

## Definition

- An index is a data structure that can be perceived as a table of contents that points us to the location where actual data lives. So when we create an index on a column of a table, we store that column and a pointer to the whole row in the index.
- Indexes are used to quickly locate data without having to search every row in a database table every time a database table is accessed.
- The goal of creating an index on a particular table in a database is to make it faster to search through the table and find the row or rows that we want.
- Indexes can be created using one or more columns of a database table, providing the basis for both rapid random lookups and efficient access to ordered records.

## Example: A library Catalog

- A library catalog is a register that contains the list of books found in a library.
- The catalog is organized like a database table generally with four columns:
  - book title
  - writer
  - subject
  - publication date
- There are usually two such catalogs: one sorted by the book title and one sorted by the writer's name.
- These catalogs are like indexes for the database of books. They provide a sorted list of data easily searchable by relevant information.
- Let's assume a table containing a list of books, the following diagram shows how an index on the 'Title' column looks like:

Index		Table		
Search Key	Pointer	Title	Writer	Date
Database indexes		Intro to databases	Michele Clark	Dec 2, 2017
Intro to computers		Database indexes	Adam Cambel	Nov, 14, 2016
Intro to databases		Intro to computers	Nickolas Homes	Feb 5, 2018
Intro to software				

- Just like a traditional relational data store, we can also apply this concept to larger datasets.
- The trick with indexes is that we must carefully consider how users will access the data. In the case of data sets that are many terabytes in size, but have very small payloads (e.g., 1 KB), indexes are a necessity for optimizing data access.
- Finding a small payload in such a large dataset can be a real challenge since we can't possibly iterate over that much data in any reasonable time.
- Furthermore, it is very likely that such a large data set is spread over several physical devices—this means we need some way to find the correct physical location of the desired data. Indexes are the best way to do this.

## How Indexes Decrease Write Performance

- An index can dramatically speed up data retrieval but may itself be large due to the additional keys, which slow down data insertion & update.
- When adding rows or making updates to existing rows for a table with an active index, we not only have to write the data but also have to update the index. This will decrease the write performance.
- This performance degradation applies to all insert, update, and delete operations for the table.
- For this reason, adding unnecessary indexes on tables should be avoided and indexes that are no longer used should be removed.
- To reiterate, adding indexes is about improving the performance of search queries.
- If the goal of the database is to provide a data store that is often written to and rarely read from, in that case, decreasing the performance of the

more common operation, which is writing, is probably not worth the increase in performance we get from reading.