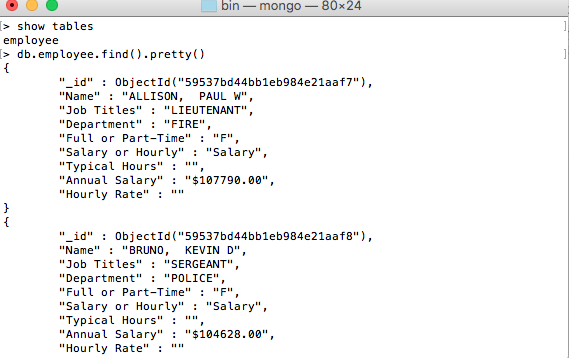
1. Downloaded CSV from <https://catalog.data.gov/dataset?res_format=csv>
2. Loading the data in MongoDB with DB name demo and collection name same as file name.

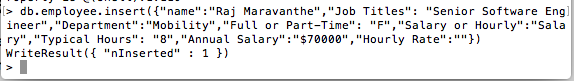
mongoimport -d demo -c employee --type csv --file /Users/rmar0010/Downloads/employee.csv –-headerline

Imported screenshot shown below.

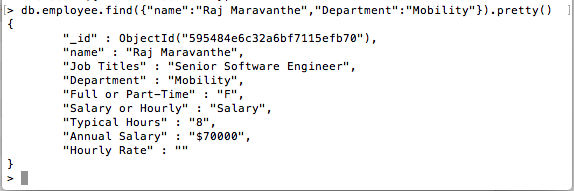


1. CRUD operations.

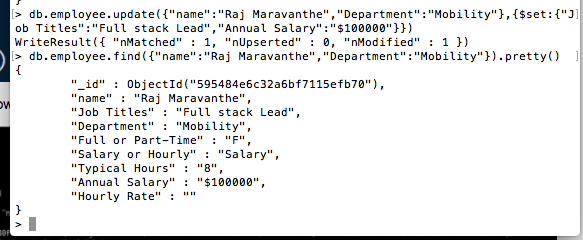
Insert operation is show below.



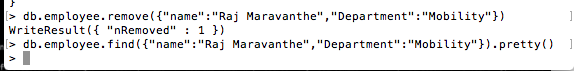
Read operation for above insertion show below.



Update operation for above insertion show below.



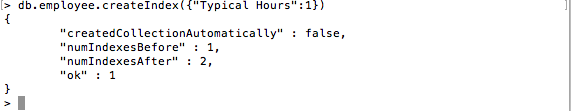
Delete operation for above data shown below.



1. Creating Indexes

Simple index is created show below.

Using command -> db.employee.createIndex({“Typical Hours”:1})



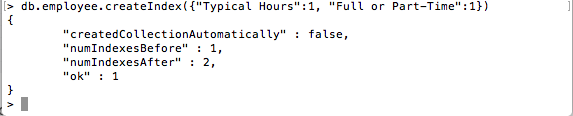
Above information will be fast in order to execute the below queries.

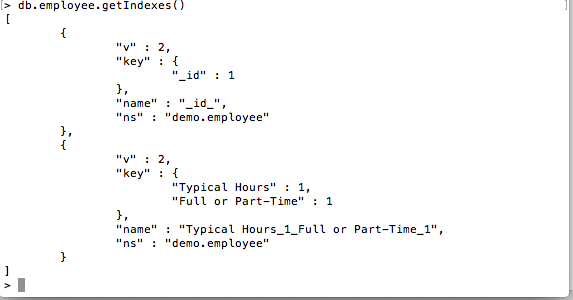
db.employee.find({“Typical Hours”:{$gt:6}})

db.employee.find({“Typical Hours”:6})

Compound Indexes : -

Db.employee.createIndex({“Typical Hours”:1, “Full or Part-Time”:1})





query example

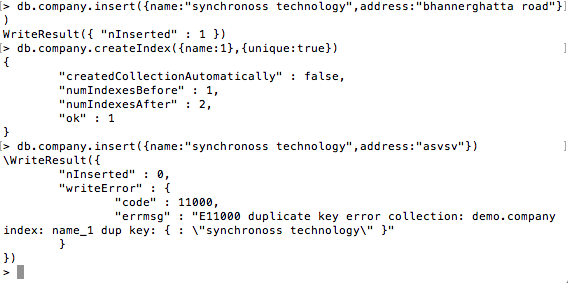
db.employee.find({"Typical Hours":{$lt:"7"},"Full or Part-Time":"F"})

Geospatial Indexes : -

Search the indexes with coordinate option. Example is given below.

db.employee.find( {coords:[50,50]} ) using index {a:’2d’}

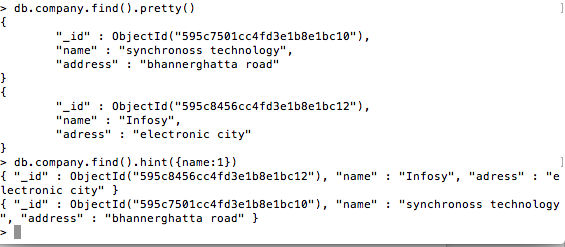
Unique Key Constraints : -



Above example clearly indicates that unique key constraints doesn’t allow duplicate values.

1. Hint Method

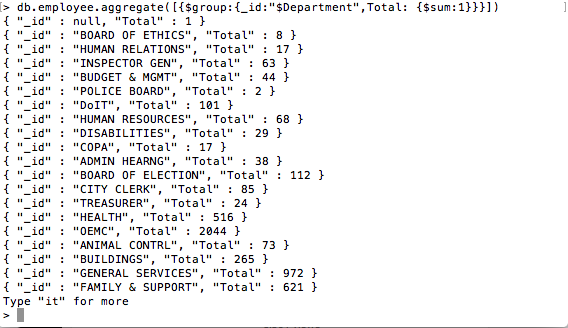
The hint operator forces the query optimizer to use a specific index to fulfill the query. Specify the index either by the index name or by document.



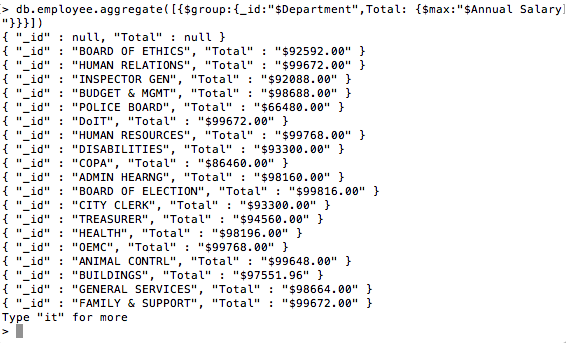
1. Aggregation activities

General Purposes

For Sum

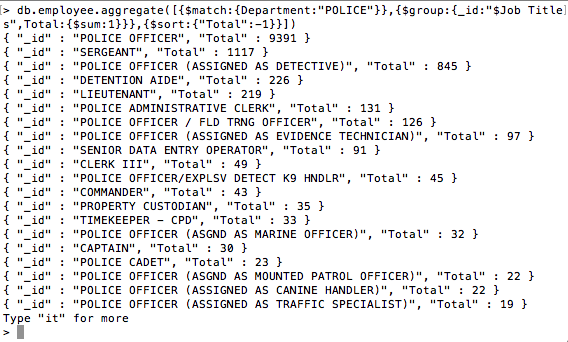


Max or Min



Pipeline

Pipeline is the aggregate which can be made in step by step.



Map Reduce

function map() {

emit(this.Department, this.Name);

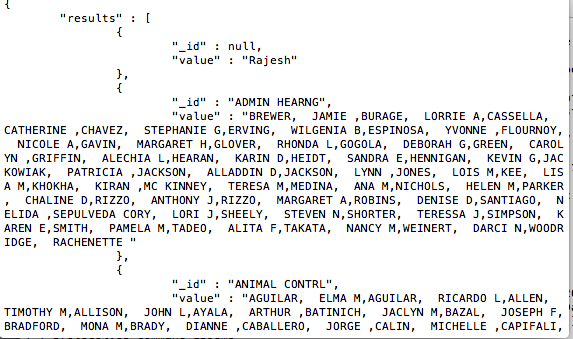
}

function reduce(key, value) {

return value.join();

}

db.employee.mapReduce(map, reduce, {out:{inline:1}})



other example.

function map() {

var x = parseInt(this.annualsalary);

emit(this.Department, x);

}

function reduce(key, value) {

return Array.sum(value);

}

db.employee.mapReduce(map, reduce, {out:{inline:1}})

