What is an index?

The principal purpose of an index is to speed up data retrieval. It does this by reducing (the less pages the better) the number of database data pages (8kb in size) that have to be visited/scanned. Similar to a phone book or a text book, each has an index to help retrieve the data fast by using the index rather than scanning the entire phone book or the text book. There are primarily two types of indexes: clustered indexes and non-clustered indexes. Let's look at an example of each.

When we talk about

In SQL Server, a clustered index determines the physical order of data in a table. There can be only one clustered index per table

An index is a copy of selected columns of data from a table that can be searched very efficiently that also includes a low-level disk block address or direct link to the complete row of data it was copied from.

Some databases extend the power of indexing by letting developers create indexes on functions or expressions.

An index is a copy of selected columns of data from a table that can be searched very efficiently that also includes a low-level disk block address or direct link to the complete row of data it was copied from.

Some databases extend the power of indexing by letting developers

What is clustered index and non-clustered index?

Therefore table can have only one clustered index. ...

A non-clustered index is a special type of index in which the logical order of the index does not match the physical stored order of the rows on disk.

The leaf node of a nonclustered index does not consist of the data pages. Instead, the leaf nodes contain index rows.

According to the above queries first_name, last_name columns can be indexed as they are located in the WHERE clause.

Also an additional field, country_id from country table, can be considered for indexing because it is in a JOIN clause.

So indexing can be considered on every field in the WHERE clause or a JOIN clause.

An index is a distinct structure in the database that is built using the create index statement. It requires its own disk space and holds a copy of the indexed table data.

That means that an index is pure redundancy. Creating an index does not change the table data; it just creates a new data structure that refers to the table.

A database index is, after all, very much like the index at the end of a book: it occupies its own space, it is highly redundant, and it refers to the actual information stored in a different place.