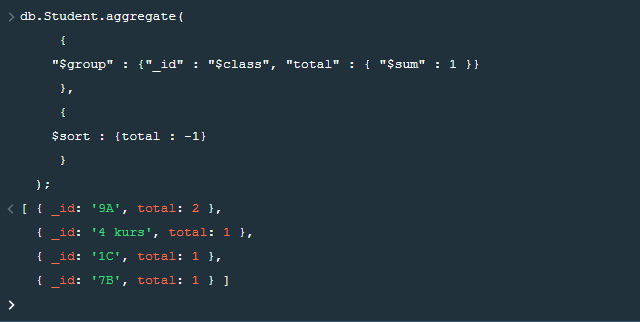
Группировка значений по полю класса и сортировка получившихся значений по убыванию.



db.Student.aggregate(

{

"$group" : {"\_id" : "$class", "total" : { "$sum" : 1 }}

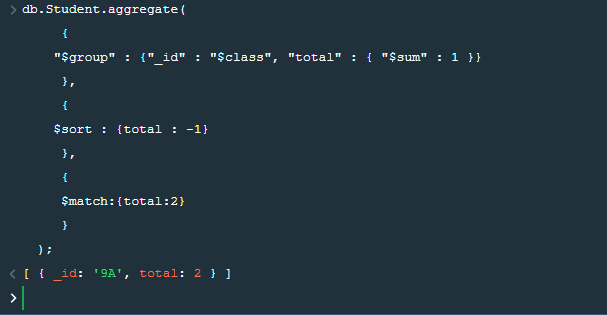
},

{

$sort : {total : -1}

}

);



db.Student.aggregate(

{

"$group" : {"\_id" : "$class", "total" : { "$sum" : 1 }}

},

{

$sort : {total : -1}

},

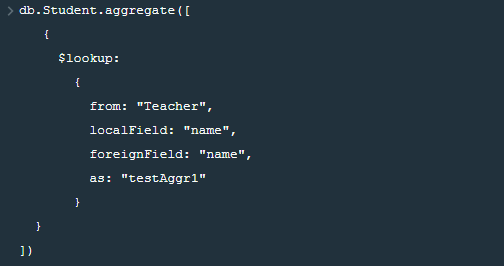
{

$match:{total:2}

}

);

Пример группировки через lookup.



db.Student.aggregate([

{

$lookup:

{

from: "Teacher",

localField: "name",

foreignField: "name",

as: "testAggr1"

}

}

])

Результат:

[ { \_id:

{ \_bsontype: 'ObjectID',

id: <Buffer 5f 74 55 7f 50 a3 e5 24 1c 30 43 b2> },

name: 'Artem',

class: '9A',

testAggr1: [] },

{ \_id:

{ \_bsontype: 'ObjectID',

id: <Buffer 5f 74 55 8b 50 a3 e5 24 1c 30 43 b3> },

name: 'Ortem',

class: '7B',

testAggr1: [] },

{ \_id:

{ \_bsontype: 'ObjectID',

id: <Buffer 5f 74 55 d4 50 a3 e5 24 1c 30 43 b4> },

name: 'Kolya',

class: '1C',

testAggr1:

[ { \_id:

{ \_bsontype: 'ObjectID',

id: <Buffer 5f 8e d4 4f 7c c9 b2 1c 70 ab 8b 0d> },

name: 'Kolya',

cabNum: 105,

exp: '0 year',

isZavUch: false } ] },

{ \_id:

{ \_bsontype: 'ObjectID',

id: <Buffer 5f 74 55 ed 50 a3 e5 24 1c 30 43 b5> },

name: 'Mikheev',

class: '4 kurs',

testAggr1: [] },

{ \_id:

{ \_bsontype: 'ObjectID',

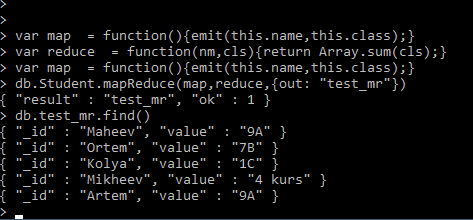
id: <Buffer 5f 74 56 06 50 a3 e5 24 1c 30 43 b6> },

name: 'Maheev',

class: '9A',

testAggr1: [] } ]

агрегированный запрос через mapReduce



> var map = function(){emit(this.name,this.class);}

> var reduce = function(nm,cls){return Array.sum(cls);}

> var map = function(){emit(this.name,this.class);}

> db.Student.mapReduce(map,reduce,{out: "test\_mr"})

{ "result" : "test\_mr", "ok" : 1 }

Результат

> db.test\_mr.find()

{ "\_id" : "Maheev", "value" : "9A" }

{ "\_id" : "Ortem", "value" : "7B" }

{ "\_id" : "Kolya", "value" : "1C" }

{ "\_id" : "Mikheev", "value" : "4 kurs" }

{ "\_id" : "Artem", "value" : "9A" }

>