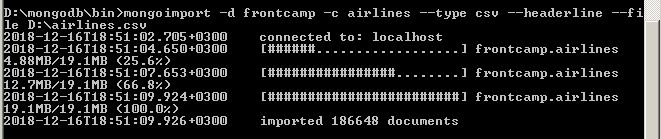
Start:



4.1

db.airlines.aggregate([

{

$group: {

\_id: {

class: "$class"

},

total: {

$sum: 1

}

}

},

{

$project: {

\_id: 0,

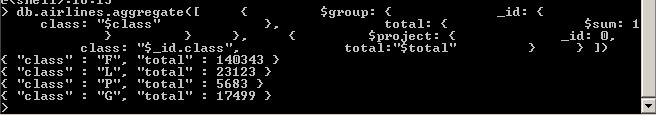
class: "$\_id.class",

total: "$total"

}

}

])



4.2

db.airlines.aggregate([

{ $match: { "destCountry" : {$ne: "United States"}}},

{

$group: {

\_id: {

destCity: "$destCity",

},

avgPass: {

$avg: "$passengers"

}

}

},

{ $sort: { avgPass: -1 }},

{ $limit: 3 },

{

$project: {

\_id: 0,

avgPassengers: "$avgPass",

city: "$\_id.destCity"

}

}

])



4.3

db.airlines.aggregate([

{ $match: { "destCountry" : "Latvia"}},

{

$group: {

\_id: {

destCountry: "$destCountry",

},

carriers: {

$push: "$carrier"

}

}

},

{

$project: {

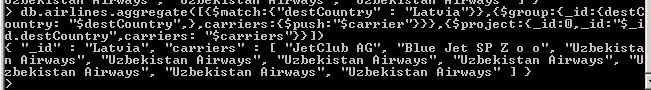
\_id: "$\_id.destCountry",

carriers: "$carriers"

}

}

])



4.4

db.airlines.aggregate([

{ $match: {$and: [

{"originCountry" : "United States"},

{"destCountry" : {$in: ["Spain", "Italia", "Greece"]}}

]}

},

{

$group: {

\_id: {

carrier: "$carrier",

},

overallPassangers: {

$sum: "$passengers"

}

}

},

{ $sort: { overallPassangers: -1 }},

{ $limit: 10 },

{ $skip : 3 },

{

$project: {

\_id: "$\_id.carrier",

total: "$overallPassangers"

}

}

])



4.5

db.airlines.aggregate([

{ $match: {"originCountry" : "United States"}},

{

$group: {

\_id: {

state: "$originState",

city: "$originCity",

},

sumPass: { $sum: "$passengers" }

}

},

{ $sort: { "\_id.state": -1 }},

{

$group: {

\_id: "$\_id.state",

sumPasCit: {

$push: {pas: "$sumPass", cit: "$\_id.city"}

}

}

},

{ $limit: 5 },

{

$project: {

cityM: {

$filter: {

input: "$sumPasCit",

as: "item",

cond: {

$eq: ["$$item.pas", {$max: "$sumPasCit.pas"}]

}

}

}

}

},

{

$project: {

\_id: "$\_id",

cityAndPassangers: { $arrayElemAt: [ "$cityM", 0 ] }

}

},

{

$project: {

\_id: 0,

totalPassangers: "$cityAndPassangers.pas",

location: {

state: "$\_id",

city: "$cityAndPassangers.cit"

}

}

}

])



5. messages

db.enron.aggregate([

{ $group: {

\_id: {

to: "$headers.To",

from: "$headers.From"

},

}},

{ $unwind: "$\_id.to" },

{

$project: {

\_id:0,

itemToFromAlfabeticRanged:{

$cond: {

if: { $gt: ["$\_id.to", "$\_id.from"]},

then: ["$\_id.to","$\_id.from"],

else: ["$\_id.from","$\_id.to"]}

}

}

},

{

$group: {

\_id: "$itemToFromAlfabeticRanged",

count: { "$sum": 1 }

}

},

{ $sort : { count : -1 } },

{ $limit: 1},

{

$project: {

\_id: 0,

writers: "$\_id",

count: "$count"

}

}

])

