~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

questions

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

sub heading

~~~~~~~~~~~~

~~~ sub heading ~~~

answer

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

HOW TO SEND MAIL FROM SQL SERVER?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

USE [YourDB]

EXEC msdb.dbo.sp\_send\_dbmail

@recipients = 'mathew@xyz.com; saadhya@xyz.com;anirudh@pqr.com’

@body = ' A warm wish for your future endeavor',

@subject = 'This mail was sent using Database Mail' ;

GO

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

EXPLAIN INLINE VARIABLE ASSIGNMENT IN SQL SERVER 2008 WITH AN EXAMPLE.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Inline variable assignment in SQL Server 2008 enables to – declare, and assign the variables in a ‘single line’.

The traditional Ex:

DECLARE @myVar int

SET @myVar = 5

need to declare the variable and assigning it to the variable to split into 2 lines and use 2 statements - DECLARE and SET.

In SQL Server 2008, as the name suggests ‘inline’, both declaration and assignment can be given in a single line:

DECLARE @myVar int = 5

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

To explain SQL server 2000 architecture & authentication?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

SQL server architecture

~~~~~~~~~~~~~~~~~~~~~--

Database having physical and logical databases, in physical databases contains file group and Transaction log in file group consists of data file - 36,767 TB data file consists of table - a single table can have 1024 Columns.

Table consists of extents - 8 contiguous pages 1 extent - 64KB

Extents consists of pages - 8KB

Pages consists of rows - 8060 bytes (max row size)

Transaction log is user defined file group by default its primary file group.

Logical - database objects like Views, sp, functions, triggers, indexes and roles etc.

~~~~~~ Authentication modes ~~~~~~

There are two types of Authentication modes.

1. Windows Authentication Mode.

2. SQL server Authentication (mixed mode)

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is procedure in SQL?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Procedure is a collection of simple DML statements used to perform certain meaningful operation stored in the database and used whenever and wherever required.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are the major improvements in SQL Server 2005 for XML supports?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

With SQL Server 2005 there are six major improvements for XML support:

• XML DML (XML Data Modification Language)

• Transact-SQL enhancements (FOR XML and OPENXML)

• HTTP SOAP Access

• New xml data type

• Indexes on xml type columns

• XQuery support

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~ StartRegion -- BASIC Interview Question ~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is normalization? Explain different levels of normalization?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Check out the article Q100139 from Microsoft knowledge base and of course, there’s much more information available in the net. It’ll be a good idea to get a hold of any RDBMS fundamentals text book, especially the one by C. J. Date. Most of the times, it will be okay if you can explain till third normal form.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is denormalization and when would you go for it?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o As the name indicates, denormalization is the reverse process of normalization. It’s the controlled introduction of redundancy in to the database design. It helps improve the query performance as the number of joins could be reduced.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• How do you implement one-to-one, one-to-many and many-to-many relationships while designing tables?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o One-to-One relationship can be implemented as a single table and rarely as two tables with primary and foreign key relationships. One-to-Many relationships are implemented by splitting the data into two tables with primary key and foreign key relationships. Many-to-Many relationships are implemented using a junction table with the keys from both the tables forming the composite primary key of the junction table. It will be a good idea to read up a database designing fundamentals text book.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What’s the difference between a primary key and a unique key?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Both primary key and unique enforce uniqueness of the column on which they are defined. But by default primary key creates a clustered index on the column, where are unique creates a nonclustered index by default. Another major difference is that, primary key doesn’t allow NULLs, but unique key allows one NULL only.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are user defined datatypes and when you should go for them?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o User defined datatypes let you extend the base SQL Server datatypes by providing a descriptive name, and format to the database. Take for example, in your database, there is a column called Flight\_Num which appears in many tables. In all these tables it should be varchar(8). In this case you could create a user defined datatype called Flight\_num\_type of varchar(8) and use it across all your tables. See sp\_addtype, sp\_droptype in books online.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is bit datatype and what’s the information that can be stored inside a bit column?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Bit datatype is used to store boolean information like 1 or 0 (true or false). Untill SQL Server 6.5 bit datatype could hold either a 1 or 0 and there was no support for NULL. But from SQL Server 7.0 onwards, bit datatype can represent a third state, which is NULL.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• Define candidate key, alternate key, composite key.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o A candidate key is one that can identify each row of a table uniquely. Generally a candidate key becomes the primary key of the table. If the table has more than one candidate key, one of them will become the primary key, and the rest are called alternate keys. A key formed by combining at least two or more columns is called composite key.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are defaults? Is there a column to which a default can’t be bound?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o A default is a value that will be used by a column, if no value is supplied to that column while inserting data. IDENTITY columns and timestamp columns can’t have defaults bound to them. See CREATE DEFAULT in books online.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is a transaction and what are ACID properties?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o A transaction is a logical unit of work in which, all the steps must be performed or none. ACID stands for Atomicity, Consistency, Isolation, Durability. These are the properties of a transaction. For more information and explanation of these properties, see SQL Server books online or any RDBMS fundamentals text book.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• Explain different isolation levels?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o An isolation level determines the degree of isolation of data between concurrent transactions. The default SQL Server isolation level is Read Committed. Here are the other isolation levels (in the ascending order of isolation): Read Uncommitted, Read Committed, Repeatable Read, Serializable. See SQL Server books online for an explanation of the isolation levels. Be sure to read about SET TRANSACTION ISOLATION LEVEL, which lets you customize the isolation level at the connection level. Read Committed - A transaction operating at the Read Committed level cannot see changes made by other transactions until those transactions are committed. At this level of isolation, dirty reads are not possible but nonrepeatable reads and phantoms are possible. Read Uncommitted - A transaction operating at the Read Uncommitted level can see uncommitted changes made by other transactions. At this level of isolation, dirty reads, nonrepeatable reads, and phantoms are all possible. Repeatable Read - A transaction operating at the Repeatable Read level is guaranteed not to see any changes made by other transactions in values it has already read. At this level of isolation, dirty reads and nonrepeatable reads are not possible but phantoms are possible. Serializable - A transaction operating at the Serializable level guarantees that all concurrent transactions interact only in ways that produce the same effect as if each transaction were entirely executed one after the other. At this isolation level, dirty reads, nonrepeatable reads, and phantoms are not possible.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• CREATE INDEX myIndex ON myTable(myColumn)What type of Index will get created after executing the above statement?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Non-clustered index. Important thing to note: By default a clustered index gets created on the primary key, unless specified otherwise.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What’s the maximum size of a row?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o 8060 bytes. Don’t be surprised with questions like ‘what is the maximum number of columns per table’. 1024 columns per table. Check out SQL Server books online for the page titled: "Maximum Capacity Specifications". Explain Active/Active and Active/Passive cluster configurations Hopefully you have experience setting up cluster servers. But if you don’t, at least be familiar with the way clustering works and the two clusterning configurations Active/Active and Active/Passive. SQL Server books online has enough information on this topic and there is a good white paper available on Microsoft site. Explain the architecture of SQL Server This is a very important question and you better be able to answer it if consider yourself a DBA. SQL Server books online is the best place to read about SQL Server architecture. Read up the chapter dedicated to SQL Server Architecture.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is lock escalation?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Lock escalation is the process of converting a lot of low level locks (like row locks, page locks) into higher level locks (like table locks). Every lock is a memory structure too many locks would mean, more memory being occupied by locks. To prevent this from happening, SQL Server escalates the many fine-grain locks to fewer coarse-grain locks. Lock escalation threshold was definable in SQL Server 6.5, but from SQL Server 7.0 onwards it’s dynamically managed by SQL Server.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What’s the difference between DELETE TABLE and TRUNCATE TABLE commands?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o DELETE TABLE is a logged operation, so the deletion of each row gets logged in the transaction log, which makes it slow. TRUNCATE TABLE also deletes all the rows in a table, but it won’t log the deletion of each row, instead it logs the deallocation of the data pages of the table, which makes it faster. Of course, TRUNCATE TABLE can be rolled back. TRUNCATE TABLE is functionally identical to DELETE statement with no WHERE clause: both remove all rows in the table. But TRUNCATE TABLE is faster and uses fewer system and transaction log resources than DELETE. The DELETE statement removes rows one at a time and records an entry in the transaction log for each deleted row. TRUNCATE TABLE removes the data by deallocating the data pages used to store the table’s data, and only the page deallocations are recorded in the transaction log. TRUNCATE TABLE removes all rows from a table, but the table structure and its columns, constraints, indexes and so on remain. The counter used by an identity for new rows is reset to the seed for the column. If you want to retain the identity counter, use DELETE instead. If you want to remove table definition and its data, use the DROP TABLE statement. You cannot use TRUNCATE TABLE on a table referenced by a FOREIGN KEY constraint; instead, use DELETE statement without a WHERE clause. Because TRUNCATE TABLE is not logged, it cannot activate a trigger. TRUNCATE TABLE may not be used on tables participating in an indexed view

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• Explain the storage models of OLAP

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Check out MOLAP, ROLAP and HOLAP in SQL Server books online for more infomation.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are the new features introduced in SQL Server 2000 (or the latest release of SQL Server at the time of your interview)? What changed between the previous version of SQL Server and the current version?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o This question is generally asked to see how current is your knowledge. Generally there is a section in the beginning of the books online titled "What’s New", which has all such information. Of course, reading just that is not enough, you should have tried those things to better answer the questions. Also check out the section titled "Backward Compatibility" in books online which talks about the changes that have taken place in the new version.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are constraints? Explain different types of constraints.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Constraints enable the RDBMS enforce the integrity of the database automatically, without needing you to create triggers, rule or defaults. Types of constraints: NOT NULL, CHECK, UNIQUE, PRIMARY KEY, FOREIGN KEY. For an explanation of these constraints see books online for the pages titled: "Constraints" and "CREATE TABLE", "ALTER TABLE"

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is an index? What are the types of indexes? How many clustered indexes can be created on a table? I create a separate index on each column of a table. What are the advantages and disadvantages of this approach?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Indexes in SQL Server are similar to the indexes in books. They help SQL Server retrieve the data quicker. Indexes are of two types. Clustered indexes and non-clustered indexes. When you create a clustered index on a table, all the rows in the table are stored in the order of the clustered index key. So, there can be only one clustered index per table. Non-clustered indexes have their own storage separate from the table data storage. Non-clustered indexes are stored as B-tree structures (so do clustered indexes), with the leaf level nodes having the index key and it’s row locater. The row located could be the RID or the Clustered index key, depending up on the absence or presence of clustered index on the table. If you create an index on each column of a table, it improves the query performance, as the query optimizer can choose from all the existing indexes to come up with an efficient execution plan. At the same t ime, data modification operations (such as INSERT, UPDATE, DELETE) will become slow, as every time data changes in the table, all the indexes need to be updated. Another disadvantage is that, indexes need disk space, the more indexes you have, more disk space is used.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is RAID and what are different types of RAID configurations?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o RAID stands for Redundant Array of Inexpensive Disks, used to provide fault tolerance to database servers. There are six RAID levels 0 through 5 offering different levels of performance, fault tolerance. MSDN has some information about RAID levels and for detailed information, check out the RAID advisory board’s homepage

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are the steps you will take to improve performance of a poor performing query?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o This is a very open ended question and there could be a lot of reasons behind the poor performance of a query. But some general issues that you could talk about would be: No indexes, table scans, missing or out of date statistics, blocking, excess recompilations of stored procedures, procedures and triggers without SET NOCOUNT ON, poorly written query with unnecessarily complicated joins, too much normalization, excess usage of cursors and temporary tables. Some of the tools/ways that help you troubleshooting performance problems are: SET SHOWPLAN\_ALL ON, SET SHOWPLAN\_TEXT ON, SET STATISTICS IO ON, SQL Server Profiler, Windows NT /2000 Performance monitor, Graphical execution plan in Query Analyzer. Download the white paper on performance tuning SQL Server from Microsoft web site. Don’t forget to check out sql-server-performance.com

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are the steps you will take, if you are tasked with securing an SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Again this is another open ended question. Here are some things you could talk about: Preferring NT authentication, using server, databse and application roles to control access to the data, securing the physical database files using NTFS permissions, using an unguessable SA password, restricting physical access to the SQL Server, renaming the Administrator account on the SQL Server computer, disabling the Guest account, enabling auditing, using multiprotocol encryption, setting up SSL, setting up firewalls, isolating SQL Server from the web server etc. Read the white paper on SQL Server security from Microsoft website. Also check out My SQL Server security best practices

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is a deadlock and what is a live lock? How will you go about resolving deadlocks?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Deadlock is a situation when two processes, each having a lock on one piece of data, attempt to acquire a lock on the other’s piece. Each process would wait indefinitely for the other to release the lock, unless one of the user processes is terminated. SQL Server detects deadlocks and terminates one user’s process. A livelock is one, where a request for an exclusive lock is repeatedly denied because a series of overlapping shared locks keeps interfering. SQL Server detects the situation after four denials and refuses further shared locks. A livelock also occurs when read transactions monopolize a table or page, forcing a write transaction to wait indefinitely. Check out SET DEADLOCK\_PRIORITY and "Minimizing Deadlocks" in SQL Server books online. Also check out the article Q169960 from Microsoft knowledge base.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is blocking and how would you troubleshoot it?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Blocking happens when one connection from an application holds a lock and a second connection requires a conflicting lock type. This forces the second connection to wait, blocked on the first. Read up the following topics in SQL Server books online: Understanding and avoiding blocking, Coding efficient transactions. Explain CREATE DATABASE syntax Many of us are used to creating databases from the Enterprise Manager or by just issuing the command: CREATE DATABAE MyDB.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• But what if you have to create a database with two filegroups, one on drive C and the other on drive D with log on drive E with an initial size of 600 MB and with a growth factor of 15%?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o That’s why being a DBA you should be familiar with the CREATE DATABASE syntax. Check out SQL Server books online for more information.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• How to restart SQL Server in single user mode? How to start SQL Server in minimal configuration mode?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o SQL Server can be started from command line, using the SQLSERVR.EXE. This EXE has some very important parameters with which a DBA should be familiar with. -m is used for starting SQL Server in single user mode and -f is used to start the SQL Server in minimal configuration mode. Check out SQL Server books online for more parameters and their explanations.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• As a part of your job, what are the DBCC commands that you commonly use for database maintenance?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o DBCC CHECKDB, DBCC CHECKTABLE, DBCC CHECKCATALOG, DBCC CHECKALLOC, DBCC SHOWCONTIG, DBCC SHRINKDATABASE, DBCC SHRINKFILE etc. But there are a whole load of DBCC commands which are very useful for DBAs. Check out SQL Server books online for more information.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are statistics, under what circumstances they go out of date, and how do you update them?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Statistics determine the selectivity of the indexes. If an indexed column has unique values then the selectivity of that index is more, as opposed to an index with non-unique values. Query optimizer uses these indexes in determining whether to choose an index or not while executing a query. Some situations under which you should update statistics: 1) If there is significant change in the key values in the index 2) If a large amount of data in an indexed column has been added, changed, or removed (that is, if the distribution of key values has changed), or the table has been truncated using the TRUNCATE TABLE statement and then repopulated 3) Database is upgraded from a previous version. Look up SQL Server books online for the following commands: UPDATE STATISTICS, STATS\_DATE, DBCC SHOW\_STATISTICS, CREATE STATISTICS, DROP STATISTICS, sp\_autostats, sp\_createstats, sp\_updatestats.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are the different ways of moving data/databases between servers and databases in SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o There are lots of options available; you have to choose your option depending upon your requirements. Some of the options you have are: BACKUP/RESTORE, dettaching and attaching databases, replication, DTS, BCP, logshipping, INSERT…SELECT, SELECT…INTO, creating INSERT scripts to generate data.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• Explain different types of BACKUPs avaialabe in SQL Server? Given a particular scenario, how would you go about choosing a backup plan?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Types of backups you can create in SQL Sever 7.0+ are Full database backup, differential database backup, transaction log backup, filegroup backup. Check out the BACKUP and RESTORE commands in SQL Server books online. Be prepared to write the commands in your interview. Books online also has information on detailed backup/restore architecture and when one should go for a particular kind of backup.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is database replication? What are the different types of replication you can set up in SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Replication is the process of copying/moving data between databases on the same or different servers. SQL Server supports the following types of replication scenarios: • Snapshot replication • Transactional replication (with immediate updating subscribers, with queued updating subscribers) • Merge replication See SQL Server books online for indepth coverage on replication. Be prepared to explain how different replication agents function, what are the main system tables used in replication etc.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• How to determine the service pack currently installed on SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o The global variable @@Version stores the build number of the sqlservr.exe, which is used to determine the service pack installed. To know more about this process visit SQL Server service packs and versions.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are cursors? Explain different types of cursors. What are the disadvantages of cursors? How can you avoid cursors?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Cursors allow row-by-row processing of the resultsets. Types of cursors: Static, Dynamic, Forward-only, Keyset-driven. See books online for more information. Disadvantages of cursors: Each time you fetch a row from the cursor, it results in a network roundtrip, where as a normal SELECT query makes only one roundtrip, however large the resultset is. Cursors are also costly because they require more resources and temporary storage (results in more IO operations). Further, there are restrictions on the SELECT statements that can be used with some types of cursors. Most of the times, set based operations can be used instead of cursors. Here is an example: If you have to give a flat hike to your employees using the following criteria: Salary between 30000 and 40000 – 5000 hike Salary between 40000 and 55000 – 7000 hike Salary between 55000 and 65000 – 9000 hike. In this situation many developers tend to use a cursor, determine each employee’s salary and update his salary according to the above formula. But the same can be achieved by multiple update statements or can be combined in a single UPDATE statement as shown below:

o UPDATE tbl\_emp SET salary = CASE WHEN salary BETWEEN 30000 AND 40000 THEN salary + 5000 WHEN salary BETWEEN 40000 AND 55000 THEN salary + 7000 WHEN salary BETWEEN 55000 AND 65000 THEN salary + 10000 END

o Another situation in which developers tend to use cursors: You need to call a stored procedure when a column in a particular row meets certain condition. You don’t have to use cursors for this. This can be achieved using WHILE loop, as long as there is a unique key to identify each row. For examples of using WHILE loop for row by row processing, check out the ‘My code library’ section of my site or search for WHILE. Write down the general syntax for a SELECT statements covering all the options. Here’s the basic syntax: (Also checkout SELECT in books online for advanced syntax).

o SELECT select\_list [INTO new\_table\_] FROM table\_source [WHERE search\_condition] [GROUP BY group\_by\_expression] [HAVING search\_condition] [ORDER BY order\_expression [ASC | DESC] ]

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is a join and explain different types of joins.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Joins are used in queries to explain how different tables are related. Joins also let you select data from a table depending upon data from another table. Types of joins: INNER JOINs, OUTER JOINs, CROSS JOINs. OUTER JOINs are further classified as LEFT OUTER JOINS, RIGHT OUTER JOINS and FULL OUTER JOINS. For more information see pages from books online titled: "Join Fundamentals" and "Using Joins".

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• Can you have a nested transaction?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Yes, very much. Check out BEGIN TRAN, COMMIT, ROLLBACK, SAVE TRAN and @@TRANCOUNT

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is an extended stored procedure? Can you instantiate a COM object by using T-SQL?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o An extended stored procedure is a function within a DLL (written in a programming language like C, C++ using Open Data Services (ODS) API) that can be called from T-SQL, just the way we call normal stored procedures using the EXEC statement. See books online to learn how to create extended stored procedures and how to add them to SQL Server. Yes, you can instantiate a COM (written in languages like VB, VC++) object from T-SQL by using sp\_OACreate stored procedure. Also see books online for sp\_OAMethod, sp\_OAGetProperty, sp\_OASetProperty, sp\_OADestroy. For an example of creating a COM object in VB and calling it from T-SQL, see ‘My code library’ section of this site.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is the system function to get the current user’s user id?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o USER\_ID(). Also check out other system functions like USER\_NAME(), SYSTEM\_USER, SESSION\_USER, CURRENT\_USER, USER, SUSER\_SID(), HOST\_NAME().

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are triggers? How many triggers you can have on a table? How to invoke a trigger on demand?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Triggers are special kind of stored procedures that get executed automatically when an INSERT, UPDATE or DELETE operation takes place on a table. In SQL Server 6.5 you could define only 3 triggers per table, one for INSERT, one for UPDATE and one for DELETE. From SQL Server 7.0 onwards, this restriction is gone, and you could create multiple triggers per each action. But in 7.0 there’s no way to control the order in which the triggers fire. In SQL Server 2000 you could specify which trigger fires first or fires last using sp\_settriggerorder. Triggers can’t be invoked on demand. They get triggered only when an associated action (INSERT, UPDATE, DELETE) happens on the table on which they are defined. Triggers are generally used to implement business rules, auditing. Triggers can also be used to extend the referential integrity checks, but wherever possible, use constraints for this purpose, instead of triggers, as constraints are much faster. Till SQL Server 7.0, triggers fire only after the data modification operation happens. So in a way, they are called post triggers. But in SQL Server 2000 you could create pre triggers also. Search SQL Server 2000 books online for INSTEAD OF triggers. Also check out books online for ‘inserted table’, ‘deleted table’ and COLUMNS\_UPDATED()

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• There is a trigger defined for INSERT operations on a table, in an OLTP system. The trigger is written to instantiate a COM object and pass the newly insterted rows to it for some custom processing. What do you think of this implementation? Can this be implemented better?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Instantiating COM objects is a time consuming process and since you are doing it from within a trigger, it slows down the data insertion process. Same is the case with sending emails from triggers. This scenario can be better implemented by logging all the necessary data into a separate table, and have a job which periodically checks this table and does the needful.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is a self join? Explain it with an example.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Self join is just like any other join, except that two instances of the same table will be joined in the query. Here is an example: Employees table which contains rows for normal employees as well as managers. So, to find out the managers of all the employees, you need a self join.

o CREATE TABLE emp ( empid int, mgrid int, empname char(10) )

o INSERT emp SELECT 1,2,’Vyas’ INSERT emp SELECT 2,3,’Mohan’ INSERT emp SELECT 3,NULL,’Shobha’ INSERT emp SELECT 4,2,’Shridhar’ INSERT emp SELECT 5,2,’Sourabh’

o SELECT t1.empname [Employee], t2.empname [Manager] FROM emp t1, emp t2 WHERE t1.mgrid = t2.empid Here’s an advanced query using a LEFT OUTER JOIN that even returns the employees without managers (super bosses)

o SELECT t1.empname [Employee], COALESCE(t2.empname, ‘No manager’) [Manager] FROM emp t1 LEFT OUTER JOIN emp t2 ON t1.mgrid = t2.empid

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~ EndRegion -- BASIC Interview Question ~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~ StartRegion -- Interview Question ~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Why are my insert, update statements failing with the following error?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Server: Msg 8152, Level 16, State 9, Line 1

String or binary data would be truncated.

The statement has been terminated.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is the T-SQL equivalent of IIF (immediate if/ternary operator) function of other programming languages?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to programmatically find out when the SQL Server service started?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to get rid of the time part from the date returned by GETDATE function?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to upload images or binary files into SQL Server tables?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to run an SQL script file that is located on the disk, using T-SQL?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to get the complete error message from T-SQL while error handling?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to get the first day of the week, last day of the week and last day of the month using T-SQL date functions?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to pass a table name, column name etc. to the stored procedure so that I can dynamically select from a table?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Error inside a stored procedure is not being raised to my front-end applications using ADO. But I get the error when I run the procedure from Query Analyzer

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to suppress error messages in stored procedures/triggers etc. using T-SQL?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to save the output of a query/stored procedure to a text file?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to join tables from different databases?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to join tables from different servers?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to convert timestamp data to date data (datetime datatype)?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Can I invoke/instantiate COM objects from within stored procedures or triggers using T-SQL?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Oracle has a rownum to access rows of a table using row number or row id. Is there any equivalent for that in SQL Server? Or how to generate output with row number in SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to specify a network library like TCP/IP using ADO connect string?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to generate scripts for repetitive tasks like truncating all the tables in a database, changing owner of all the database objects, disabling constraints on all tables etc?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Is there a way to find out when a stored procedure was last updated?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to find out all the IDENTITY columns of all the tables in a given database?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to search the code of stored procedures?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to retrieve the generated GUID value of a newly inserted row? Is there an @@GUID, just like @@IDENTITY ?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Why are my insert, update statements failing with the following error?

Server: Msg 8152, Level 16, State 9, Line 1

String or binary data would be truncated.

The statement has been terminated. <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

This error occurs, when the length of the value entered by you into a char, varchar, nchar, nvarchar column is longer than the maximum length of the column. For example, inserting 'FAQ' into a char(2) column would result in this error.

Profiler is handy in troubleshooting this error. If data truncation is okay with you and you don't want to see this error, then turn off ANSI WARNINGS by using the following SET command: SET ANSI\_WARNINGS OFF.

Steps to reproduce the problem:

CREATE TABLE MyTable

(

Pkey int PRIMARY KEY,

Col1 char(10)

)

GO

INSERT INTO MyTable (Pkey, Col1) VALUES (1, 'SQL Server Clustering FAQ')

GO

Make sure, you restrict the length of input, in your front-end applications. For example, you could use the MAXLENGTH property of the text boxes in HTML forms. E.g:

<INPUT NAME = "Name" TYPE= TEXTBOX MAXLENGTH=20>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is the T-SQL equivalent of IIF (immediate if/ternary operator) function of other programming languages? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

CASE is the equivalent of IIF function. See SQL Server Books Online for more information. Here's a quick example:

CREATE TABLE People

(

[ID] int PRIMARY KEY,

[Name] varchar(25) NOT NULL,

Sex bit NULL

)

INSERT INTO People ([ID],[Name], Sex) VALUES (1,'John Dykes', 1)

INSERT INTO People ([ID],[Name], Sex) VALUES (2,'Deborah Crook', 0)

INSERT INTO People ([ID],[Name], Sex) VALUES (3,'P S Subramanyam', NULL)

SELECT [ID], [Name],

CASE Sex

WHEN 1

THEN 'Male'

WHEN 0

THEN 'Female'

ELSE 'Not specified'

END AS Sex

FROM People

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to programmatically find out when the SQL Server service started? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Everytime SQL Server starts, it recreates the tempdb database. So, the creation date and time of the tempdb database tells us the date and time at which SQL Server service started. This information is stored in the crdate column of the sysdatabases table in master database. Here's the query to find that out:

SELECT crdate AS 'SQL Server service started approximately at:'

FROM master.dbo.sysdatabases

WHERE name = 'tempdb'

SQL Server error log also has this information (This is more accurate) and the error log can be queried using xp\_readerrorlog

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to get rid of the time part from the date returned by GETDATE function? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

We have to use the CONVERT function to strip the time off the date. Any of the following commands will do this:

SELECT CONVERT(char,GETDATE(),101)

SELECT CONVERT(char,GETDATE(),102)

SELECT CONVERT(char,GETDATE(),103)

SELECT CONVERT(char,GETDATE(),1)

See SQL Server Books Online for more information on CONVERT function.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to upload images or binary files into SQL Server tables? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

First of all, if possible, try not to stored images and other binary files in the SQL Server tables, as they slow things down. Instead, store a link (file path) to the file in the tables and let your applications directly access the files. But if you must store these files within SQL Server, use the text/ntext or image datatype columns and consider the following options:

• SQL Server 7.0 and 2000 come with a utility called textcopy.exe. You can locate this file in the Binn folder under your SQL Server installation folder. Run this file from command prompt, and it will prompt you for required input

• Use the GetChunk and AppendChunk methods of ADO Field object. MSDN has examples

• Use the ADO Stream object

• Use the Bulk Insert Image utility (BII) that ships with SQL Server 2000 (Can be found at \Program Files\Microsoft SQL

Server\80\Tools\Devtools\Samples\Utils)

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to run an SQL script file that is located on the disk, using T-SQL? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

There's no direct command to read a script file and execute it. But the isql.exe and osql.exe come in handy when you have to execute a script file from within T-SQL. Just call any of these exes using xp\_cmdshell and pass the script file name as parameter to it. See SQL Server Books Online for more information about the input parameters of these exes. Here are some quick examples:

EXEC master..xp\_cmdshell 'osql -Svaio -Usa -Pzaassds1 -ic:\MySQl.sql -n'

EXEC master..xp\_cmdshell 'isql -Svaio -Usa -Pzaassds1 -ic:\MySQl.sql -n'

See xp\_cmdshell in SQL Server Books Online if you are having permissions problems in getting this technique to work.

How to get the complete error message from T-SQL while error handling? <top>

Unfortunately, the error handling capabilities of SQL Server are limited. When an error occurs, all you can get is the error number, using the @@ERROR global variable. There is no @@ERROR\_MESSAGE global variable to get the error description.

For a complete error message, you can always query the master..sysmessages table using the error number, but most of these messages have place holders (like %s, %l etc.), and hence we can't get the complete error message.

However, the client applications using an object model such as RDO, ADO have access to the complete error message.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to get the first day of the week, last day of the week and last day of the month using T-SQL date functions? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Here's the code:

DECLARE @Date datetime

SET @Date = '2001/08/31'

SELECT DATEADD(dd,-(DATEPART(dw, @Date) - 1),@Date) AS 'First day of the week'

SELECT DATEADD(dd,-(DATEPART(dw, @Date) - 7),@Date) AS 'Last day of the week'

SELECT DAY(DATEADD(d, -DAY(DATEADD(m,1,@Date)),DATEADD(m,1,@Date))) AS 'Last day of the month'

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to pass a table name, column name etc. to the stored procedure so that I can dynamically select from a table? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Basically, SELECT and other commands like DROP TABLE won't let you use a variable instead of a hardcoded table name. To overcome this problem, you have to use dynamic sql. But dynamic SQL has some disadvantages. It's slow, as the dynamic SQL statement needs to be parsed everytime it's executed. Further, the user who is executing the dynamic SQL string needs direct permissions on the tables, which defeats the purpose of having stored procedures to mask the underlying tables. Having said that, here are some examples of dynamic SQL: (Also see sp\_executesql in SQL Server Books Online)

CREATE PROC DropTable

@Table sysname

AS

EXEC ('DROP TABLE ' + @Table)

GO

EXEC DropTable 'MyTable'

GO

CREATE PROC SelectTable

@Table sysname

AS

EXEC ('SELECT \* FROM ' + @Table)

GO

EXEC SelectTable 'MyTable'

For a complete discussion on the pros and cons of dynamic SQL check out Erland's article:

The curse and blessings of dynamic SQL

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Error inside a stored procedure is not being raised to my front-end applications using ADO. But I get the error when I run the procedure from Query Analyzer <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

This typically happens when your stored procedure is returning multiple resultsets and the offending SQL statement is executed after returning one or more resultsets. ADO will not return an error untill it processes all the recordsets returned before the offending SQL statement got executed. So, to get to the error message returned by your procedure. You have to loop through all the recordsets returned. ADO Recordset object has a method called NextRecordset, which lets you loop through the recordsets.

Having SET NOCOUNT ON at the beginning of the procedure also helps avoid this problem. SET NOCOUNT ON also helps in improving the stored procedure performance. Here's a sample procedure to simulate the problem:

CREATE PROC TestProc

AS

SELECT MAX(Col1) FROM TestTable

SELECT MIN(Col1) FROM TestTable

INSERT INTO TestTable (Col1, Col2) VALUES (1,'Oracle and SQL Server comparison')

INSERT INTO TestTable (Col1, Col2) VALUES (1,'How to configure SQL Server?') -- Dupplicate key error occurs

GO

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to suppress error messages in stored procedures/triggers etc. using T-SQL? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

It's not possible to suppress error messages from within T-SQL. Error messages are always returned to the client. If you don't want your users to see these raw error messages, you should handle them in your front-end applications. For example, if you are using ADO from ASP to connect to SQL Server, you would do something like the following:

On Error Resume Next

Set Rs = Conn.Execute ("INSERT INTO MyTable (1,'How to migrate from Oracle to SQL Server','Book'")

If Err.Number <> 0 Then Response.Write ("Error occurred while inserting new data")

On Error GoTo 0

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to save the output of a query/stored procedure to a text file using T-SQL? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

T-SQL by itself has no support for saving the output of queries/stored procedures to text files. But you could achieve this using the command line utilities like isql.exe and osql.exe. You could either invoke these exe files directly from command prompt/batch files or from T-SQL using the xp\_cmdshell command. Here are the examples:

From command prompt:

osql.exe -S YourServerName -U sa -P secretcode -Q "EXEC sp\_who2" -o "E:\output.txt"

From T-SQL:

EXEC master..xp\_cmdshell 'osql.exe -S YourServerName -U sa -P secretcode -Q "EXEC sp\_who2" -o "E:\output.txt"'

Query Analyzer lets you save the query output to text files manually. The output of stored procedures that are run as a part of a scheduled job, can also be saved to a text file.

BCP and Data Transformation Services (DTS) let you export table data to text files.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to join tables from different databases? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

You just have to qualify the table names in your SELECT queries with database name, followed by table owner name. In the following example, Table1 from pubs database and Table2 from northwind database are being joined on the column i. Both tables are owned by dbo.

SELECT a.i, a.j

FROM pubs.dbo.Table1 a

INNER JOIN

northwind.dbo.Table2 b

ON a.i = b.i

GO

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to join tables from different servers? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

To be able to join tables between two SQL Servers, first you have to link them. After the linked servers are setup, you just have to prefix your tables names with server name, database name, table owner name in your SELECT queries. The following example links SERVER\_01 to SERVER\_02. Execute the following commands in SERVER\_02:

EXEC sp\_addlinkedserver SERVER\_01

GO

/\* The following command links 'sa' login on SERVER\_02 with the 'sa' login of SERVER\_01 \*/

EXEC sp\_addlinkedsrvlogin @rmtsrvname = 'SERVER\_01', @useself = 'false', @locallogin = 'sa', @rmtuser = 'sa', @rmtpassword = 'sa password of SERVER\_01'

GO

SELECT a.title\_id

FROM SERVER\_01.pubs.dbo.titles a

INNER JOIN SERVER\_02.pubs.dbo.titles b

ON a.title\_id = b.title\_id

GO

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to convert timestamp data to date data (datetime datatype)? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

The name timestamp is a little misleading. Timestamp data has nothing to do with dates and times and can not be converted to date data. A timestamp is a unique number within the database and is equivalent to a binary(8)/varbinary(8) datatype. A table can have only one timestamp column. Timestamp value of a row changes with every update of the row. To avoid the confusion, SQL Server 2000 introduced a synonym to timestamp, called rowversion.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Can I invoke/instantiate COM objects from within stored procedures or triggers using T-SQL? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Yes. SQL Server provides system stored procedures that let you instantiate COM objects using T-SQL from stored procedures, triggers and SQL batches. Search SQL Server Books Online for sp\_OACreate and sp\_OA\* for documentation and examples. Also check out my code library for an example.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Oracle has a rownum to access rows of a table using row number or row id. Is there any equivalent for that in SQL Server? Or how to generate output with row number in SQL Server? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

There is no direct equivalent to Oracle's rownum or row id in SQL Server. Strictly speaking, in a relational database, rows within a table are not ordered and a row id won't really make sense. But if you need that functionality, consider the following three alternatives:

• Add an IDENTITY column to your table. See Books Online for more information

• Use the following query to generate a row number for each row. The following query generates a row number for each row in the authors table of pubs database. For this query to work, the table must have a unique key.

SELECT (SELECT COUNT(i.au\_id)

FROM pubs..authors i

WHERE i.au\_id >= o.au\_id ) AS RowID,

au\_fname + ' ' + au\_lname AS 'Author name'

FROM pubs..authors o

ORDER BY RowID

• Use a temporary table approach, to store the entire resultset into a temporary table, along with a row id generated by the IDENTITY() function. Creating a temporary table will be costly, especially when you are working with large tables. Go for this approach, if you don't have a unique key in your table. Search for IDENTITY (Function) in SQL Server Books Online.

For more ideas on this topic, click here to read an informative article from Microsoft Knowledgebase.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to specify a network library like TCP/IP using ADO connect string? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

To specify TCP/IP net library, append the following to your ADO connect string:

Network=dbmssocn

For more information on specifying other net libraries in ADO connect strings, click here to read the article from Microsoft Knowledgebase.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Is there a way to find out when a stored procedure was last updated? <top>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Simple answer is 'No'. The crdate column in the sysobjects table always contains the stored procedure create date, not the last updated date. You can use Profiler to trace ALTER PROC calls to the database, but you can't really afford to run a trace for ever, as it's resource intensive. Here is a simple idea! Whenever you have to alter your stored procedure, first drop it, then recreate it with the updated code. This resets the crdate column of sysobjects table. If you can make sure your developers always follow this plan, then the crdate column of sysobjects will always reflect the last updated date of the stored procedure. For example, if I have to modify a procedure named MyProc, instead of doing "ALTER PROC MyProc", here's what I would do:

- Use sp\_helptext to get the current code of MyProc.

- Change the code as needed.

- Run the following code to drop the existing version of MyProc:

IF EXISTS(SELECT 1 FROM sysobjects WHERE name = 'MyProc' AND type = 'P' AND USER\_NAME(uid) = 'dbo')

BEGIN

DROP PROC dbo.MyProc

END

- Run the updated code to recreate MyProc

There is a much more powerful way out, if you can use Visual Source Safe (VSS). VSS is a version control software, that lets you manage your code. With VSS in place, you will have to maintain all your object creation scripts as script files and check them into VSS. When you have to modify a particular stored procedure, check out that script from VSS, modify it, test it, create the stored procedure, and check the script back into VSS. VSS can show you when a script got modified, by who and a whole lot of other information.

Advantages of using VSS

- You can version control your software, as VSS maintains all your changes as different versions

- You can go back to a previous known good version of your stored procedure, if a developer makes a mistake

- Using the labelling feature, you can revert back to an entire set of scripts at a particular point in time

- You can control access to your source code by configuring permissions to your developers

- By maintaining backups of VSS database, you can secure all your code centrally, instead of worrying about individual script files

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~ EndRegion -- Interview Question ~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~ StartRegion -- Questions ~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1. How to find and delete duplicate records from a table?

2. How to generate the user defined store procedures list in a database?

3. How to obtain the description of a column?

4. How to schedule the scripting of database objects?

5. How to programmatically find the installation directory of SQL Server?

6. How to store images(JPEG,GIF...etc) in SQL server?

7. How to check if a temp table exists already?

8. How to find the number of rows affected by bcp?

9. How to add a default constraint to an existing column?

10. Records in one table but not in another?

11. How to fire triggers in SQL 7 when using bcp?

12. How to break a column to two using SUBSTRING()?

13. How to define constraints across databases?

14. How to write a script that enumerates all user tables in a SQL Server database and then set the primary key in each table?

15. How to prefix zeroes to a integer?

16. How to specify 'order by' on specific columns in a view.?

17. How to pass an array to a stored procedure?

18. How to get the current date without the time?

19. How to find the execution time of a stored procedure?

20. How to write a alphanumeric query?

21. How to create a system stored procedure that has type set to SYSTEM ?

22. How to find out the columns having index?

23. How to do a SELECT inside a CASE expression?

24. How to to find the last product purchased from each vendor?

25. How to sort data inside a string?

26. How to use PATINDEX to compare data?

27. How to catch a OUTPUT parameter?

28. How to round up a datetime value to the nearest hour?

29. How to format a datetime value like " dd-mon-yy "?

30. Which one is faster -- IN or OR?

31. How to manually insert a IDENTITY value?

32. How to display count of rows as different set of columns?

33. How to use the different DATE and TIME functions?

34. How and when to use SELF JOIN?

35. How to generate a rownumber for the query?

36. How to insert carriage return?

37. How to use dynamic sql for a nvarchar of more than 4000 characters?

38. How to avoid the appearance of 'n rows affected'?

39. How to convert a number, stored in seconds, to hours:minutes?

40. How to reset/clear a IDENTITY value?

41. SET or SELECT -- is there any performance difference?

42. How to SELECT nth row from a table?

43. Is there a performance difference between CHAR and VARCHAR?

44. How to SELECT combined DISTINCT values from two tables?

45. How to use CASE for ORDER BY clause?

~~~~~~~~~~~~~~~~~~~~~~~~

~~~ Database design ~~~

~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is normalization? Explain different levels of normalization?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Check out the article Q100139 from Microsoft knowledge base and of course, there's much more information available in the net. It'll be a good idea to get a hold of any RDBMS fundamentals text book, especially the one by C. J. Date. Most of the times, it will be okay if you can explain till third normal form.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is denormalization and when would you go for it?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

As the name indicates, denormalization is the reverse process of normalization. It's the controlled introduction of redundancy in to the database design. It helps improve the query performance as the number of joins could be reduced.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How do you implement one-to-one, one-to-many and many-to-many relationships while designing tables?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

One-to-One relationship can be implemented as a single table and rarely as two tables with primary and foreign key relationships.

One-to-Many relationships are implemented by splitting the data into two tables with primary key and foreign key relationships.

Many-to-Many relationships are implemented using a junction table with the keys from both the tables forming the composite primary key of the junction table.

It will be a good idea to read up a database designing fundamentals text book.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What's the difference between a primary key and a unique key?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Both primary key and unique enforce uniqueness of the column on which they are defined. But by default primary key creates a clustered index on the column, where are unique creates a nonclustered index by default. Another major difference is that, primary key doesn't allow NULLs, but unique key allows one NULL only.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are user defined datatypes and when you should go for them?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

User defined datatypes let you extend the base SQL Server datatypes by providing a descriptive name, and format to the database. Take for example, in your database, there is a column called Flight\_Num which appears in many tables. In all these tables it should be varchar(8). In this case you could create a user defined datatype called Flight\_num\_type of varchar(8) and use it across all your tables.

See sp\_addtype, sp\_droptype in books online.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is bit datatype and what's the information that can be stored inside a bit column?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Bit datatype is used to store boolean information like 1 or 0 (true or false). Untill SQL Server 6.5 bit datatype could hold either a 1 or 0 and there was no support for NULL. But from SQL Server 7.0 onwards, bit datatype can represent a third state, which is NULL.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Define candidate key, alternate key, composite key.?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

A candidate key is one that can identify each row of a table uniquely. Generally a candidate key becomes the primary key of the table. If the table has more than one candidate key, one of them will become the primary key, and the rest are called alternate keys.

A key formed by combining at least two or more columns is called composite key.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are defaults? Is there a column to which a default can't be bound?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

A default is a value that will be used by a column, if no value is supplied to that column while inserting data. IDENTITY columns and timestamp columns can't have defaults bound to them. See CREATE DEFUALT in books online.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~ SQL Server architecture ~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is a transaction and what are ACID properties?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

A transaction is a logical unit of work in which, all the steps must be performed or none. ACID stands for Atomicity, Consistency, Isolation, Durability. These are the properties of a transaction. For more information and explanation of these properties, see SQL Server books online or any RDBMS fundamentals text book.

Explain different isolation levels?

An isolation level determines the degree of isolation of data between concurrent transactions. The default SQL Server isolation level is Read Committed. Here are the other isolation levels (in the ascending order of isolation): Read Uncommitted, Read Committed, Repeatable Read, Serializable. See SQL Server books online for an explanation of the isolation levels. Be sure to read about SET TRANSACTION ISOLATION LEVEL, which lets you customize the isolation level at the connection level.

CREATE INDEX myIndex ON myTable(myColumn)

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What type of Index will get created after executing the above statement?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Non-clustered index. Important thing to note: By default a clustered index gets created on the primary key, unless specified otherwise.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What's the maximum size of a row?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

8060 bytes. Don't be surprised with questions like 'what is the maximum number of columns per table'. Check out SQL Server books online for the page titled: "Maximum Capacity Specifications".

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Explain Active/Active and Active/Passive cluster configurations?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Hopefully you have experience setting up cluster servers. But if you don't, at least be familiar with the way clustering works and the two clusterning configurations Active/Active and Active/Passive. SQL Server books online has enough information on this topic and there is a good white paper available on Microsoft site.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Explain the architecture of SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

This is a very important question and you better be able to answer it if consider yourself a DBA. SQL Server books online is the best place to read about SQL Server architecture. Read up the chapter dedicated to SQL Server Architecture.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is lock escalation?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Lock escalation is the process of converting a lot of low level locks (like row locks, page locks) into higher level locks (like table locks). Every lock is a memory structure too many locks would mean, more memory being occupied by locks. To prevent this from happening, SQL Server escalates the many fine-grain locks to fewer coarse-grain locks. Lock escalation threshold was definable in SQL Server 6.5, but from SQL Server 7.0 onwards it's dynamically managed by SQL Server.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What's the difference between DELETE TABLE and TRUNCATE TABLE commands?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

DELETE TABLE is a logged operation, so the deletion of each row gets logged in the transaction log, which makes it slow. TRUNCATE TABLE also deletes all the rows in a table, but it won't log the deletion of each row, instead it logs the deallocation of the data pages of the table, which makes it faster. Of course, TRUNCATE TABLE can be rolled back.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Explain the storage models of OLAP?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Check out MOLAP, ROLAP and HOLAP in SQL Server books online for more infomation.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are the new features introduced in SQL Server 2000 (or the latest release of SQL Server at the time of your interview)? What changed between the previous version of SQL Server and the current version?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

This question is generally asked to see how current is your knowledge. Generally there is a section in the beginning of the books online titled "What's New", which has all such information. Of course, reading just that is not enough, you should have tried those things to better answer the questions. Also check out the section titled "Backward Compatibility" in books online which talks about the changes that have taken place in the new version.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are constraints? Explain different types of constraints.?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Constraints enable the RDBMS enforce the integrity of the database automatically, without needing you to create triggers, rule or defaults.

Types of constraints: NOT NULL, CHECK, UNIQUE, PRIMARY KEY, FOREIGN KEY

For an explanation of these constraints see books online for the pages titled: "Constraints" and "CREATE TABLE", "ALTER TABLE"

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Whar is an index? What are the types of indexes? How many clustered indexes can be created on a table? I create a separate index on each column of a table. what are the advantages and disadvantages of this approach?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Indexes in SQL Server are similar to the indexes in books. They help SQL Server retrieve the data quicker.

Indexes are of two types. Clustered indexes and non-clustered indexes. When you craete a clustered index on a table, all the rows in the table are stored in the order of the clustered index key. So, there can be only one clustered index per table. Non-clustered indexes have their own storage separate from the table data storage. Non-clustered indexes are stored as B-tree structures (so do clustered indexes), with the leaf level nodes having the index key and it's row locater. The row located could be the RID or the Clustered index key, depending up on the absence or presence of clustered index on the table.

If you create an index on each column of a table, it improves the query performance, as the query optimizer can choose from all the existing indexes to come up with an efficient execution plan. At the same t ime, data modification operations (such as INSERT, UPDATE, DELETE) will become slow, as every time data changes in the table, all the indexes need to be updated. Another disadvantage is that, indexes need disk space, the more indexes you have, more disk space is used.

~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~ Database administration ~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is RAID and what are different types of RAID configurations?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

RAID stands for Redundant Array of Inexpensive Disks, used to provide fault tolerance to database servers. There are six RAID levels 0 through 5 offering different levels of performance, fault tolerance. MSDN has some information about RAID levels and for detailed information, check out the RAID advisory board's homepage

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are the steps you will take to improve performance of a poor performing query?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

This is a very open ended question and there could be a lot of reasons behind the poor performance of a query. But some general issues that you could talk about would be: No indexes, table scans, missing or out of date statistics, blocking, excess recompilations of stored procedures, procedures and triggers without SET NOCOUNT ON, poorly written query with unnecessarily complicated joins, too much normalization, excess usage of cursors and temporary tables.

Some of the tools/ways that help you troubleshooting performance problems are: SET SHOWPLAN\_ALL ON, SET SHOWPLAN\_TEXT ON, SET STATISTICS IO ON, SQL Server Profiler, Windows NT /2000 Performance monitor, Graphical execution plan in Query Analyzer.

Download the white paper on performance tuning SQL Server from Microsoft web site. Don't forget to check out sql-server-performance.com

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are the steps you will take, if you are tasked with securing an SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Again this is another open ended question. Here are some things you could talk about: Preferring NT authentication, using server, databse and application roles to control access to the data, securing the physical database files using NTFS permissions, using an unguessable SA password, restricting physical access to the SQL Server, renaming the Administrator account on the SQL Server computer, disabling the Guest account, enabling auditing, using multiprotocol encryption, setting up SSL, setting up firewalls, isolating SQL Server from the web server etc.

Read the white paper on SQL Server security from Microsoft website. Also check out My SQL Server security best practices

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is a deadlock and what is a live lock? How will you go about resolving deadlocks?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Deadlock is a situation when two processes, each having a lock on one piece of data, attempt to acquire a lock on the other's piece. Each process would wait indefinitely for the other to release the lock, unless one of the user processes is terminated. SQL Server detects deadlocks and terminates one user's process.

A livelock is one, where a request for an exclusive lock is repeatedly denied because a series of overlapping shared locks keeps interfering. SQL Server detects the situation after four denials and refuses further shared locks. A livelock also occurs when read transactions monopolize a table or page, forcing a write transaction to wait indefinitely.

Check out SET DEADLOCK\_PRIORITY and "Minimizing Deadlocks" in SQL Server books online. Also check out the article Q169960 from Microsoft knowledge base.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is blocking and how would you troubleshoot it?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Blocking happens when one connection from an application holds a lock and a second connection requires a conflicting lock type. This forces the second connection to wait, blocked on the first.

Read up the following topics in SQL Server books online: Understanding and avoiding blocking, Coding efficient transactions.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Explain CREATE DATABASE syntax?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Many of us are used to craeting databases from the Enterprise Manager or by just issuing the command: CREATE DATABAE MyDB. But what if you have to create a database with two filegroups, one on drive C and the other on drive D with log on drive E with an initial size of 600 MB and with a growth factor of 15%? That's why being a DBA you should be familiar with the CREATE DATABASE syntax. Check out SQL Server books online for more information.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to restart SQL Server in single user mode? How to start SQL Server in minimal configuration mode?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

SQL Server can be started from command line, using the SQLSERVR.EXE. This EXE has some very important parameters with which a DBA should be familiar with. -m is used for starting SQL Server in single user mode and -f is used to start the SQL Server in minimal confuguration mode. Check out SQL Server books online for more parameters and their explanations.

As a part of your job, what are the DBCC commands that you commonly use for database maintenance?

DBCC CHECKDB, DBCC CHECKTABLE, DBCC CHECKCATALOG, DBCC CHECKALLOC, DBCC SHOWCONTIG, DBCC SHRINKDATABASE, DBCC SHRINKFILE etc. But there are a whole load of DBCC commands which are very useful for DBAs. Check out SQL Server books online for more information.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are statistics, under what circumstances they go out of date, how do you update them?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Statistics determine the selectivity of the indexes. If an indexed column has unique values then the selectivity of that index is more, as opposed to an index with non-unique values. Query optimizer uses these indexes in determining whether to choose an index or not while executing a query.

Some situations under which you should update statistics:

1) If there is significant change in the key values in the index

2) If a large amount of data in an indexed column has been added, changed, or removed (that is, if the distribution of key values has changed), or the table has been truncated using the TRUNCATE TABLE statement and then repopulated

3) Database is upgraded from a previous version

Look up SQL Server books online for the following commands: UPDATE STATISTICS, STATS\_DATE, DBCC SHOW\_STATISTICS, CREATE STATISTICS, DROP STATISTICS, sp\_autostats, sp\_createstats, sp\_updatestats

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are the different ways of moving data/databases between servers and databases in SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

There are lots of options available, you have to choose your option depending upon your requirements. Some of the options you have are: BACKUP/RESTORE, dettaching and attaching databases, replication, DTS, BCP, logshipping, INSERT...SELECT, SELECT...INTO, creating INSERT scripts to generate data.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Explian different types of BACKUPs avaialabe in SQL Server? Given a particular scenario, how would you go about choosing a backup plan?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Types of backups you can create in SQL Sever 7.0+ are Full database backup, differential database backup, transaction log backup, filegroup backup. Check out the BACKUP and RESTORE commands in SQL Server books online. Be prepared to write the commands in your interview. Books online also has information on detailed backup/restore architecture and when one should go for a particular kind of backup.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is database replicaion? What are the different types of replication you can set up in SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Replication is the process of copying/moving data between databases on the same or different servers. SQL Server supports the following types of replication scenarios:

• Snapshot replication

• Transactional replication (with immediate updating subscribers, with queued updating subscribers)

• Merge replication

See SQL Server books online for indepth coverage on replication. Be prepared to explain how different replication agents function, what are the main system tables used in replication etc.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

How to determine the service pack currently installed on SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

The global variable @@Version stores the build number of the sqlservr.exe, which is used to determine the service pack installed. To know more about this process visit SQL Server service packs and versions.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~ Database programming ~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are cursors? Explain different types of cursors. What are the disadvantages of cursors? How can you avoid cursors?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Cursors allow row-by-row prcessing of the resultsets.

Types of cursors: Static, Dynamic, Forward-only, Keyset-driven. See books online for more information.

Disadvantages of cursors: Each time you fetch a row from the cursor, it results in a network roundtrip, where as a normal SELECT query makes only one rowundtrip, however large the resultset is. Cursors are also costly because they require more resources and temporary storage (results in more IO operations). Furthere, there are restrictions on the SELECT statements that can be used with some types of cursors.

Most of the times, set based operations can be used instead of cursors. Here is an example:

If you have to give a flat hike to your employees using the following criteria:

Salary between 30000 and 40000 -- 5000 hike

Salary between 40000 and 55000 -- 7000 hike

Salary between 55000 and 65000 -- 9000 hike

In this situation many developers tend to use a cursor, determine each employee's salary and update his salary according to the above formula. But the same can be achieved by multiple update statements or can be combined in a single UPDATE statement as shown below:

UPDATE tbl\_emp SET salary =

CASE WHEN salary BETWEEN 30000 AND 40000 THEN salary + 5000

WHEN salary BETWEEN 40000 AND 55000 THEN salary + 7000

WHEN salary BETWEEN 55000 AND 65000 THEN salary + 10000

END

Another situation in which developers tend to use cursors: You need to call a stored procedure when a column in a particular row meets certain condition. You don't have to use cursors for this. This can be achieved using WHILE loop, as long as there is a unique key to identify each row. For examples of using WHILE loop for row by row processing, check out the 'My code library' section of my site or search for WHILE.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Write down the general syntax for a SELECT statements covering all the options.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Here's the basic syntax: (Also checkout SELECT in books online for advanced syntax).

SELECT select\_list

[INTO new\_table\_]

FROM table\_source

[WHERE search\_condition]

[GROUP BY group\_by\_expression]

[HAVING search\_condition]

[ORDER BY order\_expression [ASC | DESC] ]

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is a join and explain different types of joins.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Joins are used in queries to explain how different tables are related. Joins also let you select data from a table depending upon data from another table.

Types of joins: INNER JOINs, OUTER JOINs, CROSS JOINs. OUTER JOINs are further classified as LEFT OUTER JOINS, RIGHT OUTER JOINS and FULL OUTER JOINS.

For more information see pages from books online titled: "Join Fundamentals" and "Using Joins".

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Can you have a nested transaction?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Yes, very much. Check out BEGIN TRAN, COMMIT, ROLLBACK, SAVE TRAN and @@TRANCOUNT

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is an extended stored procedure? Can you instantiate a COM object by using T-SQL?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

An extended stored procedure is a function within a DLL (written in a programming language like C, C++ using Open Data Services (ODS) API) that can be called from T-SQL, just the way we call normal stored procedures using the EXEC statement. See books online to learn how to create extended stored procedures and how to add them to SQL Server.

Yes, you can instantiate a COM (written in languages like VB, VC++) object from T-SQL by using sp\_OACreate stored procedure. Also see books online for sp\_OAMethod, sp\_OAGetProperty, sp\_OASetProperty, sp\_OADestroy. For an example of creating a COM object in VB and calling it from T-SQL, see 'My code library' section of this site.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is the system function to get the current user's user id?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

USER\_ID(). Also check out other system functions like USER\_NAME(), SYSTEM\_USER, SESSION\_USER, CURRENT\_USER, USER, SUSER\_SID(), HOST\_NAME().

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What are triggers? How many triggers you can have on a table? How to invoke a trigger on demand?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Triggers are special kind of stored procedures that get executed automatically when an INSERT, UPDATE or DELETE operation takes place on a table.

In SQL Server 6.5 you could define only 3 triggers per table, one for INSERT, one for UPDATE and one for DELETE. From SQL Server 7.0 onwards, this restriction is gone, and you could create multiple triggers per each action. But in 7.0 there's no way to control the order in which the triggers fire. In SQL Server 2000 you could specify which trigger fires first or fires last using sp\_settriggerorder

Triggers can't be invoked on demand. They get triggered only when an associated action (INSERT, UPDATE, DELETE) happens on the table on which they are defined.

Triggers are generally used to implement business rules, auditing. Triggers can also be used to extend the referential integrity checks, but wherever possible, use constraints for this purpose, instead of triggers, as constraints are much faster.

Till SQL Server 7.0, triggers fire only after the data modification operation happens. So in a way, they are called post triggers. But in SQL Server 2000 you could create pre triggers also. Search SQL Server 2000 books online for INSTEAD OF triggers.

Also check out books online for 'inserted table', 'deleted table' and COLUMNS\_UPDATED()

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

There is a trigger defined for INSERT operations on a table, in an OLTP system. The trigger is written to instantiate a COM object and pass the newly insterted rows to it for some custom processing. What do you think of this implementation? Can this be implemented better?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Instantiating COM objects is a time consuming process and since you are doing it from within a trigger, it slows down the data insertion process. Same is the case with sending emails from triggers. This scenario can be better implemented by logging all the necessary data into a separate table, and have a job which periodically checks this table and does the needful.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

What is a self join? Explain it with an example.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Self join is just like any other join, except that two instances of the same table will be joined in the query. Here is an example: Employees table which contains rows for normal employees as well as managers. So, to find out the managers of all the employees, you need a self join.

CREATE TABLE emp

(

empid int,

mgrid int,

empname char(10)

)

INSERT emp SELECT 1,2,'Vyas'

INSERT emp SELECT 2,3,'Mohan'

INSERT emp SELECT 3,NULL,'Shobha'

INSERT emp SELECT 4,2,'Shridhar'

INSERT emp SELECT 5,2,'Sourabh'

SELECT t1.empname [Employee], t2.empname [Manager]

FROM emp t1, emp t2

WHERE t1.mgrid = t2.empid

Here's an advanced query using a LEFT OUTER JOIN that even returns the employees without managers (super bosses)

SELECT t1.empname [Employee], COALESCE(t2.empname, 'No manager') [Manager]

FROM emp t1

LEFT OUTER JOIN

emp t2

ON

t1.mgrid = t2.empid

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Given an employee table, how would you find out the second highest salary?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is normalization? Explain different levels of normalization?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Check out the article Q100139 from Microsoft knowledge base and of course, there’s much more information available in the net. It’ll be a good idea to get a hold of any RDBMS fundamentals text book, especially the one by C. J. Date. Most of the times, it will be okay if you can explain till third normal form.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is denormalization and when would you go for it?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o As the name indicates, denormalization is the reverse process of normalization. It’s the controlled introduction of redundancy in to the database design. It helps improve the query performance as the number of joins could be reduced.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• How do you implement one-to-one, one-to-many and many-to-many relationships while designing tables?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o One-to-One relationship can be implemented as a single table and rarely as two tables with primary and foreign key relationships. One-to-Many relationships are implemented by splitting the data into two tables with primary key and foreign key relationships. Many-to-Many relationships are implemented using a junction table with the keys from both the tables forming the composite primary key of the junction table. It will be a good idea to read up a database designing fundamentals text book.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What’s the difference between a primary key and a unique key?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Both primary key and unique enforce uniqueness of the column on which they are defined. But by default primary key creates a clustered index on the column, where are unique creates a nonclustered index by default. Another major difference is that, primary key doesn’t allow NULLs, but unique key allows one NULL only.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are user defined datatypes and when you should go for them?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o User defined datatypes let you extend the base SQL Server datatypes by providing a descriptive name, and format to the database. Take for example, in your database, there is a column called Flight\_Num which appears in many tables. In all these tables it should be varchar(8). In this case you could create a user defined datatype called Flight\_num\_type of varchar(8) and use it across all your tables. See sp\_addtype, sp\_droptype in books online.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is bit datatype and what’s the information that can be stored inside a bit column?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Bit datatype is used to store boolean information like 1 or 0 (true or false). Untill SQL Server 6.5 bit datatype could hold either a 1 or 0 and there was no support for NULL. But from SQL Server 7.0 onwards, bit datatype can represent a third state, which is NULL.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• Define candidate key, alternate key, composite key.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o A candidate key is one that can identify each row of a table uniquely. Generally a candidate key becomes the primary key of the table. If the table has more than one candidate key, one of them will become the primary key, and the rest are called alternate keys. A key formed by combining at least two or more columns is called composite key.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are defaults? Is there a column to which a default can’t be bound?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o A default is a value that will be used by a column, if no value is supplied to that column while inserting data. IDENTITY columns and timestamp columns can’t have defaults bound to them. See CREATE DEFAULT in books online.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is a transaction and what are ACID properties?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o A transaction is a logical unit of work in which, all the steps must be performed or none. ACID stands for Atomicity, Consistency, Isolation, Durability. These are the properties of a transaction. For more information and explanation of these properties, see SQL Server books online or any RDBMS fundamentals text book. Explain different isolation levels An isolation level determines the degree of isolation of data between concurrent transactions. The default SQL Server isolation level is Read Committed. Here are the other isolation levels (in the ascending order of isolation): Read Uncommitted, Read Committed, Repeatable Read, Serializable. See SQL Server books online for an explanation of the isolation levels. Be sure to read about SET TRANSACTION ISOLATION LEVEL, which lets you customize the isolation level at the connection level. Read Committed - A transaction operating at the Read Committed level cannot see changes made by other transactions until those transactions are committed. At this level of isolation, dirty reads are not possible but nonrepeatable reads and phantoms are possible. Read Uncommitted - A transaction operating at the Read Uncommitted level can see uncommitted changes made by other transactions. At this level of isolation, dirty reads, nonrepeatable reads, and phantoms are all possible. Repeatable Read - A transaction operating at the Repeatable Read level is guaranteed not to see any changes made by other transactions in values it has already read. At this level of isolation, dirty reads and nonrepeatable reads are not possible but phantoms are possible. Serializable - A transaction operating at the Serializable level guarantees that all concurrent transactions interact only in ways that produce the same effect as if each transaction were entirely executed one after the other. At this isolation level, dirty reads, nonrepeatable reads, and phantoms are not possible.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• CREATE INDEX myIndex ON myTable(myColumn)What type of Index will get created after executing the above statement?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Non-clustered index. Important thing to note: By default a clustered index gets created on the primary key, unless specified otherwise.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What’s the maximum size of a row?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o 8060 bytes. Don’t be surprised with questions like ‘what is the maximum number of columns per table’. 1024 columns per table. Check out SQL Server books online for the page titled: "Maximum Capacity Specifications". Explain Active/Active and Active/Passive cluster configurations Hopefully you have experience setting up cluster servers. But if you don’t, at least be familiar with the way clustering works and the two clusterning configurations Active/Active and Active/Passive. SQL Server books online has enough information on this topic and there is a good white paper available on Microsoft site. Explain the architecture of SQL Server This is a very important question and you better be able to answer it if consider yourself a DBA. SQL Server books online is the best place to read about SQL Server architecture. Read up the chapter dedicated to SQL Server Architecture.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is lock escalation?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Lock escalation is the process of converting a lot of low level locks (like row locks, page locks) into higher level locks (like table locks). Every lock is a memory structure too many locks would mean, more memory being occupied by locks. To prevent this from happening, SQL Server escalates the many fine-grain locks to fewer coarse-grain locks. Lock escalation threshold was definable in SQL Server 6.5, but from SQL Server 7.0 onwards it’s dynamically managed by SQL Server.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What’s the difference between DELETE TABLE and TRUNCATE TABLE commands?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o DELETE TABLE is a logged operation, so the deletion of each row gets logged in the transaction log, which makes it slow. TRUNCATE TABLE also deletes all the rows in a table, but it won’t log the deletion of each row, instead it logs the deallocation of the data pages of the table, which makes it faster. Of course, TRUNCATE TABLE can be rolled back. TRUNCATE TABLE is functionally identical to DELETE statement with no WHERE clause: both remove all rows in the table. But TRUNCATE TABLE is faster and uses fewer system and transaction log resources than DELETE. The DELETE statement removes rows one at a time and records an entry in the transaction log for each deleted row. TRUNCATE TABLE removes the data by deallocating the data pages used to store the table’s data, and only the page deallocations are recorded in the transaction log. TRUNCATE TABLE removes all rows from a table, but the table structure and its columns, constraints, indexes and so on remain. The counter used by an identity for new rows is reset to the seed for the column. If you want to retain the identity counter, use DELETE instead. If you want to remove table definition and its data, use the DROP TABLE statement. You cannot use TRUNCATE TABLE on a table referenced by a FOREIGN KEY constraint; instead, use DELETE statement without a WHERE clause. Because TRUNCATE TABLE is not logged, it cannot activate a trigger. TRUNCATE TABLE may not be used on tables participating in an indexed view

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• Explain the storage models of OLAP

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Check out MOLAP, ROLAP and HOLAP in SQL Server books online for more infomation.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are the new features introduced in SQL Server 2000 (or the latest release of SQL Server at the time of your interview)? What changed between the previous version of SQL Server and the current version?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o This question is generally asked to see how current is your knowledge. Generally there is a section in the beginning of the books online titled "What’s New", which has all such information. Of course, reading just that is not enough, you should have tried those things to better answer the questions. Also check out the section titled "Backward Compatibility" in books online which talks about the changes that have taken place in the new version.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are constraints? Explain different types of constraints.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Constraints enable the RDBMS enforce the integrity of the database automatically, without needing you to create triggers, rule or defaults. Types of constraints: NOT NULL, CHECK, UNIQUE, PRIMARY KEY, FOREIGN KEY. For an explanation of these constraints see books online for the pages titled: "Constraints" and "CREATE TABLE", "ALTER TABLE"

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is an index? What are the types of indexes? How many clustered indexes can be created on a table? I create a separate index on each column of a table. What are the advantages and disadvantages of this approach?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Indexes in SQL Server are similar to the indexes in books. They help SQL Server retrieve the data quicker. Indexes are of two types. Clustered indexes and non-clustered indexes. When you create a clustered index on a table, all the rows in the table are stored in the order of the clustered index key. So, there can be only one clustered index per table. Non-clustered indexes have their own storage separate from the table data storage. Non-clustered indexes are stored as B-tree structures (so do clustered indexes), with the leaf level nodes having the index key and it’s row locater. The row located could be the RID or the Clustered index key, depending up on the absence or presence of clustered index on the table. If you create an index on each column of a table, it improves the query performance, as the query optimizer can choose from all the existing indexes to come up with an efficient execution plan. At the same t ime, data modification operations (such as INSERT, UPDATE, DELETE) will become slow, as every time data changes in the table, all the indexes need to be updated. Another disadvantage is that, indexes need disk space, the more indexes you have, more disk space is used.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is RAID and what are different types of RAID configurations?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o RAID stands for Redundant Array of Inexpensive Disks, used to provide fault tolerance to database servers. There are six RAID levels 0 through 5 offering different levels of performance, fault tolerance. MSDN has some information about RAID levels and for detailed information, check out the RAID advisory board’s homepage

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are the steps you will take to improve performance of a poor performing query?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o This is a very open ended question and there could be a lot of reasons behind the poor performance of a query. But some general issues that you could talk about would be: No indexes, table scans, missing or out of date statistics, blocking, excess recompilations of stored procedures, procedures and triggers without SET NOCOUNT ON, poorly written query with unnecessarily complicated joins, too much normalization, excess usage of cursors and temporary tables. Some of the tools/ways that help you troubleshooting performance problems are: SET SHOWPLAN\_ALL ON, SET SHOWPLAN\_TEXT ON, SET STATISTICS IO ON, SQL Server Profiler, Windows NT /2000 Performance monitor, Graphical execution plan in Query Analyzer. Download the white paper on performance tuning SQL Server from Microsoft web site. Don’t forget to check out sql-server-performance.com

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are the steps you will take, if you are tasked with securing an SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Again this is another open ended question. Here are some things you could talk about: Preferring NT authentication, using server, databse and application roles to control access to the data, securing the physical database files using NTFS permissions, using an unguessable SA password, restricting physical access to the SQL Server, renaming the Administrator account on the SQL Server computer, disabling the Guest account, enabling auditing, using multiprotocol encryption, setting up SSL, setting up firewalls, isolating SQL Server from the web server etc. Read the white paper on SQL Server security from Microsoft website. Also check out My SQL Server security best practices

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is a deadlock and what is a live lock? How will you go about resolving deadlocks?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Deadlock is a situation when two processes, each having a lock on one piece of data, attempt to acquire a lock on the other’s piece. Each process would wait indefinitely for the other to release the lock, unless one of the user processes is terminated. SQL Server detects deadlocks and terminates one user’s process. A livelock is one, where a request for an exclusive lock is repeatedly denied because a series of overlapping shared locks keeps interfering. SQL Server detects the situation after four denials and refuses further shared locks. A livelock also occurs when read transactions monopolize a table or page, forcing a write transaction to wait indefinitely. Check out SET DEADLOCK\_PRIORITY and "Minimizing Deadlocks" in SQL Server books online. Also check out the article Q169960 from Microsoft knowledge base.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is blocking and how would you troubleshoot it?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Blocking happens when one connection from an application holds a lock and a second connection requires a conflicting lock type. This forces the second connection to wait, blocked on the first. Read up the following topics in SQL Server books online: Understanding and avoiding blocking, Coding efficient transactions. Explain CREATE DATABASE syntax Many of us are used to creating databases from the Enterprise Manager or by just issuing the command: CREATE DATABAE MyDB.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• But what if you have to create a database with two filegroups, one on drive C and the other on drive D with log on drive E with an initial size of 600 MB and with a growth factor of 15%?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o That’s why being a DBA you should be familiar with the CREATE DATABASE syntax. Check out SQL Server books online for more information.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• How to restart SQL Server in single user mode? How to start SQL Server in minimal configuration mode?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o SQL Server can be started from command line, using the SQLSERVR.EXE. This EXE has some very important parameters with which a DBA should be familiar with. -m is used for starting SQL Server in single user mode and -f is used to start the SQL Server in minimal configuration mode. Check out SQL Server books online for more parameters and their explanations.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• As a part of your job, what are the DBCC commands that you commonly use for database maintenance?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o DBCC CHECKDB, DBCC CHECKTABLE, DBCC CHECKCATALOG, DBCC CHECKALLOC, DBCC SHOWCONTIG, DBCC SHRINKDATABASE, DBCC SHRINKFILE etc. But there are a whole load of DBCC commands which are very useful for DBAs. Check out SQL Server books online for more information.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are statistics, under what circumstances they go out of date, how do you update them?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Statistics determine the selectivity of the indexes. If an indexed column has unique values then the selectivity of that index is more, as opposed to an index with non-unique values. Query optimizer uses these indexes in determining whether to choose an index or not while executing a query. Some situations under which you should update statistics: 1) If there is significant change in the key values in the index 2) If a large amount of data in an indexed column has been added, changed, or removed (that is, if the distribution of key values has changed), or the table has been truncated using the TRUNCATE TABLE statement and then repopulated 3) Database is upgraded from a previous version. Look up SQL Server books online for the following commands: UPDATE STATISTICS, STATS\_DATE, DBCC SHOW\_STATISTICS, CREATE STATISTICS, DROP STATISTICS, sp\_autostats, sp\_createstats, sp\_updatestats

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are the different ways of moving data/databases between servers and databases in SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o There are lots of options available, you have to choose your option depending upon your requirements. Some of the options you have are: BACKUP/RESTORE, dettaching and attaching databases, replication, DTS, BCP, logshipping, INSERT…SELECT, SELECT…INTO, creating INSERT scripts to generate data.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• Explain different types of BACKUPs avaialabe in SQL Server? Given a particular scenario, how would you go about choosing a backup plan?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Types of backups you can create in SQL Sever 7.0+ are Full database backup, differential database backup, transaction log backup, filegroup backup. Check out the BACKUP and RESTORE commands in SQL Server books online. Be prepared to write the commands in your interview. Books online also has information on detailed backup/restore architecture and when one should go for a particular kind of backup.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is database replication? What are the different types of replication you can set up in SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Replication is the process of copying/moving data between databases on the same or different servers. SQL Server supports the following types of replication scenarios: ? Snapshot replication ? Transactional replication (with immediate updating subscribers, with queued updating subscribers) ? Merge replication See SQL Server books online for indepth coverage on replication. Be prepared to explain how different replication agents function, what are the main system tables used in replication etc.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• How to determine the service pack currently installed on SQL Server?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o The global variable @@Version stores the build number of the sqlservr.exe, which is used to determine the service pack installed. To know more about this process visit SQL Server service packs and versions.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are cursors? Explain different types of cursors. What are the disadvantages of cursors? How can you avoid cursors?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Cursors allow row-by-row processing of the resultsets. Types of cursors: Static, Dynamic, Forward-only, Keyset-driven. See books online for more information. Disadvantages of cursors: Each time you fetch a row from the cursor, it results in a network roundtrip, where as a normal SELECT query makes only one roundtrip, however large the resultset is. Cursors are also costly because they require more resources and temporary storage (results in more IO operations). Further, there are restrictions on the SELECT statements that can be used with some types of cursors. Most of the times, set based operations can be used instead of cursors. Here is an example: If you have to give a flat hike to your employees using the following criteria: Salary between 30000 and 40000 – 5000 hike Salary between 40000 and 55000 – 7000 hike Salary between 55000 and 65000 – 9000 hike. In this situation many developers tend to use a cursor, determine each employee’s salary and update his salary according to the above formula. But the same can be achieved by multiple update statements or can be combined in a single UPDATE statement as shown below:

o UPDATE tbl\_emp SET salary = CASE WHEN salary BETWEEN 30000 AND 40000 THEN salary + 5000 WHEN salary BETWEEN 40000 AND 55000 THEN salary + 7000 WHEN salary BETWEEN 55000 AND 65000 THEN salary + 10000 END

o Another situation in which developers tend to use cursors: You need to call a stored procedure when a column in a particular row meets certain condition. You don’t have to use cursors for this. This can be achieved using WHILE loop, as long as there is a unique key to identify each row. For examples of using WHILE loop for row by row processing, check out the ‘My code library’ section of my site or search for WHILE. Write down the general syntax for a SELECT statements covering all the options. Here’s the basic syntax: (Also checkout SELECT in books online for advanced syntax).

o SELECT select\_list [INTO new\_table\_] FROM table\_source [WHERE search\_condition] [GROUP BY group\_by\_expression] [HAVING search\_condition] [ORDER BY order\_expression [ASC | DESC] ]

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is a join and explain different types of joins.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Joins are used in queries to explain how different tables are related. Joins also let you select data from a table depending upon data from another table. Types of joins: INNER JOINs, OUTER JOINs, CROSS JOINs. OUTER JOINs are further classified as LEFT OUTER JOINS, RIGHT OUTER JOINS and FULL OUTER JOINS. For more information see pages from books online titled: "Join Fundamentals" and "Using Joins".

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• Can you have a nested transaction?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Yes, very much. Check out BEGIN TRAN, COMMIT, ROLLBACK, SAVE TRAN and @@TRANCOUNT

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is an extended stored procedure? Can you instantiate a COM object by using T-SQL?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o An extended stored procedure is a function within a DLL (written in a programming language like C, C++ using Open Data Services (ODS) API) that can be called from T-SQL, just the way we call normal stored procedures using the EXEC statement. See books online to learn how to create extended stored procedures and how to add them to SQL Server. Yes, you can instantiate a COM (written in languages like VB, VC++) object from T-SQL by using sp\_OACreate stored procedure. Also see books online for sp\_OAMethod, sp\_OAGetProperty, sp\_OASetProperty, sp\_OADestroy. For an example of creating a COM object in VB and calling it from T-SQL, see ‘My code library’ section of this site.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is the system function to get the current user’s user id?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o USER\_ID(). Also check out other system functions like USER\_NAME(), SYSTEM\_USER, SESSION\_USER, CURRENT\_USER, USER, SUSER\_SID(), HOST\_NAME().

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What are triggers? How many triggers you can have on a table? How to invoke a trigger on demand?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Triggers are special kind of stored procedures that get executed automatically when an INSERT, UPDATE or DELETE operation takes place on a table. In SQL Server 6.5 you could define only 3 triggers per table, one for INSERT, one for UPDATE and one for DELETE. From SQL Server 7.0 onwards, this restriction is gone, and you could create multiple triggers per each action. But in 7.0 there’s no way to control the order in which the triggers fire. In SQL Server 2000 you could specify which trigger fires first or fires last using sp\_settriggerorder. Triggers can’t be invoked on demand. They get triggered only when an associated action (INSERT, UPDATE, DELETE) happens on the table on which they are defined. Triggers are generally used to implement business rules, auditing. Triggers can also be used to extend the referential integrity checks, but wherever possible, use constraints for this purpose, instead of triggers, as constraints are much faster. Till SQL Server 7.0, triggers fire only after the data modification operation happens. So in a way, they are called post triggers. But in SQL Server 2000 you could create pre triggers also. Search SQL Server 2000 books online for INSTEAD OF triggers. Also check out books online for ‘inserted table’, ‘deleted table’ and COLUMNS\_UPDATED()

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• There is a trigger defined for INSERT operations on a table, in an OLTP system. The trigger is written to instantiate a COM object and pass the newly insterted rows to it for some custom processing. What do you think of this implementation? Can this be implemented better?

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Instantiating COM objects is a time consuming process and since you are doing it from within a trigger, it slows down the data insertion process. Same is the case with sending emails from triggers. This scenario can be better implemented by logging all the necessary data into a separate table, and have a job which periodically checks this table and does the needful.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• What is a self join? Explain it with an example.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

o Self join is just like any other join, except that two instances of the same table will be joined in the query. Here is an example: Employees table which contains rows for normal employees as well as managers. So, to find out the managers of all the employees, you need a self join.

o CREATE TABLE emp ( empid int, mgrid int, empname char(10) )

o INSERT emp SELECT 1,2,’Vyas’ INSERT emp SELECT 2,3,’Mohan’ INSERT emp SELECT 3,NULL,’Shobha’ INSERT emp SELECT 4,2,’Shridhar’ INSERT emp SELECT 5,2,’Sourabh’

o SELECT t1.empname [Employee], t2.empname [Manager] FROM emp t1, emp t2 WHERE t1.mgrid = t2.empid Here’s an advanced query using a LEFT OUTER JOIN that even returns the employees without managers (super bosses)

o SELECT t1.empname [Employee], COALESCE(t2.empname, ‘No manager’) [Manager] FROM emp t1 LEFT OUTER JOIN emp t2 ON t1.mgrid = t2.empid

<http://vyaskn.tripod.com/iq.htm#top#top>

<http://www.tkdinesh.com/faq/ans/orderbycase.html>

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

~~~ StartRegion -- Questions ~~~

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~