In this thread I would like to discuss MongoDB indexing. I will define what indexes are. I will discuss why indexes are used and how they are created in MongoDB. I will define what index cardinality is. I will talk about capped collections and when they should be used as well as provide an example. I will also discuss what sparse indexes are. So, what are indexes in MongoDB?

“Indexes are special data structures that store a small portion of the collection's data set in an easy to traverse form.” (Indexes — MongoDB Manual, n.d.). In MongoDB “indexes support the efficient execution of queries in MongoDB. Without indexes, MongoDB must perform a collection scan, i.e., scan every document in a collection, to select those documents that match the query statement. If an appropriate index exists for a query, MongoDB can use the index to limit the number of documents it must inspect. The index stores the value of a specific field or set of fields, ordered by the value of the field. The ordering of the index entries supports efficient equality matches and range-based query operations. In addition, MongoDB can return sorted results by using the ordering in the index.” (Indexes — MongoDB Manual, n.d.). To create an index in MongoDB one can use the flowing code below:

Text

Description automatically generated

This example will allow a database user to create a single key descending index on the “lastName” field.

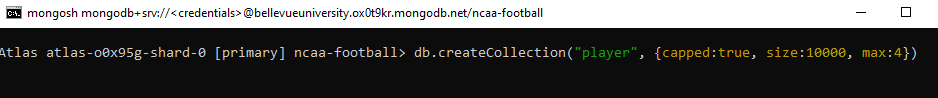
“Cardinality is defined to be number of unique elements present in a set. The lower the cardinality, the more duplicated elements. If a set has 5 elements made of boolean values, then the cardinality of the set is going to be two. So, all sets made of booleans will have a max cardinality of two and a min cardinality of one.” (Understanding Indexing and Cardinality for MongoDB, n.d.).

“[Capped collections](https://www.mongodb.com/docs/upcoming/reference/glossary/#std-term-capped-collection) are fixed-size collections that support high-throughput operations that insert and retrieve documents based on insertion order. Capped collections work in a way similar to circular buffers: once a collection fills its allocated space, it makes room for new documents by overwriting the oldest documents in the collection.” (Capped Collections — MongoDB Manual, n.d.). Capped collections should be used to:

* Store log information generated by high-volume systems. Inserting documents in a capped collection without an index is close to the speed of writing log information directly to a file system. Furthermore, the built-in first-in-first-out property maintains the order of events, while managing storage use.
* Cache small amounts of data in a capped collections. Since caches are read rather than write heavy, you would either need to ensure that this collection alwaysremains in the working set (i.e. in RAM) or accept some write penalty for the required index or indexes.

(Capped Collections — MongoDB Manual, n.d.).

In the following example, we are working with the “ncaa-football” database in which we are creating a new capped collection named “player” with a maximum document capacity “4” using createCollection() method:



“The [sparse](https://www.mongodb.com/docs/manual/core/index-sparse/) property of an index ensures that the index only contain entries for documents that have the indexed field. The index skips documents that do not have the indexed field. You can combine the sparse index option with the unique index option to prevent inserting documents that have duplicate values for the indexed field(s) and skip indexing documents that lack the indexed field(s).” (Indexes — MongoDB Manual, n.d.).

References:

*Indexes — MongoDB Manual*. (n.d.). Retrieved November 20, 2022, from <https://www.mongodb.com/docs/manual/indexes/>

*Understanding indexing and cardinality for MongoDB*. (n.d.). Bharat Kalluri. Retrieved November 20, 2022, from <https://bharatkalluri.com/posts/cardinality-and-indexing-mongodb>

*Capped Collections — MongoDB Manual*. (n.d.). Retrieved November 20, 2022, from <https://www.mongodb.com/docs/upcoming/core/capped-collections/>