

Record Management System

Milestone: Implementation in NoSQL

Group 3

Gudipalli Sri Sai Prabhath Reddy

Yashwant Dontam

857-398-7947 (Prabhath)

857-230-5413 (Yashwant)

gudipalli.s@northeastern.edu

dontam.y@northeastern.edu

Percentage of Effort Contributed by Student1: 50%

Percentage of Effort Contributed by Student2: 50%

Signature of Student 1: Gudipalli Sri Sai Prabhath Reddy

Signature of Student 2: Yashwant Dontam

Submission Date: 12/03/2023

1. Listing all the cases where the case status is "Closed"

MongoDB Compass - localhost:27017/rms_mongodb.cases

Connect Edit View Collection Help

localhost:27017 Documents rms_mongodb.ca...

My Queries Databases Search

RMS admin config local rms_mongodb

- accessrights
- administrative_docume...
- assignedto
- cases**
- evidence
- incidents
- involved
- officers
- schedules
- suspects
- users
- vehicles

rms_mongodb.cases

Documents Aggregations Schema Indexes Validation

Filter {CaseStatus: "Closed"} INSIGHT Generate query Explain Reset Find Options

ADD DATA EXPORT DATA

1 - 20 of 122

1234
CaseDescription: "nec ligula consectetur rhoncus. Nullam velit dui, semper et, lacinia _"
AssignedOfficer: "Nathan Haley"
CaseStatus: "Closed"

1237
CaseDescription: "Suspendisse commodo tincidunt nibh. Phasellus nulla."
AssignedOfficer: "Adele Drake"
CaseStatus: "Closed"

1240
CaseDescription: "velit. Aliquam nisl. Nulla eu neque pellentesque massa lobortis"
AssignedOfficer: "Xenos Pratt"
CaseStatus: "Closed"

1241

> MONGOSH

6°C Cloudy

Search

ENG US

6:49 PM 12/3/2023

2. Listing all incidents and their associated cases where the incident status is a certain value

MongoDB Compass - localhost:27017/rms_mongodb.assignedto

Connect Edit View Collection Help

localhost:27017 Documents rms_mongodb.assignedto

My Queries Databases Search

local

rms_mongodb.assignedto

Documents Aggregations Schema Indexes Validation

500 DOCUMENTS 1 INDEXES

```
>_MONGOSH
> db.incidents.aggregate([
  { $match: { "IncidentStatus": "Reported" } },
  { $lookup: {
    from: "cases",
    localField: "CaseID",
    foreignField: "CaseID",
    as: "case_details"
  }}
])
< {
  _id: ObjectId("656d0ed245dd8b9174ae320c"),
  IncidentID: 2515,
  IncidentLocation: 'Kentucky',
  IncidentStatus: 'Reported',
  IncidentDate: '27-Mar-23',
  IncidentTime: '3:37:00',
  CaseID: 1249,
  case_details: [
    {
      _id: ObjectId("656d0e3c45dd8b9174ae3017"),
      CaseID: 1249,
      CaseDescription: 'Ivenenatis vel...'
    }
  ]
}
```

6°C Cloudy Search 9:36 PM 12/3/2023

3. Listing all cases with their respective number of involved suspects

The screenshot shows the MongoDB Compass interface. The top bar indicates the connection to 'localhost:27017/rms_mongodb.assignedto'. The left sidebar shows the 'Databases' section with 'local' expanded. The main panel displays the 'Documents' tab for the 'rms_mongodb.assignedto' collection, showing 500 documents and 1 index. Below this, a MongoDB aggregation query is entered in the command line:

```
> db.cases.aggregate([
  { $lookup: {
    from: "involved",
    localField: "CaseID",
    foreignField: "CaseID",
    as: "suspects_involved"
  }},
  { $project: {
    "CaseDescription": 1,
    "NumberOfSuspects": { $size: "$suspects_involved" }
  }}
])
```

The query results are displayed as a JSON array:

```
< [
  {
    "_id": ObjectId("656d0e3c45dd8b9174ae3008"),
    "CaseDescription": 'nec ligula consectetur rhoncus. Nullam velit dui, semper et, lacinia vitae, sodales at, velit. Pellentesque ultricies dignissim lacus. Aliquam',
    "NumberOfSuspects": 1
  },
  {
    "_id": ObjectId("656d0e3c45dd8b9174ae3009"),
    "CaseDescription": 'massa. Quisque porttitor eros',
    "NumberOfSuspects": 1
  },
  ...
]
```

The bottom of the screen shows the Windows taskbar with the date and time as 9:40 PM on 12/3/2023.

4. Aggregating the types of evidence and count their occurrences in all cases

The screenshot shows the MongoDB Compass interface. The top bar indicates the connection to 'localhost:27017/rms_mongodb.assignedto'. The left sidebar shows the 'Databases' section with 'local' selected. The main panel displays the 'Documents' tab for the 'rms_mongodb.assignedto' collection, showing 500 documents and 1 index. The 'Aggregations' tab is active, showing a pipeline with one stage: '\$group'. The query is as follows:

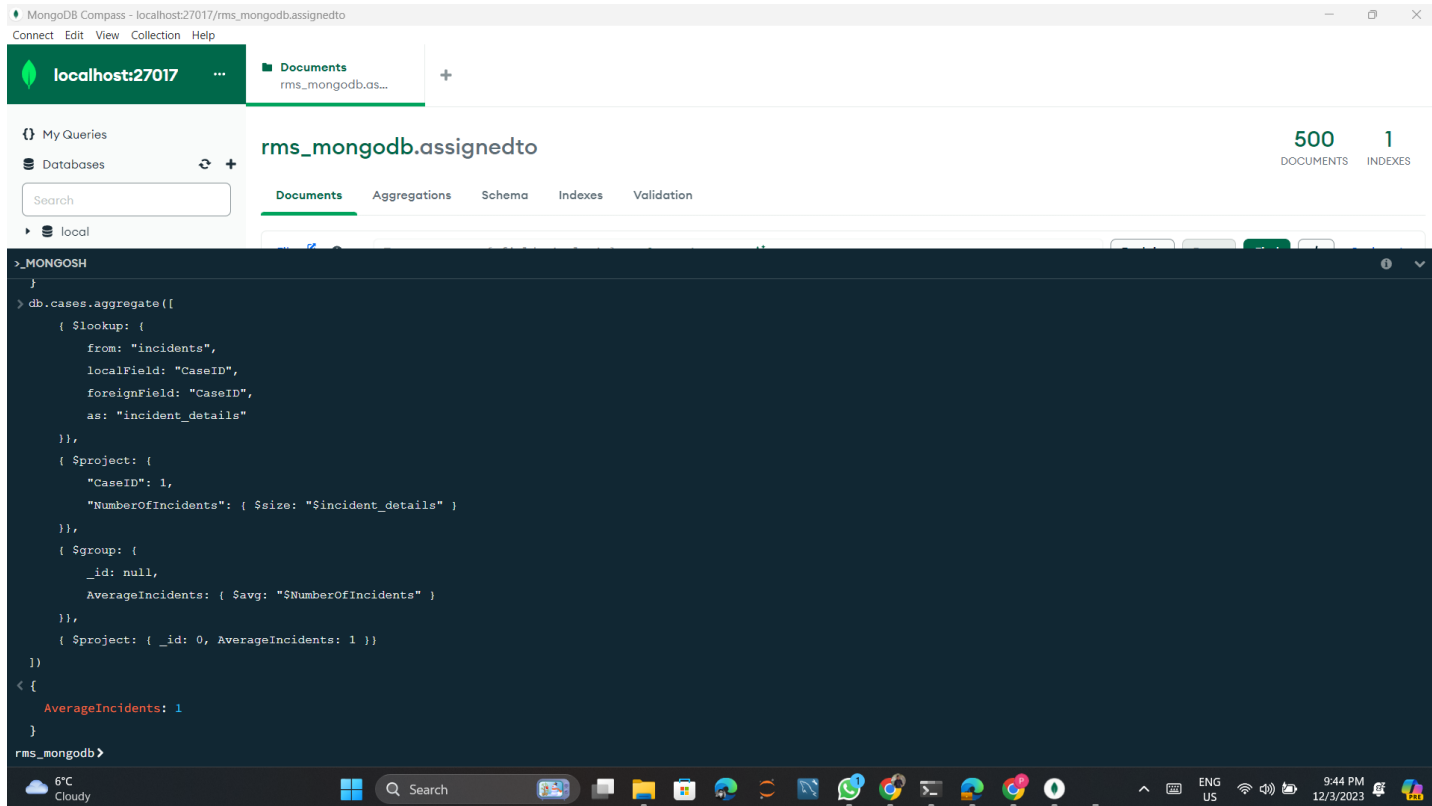
```
> db.evidence.aggregate([
  { $group: {
    _id: "$EvidenceType",
    Count: { $sum: 1 }
  } },
  { $sort: { Count: -1 } }
])
```

The results of the aggregation are displayed below the query:

```
< {
  _id: 'Explosives',
  Count: 52
}
{
  _id: 'Financial',
  Count: 52
}
{
  _id: 'Document',
  Count: 49
}
```

The Windows taskbar at the bottom shows the system clock as 9:42 PM on 12/3/2023, with a temperature of 6°C and a cloudy weather forecast.

5. Finding the average number of incidents per case



The screenshot shows the MongoDB Compass interface. The top bar indicates the connection to 'localhost:27017' and the database 'rms_mongodb.assignedto'. The left sidebar shows the 'Documents' tab selected. The main panel displays the 'Documents' tab for the 'rms_mongodb.assignedto' collection, showing 500 documents and 1 index. Below this, the 'MONGOSH' terminal window is open, displaying the following aggregation query:

```
> db.cases.aggregate([
  { $lookup: {
    from: "incidents",
    localField: "CaseID",
    foreignField: "CaseID",
    as: "incident_details"
  }},
  { $project: {
    "CaseID": 1,
    "NumberOfIncidents": { $size: "$incident_details" }
  }},
  { $group: {
    _id: null,
    AverageIncidents: { $avg: "$NumberOfIncidents" }
  }},
  { $project: { _id: 0, AverageIncidents: 1 }}
])
< {
  AverageIncidents: 1
}
```

The terminal output shows the result of the aggregation query, which is a single document with the field 'AverageIncidents' set to 1. The bottom status bar shows the system temperature as 6°C, the time as 9:44 PM, and the date as 12/3/2023.

The answer is 1 because we gave unique incidentIDs