WIKICONSOLE Home Authors Middleware OS Databases SAP DWH Tools Submit Article Forums

Sybase Interview Questions and Answers

Home / Interview Questions / Other Questions / Sybase Interview Questions and Answers

‹ Previous Next ›

Sybase Interview Questions and Answers

1. What are ***Fast BCP*** and ***Slow BCP***?
    - Fast BCP: – when you drop the triggers, indexes from the table and run the BCP command and recreate the same then that is so called FAST BCP. This BCP is fast because it need not do the checking of triggers and update the indexes.
    - Slow BCP: – Without doing any changes if you run the BCP then that is so called SLOW BCP

2. What is view? What is use of view?

Views are a logical way of looking at the physical data located in the tables. A view does not represent any physical data. Dropping a view has no effect on underlying tables. But insert and update on view affects the base tables.

Use: –

1. Restrict data access to users: – You can give access to selected columns (vertical restriction) or selected rows (horizontal restriction).
2. Ease of SQL query: – You can create a view on complex but frequently used queries so that users can access the result using simple select * from view.

4. Can I update a View if I am using 2 tables?
   - Yes you can. You can update columns of only one table using a single update statement. i.e. columns to be updated are from different tables, you have to use multiple update statements – one for each table.
   - View should not contain distinct clause or aggregate functions.

5. Can I delete from a View?
   - Yes you can. A single delete statement deletes rows from multiple base tables.
   - View should not contain distinct clause or aggregate functions.

6. Can you insert in a View if it is created on multiple base tables?
   - Yes you can. Columns being inserted into the view should belong to a single base table.
   - All the columns in the base tables, which are not included in the view, should be defined to allow null or should have a default values defined.
   - View should not contain distinct clause or aggregate functions.

7. While inserting records through a View, what ***precautions will you take***?

Explained above

8. Where does the ***text of a View creation stored***?
   - In syscomments system table

9. How will you ***see the definition of a View***?

- sp_helptext ‘NameOfTheView’

10. What are the limitations of Views? (can’t use order by, select into, union, compute by)

11. Can I use group by in views? Can I use compute by clause in a View?

You can have group by clause but you can not have compute/compute by clause in view.

12. What if *with check option* in view?

The with check option flag prevents insertion or updating of rows that will not meet the view criteria.

e.g.:

```
create view cal_publishers_chk as

select *

        from publishers

where state = “CA”

with check option
```

Update fails because, after the update, the modified rows would fail to appear in the view.

13. Tell about 5 system tables?
    - **sysobjects :-** all the objects in the database
    - **syscolumns :-** all columns of tables
    - **sysindexes :-** Index details
    - **syscomments :-** comments and texts of stored procedure, view, trigger, rule, default and other database objects.
    - **sysdepends :-** details of a objects which depends on which table or stored procedures
    - **sysprotects :-** permission list to a particular object to users or alias
    - **sysusers :-** user and alias details in the database.
    - **sysprocedures :-** Details of procedure, defaults, rules, trigger and views

14. What is stored in *sysprocedures*?
    - Details of procedure, defaults, rules, trigger and views

15. What is ***normalization***?

Normalization in a relational database is an approach to structuring information in order to avoid redundancy and inconsistency. This is done to promote efficient maintenance, storage, and updating.

There are three main rules or levels of normal forms. First, Second and Third.

First normal forms, the least structured, are groups of records in which each field (column) contains unique and non-repeating information. Second and third normal forms break down first normal forms, separating them into different tables by defining successively finer interrelationships between fields.

16. What is the need for ***de-normalization***?

- Nearly all of the most frequent queries require access to the full set of joined data.
- A majority of applications perform table scans when joining tables.
- Computational complexity of derived columns requires temporary tables or excessively complex queries.

17. What are triggers? Why do we use them? Diff types of triggers? What are magic tables?

- A **trigger** is a stored procedure bound to a table, which is implicitly executed when a user attempts to modify data with an insert, delete, or update command.
    - Use: –

1. i. Validation purpose
2. ii. Referential integrity

- Type of trigger: – update/delete/insert
- Two tables are created at the time of trigger execution 1. Inserted 2. Deleted which are called magic tables.

18. Can I commit transactions in a Trigger?

No

19. What is the concept of ***magic-tables***?

Statement	Inserted	Deleted
Insert	New row being inserted	Blank
Update	New values of the row being updated	Old values of the row being updated
Delete	Blank	Row being deleted

20. What is the ***depth of Triggers***?
    - 16 Levels i.e. 16 nested triggers can be there in Sybase

21. How do you check that the *depth of the Trigger* doesn't *cross the limit*?
    - Just check the @@nestlevel global variable for the same

22. Can u call triggers explicitly? Can u disable triggers?

No you cannot, no you cannot in Sybase

23. I am having a table with no rows and an update trigger on it. If I fire an update statement, will the trigger be fired?

yes

24. What are stored procedures?

25. What are the advantages and disadvantages of S.P? What are diff parameters we can use in SP? Can I use my output parameter as an input parameter in Sybase?

Advantages: –

- Fast execution: – Stored procedures, after their first execution, become memory resident and do not need to be reparsed, reoptimized, or recompiled. So they are faster.
- Reduced network traffic: – As they are memory resident, less SQL need to cross busy network lines.
- Modular Programming: – A way to brake up the logic in small modules.
- Restricted, function based access to tables: – You can grant permissions in such a way as to allow a user access to tables only through the stored procedures.
- Enforced consistency: – If users are accessing tables only through stored procedures, ad-hoc modifications can be easily avoided.

Disadvantages: –

- A stored procedure can not create views, defaults, rules or triggers.
- You can create and use temporary tables. These tables are dropped at procedure termination. A table can’t be created, dropped and re-created with the same name in a single procedure.
- Stored procedures are parsed in singe pass and will not resolve forward or backward references. I.e. a stored procedure must create temporary table before referencing it, or it must exist at the time of stored procedure creation.

- Types of Parameters: – Input and Output

- Maximum number of parameters: – 255
- A single parameter can’t be input as well as output.

OR

How will you ***debug a SP***?

- Debug it by using print /select statement.

27. Can u rename a column name/ table name? If yes, then will the S.P Work?

Yes you can and the SP will work fine. But if you try to drop and recreate the procedure with same code then it will give error if column name is hard coded.

28. What is ***roll-forward***?
    - **(Ans:- when the server starts it see whether a checkpoint has been fired or not. If the it found the check-point it means data has to write into the physical device, this is so called roll-forward)**

29. How will you tell the ***row-size***?
    - Row size of a table can be estimated for that you have to know the storing structure of Sybase means what space Sybase takes to store fixed length column or variable length column. Ex varchar, char, int, …..

```
 Select

TableName = obj.name,

RowSize   =

Case col.status

when 0 then (sum(col.length))
```

```
from sysobjects obj,syscolumns col

where obj.id = col.id

and   obj.type = ‘U’

–and id = object_id(‘NameOfTheTable’)

group by obj.name,col.status
```

you can add one more clause for fix and variable length of columns but please do it by you own.

OR

Sp_spaceused table_name get the space used by the table say 2GB.

Select count(*) from table_name say 30000 Rows

Now the size of the column can be :- 2GB/30000 Rows = 70 KB (approx)

30. How will you ***optimize a query***?
    - Check the query plan.
    - Make sure that the preceding columns of index are available in search criteria.
    - Make sure the Statistics are up-to-date: – Run update statistics on the tables in question if the data in these tables is changed drastically.
    - Check Search Arguments: – If possible avoid inequality operator. Avoid operations on columns and constant expressions that can not be evaluated at query compile time.
    - If possible, use index covering.
    - If stored procedure accepts parameters, it is better to recompile it at every execution.
    - Check the indexes are on proper columns.

31. What is the diff. between a ***sub-query*** and a ***correlated sub-query***?

Sub-query: – A **sub-query** is a select statement that is nested inside another select, insert, update, or delete statement, inside a conditional statement.

Correlated sub-query: – The sub-query, which depends on the outer query for its values, is called as correlated sub-query.

- The *subquery_select_list* can consist of only one column name, except in the 'exists' sub-query, where an (*) is usually used in place of the single column name.
- text and image datatypes are not allowed in sub-queries.
- Sub-query cannot include the order by clause, the compute clause,
- There is a limit of 16 nesting levels.
- The maximum number of sub-queries on each side of a union is 16.

32. If I have a main query and a sub-query, which would execute first?

If a sub query is non-correlated then it will be executed first if it is a correlated sub-query then that will be executed in parallel with the main query

33. What is an ***Index***? How many types of indexes are there?

Index is used to provide faster access to the data in a table than scanning every page. Sometimes index can be used as mechanism for enforcing uniqueness.

There are 2 types of indexes: Clustered and Non clustered

34. What is the diff. Between clustered and non-clustered indexes?

contains pointers to rows on data pages. More precisely, each leaf page contains an indexed value and a pointer to the row with that value. Up to 249 non-clustered indexes are allowed in a table.

35. Why are there ***only 249 non-clustered Indexes*** in SQL-Server? Explain.

Values defined for indid (Index ID) column in sysindexes table are as follows:

0 – table does not have clustered index

1 – clustered index

>1 and <= 250 (i.e. **249**) – non clustered indexes

255 – text or image data.

36. What is ***Index Covering***?

When all the keys of non-clustered index are used in select statement and also in where clause then the non-clustered index does not need to go to last level of index it just pick the data from leaf-level so it takes less time to execute the query this is called Index-Covering.

37. How can you ***force an Index***?
    - Just write the name or the index number in the bracket to force the index.

38. What is Isolation? How many Isolation levels are there? Explain?

levels include the restrictions imposed by the lower levels.

- Global variable: – @@isolation
- Command: – set transaction isolation level
- **Level 0 (READ UNCOMMITTED)**: – Allows dirty reads. Such reads occur when one transaction modifies a row, and a second transaction reads that row before the first transaction commits the change. If the first transaction rolls back the change, the information read by the second transaction becomes invalid. Though second transaction is allowed to read the data, it prevents from changing the data that has already been modified by the first uncommitted transaction

To allow this SQL Server does not attempt to acquire read locks on data pages for a select, so the select command is not blocked by any exclusive locks. In addition, as no read locks are acquired, no update operations attempting to acquire exclusive locks are blocked by the read operation.

- **Level 1 (READ COMMITTED)**: – Prevents dirty reads. If one transaction is modifying the data, no other transaction is allowed to read or change the same unless the first transaction is either committed or rolled back. This is the **default isolation level supported** by Adaptive Server.
- **Level 2 (REPEATABLE READS)**: – allows a single page to be read for many times within the same transaction and guarantees that the same value is read each time. This option prevents other transactions from updating a data row that has been read until the transaction in which it was read is committed or rolled back.
- **Level 3 (SERIALIZABLE READS)**: – ensures that data read by one transaction is valid until the end of that transaction, hence preventing phantom rows. Adaptive Server supports this level through the holdlock keyword of the select statement, which applies a read-lock on the specified data. Phantom rows occur when one transaction reads a set of rows that satisfy a search condition, and then a second transaction modifies the data (through an insert, delete, update, and so on). If the first transaction repeats the read with the same search conditions, it obtains a different set of rows.

39. What are diff. ***types of locks***, Explain?
    - **Shared Lock**: – used by process that is reading pages. Multiple shared locks can be held on one page; a single shared lock prevents any exclusive lock from being acquired. Shared locks typically are held only for the duration of the read on a particular page. If holdlock is specified, shared locks are held until the completion of the command or the transaction it is in.
    - **Update lock**: – used by process that updates or deletes data, but have not yet done so. Update locks are acquired as the set of rows to be updated or deleted is being determined. Update locks are read-compatible with shared locks during the pre modification phase, but no other update lock or exclusive lock can be acquired on the page. Update locks automatically are updated to exclusive locks when the data change occurs.
    - **Exclusive Lock**: – used by process that currently adding, changing or deleting information on data pages. Exclusive locks prevent any other type of lock (exclusive, update or shared) from being acquired. Exclusive locks are held on all affected pages until an explicit transaction or a command in the implicit transaction is complete.

- Lock acquired by update or delete statement at the time determining rows to be updated or deleted by a particular query.

41. What is deadlock? How SQL Server handles it? How to avoid deadlock?

Deadlock is the situation when two different user connections fight for the same resources. One user connection acquires a lock on a particular page. The next step is to acquire a lock on the next affected by the transaction. However, a different user connection has already locked that page and is requesting a lock on the page the first user connection has locked. In this situation, neither user connection can continue until the other has continued. This is deadlock.

The SQL Server continually checks for deadlocks and automatically aborts one of the conflicting transactions. The transaction that is aborted is the one belonging to the process that has accumulated the least amount of CPU time since the start of its session, not the amount of CPU time for the specific transaction.

Methods for reducing lock contention, such as **changing the locking scheme**, **avoiding table locks**, and **not holding shared locks**

Some of other methods are

Acquire locks on objects in the same order

Well-designed applications can minimize deadlocks by always acquiring locks in the same order. Updates to multiple tables should always be performed in the same order. In applications with large numbers of tables and transactions that update several tables, establish a locking order that can be shared by all application developers.

Delaying deadlock checking

Adaptive Server performs deadlock checking after a minimum period of time for any process waiting for a lock to be released (sleeping). This deadlock checking is time-consuming overhead for applications that wait without a deadlock. If your applications deadlock infrequently, Adaptive Server can delay deadlock checking and reduce the overhead cost. You can specify the minimum amount of time (in milliseconds) that a process waits before it initiates a deadlock check using the configuration parameter deadlock checking period. Valid values are 0–2147483. The default value is 500. deadlock checking period is a dynamic configuration value, so any change to it takes immediate effect. If you set the value to 0, Adaptive Server initiates deadlock checking when the process begins to wait for a lock. If you set the value to 600, Adaptive Server initiates a deadlock check for the waiting process after at least 600 ms. For example:

Setting deadlock checking period to a higher value produces longer delays before deadlocks are detected. However, since Adaptive Server grants most lock requests before this time elapses, the deadlock checking overhead is avoided for those lock requests. Adaptive Server performs deadlock checking for all processes at fixed intervals, determined by deadlock checking period. If Adaptive Server performs a deadlock check while a process's deadlock checking is delayed, the process waits until the next interval. Therefore, a process may wait from the number of milliseconds set by deadlock checking period to almost twice that value before deadlock checking is performed. sp_sysmon can help you tune deadlock checking behavior.

42. How many types of locks are there? (3, Shared , Update, Exclusive)

43. How many levels of locking is there? (2, Table level, Column level) How does the server change the levels? (lock escalation)

Adaptive Server has two levels of locking:

- For tables that use allpages locking or datapages locking, either page

locks or table locks.

- For tables that use datarows locking, either row locks or table locks

Page or row locks are less restrictive (or smaller) than table locks. A page lock locks all the rows on data page or an index page; a table lock locks an entire table. A row lock locks only a single row on a page. Adaptive Server uses page or row locks whenever possible to reduce contention and to improve concurrency.

Adaptive Server uses a table lock to provide more efficient locking when an entire table or a large number of pages or rows will be accessed by a statement.

Page and row locks

The following describes the types of page and row locks:

- *Shared locks :* Adaptive Server applies **shared loc**ks for read operations. If a shared lock has been applied to a data page or data row or to an index page, other transactions can also acquire a shared lock, even when the first transaction is active. However, no transaction can acquire an exclusive lock on the page or row until all shared locks on the page or row are released. This means that many transactions can simultaneously read the page or row, but no transaction can change data on the page or row while a shared lock exists. Transactions that need an exclusive lock wait or "block" for the release of the shared locks before continuing. By default, Adaptive Server releases shared locks after it finishes scanning the page or row. It does not hold shared locks until the statement is completed or until the end of the transaction unless requested to do so by the user. For more details on how shared locks are applied.
- *Exclusive locks :* Adaptive Server applies an **exclusive lock** for a data modification operation. When a transaction gets an exclusive lock, other transactions cannot acquire a lock of any kind on the page or row until the exclusive lock is released at the end of its transaction. The other transactions wait or "block" until the exclusive lock is released.
- *Update locks :* Adaptive Server applies an **update lock** during the initial phase of an update, delete, or fetch (for cursors declared for update) operation while the page or row is being read. The update lock allows shared locks on the page or row, but does not allow other update or exclusive locks. Update locks help avoid deadlocks and lock contention. If the page or row needs to be changed, the update lock is promoted to an exclusive lock as soon as no other shared locks exist on the page or row. In general, read operations acquire shared locks, and write operations acquire exclusive locks. For operations that delete or update data, Adaptive Server applies page-level or row-level exclusive and update locks only if the column used in the search argument is part of an index. If no index exists on any of the search arguments, Adaptive Server must acquire a table-level lock.

Table locks

The following describes the types of table locks.

- *Intent lock :* An **intent lock** indicates that page-level or row-level locks are currently held on a table. Adaptive Server applies an intent table lock with each shared or exclusive page or row lock, so an intent lock can be either an exclusive lock or a shared lock. Setting an intent lock prevents other transactions from subsequently acquiring conflicting table-level locks on the table that contains that locked page. An intent lock is held as long as page or row locks are in effect for the transaction.

during a create clustered index command. update and delete statements require exclusive table locks if their search arguments do not reference indexed columns of the object.

44. What are global variables? Can I update them?

Global variable are Sybase defined variables. We cannot update the global variable.

45. Inside a SP what ***global variables*** are used? Explain them.
    - @@error : – Error number reported for last SQL statement
    - @@sqlstatus : – Status of previous fetch statement in a cursor
    - @@rowcount : – Number of rows processed by preceding command
    - @@trancount : – Transaction nesting level
    - @@transtate :- Current state of the transaction
    - @@tranchained : – Current transaction mode (1 – Chained, 0 – Unchained)
    - @@servername : – Name of the local SQL Server
    - @@version : – SQL Server and O/S release level
    - @@spid : – Current process ID
    - @@identity : – Last identity value used in an insert
    - @@nestlevel : – Nuber of levels nested in a stored procedure or trigger

46. Apart from the global variables you listed are there any other variables?
    - @@isolation : – Current isolation level

47. What are diff. ***Types of temporary tables***? What is the ***diff. Between them***?

user session or the stored procedure in which they are created.

- Work tables: – These tables are created by Sybase to resolve complex queries where in the data is taken to workable tables and then further manipulation is done on to it for achieving the result. These tables are dropped when query execution completes. User can’t access these tables.

48. What is a ***shared variable***?

49. What is the ***diff. Between Native driver and ODBC driver***?

50. Explain Union, Union all, what is the difference between them?

Used to combine results of two or more SQL statements provided that the datatypes of respective columns in all the queries match exactly or are implicitly convertible. ‘Union’ removes duplicate rows but ‘Union all’ does not.

51. What alternatives do you have for Union?

You can use the temp table. You can use the ‘OR’ clause in the SQL statement.

52. Can u use select into statement in a Union?

Possible if it is stated in the first select statement.

Eg:

```
Select * into tab1 from A1
```

53. I am having a table A with 500 rows, I write select * from A union select * from A, what will be the output? (i.e. How many rows will be displayed) ?

Same number of rows but if you use the union all command you will get the double number of records.

54. Can u change the data type of a column when there are records, without dropping the table?

In real scenario you can never update the data type of a column but in 11.9.x if you are having sa role, you can go to syscolumns and can update the type column without dropping the table or deleting the table but this is very dangerous because it may corrupt the Sybase storage files. Just think what will happen if you update the data type of char to int or vice-versa.

We can modify the datatype of a column in 12.5. You can only convert datatypes that are either implicitly or explicitly convertible to the new datatype, or if there is an explicit conversion function in Transact-SQL.

55. Can u drop a column if it is not the last column of the table?

Yes you can by same alter table.

56. If I need a few rows from the table, how can I best achieve it?

I – create a view and then call BCP

II – create a tempdb..table and then call BCP.

58. What is the diff between char and varchar data type?

Char is fix length data type and varchar is variable length data type.

59. What is the diff between money and numeric?

Money can store a number up to 4 decimal places only. Numeric can save a number up to 37 decimal places.

60. What is the meaning of precision and scale of a data type?

Precision is maximum number of digit you can store in column. We can have 1-38 precision the default is 18

Scale is the number of digit we can have to the right of decimal point. We can have 0-38 scale. The default is 0.

E.g.: declare @var numeric(10,4)

Here 10 is precision and 4 is scale. We can have 10 digits and 4 decimal points in this variable

61. What are the diff types of cursors? (Read only & Update)

62. What is a cursor? Why should we avoid it? How should we avoid it?

By means of cursor, we can address a set of rows individually, one row at a time. A cursor is a pointer that identifies a specific working row within a set.

63. Can I use temporary tables in cursors?

Yes.

```
Eg : create proc p1

as

begin

    declare @id int

    select * into #a from an1

    declare  c1 cursor  for select id from #a

    open c1

    fetch c1 into @id

    while @@sqlstatus != 2

    begin

        print ‘%1!’,@id

        fetch c1 into @id

    end

end
```

- To see the status of sql query executed.

65. Explain what are the diff values of @@ sqlstatus and what do they mean?

0	Successful fetching of row
1	Error while fetching the row
2	No more rows to fetch. Ie cursor pointer is on last row of result

66. When there is a compute by clause, can we have an order by clause in the query?

Yes you can.

67. What are joins? What are the diff types of Joins? (Equi, Outer, Self) (Full, Left, Right outer join)

68. What is the diff between equi join and natural join?

Joins based on equality (=) are called **equijoin**s. Equijoins compare the values in the columns being joined for equality and then include all the columns in the tables being joined in the results.

This query is an example of an equijoin:

select *

from authors, publishers

where authors.city = publishers.city

The query that results in the natural join of publishers and authors on the city column is:

```
select publishers.pub_id, publishers.pub_name,

publishers.state, authors.*

from publishers, authors

where publishers.city = authors.city
```

The column publishers.city does not appear in the results.

69. What are constraints/rules/defaults? What is the use?

All are used to maintain domain integrity.

- Rule: – Rule provide a mechanism for enforcing domain constraints for columns or user-defined datatypes. The rules are applied before an insert or update statement is executed.

You have to create a rule and then bind it to a column. To drop a rule you have to unbind it from all the columns. If you bind a rule to a column and there is an existing rule bound to a column, it is replaced by the new rule. Rules are not applied when you bulk copy data into the system.

- Defaults: -provide a value for a column when it is not supplied at the time of insert.

A default can be created as a database object and then bound to a column or it can be declared at the time of table creation. Only one default may be bound per column. If you try to bind a second default to a column, SQL server will give error. You can’t bind a default to a column that was created or altered to have a default value. The default is applied before a rule is checked. Defaults are applied during bulk copy.

- Constraints: – are used for defining data integrity requirements. There are 3 types

1. Primary key and unique constraint: – Unique constraints require that all non-null values be unique and allow a single null value for the column in the table.

1. Check Constraints: – Similar to rules. Can be defined at column level or table level. Multiple constraints can be associated with a single column. Check constraints can not compare column values to values in other tables, but can look at other columns in the current row (table level constraints). Check constraints can't contain aggregates.
2. Referential-integrity constraints: – This is implemented by declaring primary key and foreign key (references clause) at the time of table creation.

70. What is the diff between primary key and unique key?

Primary key does not allow null values. Unique key allows only one null value for the specified column in the table.

71. Can we create a non-clustered, non-unique index on the primary key?

alter table tt add constraint pk primary key nonclustered (c1)

72. What are constraints? Can I drop constraints? How?

Alter table <table-name> drop constraint <constraint-name>

73. How will I create a primary key constraint with non-clustered index ?(default is clustered)

alter table tt add constraint pk primary key nonclustered (c1)

74. What is the meaning of User-defined datatypes?

You can bind rules and defaults directly to user-defined datatypes. Any column created with such datatypes automatically inherits the rule and default but a rule or default bound explicitly to a column overrides a rule or default bound to a datatype.

User-defined datatype is not a database object, so not listed in *sysobjects*. It is listed in *systypes*.

75. Why is save tran used for?

To save the transaction if rollback occurred. Commit the transaction upto save point.

76. What is a transaction? What are diff modes of transaction?

A **transaction** is a set of Transact-SQL statements so that they are treated as a unit. Either all statements in the group are executed or no statements are executed. Adaptive Server automatically manages all data modification commands, including single-step change requests, as transactions. By default, each insert, update, and delete statement is considered a single transaction.

Adaptive Server supports the following transaction modes:

- The SQL standards-compatible mode, called **chained** mode, implicitly begins a transaction before any data retrieval or modification statement. These statements include: delete, insert, open, fetch, select, and update. You must still explicitly end the transaction with commit transaction or rollback transaction.
- The default mode, called **unchained** mode or Transact-SQL mode, requires explicit begin transaction statements paired with commit transaction or rollback transaction statements to complete the transaction. You can set either mode using the chained option of the set command. However, do not mix these transaction modes in your applications. The behavior of stored procedures and triggers can vary, depending on the mode, and you may require special action to run a procedure in one mode that was created in the other. The SQL standards require every SQL data-retrieval and data-modification statement to occur inside a transaction, using chained mode. A transaction automatically starts with the first data-retrieval or data-modification statement after the start of a session or after the previous transaction commits or aborts. This is the chained transaction mode. You can set this mode for your current session by turning on the chained option of the set statement:

transaction mode is unchained. In chained transaction mode, Adaptive Server implicitly executes a begin transaction statement just before the following data retrieval or modification statements: delete, insert, open, fetch, select, and update. For example, the following group of statements produce different results, depending on which mode you use:

In unchained transaction mode, the rollback affects only the delete statement, so publishers still contains the inserted row. In chained mode, the insert statement implicitly begins a transaction, and the rollback affects all statements up to the beginning of that transaction, including the insert. All application programs and ad hoc user queries should know their current transaction mode. Which transaction mode you use depends on whether or not a particular query or application requires compliance to the SQL standards. Applications that use chained transactions (for example, the Embedded SQL precompiler) should set chained mode at the beginning of each session.

77. How many types of Transactions are there? (Chained/Unchained)

78. When will Sp_recompile not work?

If object related to the table exists in different database then sp_recompile won't work. Sp_recompile will also not work for system tables

In the following cases sp_recompile will not work: –

A stored procedure contains select * statement. Alter table to add new columns. The procedure will not pick up new columns even if executed by using the with recompile option. This is because the old columns list is stored in parse tree. To pick up the new columns, a new parse tree should be created. The only way to do this is to drop and re-create the procedure.

A procedure refers customer table. Rename customer table as old_customer. Create new customer table with diff values. Procedure will continue to use old_customer table though re-compiled with sp_recompile option or with recompile option is used. This is because stored procedure resolves object references by object ID rather than name. To refer the new customer table, you have to either drop old_customer table (now the stored procedure will be compiled and a new parse tree is generated to refer new table) or drop and recreate the procedure (if you want to keep old_customer table)

80. Can I add a column? How? What things to be kept in mind?

1. Yes you can
2. Use alter table command
3. For alter table you must be the table owner or SA
4. New column should allow null values or to be not null default value should be specified.

81. I want to add a column in a table but it should be not null? Is it possible?

```
Ans:- create table tt

(

c1 int not null,

c2 int not null

)



insert tt values(2,2)



alter table tt

add c3 int default  0 not null
```

82. I have 1 million records and I will have 500 updates and deletes every day, Which Index will I use? Why?

83. How will you give access to a particular table only through a stored procedure?

Revoke all rights on of the table from all users and Grant rights for the procedure to the users. Hence now the table will be accessed by using procedure only

84. I have created a raiseerror message. I want to show that message in diff procedures or location. How will I do it?

Using sp_getmessage and then print statement you can get the user defined error message.

Sp_getmessage will get the userdefined message in an output variable.

Print will print the message.

Eg :-

```
sp_addmessage 25123 , “Message displaying one : ‘%1!’  two : ‘%2!’ “

go

declare @message varchar(200)

exec sp_getmessage 25123 ,  @message output

print  @message ,’one’,’two’
```

Message displaying one : one two : two

85. Why do we use group by?

If we want to have an vector aggregate value we use group by clause

(Vector aggregate :- Where we are getting the aggregated value in relation with some other field.

Eg. Select name_fs , sum(amount) from TableA group by name_fs

Scalar aggregate : This aggregate is independent value and only one value will be returned.

Eg. Select sum(amount) from TableA)

86. Can I have two different triggers on the same table for insert operations? One insert trigger inserts records into history table. One insert trigger updates records in other table. Is it possible? In that case which trigger gets executed?

You can define as many triggers as u want. But only the last defined trigger remain active for the table. If u drop that trigger, the previous latest should fire.

87. Is there anyway with which we can display the output of a single record in two lines. (ex.)

```
Emp ID                    Emp Name           Designation

_________-   _____________      ____________
```

E2	Santosh	SSE

Output what I want is:

Employee Name := Solomon

Employee Desig := SSE

____

Employee Name := Santosh

Employee Desig := SSE

____

So the Name in first line followed by Designation in next line and finally the "—-" characters in separate line.

Can this be done using a Select Statement and some special characters? NO CURSOR.

```
select ‘Employee Name = ‘+emp_name +char(13)+ ‘Employee Desig = ‘+emp_desg
from tmp1
```

88. Can I have a Select statement which has an aggregate function but does not have a “Group By” clause?

For ex:

Select type,count(*)

From titles

Select type,count(*)

From titles

Having count(*) > 1

Can such a statements work and give output.

It does not work in SQL as it does not accept aggregate functions without Group By clause. In Sybase, both will give same result but the result will be wrong.

89. I want to retrieve alternate records from a table:

Sample records are

Col1	col2

3	solomon
5	XMY
6	ABC
8	PQR
15	DEF

So the Output what I want is

Col1	col2
1	solo
5	XMY
8	PQR

How can we get that?

Ans: – select id,val

from temp4 a

90. Can I have an Update cursor without a unique record in the select statement

No. It will return an error saying Optimizer could not find a unique index.

91. What should be done to make a cursor statement return unique value without specifying any primary/unique key

Use distinct clause while declaring the cursor.

92. Can I get the showplan of a procedure without breaking it or without executing the procedure?

set showplan on

set fmtonly on

set exec off

93. I have got a 50,000 records table in tempdb and want to remove 10,000 records. Ii have got the where condition but when I try to delete it will full the tempdb log. Is there any way of deleting these records without making the log full?

Use set rowcount function. This will limit your number of rows.

94. I have got 10,00,000 records in a table and out of that got 1,00,000 duplicate records (complete row). So in all I have got 9,00,000 unique records in the table. Now I want to remove the duplicate from that table.

95. I have a table that has multiple records for a given entity, say for exp.

Value	Name	Address
1	ABC	xyz
0	ABC	pqr
0	ABC	fgh
0	DEF	rst
0	DEF	opq

There are two people actually ABC and DEF

ABC has 3 addresses and DEF has only one….

The 'value' specifies, which address is preferable. In this case even if ABC has 3 addresses since one of its value is 1(preferred), I want the query to return only that record that has the value 1 and its corresponding address. And if a person does not have any preferred address, then the first record should be displayed.

1	ABC	xyz
0	DEF	rst

I don’t want to use a cursor.

Ans: –

```
create table an1 ( id int , fn char(3) , sn char(3) )

insert an1 select 1 ,’ABC’,’xyz’

insert an1 select 0 ,’ABC’, ‘pqr’

insert an1 select 0, ‘ABC’,’fgh’

insert an1 select 0, ‘DEF’,’rst’

insert an1 select 0, ‘DEF’ ,’aaa’

insert an1 select 0, ‘D1F’ ,’1aa’

insert an1 select 0, ‘D1F’ ,’2aa’
```

```
alter table an1 add id1 int null



declare @ff char(3)

select @ff = ”

update an1 set id1 = case when @ff = fn then 0 else 1 end , @ff = case when @ff != fn then fn else @ff end

select * from an1 where id = 1 union  select * from an1 where id1 = 1 and id not in ( select 1 from an1 where id = 1 )
```

Some personal questions.

1. How will you scale yourself on writing syntax, on a scale of 1-10?
2. Tell me something about yourself.
3. Why do you want to leave your company?
4. Tell me the workflow of your firm.
5. Do you want to ask any questions from me?
6. What do you think about your Interview? Assess it.
7. If you are selected then are you confident that you will perform well with cursors, triggers, views etc.

@ Like 0 Share Tweet G+1 0 Share 0 submit

- Stur

By admin | February 25th, 2013 | Interview Questions, Other Questions | 0 Comments

Share This Story, Choose Your Platform!

About the Author: admin

Related Posts

Leave A Comment

You must be logged in to post a comment.

WIKICONSOLE

Welcome to wikiconsole...

SIGN IN

- Register
- Log in
- Entries RSS
- Comments RSS
- WordPress.org