Sean Lee

Professor Imielinski

CS 336

5 December 2022

**Mongo HW**

1. Return the bar if either its phone or address is an empty string

db.bars.find({$or: [{phone:"" }, {addr: ""}]})

2. Find the city that has more than 4 bars. Return the city name and the number of bars it has.

db.bars.aggregate(

[{

$group: {

\_id: "$city",

count: {

$sum: 1

},

},

},

{

$match: {

count: {

$gt: 4

},

},

},

])

3. Return how many bars sell more than 5 kinds of beers.

db.bars.aggregate(

[

{

$unwind: {

path: "$beers",

},

},

{

$group: {

\_id: "$name",

count\_of\_beers: { $sum: 1 },

},

},

{

$match: {

count\_of\_beers: { $gt: 5 },

},

},

{

$count: "Number of bars that sell more than 5 beers",

},

])

4. Find the drinkers that have visited any bars either on Saturday or Sunday (or both) [hint: go check out "$elemMatch" function]:

db.bars.aggregate(

[

[

{

$unwind: {

path: "$history",

},

},

{

$match: {

$or: [

{ "history.day": "Saturday" },

{ "history.day": "Sunday" },

],

},

},

{

$group: {

\_id: "$name",

},

},

])

5. Find the drinker who has ordered "Blue Tattoo" beer more than once (in the drinker's history)

db.bars.aggregate(

[

{

$unwind: {

path: "$history",

},

},

{

$match: {

"history.set\_of\_beers": "Blue Tattoo",

},

},

{

$group: {

\_id: "$name",

"number of times ordered": { $sum: 1 },

},

},

{

$match: {

"number of times ordered": { $gt: 1 },

},

},

])

6. Insert Lucy to Drinker collection. Lucy is from Edison, lives at "433 river Road" with phone number 732-571-9871, she is 23 years old and her list of favorite bar foods consists of French fries, Onion rings, Nachos. and Wings

db.Drinkers.insert\_one({

"name":"lucy",

"City":"Edison",

"Addr":"433 river Road",

"Phone":"732-571-9871",

"Age": "23 ",

"Favorite\_foods":[

{"name":"French fries"},

{"name":"Onion Rings"},

{"name":"Nachos"},

{"name":"Wings"}]

})

7.

Original:

R=

[{ "A": 1, "B": 1, "C": 0, "D": 0 },

{ "A": 1, "B": 1, "C": 0, "D": 0 },

{ "A": 1, "B": 1, "C": 0, "D": 0 },

{ "A": 1, "B": 1, "C": 1, "D": 0 },

{ "A": 0, "B": 1, "C": 1, "D": 0 },

{ "A": 0, "B": 1, "C": 0, "D": 0 },

{ "A": 0, "B": 1, "C": 1, "D": 1 }]

Original Value Count = 28

New:

[{ "ABD": 11, "C": [0, 1]},

{ "ABD": 010, "C": [1, 0]}

{ "ABD": 011, "C": [1]}]

New Value Count = 13

8.

from pymongo import MongoClient

client = MongoClient('localhost', 27017)

collection = client['db']['penna']

sum\_at\_end = 0

for doc in collection.aggregate([

{

'$unwind': {

'path': '$vote'

}

}, {

'$match': {

'vote.timestamp': '2020-11-11 21:50:46'

}

}, {

'$group': {

'\_id': 'TotIncrement',

'Sum': {

'$sum': '$vote.totalvotes'

}

}

}

]):

print (doc)

