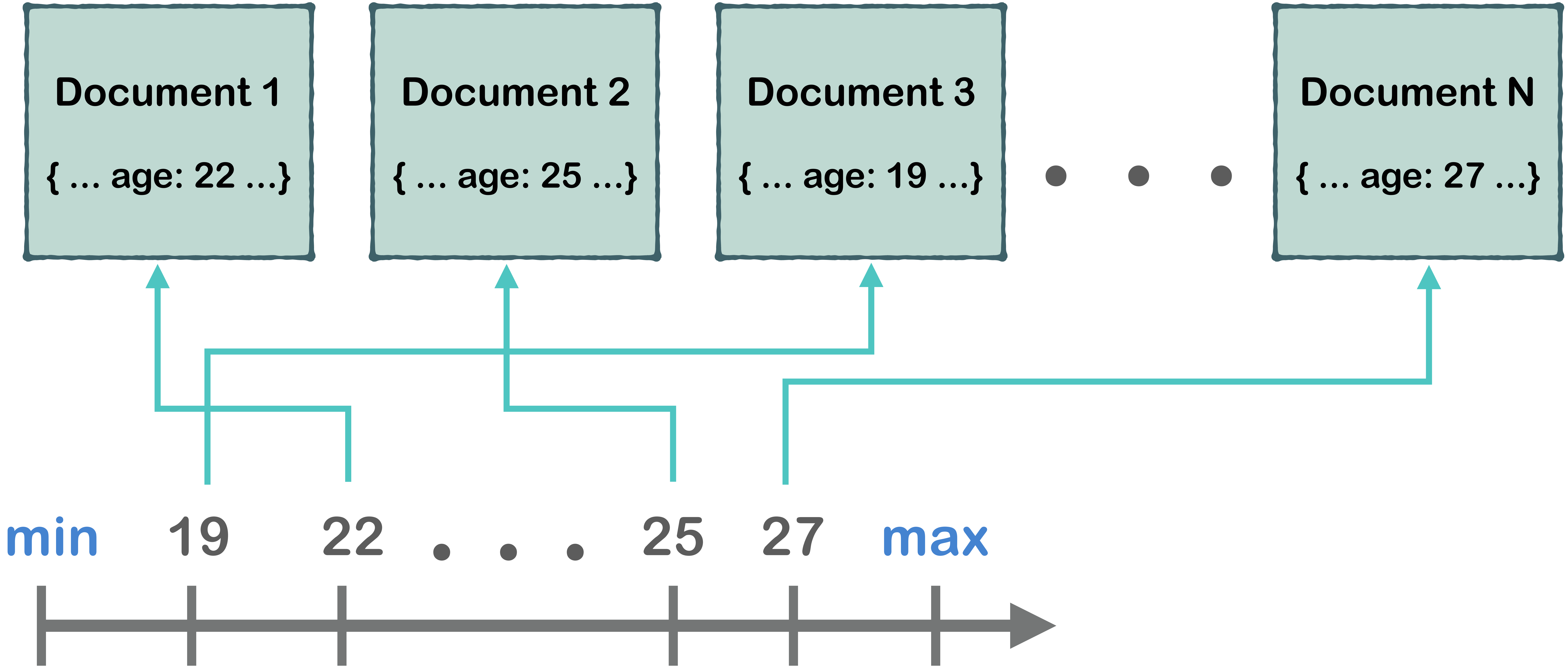
INDEXES

Indexes Overview

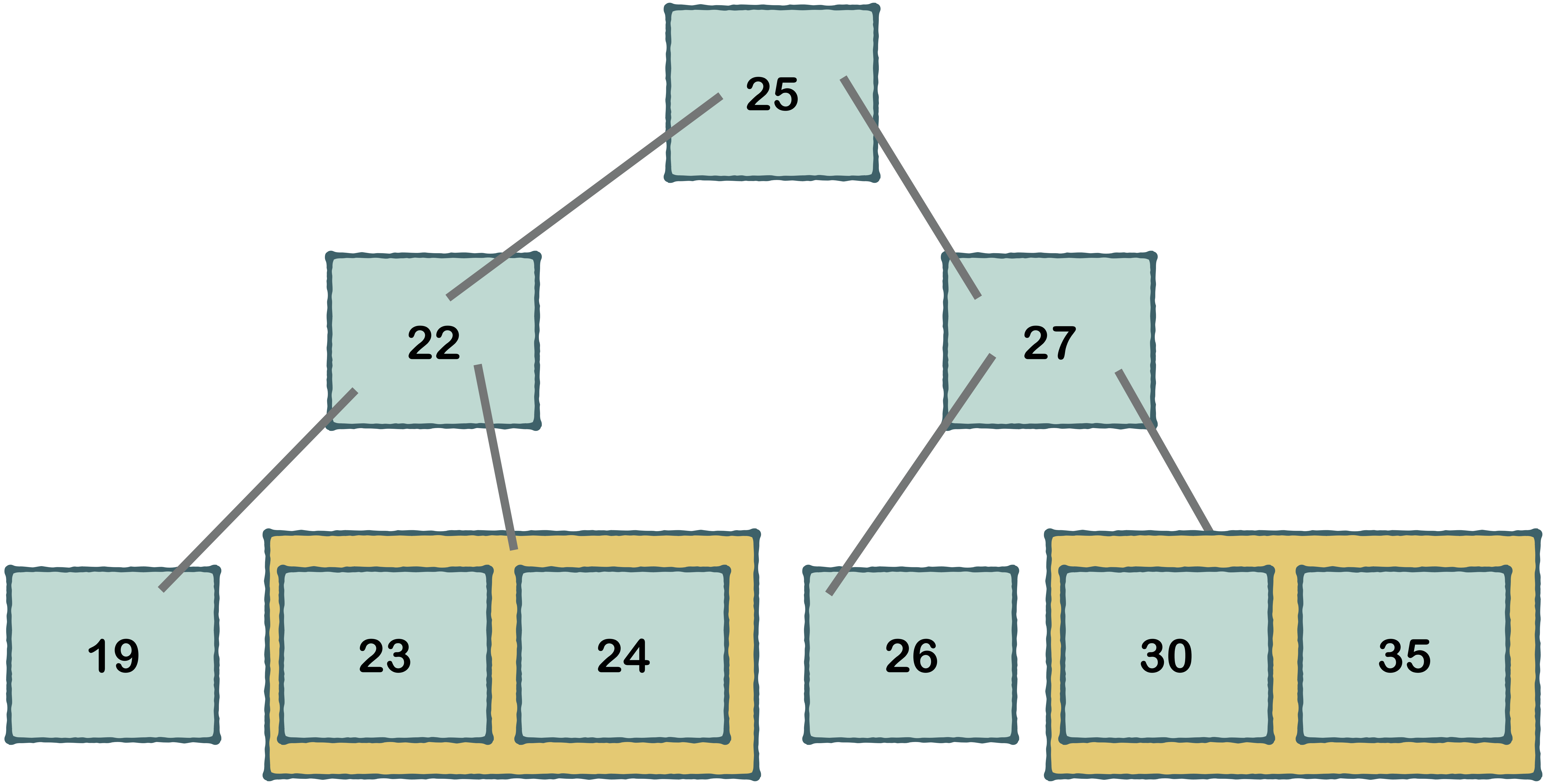
- Indexes improve MongoDB query execution
- Without index whole collection must be scanned (COLLSCAN)
- Index stores sorted field values
- If appropriate index exists, MongoDB performs only index scan (IXSCAN)

Index Creation Process

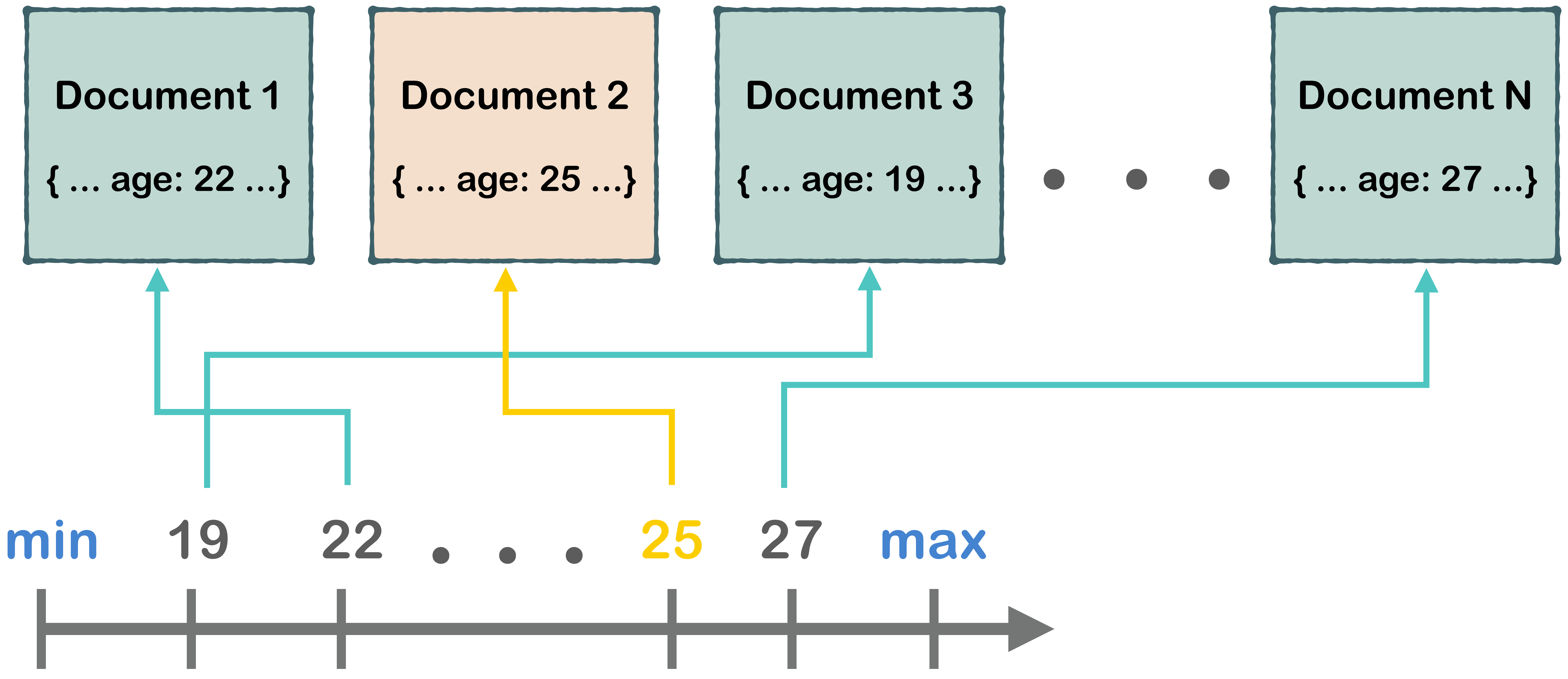


Key { age: 1 }

B-Tree index data structure

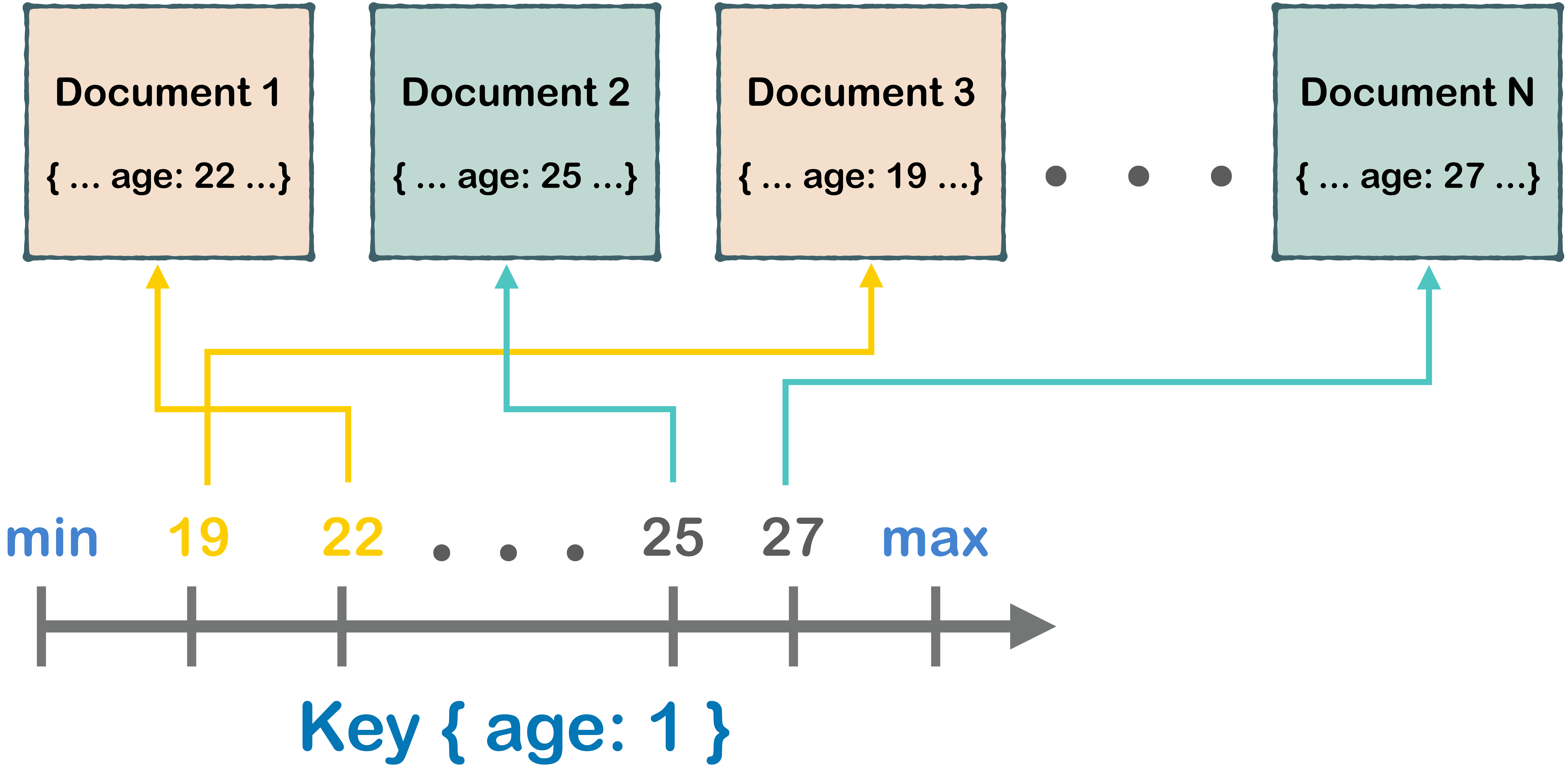


Query { age: 25}



Key { age: 1 }

Query { age: { $lte: 22 } }



Default _id index

- { _id: 1 } is default index in each MongoDB collection

- Name of this index is _id_

- Default _id index is unique

```
{
    "v" : 2,
    "key" : {
        "_id" : 1
    },
    "name" : "_id_",
    "ns" : "myDb.persons"
}
```

getIndexes()

- Returns current indexes for certain collection

```
db.<collectionName>.getIndexes()
```

↓

```
[
    {
        "v" : 2,
        "key" : {
            "_id" : 1
        },
        "name" : "_id_",
        "ns" : "myDb.persons"
    }
]
```

Create new Index

- **Writes resulting documents to the MongoDB collection**

```
db.collection.createIndex( { <keyname>: [-1 | 1] }, <options> )
```

- **Examples**

```
db.persons.createIndex( { age: 1 } )
db.persons.createIndex( { name: 1 } )
```

Index Creation Options

- Create index in the background. Other operations will not be blocked

```
{ background: true }
```

- Create unique index

```
{ unique: true }
```

- Specify name for the index

```
{ name: "<indexName>" }
```

Example 1: Create Unique index

```
db.persons.createIndex(
    { index: 1 },
    { unique: true }
)
```

```
{
  "v" : 2,
  "unique" : true,
  "key" : {
      "index" : 1
  },
  "name" : "index_1",
  "ns" : "myDb.persons"
}
```

```
{
  "createdCollectionAutomatically" : false,
  "numIndexesBefore" : 1,
  "numIndexesAfter" : 2,
  "ok" : 1
}
```

Example 2: Create index in background

```
db.persons.createIndex(
    { name: 1 },
    { background: true }
)
```

```
{
  "v" : 2,
  "key" : {
      "name" : 1
  },
  "name" : "name_1",
  "ns" : "myDb.persons",
  "background" : true
}
```

```
{
  "createdCollectionAutomatically" : false,
  "numIndexesBefore" : 2,
  "numIndexesAfter" : 3,
  "ok" : 1
}
```

Example 3: Index with custom name

```
db.persons.createIndex(
    { age: 1 },
    { name: "customAgeIndex" }
)
```

```
{
  "v" : 2,
  "key" : {
      "age" : 1
  },
  "name" : "customAgeIndex",
  "ns" : "myDb.persons"
}
```

```
{
  "createdCollectionAutomatically" : false,
  "numIndexesBefore" : 3,
  "numIndexesAfter" : 4,
  "ok" : 1
}
```

Analyze query performance

- **Returns information about the query**

```
db.<collectionName>.explain().<method>

db.<collectionName>.explain("executionStats").<method>
```

- **Examples**

```
db.persons.explain().find({"age": {$gt: 25}})

db.persons.explain().aggregate([{$group: {_id:

"$country"}}])
```

Example 4: Query performance with index

```
{
  "queryPlanner" : {
    "plannerVersion" : 1,
    "namespace" : "myDb.persons",
    "indexFilterSet" : false,
    "parsedQuery" : {
        "age" : {
            "$gt" : 25.0
        }
    },
    "winningPlan" : {
        "stage" : "FETCH",
        "inputStage" : {
            "stage" : "IXSCAN",
            "keyPattern" : {
                "age" : 1.0
            },
      ...
```

```
db.persons
  .explain()
  .find({"age": {$gt: 25}})
```

Example 5: ExecutionStats with Index

```
"executionStats" : {
        "executionSuccess" : true,
        "nReturned" : 258,
        "executionTimeMillis" : 0,
        "totalKeysExamined" : 258,
        "totalDocsExamined" : 258,
        "executionStages" : {
            "stage" : "FETCH",
            "nReturned" : 258,
            "executionTimeMillisEstimate" : 0,
            "works" : 259,
            "advanced" : 258,
            "needTime" : 0,
            "needYield" : 0,
         ...
```

```
db.persons
   .explain("executionStats")
   .find({"age": {$lt: 25}})
```

Example 6: Query performance without index

```
{
  "queryPlanner" : {
      "plannerVersion" : 1,
      "namespace" : "myDb.persons",
      "indexFilterSet" : false,
      "parsedQuery" : {
          "gender" : {
              "$eq" : "female"
          }
      },
      "winningPlan" : {
          "stage" : "COLLSCAN",
          "filter" : {
              "gender" : {
                  "$eq" : "female"
              }
          },
          "direction" : "forward"
      },
        ...
```

```
db.persons
   .explain()
   .find({"gender": "female"})
```

Example 7: ExecutionStats without Index

```
"executionStats" : {
    "executionSuccess" : true,
    "nReturned" : 507,
    "executionTimeMillis" : 0,
    "totalKeysExamined" : 0,
    "totalDocsExamined" : 1000,
    "executionStages" : {
        "stage" : "COLLSCAN",
        "filter" : {
            "gender" : {
                "$eq" : "female"
            }
        },
        "nReturned" : 507,
        "executionTimeMillisEstimate" : 0,
        "works" : 1002,
        "advanced" : 507,
        "needTime" : 494,
        "needYield" : 0,
      ...
```

```
db.persons
  .explain("executionStats")
  .find({"gender": "male"})
```

Example 8: ExecutionStats with RegExp

```
"executionStats" : {
    "executionSuccess" : true,
    "nReturned" : 148,
    "executionTimeMillis" : 0,
    "totalKeysExamined" : 1000,
    "totalDocsExamined" : 148,
    "executionStages" : {
        "stage" : "FETCH",
        "nReturned" : 148,
        "executionTimeMillisEstimate" : 1,
        "works" : 1001,
        "advanced" : 148,
        "needTime" : 852,
        "needYield" : 0,
      ...
```

```
db.persons
  .explain("executionStats")
  .find({"name": /el/i})
```

Delete Indexes

- Deletes certain index

```
db.collection.dropIndex( { <fieldName>: 1 } )
db.collection.dropIndexes()
```

- Examples

```
db.persons.dropIndex( { age: 1 } )
db.persons.dropIndexes()
```

SUMMARY

- MongoDB Indexes
- Default _id Index
- Create custom single-field Indexes
- Unique Indexes
- Queries Performance
- Delete indexes