

Fara indexi:

- MongoDB trebuie sa citeasca fiecare document din collectie

- daca e necesar sa se faca sort, trebuie sa sorteze datele in memorie



* In mod default, MongoDB creaza doar un index per collection, si logic ca la \_id
* Totusi, indexes vin si cu aspecte negative:

- fiecare insert sau update la un document face ca index data structure sa fie modificata

- prea multi indexi pot avea consecinte negative

**Index types**

* **Single field** – index pe un single field doar
* **Compound** – index pe mai multe fields
* **Multikey** – index care are un field array

- db.collection.getIndexes() – arata indexii colectiei

- db.collection.explain().find(...) –arata ce indexi s-au folosit in query. Arata defapt execution plan si putem vedea anumite sugestii bune.

De ex:

....

**winningPlan: {**

**stage: 'FETCH',**

**inputStage: {**

**stage: 'IXSCAN',**

**keyPattern: { name: 1 },**

**indexName: 'name\_1',**

**isMultiKey: false,**

**multiKeyPaths: { name: [] },**

**isUnique: false,**

**isSparse: false,**

**isPartial: false,**

**indexVersion: 2,**

**direction: 'forward',**

**indexBounds: { name: [ '["Facebook", "Facebook"]' ] }**

**}**

**.....**

Asta inseamna ca intai se incearca ‘FETCH’, adica sa se citeasca din toate documentele rand pe rand, insa s-a gasit un index si s-au scanat datele din el, de aia avem ‘IXSCAN’. Deci, nu au trebuit scanate documentele rand pe rand

stage:

* The **IXSCAN** stage indicates the query is using an index and what index is being selected.
* The **COLLSCAN** stage indicates a collection scan is perform, not using any indexes. So, there’s no index uses for this query
* The **FETCH** stage indicates documents are being read from the collection.
* The **SORT** stage indicates documents are being sorted in memory. An index for this column should be created

**Single field index**

* **db.collection.createIndex({fieldName : 1})**

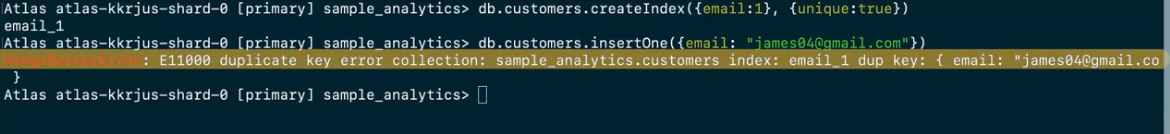
1 – inseamna ca index va stoca datele in oride ascendenta

0 – descendent

* **db.collection.createIndex({fieldName : 1}, {unique: true})**

- unique : true spune ca valorile acestui field sunt unice per document si asa index e si mai rapid

Totodata, daca folosim unique : true, nu se va mai permite inserarea unui document ce are acest field cu o valoare deja existenta



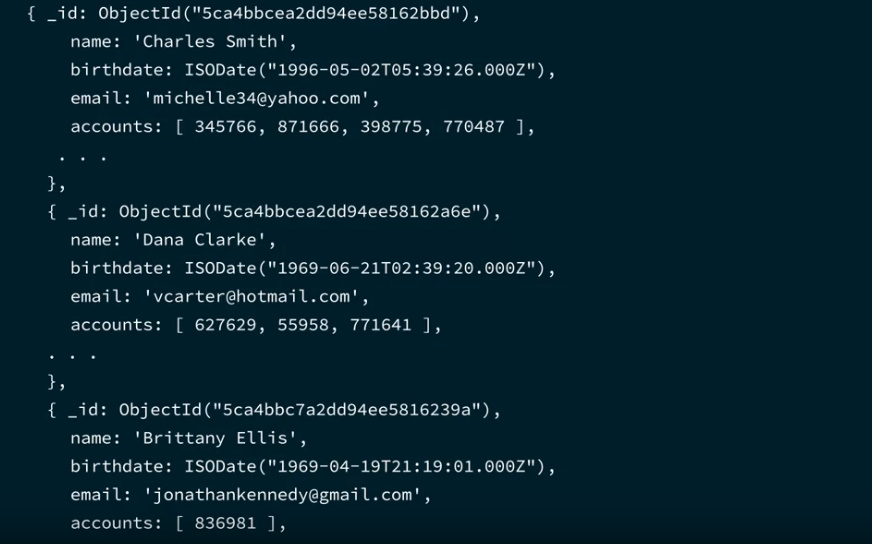
**MultiKey index**

* Este cand avem un array field printre fieldurile din index
* Array poate fi de :

- date primitive

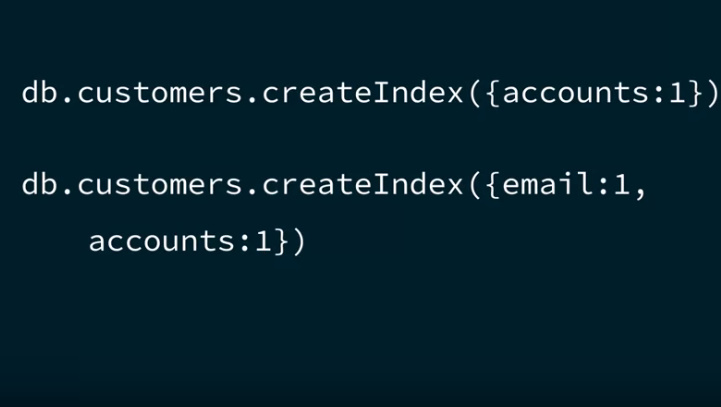
- subdocumente(objects)

- subarrays





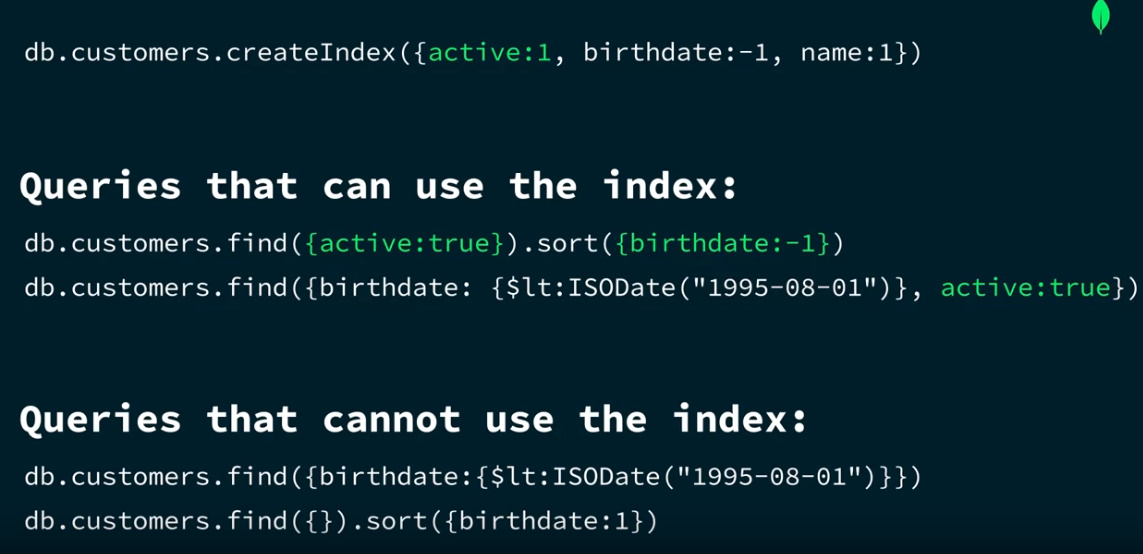
* Cream index la fel cu .createIndex() si includem in el field array





* Vedem ca putem include mai multe fielduri intr-un index, si un field sa fie array
* **ATENTIE! Intr-un index putem avea doar un array field**

**Compound indexes**

* Sunt indexes ce contin mai multe fields
* Pot contine maxim un field array
* 

Primul field e cel principal. Urmatoarele se folosesc daca se intalnesc documents ce au primul field egal.

Deci, un query care nu foloseste primul field din index nu va folosi acest index