NON CLUSTERED INDEX GUIDE

- A nonclustered index contains the index key values and row locators that point to the storage location of the table data
- As discussed, we can have multiple nonclustered indexes on a table or indexed view
- Nonclustered indexes should be designed for frequently used queries
- Nonclustered indexes are optimal choice for EXACT match queries because the index contains entries describing the exact location in the table
- Those tables with low update requirements -databases that contain heavily updated tables should avoid over-indexing.
- Read-only data can benefit from many nonclustered indexes
- OLTP applications and large numbers of indexes on a table affect the performance of INSERT, UPDATE, DELETE, and MERGE statements BECAUSE all indexes must be adjusted appropriately as data in the table changes
- Use JOIN or GROUP BY clauses
- Create multiple nonclustered indexes on columns involved in join and grouping operations
- Create a non cluster a column(s) frequently involved in search conditions of a query using WHERE clause
- Create a non cluster a column(s) to cover the query (meaning if the where clause has more than one predicate, cover those columns)
- When the query optimizer can locate all the column values within the index; table or clustered index data is not accessed resulting in fewer disk I/O operations.
- Create non clustered on distinct values, such as a combination of last name and first name
- other