

## What Is a Graph Database?

A graph database is a type of NoSQL database that stores data in collections of nodes and edges instead of a series of columns and rows. Graphs contain nodes, edges, and properties

## What is the most popular graph database?

Neo4j, Cassandra, Microsoft SQL Server 2017 etc...

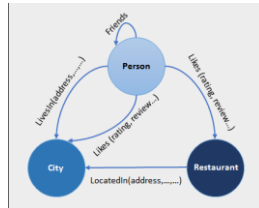


Diagram for graph database

## What Are the Major Advantages OR POWER of Using a Graph Database?

- Object-Oriented Thinking
- Performance
- Better Problem-Solving
- Update Data in Real-Time and Support Queries Simultaneously
- Flexible Online Schema Environment
- Group by Aggregate Queries
- Combine and Hierarchize Multiple Dimensions

## What is Indexing?

Indexing is used to optimize the performance of a database by minimizing the number of disk accesses required when a query is processed. It is a data structure technique which is used to quickly locate and access the data in a database.

## Types of Single-Level Ordered indexes

- Primary Indexes
- Clustering Indexes
- Secondary Indexes

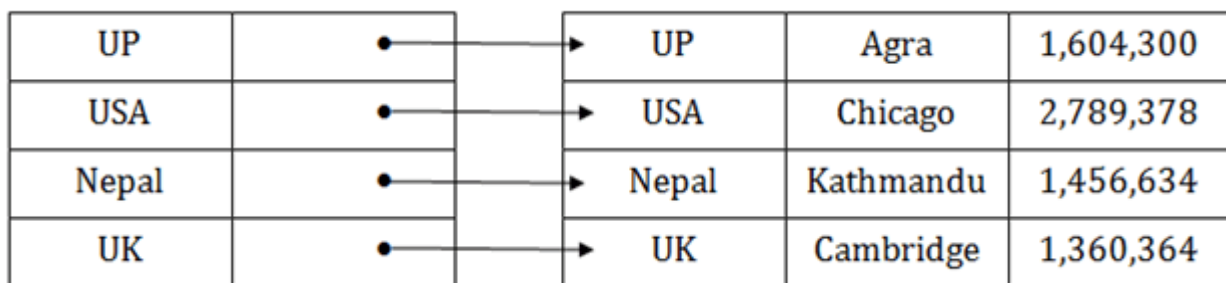
## What is Primary Index

If the index is based on the primary key, it is called the primary index. Those keys are unique to each record. Primary index is defined on an ordered data file. The data file is ordered on a key field. The key field is generally the primary key of the relation.

The primary index can be classified into two types: Dense index and sparse index.

## Dense index

- The dense index contains an index record for every search key value in the data file. It makes searching faster.
- In this, the number of records in the index table is same as the number of records in the main table.

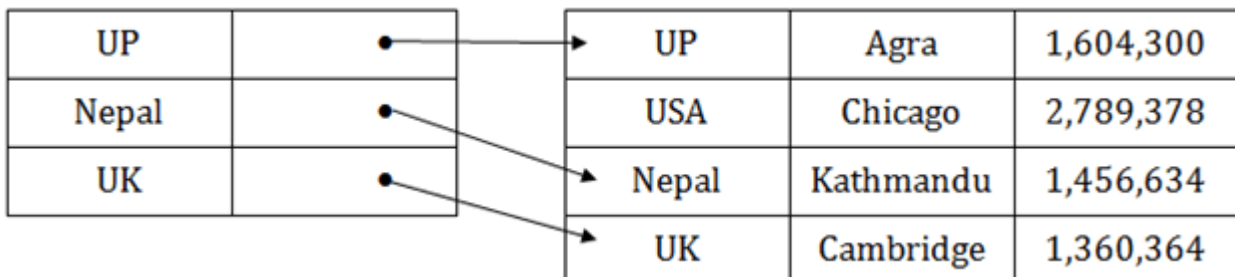


The diagram illustrates a dense index. On the left is an index table with four rows, each containing a search key and a pointer (represented by a black dot). Arrows point from each pointer to a corresponding row in the main data table on the right. The main table has five columns: the first column contains the search key, the second contains the city name, and the third contains the population.

UP	●	→	UP	Agra	1,604,300
USA	●	→	USA	Chicago	2,789,378
Nepal	●	→	Nepal	Kathmandu	1,456,634
UK	●	→	UK	Cambridge	1,360,364

## Sparse index

- In the data file, index record appears only for a few items. Each item points to a block.
- In this, instead of pointing to each record in the main table, the index points to the records in the main table in a gap.



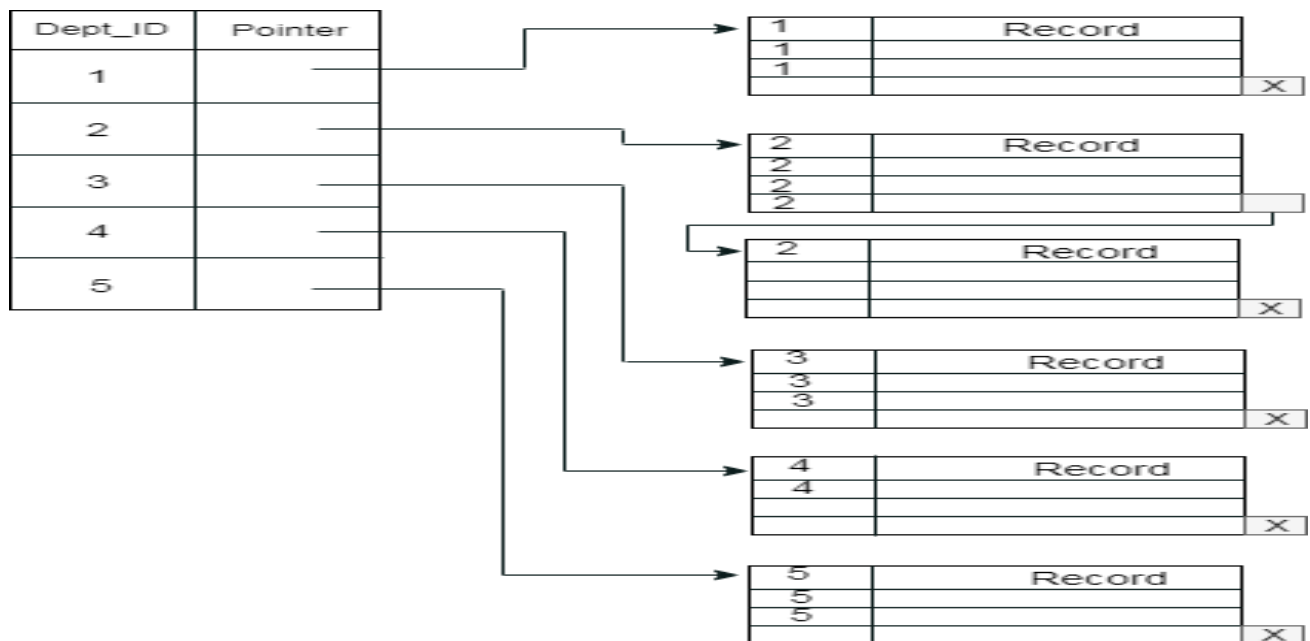
The diagram illustrates a sparse index. The index table on the left has three rows, each with a search key and a pointer. Arrows point from the pointers to specific rows in the main data table on the right. Notably, the pointer from 'Nepal' in the index points to the row for 'UK' in the main table, indicating a gap in the index.

UP	●	→	UP	Agra	1,604,300
Nepal	●	→	UK	Cambridge	1,360,364
UK	●	→	Nepal	Kathmandu	1,456,634

## Clustering Index

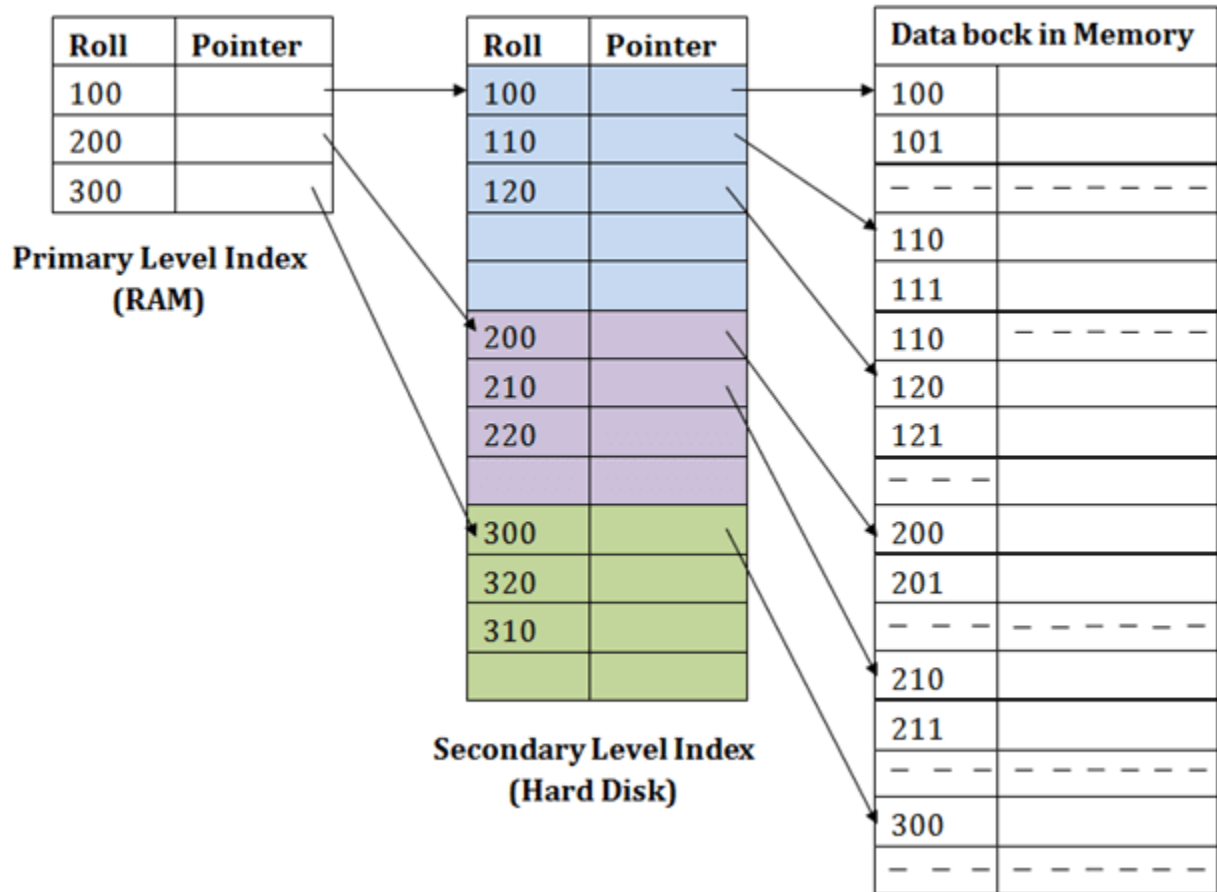
- A clustered index can be defined as an ordered data file. Sometimes the index is created on non-primary key columns which may not be unique for each record.
- In this case, to identify the record faster, we will group two or more columns to get the unique value and create index out of them. This method is called a clustering index.
- The records which have similar characteristics are grouped, and indexes are created for these group.

**Example:** suppose a company contains several employees in each department. Suppose we use a clustering index, where all employees which belong to the same Dept\_ID are considered within a single cluster, and index pointers point to the cluster as a whole. Here Dept\_Id is a non-unique key.



## Secondary Index

In secondary indexing, to reduce the size of mapping, another level of indexing is introduced. In this method, the huge range for the columns is selected initially so that the mapping size of the first level becomes small. Then each range is further divided into smaller ranges. The mapping of the first level is stored in the primary memory, so that address fetch is faster. The mapping of the second level and actual data are stored in the secondary memory (hard disk).



### What is Multilevel Index?

The multilevel indexing segregates the main block into various smaller blocks so that the same can be stored in a single block. The outer blocks are divided into inner blocks which in turn are pointed to the data blocks. This can be easily stored in the main memory with fewer overheads.

