**Types of Indexes:**

* **Bitmap Index:**

Bitmap Indexing is a special type of database indexing that uses **bitmaps**. This technique is used for huge databases, when column is of low cardinality and these columns are most frequently used in the query.

In Bitmap Indexing these bits are used to represent the unique values in those low cardinality columns. This technique of storing the low cardinality rows in form of bits are called bitmap indices.

Table

Description automatically generatedTable

Description automatically generated

Table

Description automatically generatedGraphical user interface

Description automatically generated with medium confidence

**Advantages –**

* Efficiency in terms of insertion deletion and updation.
* Faster retrieval of records

**Disadvantages –**

* Only suitable for large tables
* Bitmap Indexing is time consuming
* **Clustered and Non-Clustered Index:**

Clustered index sorts and stores the rows data of a table / view based on the order of clustered index key. Clustered index key is implemented in B-tree index structure.

 A non-clustered index is created using clustered index. Each index row in the non-clustered index has non clustered key value and a row locator. Locator positions to the data row in the clustered index that has key value.

Clustered vs. Non-clustered index in SQL server is that the non-clustered index stores the data at one area and indices at another area, while the clustered index is a kind of index that sorts the data rows in the table on their key values.

🡪 Both clustered and non-clustered indexes contain only keys and record identifiers in the index structure. The record identifiers always point to rows in the data pages. With clustered indexes, the database manager attempts to keep the data in the data pages in the same order as the corresponding keys in the index pages.

|  |  |
| --- | --- |
| Clustered index is faster | Non-clustered index is slower. |
| A table can have only one clustered index | A table can have multiple non-clustered index. |
| Clustered index store pointers to block not data | Non-Clustered index store both value and a pointer to actual row that holds data |
| Primary Keys of the table by default are clustered index | Composite key when used with unique constraints of the table act as non-clustered index. |

* **Local and Global Index:**

The first partitioned index method is called a LOCAL partition. A local partitioned index creates a one-for-one match between the indexes and the partitions in the table. Of course, the key value for the table partition and the value for the local index must be identical. The second method is called GLOBAL and allows the index to have any number of partitions.

Local partitioned indexes allow the DBA to take individual partitions of a table and indexes offline for maintenance (or reorganization) without affecting the other partitions and indexes in the table.

* **Spatial Index:**

A spatial index provides the ability to perform certain operations more efficiently on spatial objects (data) in a column of the geometry data type . The spatial index reduces the number of objects on which relatively costly spatial operations need to be applied

References :

[Difference between Clustered and Non-clustered index - GeeksforGeeks](https://www.geeksforgeeks.org/difference-between-clustered-and-non-clustered-index/)

[Bitmap Indexing in DBMS - GeeksforGeeks](https://www.geeksforgeeks.org/bitmap-indexing-in-dbms/?ref=lbp)

[6 Indexes (oracle.com)](https://docs.oracle.com/cd/B12037_01/server.101/b10736/indexes.htm#i1006549)

[Oracle Global Index vs. Local Index (dba-oracle.com)](http://dba-oracle.com/t_global_local_partitioned_index.htm)

[Partitioned tables and indexes - SQL Server, Azure SQL Database, Azure SQL Managed Instance | Microsoft Learn](https://learn.microsoft.com/en-us/sql/relational-databases/partitions/partitioned-tables-and-indexes?view=sql-server-ver16)