifies data in a table as well as the corresponding indexes, then if too many indexes are updating can actually hurt the performance of data modifications.

3. What are the types of Indexes?

There are two types of index, clustered and nonclustered. Clustered indexes sort and store the data rows in the table or view based on their key values. But there can be only one clustered index per table. Nonclustered indexes have a structure separate from the data rows. A nonclustered index contains the nonclustered index key values and each key value entry has a pointer to the data row that contains the key value.

4. Does SQL Server automatically create indexes when a table is created? If yes, under which constraints?

Indexes can be automatically created, when PRIMARY KEY and UNIQUE constraints are defined on table columns.

5. Can a table have multiple clustered index? Why?

NO. There can be only one clustered index per table. Because clustered index is the data rows themselves can be stored in only one order.

6. Can an index be created on multiple columns? Is yes, is the order of columns matter?

YES. An index be created on multiple columns. And the order of columns is really matter, depends on the search condition, it will have a huge difference on the performance. One type of query may need a certain column order for the index. If we have several types of queries, it might need several indexes to help, with columns in different orders.

7. Can indexes be created on views?

YES. An index is an on-disk structure associated with a table or view which is used to speed up the data retrieval.

8. What is normalization? What are the steps (normal forms) to achieve normalization?

Database Normalization is a process of organizing data to minimize redundancy and ensure the data consistency.

The first normal form is data in each column should be atomic, no multiples values separated by comma. The table does not contain any repeating column group. Identify each record using primary key.

The second normal form is the table must meet all the conditions of normal form 1. Move redundant data to separate table. Create relationships between these tables using foreign keys.

The third normal form is the table must meet all the conditions of normal form 1 and 2. Does not contain columns that are not fully dependent on primary key.

9. What is denormalization and under which scenarios can it be preferable?

Denormalization is a database optimization technique by adding redundant data to one or more tables. This can help us avoid costly joins in a relational database when the tables are large.

10. How do you achieve Data Integrity in SQL Server?

There are some rules used to achieve data integrity, those are include: limit access to data and change permissions to constrain modifications to data by unapproved parties. Focus on data validation to ensure the accuracy of data when collected or integrated. Maintain a regular backup of data. Use logs to monitor when data is entered, altered, or erased. Conduct systematic internal audits to ensure that information is up to date.

11. What are the different kinds of constraint do SQL Server have?

SQL Server contains 6 types of constraints: Not Null Constraint, Check Constraint, Default Constraint, Unique Constraint, Primary Constraint, Foreign Constraint

12. What is the difference between Primary Key and Unique Key?

Unique Key ensures that all values in a column are different.

And the Primary Key is a combination of Not Null and Unique. It can uniquely identifies each row in a table.

13. What is foreign key?

A Foreign Key is a field or fields in one table, which refers to the Primary Key in another table.

13. Can a table have multiple foreign keys?

YES. A particular table can have more than one foreign key on it. Since, a foreign key allows us to link up or define relationships between tables.

14. Does a foreign key have to be unique? Can it be null?

A foreign key stores the primary key in another database table. So it cannot be null and it has to be unique.

15. Can we create indexes on Table Variables or Temporary Tables?

YES. We can create indexes on table variables or temporary tables by adding the Primary Key or Unique constraints.

16. What is Transaction? What types of transaction levels are there in SQL Server?

Transaction is a set of SQL statements executed against a database. Usually it include at least one statement. And it changes the database from one consistent state to another. There are four level of isolation for transactions: Read Uncommitted (Lowest level), Read Committed, Repeatable Read, Serializable (Highest Level), and Snapshot Isolation.

# Write queries for following scenarios

1. Write an sql statement that will display the name of each customer and the sum of order totals placed by that customer during the year 2002

Create table customer(cust\_id int, iname varchar (50)) create table order(order\_id int,cust\_id int,amount money,order\_date smalldatetime)

SELECT c.iname, SUM(o.amount)

FROM customer c

JOIN order o

ON c.cust\_id = o.cust\_id AND o.order\_date BETWEEN 2002.01.01 AND 2002.12.31

2. The following table is used to store information about company’s personnel:

Create table person (id int, firstname varchar(100), lastname varchar(100)) write a query that returns all employees whose last names start with “A”.

SELECT p.id, p.firstname, p.lastname

FROM person p

WHERE p.lastname LIKE “A%”

3. The information about company’s personnel is stored in the following table:

Create table person(person\_id int primary key, manager\_id int null, name varchar(100)not null) The filed managed\_id contains the person\_id of the employee’s manager.

Please write a query that would return the names of all top managers(an employee who does not have a manger, and the number of people that report directly to this manager.

SELECT p.name, COUNT(m.person\_id)

FROM person p

JOIN person m

ON p.person\_id = m.manager\_id AND p.manager\_id IS NULL

GROUP BY p.name

4. List all events that can cause a trigger to be executed.

Triggers are automatically fired on an event. As the DML Statements like Insert, Delete or Update.

5. Generate a destination schema in Normal Form. Include all necessary fact, join, and dictionary tables, and all Primary and Foreign Key relationships. The following assumptions can be made:

a. Each Company can have one or more Divisions.

b. Each record in the Company table represents a unique combination

c. Physical locations are associated with Divisions.

d. Some Company Divisions are collocated at the same physical of Company Name and Division Name.

e. Contacts can be associated with one or more divisions and the address, but are differentiated by suite/mail drop records. status of each association should be separately maintained and audited

