T4: EJ 1 MongoDB

lunes, 12 de mayo de 2025 13:24

1. En `sample_training.zips` ¿Cuántas colecciones tienen menos de 1000 personas en el campo pop? (sol. 8065)

```
sample_airbnb> show dbs
admin               40.00 KiB
config             108.00 KiB
local               72.00 KiB
sample_airbnb       52.51 MiB
sample_analytics     8.70 MiB
sample_geospatial  788.00 KiB
sample_mflix        27.89 MiB
sample_supplies    960.00 KiB
sample_training     60.85 MiB
sample_weatherdata   2.55 MiB
sample_airbnb> use sample_training
switched to db sample_training
sample_training> show collections
companies
grades
inspections
posts
routes
stories
trips
tweets
zips
sample_training> db.zips.findOne().pretty()
TypeError: db.zips.findOne().pretty is not a function
sample_training> db.zips.findOne()
{
  _id: ObjectId('5c8eccc1caa187d17ca6ed18'),
  city: 'ACMAR',
  zip: '35004',
  loc: { y: 33.584132, x: 86.51557 },
  pop: 6055,
  state: 'AL'
}
sample_training> db.zips.find({"pop": {"$lt":1000}}).count()
8065
sample_training>
```

2. En `sample_training.trips` ¿Cuál es la diferencia entre la gente que nació en 1998 y la que nació después de 1998? (sol. 6)

```
sample_training> show collections
companies
grades
inspections
posts
routes
stories
trips
tweets
zips
sample_training> db.trips.findOne()
{
  _id: ObjectId('5732bb8222b288919b68abf5b'),
  tripduration: 889,
  'start station id': 268,
  'start station name': 'Howard St & Centre St',
  'end station id': 3002,
  'end station name': 'South End Ave & Liberty St',
  bikeid: 22794,
  usertype: 'Subscriber',
  'birth year': 1961,
  gender: 2,
  'start station location': { type: 'Point', coordinates: [ -73.99973337, 40.71910537 ] },
  'end station location': { type: 'Point', coordinates: [ -74.015756, 40.711512 ] },
  'start time': ISODate('2016-01-01T00:01:06.000Z'),
  'stop time': ISODate('2016-01-01T00:15:56.000Z')
}
sample_training> db.trips.find({"birth year": 1998},{"birth year": {"$gt": 1998}}).count()
12
sample_training> db.trips.find({ "birth year": 1998 }).count()
12
sample_training> db.trips.find({ "birth year": { "$gt": 1998 } }).count()
18
sample_training>
```

18-12=6

3. En `sample_training.routes` ¿Cuántas rutas tienen al menos una parada? (sol. 11)

```
sample_training> show collections
companies
grades
inspections
posts
routes
stories
trips
tweets
zips
sample_training> db.routes.findOne()
{
  _id: ObjectId('56e9b39b732b6122f877fa34'),
  airline: { id: 410, name: 'Aerocondor', alias: '2B', iata: 'ARD' },
  src_airport: 'DME',
  dst_airport: 'KZN',
  codeshare: '',
  stops: 0,
  airplane: 'CR2'
}
sample_training> db.routes.find({"stops": {"$gte": 1}}).count()
Uncaught:
SyntaxError: Unexpected token, expected "," (1:36)

> 1 | db.routes.find({"stops": {"$gte":  }).count()
    |                                    ^
  2 |

sample_training> db.routes.find({"stops": {"$gte": 1}}).count()
11
sample_training>
```

2. En `sample_training.companies`, ¿en cuántas empresas coinciden su `permalink` con su `twitter_username`? (sol. 1299)

```
sample_training> db.companies.find({"$expr": {"$eq": ["$twitter_username", "$permalink"]}}).count()
1299
sample_training>
```

3. En `sample_airbnb.listingsAndReviews`, ¿cuál es el nombre del alojamiento en el que pueden estar más de 6 personas alojadas y tiene exactamente 50 reviews? (sol. Sunset Beach Lodge Retreat)

```
sample_airbnb> db.listingsAndReviews.find({ "$expr" : { "$and": [{ "$gt" : [ "$accommodates", 6]}, { "$eq": [{"$size": "$reviews"}, 50]}]}}, { "name" : 1, "_id" : 0})
[ { name: 'Sunset Beach Lodge Retreat' } ]
sample_airbnb>
```

4. En `sample_airbnb.listingsAndReviews`, ¿cuántos documentos tienen el `"property_type" "House"` e incluyen `"Changing table"` como una de las `"amenities"`? (sol. 11)

```
sample_airbnb> db.listingsAndReviews.find({ $and": [{"property_type": "House"}, { "amenities" : {"$all": ["Changing table"]}}]}).count()
11
```

Cosas aprendidas: el $eq y esos, no funcionan sin un $expr... pero entonces ahí se puede evaluar el valor de una clave valor {"clave": "valor"}; y no hace falta hacer "$clave", a menos que los hagamos con las expresiones

4. En `sample_training.inspections` ¿Cuántos negocios tienen un resultado de inspección `"Out of Business"` y pertenecen al sector `"Home Improvement Contractor - 100"`? (sol. 4)

```
sample_training> show collections
companies
grades
inspections
posts
routes
stories
trips
tweets
zips
sample_training> db.inspections.findOne()
{
  _id: ObjectId('56dd1033a378eccde8a83554'),
  id: '10172-2015-CMPL',
  certificate_number: 9304489,
  business_name: 'UNNAMED HOT DOG VENDOR LICENSE NUMBER TA01158',
  date: 'Aug 21 2015',
  result: 'No Violation Issued',
  sector: 'Mobile Food Vendor - 881',
  address: { city: '', zip: '', street: '', number: '' }
}
sample_training> db.inspections.find().count()
80047
sample_training> db.inspections.find({"result": "Out of Business"} , {"sector": "Home Improvement Contractor -"}).count()
7038
sample_training> db.inspections.find({"result": "Out of Business"} , {"sector": "Home Improvement Contractor - 100"}).count()
7038
sample_training> db.inspections.find({"$and" : [{"result": "Out of Business"} , {"sector": "Home Improvement Contractor - 100"}]}).count()
4
sample_training>
```

5. En `sample_training.inspections` ¿Cuántos documentos hay con fecha de inspección `"Feb 20 2015"` o `"Feb 21 2015"` y cuyo sector no sea `"Cigarette Retail Dealer - 127"`? (sol. 204)

```
sample_training> show collections
companies
grades
inspections
posts
routes
stories
trips
tweets
zips
sample_training> db.inspections.findOne()
{
  _id: ObjectId('56dd1033a378eccde8a83554'),
  id: '10172-2015-CMPL',
  certificate_number: 9304489,
  business_name: 'UNNAMED HOT DOG VENDOR LICENSE NUMBER TA01158',
  date: 'Aug 21 2015',
  result: 'No Violation Issued',
  sector: 'Mobile Food Vendor - 881',
  address: { city: '', zip: '', street: '', number: '' }
}
sample_training> db.inspections.find({"$and": [{"$or": [{"date": "Feb 20 2015"}, {"date": "Feb 21 2015"}]}, {"sector" : {"$not": "Cigarette Retail Dealer - 127"}}]}).count()
MongoServerError[BadValue]: $not argument must be a regex or an object
sample_training> db.inspections.find({"$and": [{"$or": [{"date": "Feb 20 2015"}, {"date": "Feb 21 2015"}]}, {"sector" : {"$nor": "Cigarette Retail Dealer - 127"}}]}).count()
MongoServerError[BadValue]: unknown operator: $nor
sample_training> db.inspections.find({"$and": [{"$or": [{"date": "Feb 20 2015"}, {"date": "Feb 21 2015"}]}, {"sector" : {"$ne": "Cigarette Retail Dealer - 127"}}]}).count()
204
sample_training>
```

Ej. 1: Número de documentos de sample_training.trips donde el viaje empieza y termia en la misma estación:

```
grades
inspections
posts
routes
stories
trips
tweets
zips
sample_training>  db.trips.findOne()
{
  _id: ObjectId('5732bb8222b288919b68abf5b'),
  tripduration: 889,
  'start station id': 268,
  'start station name': 'Howard St & Centre St',
  'end station id': 3002,
  'end station name': 'South End Ave & Liberty St',
  bikeid: 22794,
  usertype: 'Subscriber',
  'birth year': 1961,
  gender: 2,
  'start station location': { type: 'Point', coordinates: [ -73.99973337, 40.71910537 ] },
  'end station location': { type: 'Point', coordinates: [ -74.015756, 40.711512 ] },
  'start time': ISODate('2016-01-01T00:01:06.000Z'),
  'stop time': ISODate('2016-01-01T00:15:56.000Z')
}
sample_training>  db.trips.find({"$expr": {"$eq": ["$end station id", "$start station id"]}}).count()
316
```

Ej. 2: Find all documents where the trip lasted longer than 1200 seconds, and started and ended at the same station:

```
sample_training> db.trips.find({"$and": [{"tripduration":{"$gt": 1200}}, {"$expr": {"$eq": ["$start station id", "$end station id"]}}]}).count()
173
sample_training> db.trips.find({ "$expr": { "$and": [ { "$gt": ["$tripduration", 1200] }, { "$eq": ["$end station id", "$start station id"] } ] } }).count()
173
sample_training>
```

5. En `sample_training.companies`, ¿Cuántas empresas tienen oficinas en Seattle? (sol. 117)

```
sample_training> db.companies.find({ "offices.city": "Seattle"}).count()
117
```

6. En `sample_training.companies`, haga una query que devuelva únicamente el nombre de las empresas que tengan exactamente 8 `"funding_rounds"`

```
sample_training> db.companies.find({"$expr" :{"$eq": [{"$size": "$funding_rounds"}, 8]}}, {"name": 1, "_id": 0})
[
  { name: 'Twitter' },
  { name: 'LinkedIn' },
  { name: 'PayScale' },
  { name: 'Xobni' },
  { name: 'Zynga' },
  { name: 'ShareThis' },
  { name: 'TicketLeap' },
  { name: 'Nobliyng' },
  { name: 'PlumChoice' },
  { name: 'SolFocus' },
```

```
le`]]]]]).count()
11
```

Cosas aprendidas: el $eq y esos, no funcionan sin un $expr... pero entonces ahí se puede evaluar el valor de una clave valor {"clave": "valor"}; y no hace falta hacer "$clave", a menos que los hagamos con las expresiones

```
  { name: 'PayScale' },
  { name: 'Xobni' },
  { name: 'Zynga' },
  { name: 'Sharethis' },
  { name: 'TicketLeap' },
  { name: 'Moblyng' },
  { name: 'PlumChoice' },
  { name: 'SolFocus' },
  { name: 'HyperWeek' },
  { name: 'Virident Systems' },
  { name: 'Extreme Enterprises' },
  { name: 'CipherMax' },
  { name: 'Stemgent' },
  { name: 'Sonos' },
  { name: 'BridgeLux' },
  { name: 'Silicor Materials' },
  { name: '1366 Technologies' },
  { name: 'Biolex Therapeutics' }
]
```