What Is Your Greatest Weakness Job Interview Question

1)I used to like to work on one project to its completion before starting on another, but I've learned to work on many projects at the same time, and I think it allows me to be more creative and effective in each one

Sometimes, I spend more time than necessary on a task or take on tasks personally that could easily be delegated to someone else. Although I've never missed a deadline, it is still an effort for me to know when to move on to the next task, and to be confident when assigning others work.

2) Why Should We Hire You?

I have top-notch administrative skills and I believe I'd be an asset for the office. My skill set seems to be a perfect match for what you're looking for. In addition, I enjoy working with people, and would welcome the opportunity to be a part of your team.”

# 3) What is Your Greatest Strength?

* I have an extremely strong work ethic. When I'm working on a project, I don't want just to meet deadlines. Rather, I prefer to complete the project well ahead of schedule. Last year, I even earned a bonus for completing my three most recent reports one week ahead of time.

4) where do you see yourself in five years as sql developer

I have a long time plans to stick with the company and learn their domain/Product for the initial months.  
  
Going forward I would like to apply my knowledge among the team and as well as learn from the team.  
  
I would like to see my company growing so that I can gro with the company.

5) The difference between SP and UDF is listed below:

+---------------------------------+----------------------------------------+

| Stored Procedure (SP) | Function (UDF - User Defined |

| | Function) |

+---------------------------------+----------------------------------------+

| SP can return zero , single or | Function must return a single value |

| multiple values. | (which may be a scalar or a table). |

+---------------------------------+----------------------------------------+

| We can use transaction in SP. | We can't use transaction in UDF. |

+---------------------------------+----------------------------------------+

| SP can have input/output | Only input parameter. |

| parameter. | |

+---------------------------------+----------------------------------------+

| We can call function from SP. | We can't call SP from function. |

+---------------------------------+----------------------------------------+

| We can't use SP in SELECT/ | We can use UDF in SELECT/ WHERE/ |

| WHERE/ HAVING statement. | HAVING statement. |

+---------------------------------+----------------------------------------+

| We can use exception handling | We can't use Try-Catch block in UDF. |

| using Try-Catch block in SP. | |

+---------------------------------+----------------------------------------+

SELECT SERVERPROPERTY('Edition') AS Edition,  
SERVERPROPERTY('ProductLevel') AS ProductLevel,  
SERVERPROPERTY('ProductVersion') AS ProductVersion

## **What is Log Shipping?**

Log shipping is the process of automating the backup of database and transaction log files on a production SQL server and then restoring them onto a standby server. All Editions (except Express Edition) supports log shipping. In log shipping, the transactional log file from one server is automatically updated into the backup database on the other server. If one server fails, the other server will have the same db and can be used this as the Disaster Recovery plan. The key feature of log shipping is that it will automatically backup transaction logs throughout the day and automatically restore them on the standby server at defined intervals. (Courtney: [**Rhys**](https://blog.sqlauthority.com/2011/07/08/sql-server-interview-questions-and-answers-frequently-asked-questions-day-8-of-31/#comment-146895)**)**

## **What is a Scheduled Job or What is a Scheduled Task?**

Scheduled tasks let user automate processes that run on regular or predictable cycles. User can schedule administrative tasks, such as cube processing, to run during times of slow business activity. User can also determine the order in which tasks run by creating job steps within a SQL Server Agent job, e.g. back up database and update statistics of the tables. Job steps give user control over flow of execution. If one job fails, then the user can configure SQL Server Agent to continue to run the remaining tasks or to stop execution.

## **What is an Execution Plan? When would you Use it? How would you View the Execution Plan?**

An execution plan is basically a road map that graphically or textually shows the data retrieval methods chosen by the SQL Server query optimizer for a stored procedure or ad-hoc query, and it is a very useful tool for a developer to understand the performance characteristics of a query or stored procedure since the plan is the one that SQL Server will place in its cache and use to execute the stored procedure or query. Within the Query Analyzer, there is an option called “Show Execution Plan” (in the Query drop-down menu). If this option is turned on, it will display query execution plan in a separate window when the query is ran again.

## **What is a Table Called, if it has neither Cluster nor Non-cluster Index? What is it Used for?**

Unindexed table or Heap. Microsoft Press Books and Book on Line (BOL) refers it as Heap. A heap is a table that does not have a clustered index and therefore, the pages are not linked by pointers. The IAM pages are the only structures that link the pages in a table together. Unindexed tables are good for fast storing of data. Many times, it is better to drop all the indexes from table and then do bulk of INSERTs and restore those indexes after that.

## **SQL Injection Attack?**

SQL injection is an attack in which malicious code is inserted into strings that are later passed to an instance of SQL Server for parsing and execution. Any procedure that constructs SQL statements should be reviewed for injection vulnerabilities because SQL Server will execute all syntactically valid queries that it receives. Even parameterized data can be manipulated by a skilled and determined attacker.

Here are few methods which can be used to protect again SQL Injection attack:

* Use Type-Safe SQL Parameters
* Use Parameterized Input with Stored Procedures
* Use the Parameters Collection with Dynamic SQL
* Filtering Input parameters
* Use the escape character in LIKE clause

## **What is CHECKPOINT Process in the SQL Server?**

CHECKPOINT process writes all dirty pages for the current database to disk. Dirty pages are data pages that have been entered into the buffer cache and modified, but not yet written to disk.

* Wrapping Parameters with QUOTENAME() and REPLACE()

## **What is the Difference between Index Seek vs. Index Scan?**

An index scan means that SQL Server reads all the rows in a table, and then returns only those rows that satisfy the search criteria. When an index scan is performed, all the rows in the leaf level of the index are scanned. This essentially means that all the rows of the index are examined instead of the table directly. This is sometimes compared to a table scan, in which all the table data is read directly. However, there is usually little difference between an index scan and a table scan.

An index seek, on the other hand, means that the Query Optimizer relies entirely on the index leaf data to locate rows satisfying the query condition. An index seek will be most beneficial in cases where a small percentage of rows will be returned. An index seek will only affect the rows that satisfy a query condition and the pages that contain these qualifying rows; in terms of performance, this is highly beneficial when a table has a very large number of rows. [(Read more here)](https://blog.sqlauthority.com/2009/08/24/sql-server-index-seek-vs-index-scan-diffefence-and-usage-a-simple-note/)

## **What are the Advantages of Using CTE?**

* Using CTE improves the readability and enables easy maintenance of complex queries.
* The query can be divided into separate, simple, and logical building blocks, which can be then used to build more complex CTEs until the final result set is generated.
* CTE can be defined in functions, stored procedures, triggers or even views.
* After a CTE is defined, it can be used as a Table or a View and can SELECT, INSERT, UPDATE or DELETE Data.

CLR

In SQL Server 2008, SQL Server objects such as user-defined functions can be created using such CLR languages. This CLR language support extends not only to user-defined functions, but also to stored procedures and triggers. You can develop such CLR add-ons to SQL Server using Visual Studio 2008. ([Read m](https://blog.sqlauthority.com/2008/03/14/sql-server-2005-clr/)

## **What are Synonyms?**

Synonyms give you the ability to provide alternate names for database objects. You can alias object names; for example, using the Employee table as Emp. You can also shorten names. This is especially useful when dealing with three and four part names; for example, shortening server.database.owner.object to object. ([Read more here](https://blog.sqlauthority.com/2008/01/07/sql-server-2005-introduction-and-explanation-to-synonym-helpful-t-sql-feature-for-developer/))

## **What is the Difference between Update Lock and Exclusive Lock?**

When Exclusive Lock is on any process, no other lock can be placed on that row or table. Every other process have to wait till Exclusive Lock completes its tasks.

Update Lock is a type of Exclusive Lock, except that it can be placed on the row which already has Shared Lock on it. Update Lock reads the data of the row which has the Shared Lock as soon as the Update Lock is ready to change the data it converts itself to the Exclusive Lock.

## **What is the ‘FILLFACTOR’?**

A “FILLFACTOR” is one of the important arguments that can be used while creating an index.

According to MSDN, FILLFACTOR specifies a percentage that indicates how much the Database Engine should fill each index page during index creation or rebuild. Fill-factor is always an integer valued from 1 to 100. The fill-factor option is designed for improving index performance and data storage. By setting the fill-factor value, you specify the percentage of space on each page to be filled with data, reserving free space on each page for future table growth.

Specifying a fill-factor value of 70 would imply that 30 percent of each page will be left empty, providing space for index expansion as data is added to the underlying table. The fill-factor setting applies only when the index is created or rebuilt

## **What is a Covered index?**

It is an index that can satisfy a query just by its index keys without having needed to touch the data pages.

It means that when a query is fired, SQL Server doesn’t need to go to the table to retrieve the rows, but can produce the results directly from the index as the index covers all the columns used in query. [(Read more her](https://blog.sqlauthority.com/2010/03/09/sql-server-improve-performance-by-reducing-io-creating-covered-index/)

69. What are Subscriptions?  
Subscriptions are a functionality with SSRS that pushes or delivers reports to multiple users on an automatic and scheduled basis.

70. What are the types of Subscriptions?  
The types of subscriptions include:  
· Standard Subscriptions-The report is delivered to multiple users in a single rendering format and the recipient information is hardcoded  
· Data Driven Subscriptions- The report is delivered to multiple users in multiple rendering formats and the recipient information is not readily available. The recipient information is in the data source where it must be retrieved.