

Importance of Indices

Group Name: SYS_DB

Owner: Venkata Praveen Abburi

Problem Statement:

The execution of complex queries in database system with large data will be overhead in terms of performance issues. The main performance issues will be the execution speed and time taken to execute a complex query.

Tentative Solution:

This type of performance issues in the real time databases can be overcome by using indices concept. Ordered Indices help us to improve the performance of system in terms of speed and time. In our scenario we will use joins and filters to implement the complex query, usage of Non-clustered indices helps to improve the performance cost [1].

Planned Tasks:

- In this project we will create Non-clustered indices on adventure works [2] (Sample data provided by Microsoft) database 2017 in SQL server.
- After creating the Non-clustered indices, we will implement complex queries by using following concepts [3].
 1. Joins
 2. Sub query and co-related sub query
 3. Ranking functions
 4. Set operators (union, intersect, union all)
- Evaluating the performance cost.

Tools:

Microsoft SQL Server Management Studio

Conclusion:

Our project will illustrate the improvement of performance cost in terms of speed and time in executing complex query with the implementation of non-clustered indices.

References:

- [1] Bert Wagner, "Clustered vs Nonclustered: "What Index Is Right For My Data?", hackernoon.com , Dec. 18, 2017. [Online]. Available: <https://hackernoon.com/clustered-vs-nonclustered-what-index-is-right-for-my-data-717b329d042c> [Accessed: Feb. 24, 2019].
- [2] Microsoft Developer Network , "Adventure Works Installation and configuration" Available: " <https://docs.microsoft.com/en-us/sql/samples/adventureworks-install-configure?view=sql-server-2017> [Accessed: Feb. 24, 2019].
- [3] Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, *Database System Concepts Sixth Edition*.