**Interview Questions:**

1. Explain Current project and Roles/Responsibilities?
2. Explain Primary key and unique key?
3. What are indexes? Why we use it?
4. Difference between Temporary table and Table Variable.
5. How do you improve Query performance in SQL?
6. Write the SQL query based on given two employee related tables (Employee and Employee Salary).
7. Find the most recent salary of each employee
8. Inner join and Left Join
9. Second recent salary
10. What are Cursors?
11. Explain Polymorphism and Abstraction?
12. Explain new method and overriding?
13. What is the use of virtual and override?
14. What are Abstract Classes and Interfaces? Explain with real time example.
15. Write the program to find the even/odd numbers from list and display in given order.

C#

C#.NET

---------

- Explain .Net framework architecture.

- what are the .Net Framework components? Explain .Net Standard.

- Explain components like : CLS, CTS, JIT, MSIL, Profiling etc..

- Explain managed code and unmanaged code.

- How to call COM components inside .NET framework?

- Execution process of Grabage collector.

- What is CAS (Code Access Security) & different components of CAS?

- What is GAC and how to put assembly inside GAC, settelite assembly

- What is Assembly and types of assembly?

- What is Globalization and Localization?

- Explain Garbage collector and Generation 0, Generation 1 and Generation 2.

- What is Reflection and Why we need reflection in c#?

- Differnece between out and ref. There is new "in" parameter came into C# 8.0.

- Difference between var and dynamic?

- Difference between reflection and dynamic?

- What does the "volatile" keyword in C# mean?

- Why serialization and de-serialization?

- Explain types of serialization? Like Binary, SOAP and XML serialization.

- Explain OOPS concetps like Abstraction, Encapsulation, Polymorphism and Inheritance.

- Explain Generics and how it helps to our C# programming?

- Explain using directive and using statement.

- Why to use Generic method if we can overload a method for multiple data types?

- What is unsafe code and how to compile unsafe code and use in un-sage environment?

- When to use abstract class and when to use Interface?

- Diff between readonly and const. Explain real time use.

- how run time polymorphism works, explain with example how did u used?

- Explain covariance and contravariance c#.

- What is delegate and why we need delegate to call any method? Explain multicast delegate in c#?

- Explain Events and how did you used in real time?

- Explain MarshalByRef and MarshalByValue in C#.

- Explain Object pooling and how to achive?

- Explain Method hiding and how to use?

- How to implement two interface with same method in C#?

- Explain sealed class and real time use.

- How to restict a class, To not to create an object of the class? Explain real time use of this case.

- What is Constructor Chaining in C#?

- Difference between is and as operator in C#

- Is it possible to serialise hashtables?

- Explain Thread, real time use of multi thread.

- What is association, aggregation and composition?

- Explain IDisposable and finalizers in C#.

- Write a method to perform a binary search on an array without using built-in methods.

- Explain Dependency injection with example.

- Explain Singleton and Factory Pattern. What are the Design Pattern did you used?

SQL Server

**What is the Execution Order of a SQL Query in SQL Server?**

Execution Order of a SQL Query:

FROM ON JOIN

WHERE

GROUP BY

WITH CUBE or WITH ROLLUP

HAVING

SELECT

DISTINCT

ORDER BY

TOP

**How many columns can be included in an Index in SQL Server?**

You can create Indexes with 16 KEY COLUMNS in both Clustered and NonClustered Indexes. You can also INCLUDE 1023 NON KEY COLUMNS in NonClustered Index. Create NonClustered Index Ix\_NC\_Test On Test(Col1) Include(Col2,Col3) Col1 is Key Column(Upto 16) and Col2 and 3 are NonKey Columns(Upto 1023)

**How many Indexes you can create on a Table in SQL Server?**

Clustered Indexes : 1 and NonClustered Indexes : 249(Till 2005) and 999(After 2005)

**How to find and remove duplicates from a table?**

We can remove duplicates using a Ranking function Row\_Number() with partition by clause.

CREATE TABLE TEST\_WF(ID INT, NAME VARCHAR(5))

INSERT INTO TEST\_WF VALUES (1,'A'),(2,'B'),(3,'B'),(4,'C'),(5,'C'),(6,'C'),(7,'D')

SELECT \* INTO #TEMP FROM TEST\_WF;

WITH CTE AS (SELECT NAME, ROW\_NUMBER() OVER(PARTITION BY NAME ORDER BY NAME) AS RN FROM #TEMP) DELETE FROM CTE WHERE RN > 1

**Differenciate between Truncate and Delete.**

1. Truncate is a DDL command whereas Delete is a DDL command.

2. Truncate applies Lock on the entire table whereas Delete applies lock on the specific rows which are going to be deleted.

3. Truncate removes all rows from the table, whereas Delete removes the selected rows depending upon the predicate(Condition).

4. Truncate Minimally Logged the Transaction Log and hence are faster but cannot be rolled back, whereas Delete Logs every row in Transaction Log it deletes and are slower but can be rolled back.

5. Truncate RESET the Identity Seed, where as in Delete we have to Reset the identity seed using DBCC CHECKIDENT(‘TableName’, RESEED, 1(Value from which you want to reset))

6. Truncate cannot be used with the tables having Foreign Key references to any other table, whereas you can use Delete with such tables.

**How do you optimize a long running stored procedure in SQL Server?**

To optimize the long running SP, First we need to analyze the joins(ex : inner,outer apply) in query one by one removing condition from SP and notice the time and records it takes to fetch the records. you can apply the index on the table columns which takes more time an above first condition. can be removed unwanted where condition in query, can be call some condition in if-else condition.

**Where does Temporary table and Table variables get stored in SQL db?**

Both gets created in tempdb database , Create temporary table or table variable and open tempdb database and run following command , you will be able to see objects created there.

SELECT \* FROM SYS.OBJECTS ORDER BY OBJECT\_ID DESC By running above command you will be able to see temporary table although you will not be able to see table variable because it get decomposed as soon as it gets created. Though it gets created in Tempdb database

**Why not we should prefix stored procedure names with “sp\_”?**

The prefix sp\_ is reserved keyword for system stored procedures that ship with SQL Server. Whenever SQL Server encounters a procedure name starting with sp\_, it first tries to locate the procedure in the master database and then it looks for any qualifiers (database, owner) provided and then it tries dbo as the owner. So you can really save time in locating the stored procedure by avoiding the sp\_ prefix.

**What is difference between Select and set statement for assign a value in variable in SQL Server?**

SET is the ANSI standard for variable assignment, SELECT is not. SET can only assign one variable at a time, SELECT can make multiple assignments at once. If assigning from a query, SET can only assign a scalar value. If the query returns multiple values/rows then SET will raise an error. SELECT will assign one of the values to the variable and hide the fact that multiple values were returned (so you'd likely never know why something was going wrong elsewhere - have fun troubleshooting that one) When assigning from a query if there is no value returned then SET will assign NULL, where SELECT will not make the assignment at all (so the variable will not be changed from it's previous value) As far as speed differences - there are no direct differences between SET and SELECT. However SELECT's ability to make multiple assignments in one shot does give it a slight speed advantage over SET.

**Difference between table variable and temporary tables**

There is not so much difference between table variable and temp table. Both are using for capturing data temporary .Both are getting stored in tempdb.On Temp table we can have multiple index but in table variable we can have only one index which depends on primary key.Global temp table (type of temp table) can be accessed by multiple session but table variable is for current scope only.

**What are magic tables in SQL and how can we use them?**

when we use trigger inserted and deleted data are stored in temporary table those table are called magic tables.

**How to define a heap?**

A table which does not contains any physical ordered index aka clustered index is called as the heap. The data in heap is not sorted physically. If a table contains any non-clustered index but no clustered index, it means there is no physical ordering of data only logical ordering happens hence it is a heap.

**Find Tables From DataBase With Same Prefix Name?**

SELECT \* FROM sys.objects WHERE type='u' AND name LIKE 'Prefix%'

**How to execute sql query after specific time given In SQL SERVER?**

WAITFOR DELAY '00:00:20'; select \* from tablename For delaying the execution of 20 secand WAITFOR Time '00:04:20'; select \* from tablename For delaying the execution particular time

**How to make a column which accepts unique values or multiple null without using a trigger?**

Here we will use filtered unique index. Create unique nonclustered index Ix\_table\_column on table(column) where column is not null

**What is the optimization Techniques in SQL?**

<https://www.codeproject.com/Articles/35665/Top-steps-to-optimize-data-access-in-SQL-Serv>

**How to encrypt / decrypt password in SQL server 2012?**

There are many methods in Sql server 2012 which allows you to encrypt password easily. ENCRYPTBYPASSPHRASE(YOURKEYPHRASE,PASSWORD)will give you varbinary result. To Get the actual result you need to convert using convert(varchar(50),DECRYPTBYPASSPHRASE(YOURKEYPHRASE,ENCRYPTED\_VALUE)).

**Find the 5th highest salary of employee.**

WITH RESULT AS (

SELECT SALARY,DENSE\_RANK() OVER (ORDER BY SALARY DESC) AS DENSERANK

FROM EMPLOYEES )

SELECT TOP 1 SALARY FROM RESULT WHERE DENSERANK = 5

**What is the difference between Commit and rollback command?**

Commit command is used to make all the changes permanent to the underlying database. Rollback command is used to end the current transaction and undo all the changes we made since the current transaction began.We can't Rollback after the Commit.

**What types of Joins are possible with Sql Server?**

Join helps you to get the details from two to more than two tables. Types Of join :

1.) Inner join

2.)Outer join

2.a) left Outer join

2.b) Right Outer Join

2.c) Full Outer Join

3.)Cross Join

4.)Self join

**What is a NOLOCK?**

Using the NOLOCK query optimiser hint is generally considered good practice in order to improve concurrency on a busy system. When the NOLOCK hint is included in a SELECT statement, no locks are taken when data is read. The result is a Dirty Read, which means that another process could be updating the data at the exact time you are reading it. There are no guarantees that your query will retrieve the most recent data. The advantage to performance is that your reading of data will not block updates from taking place, and updates will not block your reading of data. SELECT statements take Shared (Read) locks. This means that multiple SELECT statements are allowed simultaneous access, but other processes are blocked from modifying the data. The updates will queue until all the reads have completed, and reads requested after the update will wait for the updates to complete. The result to your system is delay(blocking).

**What is Collation?**

Collation refers to a set of rules that determine how data is sorted and compared. Character data is sorted using rules that define the correct character sequence, with options for specifying case-sensitivity, accent marks, kana character types and character width.

**What is cursors?**

Cursor is a database object used by applications to manipulate data in a set on a row-by-row basis, instead of the typical SQL commands that operate on all the rows in the set at one time. In order to work with a cursor we need to perform some steps in the following order: Declare cursor,Open cursor,Fetch row from the cursor,Process fetched row,Close cursor,Deallocate cursor

**Can we return Data from 4(more than 1) tables in stored procedure?**

Yes- create stored procedure and write select statement separated by ;

**Display Employee name and Manager Name from Employee Table(Self Join)?**

select E2.Name,E1.Name from Employee E1 JOIN Employee E2 on E1.EmpId=E2.ManagerId

**Where does Uncommitted transaction resides in File System?**

Uncommitted transactions are present in LDF file when the transaction is committed it is then moved to MDF file.

**What is a View ?Can we insert,Update and delete a view?**

View is a result of a query saved in DB. View is used when we have to give access to limited amount of data. We can give permission to query a view for a table while denied access to the original table. It acts as a security measure.We can insert,update,delete an updatable view only. Read-only views are not updatable.

**What is the difference between implicit and explicit transaction?**

1. Implicit transaction is auto commit, there are no beginning and ending of the transaction while explicit transaction has beginning and end and rollback command.

2. In explicit transaction, if error occurs between transaction then it can be roll back where as it is not possible in implicit transaction.

**What is #temp and @table variable in SQL server?**

One of the most common MYTH about Temporary Table & Table Variable is that: Temporary Tables are created in TempDB and Table Variables are created In-Memory. Fact is that both are created in TempDB, below Demos prove this reality.Temporary Tables honor the explicit transactions defined by the user.

Table variables doesn’t participate in the explicit transactions defined by the user.Temporary Tables are not allowed in User Defined Functions.

Table Variables can be used in User Defined Functions.

Temporary table supports adding Indexes explicitly after Temporary Table creation and it can also have the implicit Indexes which are the result of Primary and Unique Key constraint.

Table Variables doesn’t allow the explicit addition of Indexes after it’s declaration, the only means is the implicit indexes which are created as a result of the Primary Key or Unique Key constraint defined during Table Variable declaration.

**What is happened when we create a field a primary key?**

if primary key created internally clustered index will be created

**What is the type of trigger ?**

There is 2 type of trigger and these are After trigger and Instead of trigger

**What is syntax for creating triggers ?**

FOR UPdate:

CREATE TRIGGER [dbo].[TRG\_User\_Updated] ON

[dbo].[PATIENT\_ASSESSMENTS] AFTER UPDATE

AS BEGIN

SET NOCOUNT ON;

INSERT INTO user(userID )

SELECT d.userID FROM INSERTED as d

END

**What is a NATURAL JOIN in SQL?**

A NATURAL JOIN is a JOIN operation that creates an implicit join clause for you based on the common columns in the two tables being joined. Common columns are columns that have the same name in both tables.

A NATURAL JOIN can be an INNER join, a LEFT OUTER join, or a RIGHT OUTER join. The default is INNER join.

If the SELECT statement in which the NATURAL JOIN operation appears has an asterisk (\*) in the select list, the asterisk will be expanded to the following list of columns (in this order):

* All the common columns
* Every column in the first (left) table that is not a common column
* Every column in the second (right) table that is not a common column

An asterisk qualified by a table name (for example, COUNTRIES.\*) will be expanded to every column of that table that is not a common column.

If a common column is referenced without being qualified by a table name, the column reference points to the column in the first (left) table if the join is an INNER JOIN or a LEFT OUTER JOIN. If it is a RIGHT OUTER JOIN, unqualified references to a common column point to the column in the second (right) table.

Examples

If the tables COUNTRIES and CITIES have two common columns named COUNTRY and COUNTRY\_ISO\_CODE, the following two SELECT statements are equivalent:

SELECT \* FROM COUNTRIES NATURAL JOIN CITIES

SELECT \* FROM COUNTRIES JOIN CITIES

USING (COUNTRY, COUNTRY\_ISO\_CODE)

The following example is similar to the one above, but it also preserves unmatched rows from the first (left) table:

SELECT \* FROM COUNTRIES NATURAL LEFT JOIN CITIES

**What is a FULL OUTER JOIN in SQL?**

The FULL OUTER JOIN keyword return all records when there is a match in either left (table1) or right (table2) table records.

syntax:

SELECT column\_name(s)

FROM table1

FULL OUTER JOIN table2

ON table1.column\_name = table2.column\_name

WHERE condition;

**Difference between Count(\*) and Count(1) in sql?**

The parameter to the COUNT function is an expression that is to be evaluated for each row. The COUNT function returns the number of rows for which the expression evaluates to a non-null value. ( \* is a special expression that is not evaluated, it simply returns the number of rows.)There are two additional modifiers for the expression: ALL and DISTINCT. These determine whether duplicates are discarded. Since ALL is the default, your example is the same as count(ALL 1), which means that duplicates are retained.Since the expression "1" evaluates to non-null for every row, and since you are not removing duplicates, COUNT(1) should always return the same number as COUNT(\*).

**Update "Male" into "Female" and "Female" into "Male" in Gender Column of a table in single line query.?**

update table\_Name set Gender= Case when Gender='Male' then 'Female' else'Male' end

**Isolation levlels and transanctions in sql. What are they and diffrences?**

Five type of transanctions isolation level in sql server

1 READ UNCOMMITTED

2 READ COMMITTED

3 REPEATABLE READ

4 SERIALIZABLE

5 SNAPSHOT

**what are the ways to loop through records in sql qurey ?**

We can use Cursors to loop with this Sql Server has provisions to use while loop as well. Using these we can loop through all records in sql server.

**What are new features in SQL Serve 2014 in comparison with SQL Server 2012?**

Difference Between 2012 and 2014 SQL Server Enterprise Edition

1. Memory Support: The earlier versions of Sql Server (i.e. 2008 R2 and 2012 only support 64 GB. On 2014 the memory support is increased to 128 GB of memory.

2. Indexes: In 2012 columnstore indexes was introduced which was a great help to datawarehouse workloads. But they were limited to non clustered indexes, supported very few number of datatypes. But in 2014 this feature was expanded to support clustered columnstore indexes and many of the datatypes restrictions was removed.

3. Buffer Pool Extension: Use of fast SSD drives that can really prove as a perfect substitutes for memory. This proves to be very grateful when the server does not support more addition to RAM memory and database size exceeding the amount of RAM.

4. The New Cardinality Estimator: The introduction to this estimator which is great at fixing a lot of queries and providing better plans without re-writing the queries themselves.

5. Always on Availability Groups: SQL Server 2014's AlwaysOn Availability Groups has been enhanced with support for additional secondary replicas and Windows Azure integration.

SQL Server Continues

**Sql query releted Questions**

Sample tables are listed to visualize the data and associate with query answers given.

1. ==================
2. Consider below tables
3. ==================
4. EMPLOYEE
5. empid   empname managerid   deptid  salary  DOB
6. 1       emp 1       0       1       6000    1982-08-06 00:00:00.000
7. 2       emp 2       0       5       6000    1982-07-11 00:00:00.000
8. 3       emp 3       1       1       2000    1983-11-21 00:00:00.000
9. 13      emp 13      2       5       2000    1984-03-09 00:00:00.000
10. 11      emp 11      2       1       2000    1989-07-23 00:00:00.000
11. 9       emp 9       1       5       3000    1990-09-11 00:00:00.000
12. 8       emp 8       3       1       3500    1990-05-15 00:00:00.000
13. 7       emp 7       2       5       NULL    NULL
14. 3       emp 3       1       1       2000    1983-11-21 00:00:00.000
16. --DEPARTMENT TABLE
17. deptid  deptname
18. 1       IT
19. 2       Admin

1. Employee and Manager ID are in the same table; can you get manager names for employees?

**Answer:**  
With the help of Common table expressions, we can achieve this.

1. ;**with** empCTE **as**
2. (
3. **select** e.empid, e.empname, e.managerid,
4. CAST('' **as** **varchar**(50)) **as** Mname **from** employee e
5. **where** managerid = 0
7. **union** all
9. **select** e1.empid, e1.empname, e1.managerid,
10. CAST(c.empname **as** **varchar**(50)) **as** Mname **from** employee e1
11. **inner** join empCTE **as** C **on** e1.managerid=c.empid
12. **where** e1.managerid>0
13. ) **select** \* **from** empCTE

2. Can you get employee details whose department id is not valid or department id not present in department table?

**Answer**Identifying Department IDs in employee table, which are not available in master.  
  
There are multiple ways to do this.

Using Left JOIN

1. **SELECT** E.EMPID,E.EMPNAME, E.DEPTID **FROM** EMPLOYEE E
2. left outer join DEPARTMENT d
3. **on** E.DEPTID = D.DEPTID
4. **WHERE** D.DEPTID **IS** NULL

Using NOT IN

1. **SELECT** E.EMPID,E.EMPNAME, E.DEPTID **FROM** EMPLOYEE E
2. **where** e.deptid not in (**select** deptid **from** department)

Using NOT Exists

1. **SELECT** E.EMPID,E.EMPNAME, E.DEPTID **FROM** EMPLOYEE E
2. **where** NOT EXISTS (**select** deptid **from** department **where** e.deptid=department.deptid)

**Note**  
"Not In"  is the least recommended, considering performance. Outer join and Not Exists are preferred.  
  
Using EXCEPT KEYWORD  
  
if you want to list Department IDs only. INTERSECT and EXCEPT keywords have rules

1. **SELECT** deptid **FROM** EMPLOYEE
2. **EXCEPT**
3. **SELECT** DEPTID **FROM** DEPARTMENT

**3. Can you get the list of employees with same salary?**

**Answer**

With where clause

1. **Select** **distinct** e.empid,e.empname,e.salary
2. **from** employee e, employee e1
3. **where** e.salary =e1.salary
4. and e.empid != e1.empid

4. How can you find duplicate records in Employee table?

**Answer**

1. **SELECT** EMPID,EMPNAME, SALARY, COUNT(\*) **AS** CNT
2. **FROM** EMPLOYEE
3. **GROUP** **BY** EMPID,EMPNAME, SALARY
4. **HAVING** COUNT(\*)>1

5. How can you  DELETE DUPLICATE RECORDS?

**Answer**

There are multiple options to perform this operation.   
  
Using row count to restrict delete only 1 record

1. **set** rowcount 1
2. **DELETE** **FROM** EMPLOYEE **WHERE** EMPID IN (
3. **SELECT** EMPID
4. **FROM** EMPLOYEE
5. **GROUP** **BY** EMPID,EMPNAME, SALARY
6. **HAVING** COUNT(\*)>1
7. )
8. **set** rowcount 0

Use auto increment primary key "add" if not available in the table, as in given example.

1. **alter** **table** employee
2. **add** empidpk **int** identity (1,1)

Now, perform query on min of auto pk id, group by duplicate check columns - this will give you latest duplicate records

1. **select** \* **from** employee **where**
2. empidpk not in ( **select** **min**(empidpk) **from** employee
3. **group** **by** EMPID,EMPNAME, SALARY )

Now, delete.

1. **Delete** **from** employee **where**
2. empidpk not in ( **select** **min**(empidpk) **from** employee
3. **group** **by** EMPID,EMPNAME, SALARY )

6. Find the second highest salary.

**Answer**

1. **Select** **max**(Salary) **from** employee
2. **where** Salary not in (**Select** **max**(Salary) **from** employee)

7. Now, can you find 3rd, 5th or 6th i.e. N'th highest Salary?

**Answer**

Query for 3rd highest salary

1. **SELECT** \* **FROM** EMPLOYEE E
2. **WHERE** 2 = (**SELECT** COUNT(**DISTINCT** E1.SALARY)
3. **FROM** EMPLOYEE E1
4. **WHERE** E1.SALARY>E.SALARY)

Here, 2= 3-1 i.e. N-1 ; can be applied for any number.

8. Can you write a query to find employees with age greater than 30?

**Answer**

1. **select** \* **from** employee
2. **where** datediff(year,dob, getdate()) >30

9. Write an SQL Query to print the name of the distinct employees whose DOB is between 01/01/1960 to 31/12/1987

**Answer**

1. **SELECT** **DISTINCT** EmpName **FROM** Employee
2. **WHERE** DOB BETWEEN '01/01/1960' AND '12/31/1987'

10.  Please write a query to get the maximum salary from each department.

**Answer**

1. **select** DeptId, **max**(salary) **as** Salary **from** employee **group** **by** deptid

11. What is wrong with the following query?

1. **SELECT** empName **FROM** employee **WHERE** salary <> 6000

**Answer**

The following query will not fetch record with the salary of 6000 but also will skip the record with NULL.  
  
As per SQL Server logic, it works on 3 values in matching conditions. TRUE or FALSE and UNKNOWN. Here,  NULL implies UNKNOWN.

to fix this:

1. **SELECT** empName  **FROM**
2. employee **WHERE** salary **is** NULL or salary <> 6000

12. Can you show one row twice in results from a table?

**Answer**Yes. We can use union all or cross join to obtain this.

1. **select** deptname **from** department d **where** d.deptname='it'
2. **union** all
3. **select** deptname **from** department d1 **where** d1.deptname='it'

 -- also cross join alias same table

1. **select** d.deptname **from** department d, department d1
2. **where** d.deptname='it'

13. Could you tell the output or result of the following SQL statements?

**Answer**

1. **select** '7'
2. -- output = 7
3. **select** 7
4. -- output = 7
5. **select** count (7)
6. -- output = 1
7. **SELECT** COUNT('7')
8. -- output = 1
9. **SELECT** COUNT(\*)
10. -- output = 1

14. What is an alternative for TOP clause in SQL?

**Answer**

- There can be two alternatives for the top clause in SQL.  
  
#1

-- Alternative - ROWCOUNT function

1. **Set** rowcount 3
2. **Select** \* **from** employee **order** **by** empid **desc**
3. **Set** rowcount 0

#2

-- Alternative and  WITH and ROWNUMBER function

-- between 1 and 2

1. **With** EMPC **AS**
2. ( **SELECT** empid, empname,salary,
3. ROW\_NUMBER() OVER (**order** **by** empid **desc**) **as** RowNumber
4. **FROM** employee )
5. **select** \*
6. **from** EMPC
7. **Where** RowNumber Between 1 and 7

15.  Will the following statements  run or give error?

**Answer**NO error.

1. **SELECT** COUNT(\*) + COUNT(\*)
2. **Output**  = 2
3. **SELECT** (**SELECT** 'c#')
4. **Output** = c#

16. Can you write a query to get employee names starting with a vowel?

**Answer**

Using like operator and expression,

1. **Select** empid, empname **from** employee **where** empname like '[aeiou]%'

17. Can you write a query to get employee names ending with a vowel?

**Answer**

1. **Select** empid, empname **from** employee **where** empname like '%[aeiou]'

18. Can you write a query to get employee names starting and ending with a vowel?

**Answer**

Here you will get only one record of "empone".

1. **select** empid, empname **from** employee **where** empname like '[aeiou]%[aeiou]'

19.  Write a query to get employees whos ID is even.

**Answer**

1. **select** \* **from** employee
2. **where** empid %2 =0

20. Write a query to get employees whos ID is an odd number.

**Answer**

1. **select** \* **from** employee
2. **where** empid %2 !=0

21. How can you get random employee record from the table?

**Answer**

1. **select** **top** 1 \* **from** employee **order** **by** newid()

22.(Tricky) Below is the table data which has 1 columns and 7 rows

1. **Table** -TESTONE
2. DATACOL
3. 10/12
4. 1a/09
5. 20/14
6. 20/1c
7. 3112
8. 11/16
9. mm/pp

Give data in a table is of format 'NN/NN', verify that the first and last two characters are numbers and that the middle character is '/'.

**Answer**

Print the expression 'NUMBER' if valid, 'NOT NUM' if not valid.  
  
This can be done using like operator and expression. Checking numbers and not characters.

1. **SELECT** DataCol, 'CHECK' =
2. CASE
3. **WHEN** datacol like '%[0-9]%[^A-Z]%/%[^A-Z]%[0-9]%' **then** 'NUMBER'
4. **else** 'NOT NUM'
5. **end**
6. **from** TestOne

23. Consider following 3 tables with one column

1. Tbl1
2. col1
3. 1
4. 1
5. 1
6. Tbl2
7. col1
8. 2
9. 2
10. 2
11. Tbl3
12. col1
13. 3
14. 3
15. 3

**How many rows will following query return? (0, 3 or 9)**

1. **Select** \* **from** Tbl1 **inner** join tbl2 **on** tbl1.col1=tbl2.col1
2. Left outer join Tbl3 **on** Tbl3.Col1=Tbl2.Col1

**Answer- 0 .**

24. If all values from tbl2 are deleted. What will be the output of the following query?

**Answer**

select Tbl1.\* from tbl1,tbl2

Ans - 0 Rows.

25. Can you write a query to print prime numbers from 1 to 100?

**Answer**  
For this, we have to use a loop as in other programming languages.

1. **DECLARE**
3. @i **INT**,
4. @a **INT**,
5. @count **INT**,
6. @result **varchar**(**Max**)
8. **SET** @i = 1
9. **set** @result=''
11. WHILE (@i <= 100)
12. **BEGIN**
13. **SET** @count = 0
14. **SET** @a = 1
15. -- logic to check prime number
16. WHILE (@a <= @i)
17. **BEGIN**
18. IF (@i % @a = 0)
19. **SET** @count = @count + 1
21. **SET** @a = @a + 1
22. **END**
23. IF (@count = 2)
24. **set** @result = @result+cast(@i **as** **varchar**(10))+' , '
26. **SET** @i = @i + 1
27. **END**
29. **set** @result = (**select** substring(@result, 1, (len(@result) - 1)))
30. print(@result)

26. Write query to print numbers from 1 to 100 without using loops

**Answer**

This can be done using Common Table Expression without using a loop.

1. ;**with** numcte
2. **AS**
3. (
4. **SELECT** 1 [**SEQUENCE**]
6. **UNION** ALL
8. **SELECT** [**SEQUENCE**] + 1 **FROM** numcte **WHERE** [**SEQUENCE**] <100
9. )
11. **SELECT** \* **FROM** numcte

27. What will be the output of following SQL?(tricky)

1. **Select** $
3. Options  -
4. a. 0.00,
5. b. $,
6. c. 0,
7. d. Syntax Error
9. Answer  = 0.00

28. What will be the output of following SQL queries?

1. **Select** **select** 'TD'
2. Options -
3. 1. TD,
4. 2. Syntax Error,
5. **3. select** TD
7. Answer - Syntax Error. (Incorrect syntax near the keyword 'select'. )
9. **select** \* **from** 'Employee'
11. Answer -  Incorrect syntax near 'Employee' .

29. What will be the outputs in the following SQL queries with aggregate functions?

1. **SELECT** SUM (1+4\*5)
3. Options - a.21,      b.25,         c.Error        d.10
5. Answer -: 21

8. **SELECT** **MAX**  (1,3,8)
10. Options - a.8,        b. 12,        c.Error        d.1
12. Answer -: Error. **Max** **function** takes **only** 1 argument.

15. **SELECT** **Max** ('TD')
17. Options -  a.TD         b. Error      c. 1       d.0
19. Answer-: TD

22. **SELECT** **Max** ('TD'+'AD')
24. Options -  a.TDAD         b. Error      c. T2D       d.0
25. Answer-: TDAD

30. What will be the output of following queries? [Tricky involving 0]

1. **SELECT** 0/0
3. A. Divide **by** 0 error,   B. 0
4. C. NULL,                   D. Incorrect syntax error
6. Answer -:  Divide **by** 0 error
8. **SELECT**  0/6
10. A. Divide **by** 0 error,   B. 0
11. C. 6,                         D. Incorrect syntax error
13. Answer -:  0

31 What will be the output of given statement?

SELECT SUM (NULL)   
  
**Answer** = Error. Cannot pass null type in SUM function.  
  
Operand data type NULL is invalid for avg operator. 

32. What will be the output of given statement?

SELECT

MAX (NULL)  
  
**Answer** = Error. Operand data type NULL is invalid for MAX operator.

33. Will following statement give error or 0 as output?

SELECT

AVG (NULL)  
  
Answer = Error. Operand data type NULL is invalid for Avg operator.   
  
**Note**MIN, MAX,SUM,AVG none of these function takes NULL parameter/argument. Also, these functions accept only one argument.

34. Will the following statements execute? if yes what will be output?

SELECT NULL+1

SELECT NULL+'1'  
  
**Answer** - Yes, no error. The output will be NULL. Perform any operation on NULL will get the NULL result.

35. Will following statement give Divide by Zero error?

SELECT NULL/0  
  
**Answer**- No. It will execute and result will be NULL.

**SQL Server**

**What is RDBMS?**

Ans. RDBMS or Relational Data Base Management Systems are database management systems that maintain data in the form of tables. We can create relationships between the tables. An RDBMS has the capability to recombine the data items from different files, providing powerful tools for data usage.

**What is the difference between a primary key and a unique key?**

The primary key is a column whose values uniquely identify every row in a table. Primary key values can never be reused. They create a clustered index on the column and cannot be null.

A Unique key is a column whose values also uniquely identify every row in a table but they create a non-clustered index by default and it allows one NULL only.

**What is the difference between clustered and a non-clustered index?**

A clustered index is an index that rearranges the table in the order of index itself. Its leaf nodes contain data pages. A table can have only one clustered index.

A non-clustered index is an index that does not re-arranges the table in the order of index itself. Its leaf nodes contain index rows instead of data pages. A table can have many non-clustered indexes.

**What is the difference between a HAVING CLAUSE and a WHERE CLAUSE?**

Both specify a search condition but Having clause is used only with the SELECT statement and typically used with GROUP BY clause.

If GROUP BY clause is not used then Having behaved like WHERE clause only.

**What is Mirroring?**

Mirroring is a high availability solution. It is designed to maintain a hot standby server which is consistent with the primary server in terms of a transaction. Transaction Log records are sent directly from principal server to a secondary server which keeps a secondary server up to date with the principal server.

**What are the advantages of the Mirroring?**

It is more robust and efficient that Log shipping.

It has an automatic failover mechanism.

The secondary server is synced with the primary in near real time.

**What is Log Shipping?**

Log shipping is nothing but the automation of backup and restores of a database from one server to another standalone standby server. This is one of the disaster recovery solutions. If one server fails for some reason we will have the same data available on the standby server.

**Can we take the full database backup in Log shipping?**

Yes, we can take the full database backup. It won’t affect the log shipping.

**What is an execution plan?**

An execution plan is a graphical or textual way of showing how the SQL server breaks down a query to get the required result. It helps a user to determine why queries are taking more time to execute and based on the investigation user can update their queries for the maximum result.

In Query Analyzer is an option called “Show Execution Plan” (located on the Query drop-down menu). If this option is turned on it will display a query execution plan in a separate window when a query is run again.

**What is the Stored Procedure?**

A stored procedure is a set of SQL queries which can take input and send back output. And when the procedure is modified, all clients automatically get the new version. Stored procedures reduce network traffic and improve performance. Stored procedures can be used to help ensure the integrity of the database.

**List the advantages of using Stored Procedures?**

* Stored procedure boosts application performance.
* Stored procedure execution plans can be reused as they cached in SQL Server's memory which reduces server overhead.
* Stored procedures can be reused.
* Stored procedures can encapsulate logic. You can change the stored procedure code without affecting clients.
* Stored procedures provide better security for your data.

**What is identity in SQL?**

An identity column in the SQL automatically generates numeric values. We can be defined as a start and increment value of the identity column. Identity columns do not need to be indexed.

**What are the common performance issues in SQL Server?**

Deadlocks

Blocking

Missing and unused indexes.

I/O bottlenecks

Poor Query plans

Fragmentation

**What is a performance monitor?**

Windows performance monitor is a tool to capture metrics for the entire server. We can use this tool for capturing events of SQL server also.

Some useful counters are – Disks, Memory, Processors, Network etc.

**What is the difference between a Local and a Global temporary table?**

If defined in inside a compound statement a local temporary table exists only for the duration of that statement but a global temporary table exists permanently in the database but its rows disappear when the connection is closed.

**What is the SQL Profiler?**

SQL Profiler provides a graphical representation of events in an instance of SQL Server for the monitoring and investment purpose. We can capture and save the data for further analysis. We can put filters as well to captures the specific data we want.

**How can we check the SQL Server version?**

Ans. By running the following command: SELECT @@Version

**Is it possible to call a stored procedure within a stored procedure?**

Yes, we call a stored procedure within a stored procedure It is called recursion property of SQL server and these type of stored procedures are called nested stored procedures.

**What is SQL Server Agent?**

SQL Server agent allows us to schedule the jobs and scripts. It helps is implementing the day to day DBA tasks by automatically executing them on a scheduled basis.

**What is the PRIMARY KEY?**

The primary key is a column whose values uniquely identify every row in a table. Primary key values can never be reused.

**What is a UNIQUE KEY constraint?**

A UNIQUE constraint enforces the uniqueness of the values in a set of columns, so no duplicate values are entered. The unique key constraints are used to enforce entity integrity as the primary key constraints.

**What is FOREIGN KEY**

When a one table’s primary key field is added to related tables in order to create the common field which relates the two tables, it called a foreign key in other tables.Foreign Key constraints enforce referential integrity.

**What is a CHECK Constraint?**

A CHECK constraint is used to limit the values or type of data that can be stored in a column. They are used to enforce domain integrity.

**What are a Scheduled Jobs?**

The scheduled job allows a user to run the scripts or SQL commands automatically on the scheduled basis. The user can determine the order in which commands need to execute and the best time to run the job to avoid the load on the system.

**What is BCP or Bulk Copy?**

BCP or Bulk Copy is a tool by which we can copy a large amount of data to tables and views. BCP does not copy the structures same as source to destination. BULK INSERT command helps to import a data file into a database table or view in a user-specified format.

**What is Normalization?**

The process of table design to minimize the data redundancy is called normalization. We need to divide a database into two or more table and define relationships between them. Normalization usually involves dividing a database into two or more tables and defining relationships between the tables.

**1NF (Eliminate Repeating Groups)**: Make a separate table for each set of related attributes, and give each table a primary key. Each field contains at most one value from its attribute domain.

**2NF (Eliminate Redundant Data):** If an attribute depends on only part of a multi-valued key, remove it to a separate table.

**3NF (Eliminate Columns Not Dependent On Key):** If attributes do not contribute to a description of the key, remove them to a separate table. All attributes must be directly dependent on the primary key.

**BCNF (Boyce-Codd Normal Form):** If there are non-trivial dependencies between candidate key attributes, separate them out into distinct tables.

**4NF (Isolate Independent Multiple Relationships):** No table may contain two or more 1:n or n:m relationships that are not directly related.

**5NF (Isolate Semantically Related Multiple Relationships):** There may be practical constraints on information that justifies separating logically related many-to-many relationships.

**What is De-normalization?**

De-normalization is the process of adding redundant data to a database to enhance the performance of it. It is a technique to move from higher to lower normal forms of database modeling in order to speed up database access.

**What is a Trigger and types of a trigger?**

The trigger allows us to execute a batch of SQL code when a table event occurs (Insert, update or delete command executed against a specific table). Triggers are stored in and managed by DBMS. It can also execute a stored procedure.

2 types of triggers that are available in the SQL Server are as follows:

**DML Triggers**: DML or Data Manipulation Language triggers are invoked whenever any of the DML commands like INSERT, DELETE or UPDATE happens on the table or on the view.

**DDL Triggers**: DDL or Data Definition Language triggers are invoked whenever any changes occur in the definition of any of the database objects instead of actual data. These are very helpful to control the production and development of database environments.

**Logon Triggers**: These are very special triggers which fire in case of the logon event of the SQL Server. This is fired before setup of a user session in the SQL Server.

**What is the Subquery?**

A Subquery is a subset of select statements whose return values are used in filtering conditions of the main query. It can occur in a SELECT clause, FROM clause and WHERE clause. It nested inside a SELECT, INSERT, UPDATE, or DELETE statement or inside another subquery.

Types of Sub-query:

**Single-row sub-query**: where the sub-query returns only one row

**Multiple-row sub-query**: where the subquery returns multiple rows, and

**Multiple column sub-query**: where the sub-query returns multiple columns

**What is a Linked Server?**

Linked Servers is a concept by which we can connect another SQL server to a Group and query both the SQL Servers database using T-SQL Statements

**What is Collation?**

Collation refers to a set of rules that determine how data is sorted and compared. Character data is sorted using rules that define the correct character sequence, with options for specifying case sensitivity, accent marks, kana character types and character width.

**What is View?**

A view is a virtual table which contains data from one or more tables. Views restrict data access of the table by selecting only required values and make complex queries easy.

Rows updated or deleted in the view are updated or deleted in the table the view was created with. It should also be noted that as data in the original table changes, so does data in the view, as views are the way to look at part of the original table. The results of using a view are not permanently stored in the database

**Where SQL server usernames and passwords are stored in a SQL server?**

They get stored in System Catalog Views sys.server\_principals and sys.sql\_logins.

**What are the properties of a transaction?**

Generally, these properties are referred to as ACID properties.

They are:Atomicity

Consistency

Isolation

Durability

**Define UNION, UNION ALL, MINUS, INTERSECT?**

**UNION** – returns all distinct rows selected by either query.

**UNION ALL** – returns all rows selected by either query, including all duplicates.

**MINUS** – returns all distinct rows selected by the first query but not by the second.

**INTERSECT** – returns all distinct rows selected by both queries.

**What are functions in the SQL Server?**

Functions are the sequence of the statements which accepts inputs, process the inputs to perform some specific task and then provides the outputs.

**What is a User-Defined function in the SQL Server and what is its advantage?**

User-Defined Function is a function which can be written as per the needs of the user by implementing your own logic. The biggest advantage of this function is that the user is not limited to pre-defined functions and can simplify the complex code of pre-defined function by writing a simple code as per the needs. This returns Scalar value or a table.

**What are the Pre-Defined functions in the SQL Server?**

These are Built-In functions of the SQL Server like String functions which are provided by SQL Server like ASCII, CHAR, LEFT, etc. string functions.

**What is TCL in SQL Server?**

TCL is Transaction Control Language Commands which are used to manage the transactions in the SQL Server. There are 3 TCL Commands in the SQL Server. These are as follows:

**Commit**: This command is used to save the transaction permanently in the database.

**Rollback**: This is used to roll back the changes that are done i.e. to restore the database in the last committed state.

**Save Tran**: This is used for saving the transaction so as to provide the convenience that the transaction can be rolled back to the point wherever required.

**Why is replication required on the SQL Server?**

Ans. Replication is the mechanism which is used to synchronize the data among the multiple servers with the help of a replica set.

This is mainly used to increase the capacity of the reading and to provide an option to its users to select among various different servers to perform the read/write operations.

**What are the advantages of having an index on the SQL Server?**

Index supports the mechanism of having faster data retrieval from the database.

This forms a data structure in a way which helps in minimizing the data comparisons.

This improves the performance of the retrieval of the data from the database.