

Document Based Database

Ice Ybanez

R00176611

Part 1:

1.

```
test> show dbs
ServerSideWebDevProject  24.00 KiB
TrainingSessionDB          252.00 KiB
admin                      40.00 KiB
config                     60.00 KiB
inventorydb                8.00 KiB
local                      80.00 KiB
test> use banksdb
switched to db banksdb
banksdb> db.createCollection("clients")
{ ok: 1 }
banksdb> show dbs
ServerSideWebDevProject  24.00 KiB
TrainingSessionDB          252.00 KiB
admin                      40.00 KiB
banksdb                    8.00 KiB
config                     60.00 KiB
inventorydb                8.00 KiB
local                      80.00 KiB
banksdb> show collections
clients
banksdb>
```

Create new database “banksdb” and a new collection “clients”

```
banksdb> db.clients.insertMany([
...   {
...     "isActive": false,
...     "balance": 3960.64,
...     "age": 30,
...     "eyeColor": "blue",
...     "name": "Dawn Keith",
...     "gender": "female",
...     "company": "COSMOSIS",
...     "email": "dawnkeith@cosmosis.com",
...     "phone": "+1 (839) 437-3421",
...     "address": "392 Clifford Place, Fontanelle, Arizona, 2687"
...   },
...   {
...     "isActive": false,
...     "balance": 1280.14,
...     "age": 31,
...     "eyeColor": "green",
...     "name": "Bettie Eaton",
...     "gender": "female"
...   }
... ])
```

Manually insert all documents in bank.json file into “clients” collection

```
{
  acknowledged: true,
  insertedIds: [
    '0': ObjectId('682dbb4f2707ce81f16c4bd0'),
    '1': ObjectId('682dbb4f2707ce81f16c4bd1'),
    '2': ObjectId('682dbb4f2707ce81f16c4bd2'),
    '3': ObjectId('682dbb4f2707ce81f16c4bd3'),
    '4': ObjectId('682dbb4f2707ce81f16c4bd4'),
    '5': ObjectId('682dbb4f2707ce81f16c4bd5'),
    '6': ObjectId('682dbb4f2707ce81f16c4bd6'),
    '7': ObjectId('682dbb4f2707ce81f16c4bd7'),
    '8': ObjectId('682dbb4f2707ce81f16c4bd8'),
    '9': ObjectId('682dbb4f2707ce81f16c4bd9'),
    '10': ObjectId('682dbb4f2707ce81f16c4bd10'),
    '11': ObjectId('682dbb4f2707ce81f16c4bd11'),
    '12': ObjectId('682dbb4f2707ce81f16c4bd12'),
    '13': ObjectId('682dbb4f2707ce81f16c4bd13'),
    '14': ObjectId('682dbb4f2707ce81f16c4bd14'),
    '15': ObjectId('682dbb4f2707ce81f16c4bd15'),
    '16': ObjectId('682dbb4f2707ce81f16c4bd16'),
    '17': ObjectId('682dbb4f2707ce81f16c4bd17'),
    '18': ObjectId('682dbb4f2707ce81f16c4bd18'),
    '19': ObjectId('682dbb4f2707ce81f16c4bd19'),
    '20': ObjectId('682dbb4f2707ce81f16c4bd1a'),
    '21': ObjectId('682dbb4f2707ce81f16c4bd1b'),
    '22': ObjectId('682dbb4f2707ce81f16c4bd1c'),
    '23': ObjectId('682dbb4f2707ce81f16c4bd1d'),
    '24': ObjectId('682dbb4f2707ce81f16c4bd1e'),
    '25': ObjectId('682dbb4f2707ce81f16c4bd1f'),
    '26': ObjectId('682dbb4f2707ce81f16c4bd1g'),
    '27': ObjectId('682dbb4f2707ce81f16c4bd1h'),
    '28': ObjectId('682dbb4f2707ce81f16c4bd1i'),
    '29': ObjectId('682dbb4f2707ce81f16c4bd1j'),
    '30': ObjectId('682dbb4f2707ce81f16c4bd1k'),
    '31': ObjectId('682dbb4f2707ce81f16c4bd1l'),
    '32': ObjectId('682dbb4f2707ce81f16c4bd1m'),
    '33': ObjectId('682dbb4f2707ce81f16c4bd1n'),
    '34': ObjectId('682dbb4f2707ce81f16c4bd1o')
  ]
}
banksdb> show collections
clients
```

```
banksdb> db.clients.find()
[ {
  _id: ObjectId('682dbb4f2707ce81f16c4bd0'),
  isActive: false,
  balance: 3960.64,
  age: 30,
  eyeColor: 'blue',
  name: 'Dawn Keith',
  gender: 'female',
  company: 'COSMOSIS',
  email: 'dawnkeith@cosmosis.com',
  phone: '+1 (839) 437-3421',
  address: '392 Clifford Place, Fontanelle, Arizona, 2687'
},
{
  _id: ObjectId('682dbb4f2707ce81f16c4bd1'),
  isActive: false,
  balance: 1280.14,
  age: 31,
  eyeColor: 'green',
  name: 'Bettie Eaton',
  gender: 'female',
  company: 'COMTREK',
  email: 'bettieeaton@comtrek.com',
  phone: '+1 (861) 460-2317',
  address: '203 Allen Avenue, Elrama, North Carolina, 4453'
},
{
  _id: ObjectId('682dbb4f2707ce81f16c4bd2'),
  isActive: true,
  balance: 2042.37,
  age: 38,
  eyeColor: 'green',
  name: 'Margie Ayala',
  gender: 'female',
  company: 'VOIPA',
  email: 'margieayala@voipa.com',
  phone: '+1 (941) 569-2231',
  address: '111 Schroeders Avenue, Suitland, Louisiana, 7042'
},
{
  _id: ObjectId('682dbb4f2707ce81f16c4bd3'),
  isActive: false,
```

Sample of listing documents in collection using db.clients.find()

```

banksdb> db.clients.insertOne({
...   "isActive": true,
...   "balance": 2042.37,
...   "age": 38,
...   "eyeColor": "green",
...   "name": "Margie Ayala",
...   "gender": "female",
...   "company": "VOIPA",
...   "email": "margieayala@voipa.com",
...   "phone": "+1 (941) 569-2231",
...   "address": "111 Schroeders Avenue, Suitland, Louisiana, 7042"
... })
{
  acknowledged: true,
  insertedId: ObjectId('682dbf742707ce81f16c4c17')
}

```

Insert client

```

banksdb> db.clients.findOne({_id:ObjectId('682dbf742707ce81f16c4c17')})
{
  _id: ObjectId('682dbf742707ce81f16c4c17'),
  isActive: true,
  balance: 2042.37,
  age: 38,
  eyeColor: 'green',
  name: 'Margie Ayala',
  gender: 'female',
  company: 'VOIPA',
  email: 'margieayala@voipa.com',
  phone: '+1 (941) 569-2231',
  address: '111 Schroeders Avenue, suitland, Louisiana, 7042'
}

```

To check if inserted in the correct collection

2.

Margie has been inserted twice because the server automatically created different primary keys (_id) for each document. This means two documents with the same contents can exist at any one time as long as their _id's are different, and this case, it is.

To fix this problem we insert a unique _id for each document by stating it when inserting a new document.

The server will check if there is a _id key:value on the re-inserted document. If it finds one, it will stop insertion.

3.

```

banksdb> db.clients.updateMany({$isActive:true,gender:"female",age:{$lt:30}},[{$set:{balance:{$multiply:[ "$balance", 1.01 ]}}])
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 2,
  modifiedCount: 2,
  upsertedCount: 0
}
banksdb> db.clients.find({$isActive:true,gender:"female",age:{$lt:30}},{_id:0,isActive:1,gender:1,age:1,balance:1})
[
  { isActive: true, balance: 1595.5374, age: 24, gender: 'female' },
  { isActive: true, balance: 2061.5817, age: 28, gender: 'female' }
]

```

Applying one year's 1% interest rate to the account of all ACTIVE FEMALE clients whose age BELOW 30 (with projection)

4.

```
banksdb> db.clients.updateMany({isActive:false,gender:"male",age:{$gte:30,$lte:45}},{$set:{hobbies:["reading", "video games"],favFoods:["Chinese", "Korean", "French"]}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 7,
  modifiedCount: 7,
  upsertedCount: 0
}
```

Using \$set to create hobbies and favFoods information in clients' profiles.

```
banksdb> db.clients.find({isActive:false,gender:"male",age:{$gte:30,$lte:45}},{_id:0,isActive:1,gender:1,age:1,hobbies:1,favFoods:1})
[ {
  isActive: false,
  age: 37,
  gender: 'male',
  favFoods: [ 'Chinese', 'Korean', 'French' ],
  hobbies: [ 'reading', 'video games' ]
},
{
  isActive: false,
  age: 36,
  gender: 'male',
  favFoods: [ 'Chinese', 'Korean', 'French' ],
  hobbies: [ 'reading', 'video games' ]
},
{
  isActive: false,
  age: 37,
  gender: 'male',
  favFoods: [ 'Chinese', 'Korean', 'French' ],
  hobbies: [ 'reading', 'video games' ]
},
{
  isActive: false,
  age: 32,
  gender: 'male',
  favFoods: [ 'Chinese', 'Korean', 'French' ],
  hobbies: [ 'reading', 'video games' ]
},
{
  isActive: false,
  age: 30,
  gender: 'male',
  favFoods: [ 'Chinese', 'Korean', 'French' ],
  hobbies: [ 'reading', 'video games' ]
},
{
  isActive: false,
  age: 34,
  gender: 'male',
  favFoods: [ 'Chinese', 'Korean', 'French' ],
  hobbies: [ 'reading', 'video games' ]
},
{
  isActive: false,
  age: 36,
  gender: 'male',
  favFoods: [ 'Chinese', 'Korean', 'French' ],
  hobbies: [ 'reading', 'video games' ]
}]
```

```
{
  isActive: false,
  age: 36,
  gender: 'male',
  favFoods: [ 'Chinese', 'Korean', 'French' ],
  hobbies: [ 'reading', 'video games' ]
}
]
```

To check if inserted

5.

```
banksdb> db.clients.updateMany({isActive:false,gender:"male",age:{$gte:30,$lte:45},eyeColor:"green"},{$pull:{favFoods:"Korean"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 5,
  modifiedCount: 5,
  upsertedCount: 0
}
banksdb> db.clients.find({isActive:false,gender:"male",age:{$gte:30,$lte:45},eyeColor:"green"},{_id:0,isActive:1,gender:1,age:1,eyeColor:1,favFoods:1})
[
  {
    isActive: false,
    age: 36,
    eyeColor: 'green',
    gender: 'male',
    favFoods: [ 'Chinese', 'French' ]
  },
  {
    isActive: false,
    age: 37,
    eyeColor: 'green',
    gender: 'male',
    favFoods: [ 'Chinese', 'French' ]
  },
  {
    isActive: false,
    age: 32,
    eyeColor: 'green',
    gender: 'male',
    favFoods: [ 'Chinese', 'French' ]
  },
  {
    isActive: false,
    age: 30,
    eyeColor: 'green',
    gender: 'male',
    favFoods: [ 'Chinese', 'French' ]
  },
  {
    isActive: false,
    age: 36,
    eyeColor: 'green',
    gender: 'male',
    favFoods: [ 'Chinese', 'French' ]
  }
]
```

Updating profiles of inactive male clients with green eyes

Part 2:

1.

```
studentsdb> db.students_usa.find({region:"Southwest"},{_id:0,city:1})
[
  { city: 'Austin' }, { city: 'Austin' },
  { city: 'San Diego' }, { city: 'Austin' },
  { city: 'Austin' }, { city: 'Los Angeles' },
  { city: 'Palm Springs' }, { city: 'Austin' },
  { city: 'Honolulu' }, { city: 'Palm Springs' },
  { city: 'Austin' }, { city: 'Honolulu' },
  { city: 'Palm Springs' }, { city: 'Palm Springs' },
  { city: 'Lahaina' }, { city: 'Austin' },
  { city: 'Palm Springs' }, { city: 'Los Angeles' },
  { city: 'Lahaina' }, { city: 'Palm Springs' }
]
Type "it" for more
studentsdb> it
[
  { city: 'Lahaina' }, { city: 'Palm Springs' },
  { city: 'Honolulu' }, { city: 'Los Angeles' },
  { city: 'San Diego' }, { city: 'Honolulu' },
  { city: 'San Diego' }, { city: 'Austin' },
  { city: 'San Diego' }, { city: 'Honolulu' },
  { city: 'Honolulu' }, { city: 'Los Angeles' },
  { city: 'Lahaina' }, { city: 'Lahaina' },
  { city: 'Los Angeles' }, { city: 'Los Angeles' },
  { city: 'Los Angeles' }, { city: 'Honolulu' },
  { city: 'San Diego' }, { city: 'Los Angeles' }
]
Type "it" for more
studentsdb> it
[
  { city: 'Palm Springs' },
  { city: 'Palm Springs' },
  { city: 'Lahaina' },
  { city: 'Austin' },
  { city: 'Palm Springs' }
]
```

Listing all cities in the Southwestern region using projection.

From dataset provided in the question, Austin, San Diego, Los Angeles, Palm Springs, Lahaina, Sedona, San Francisco, Nashville, Las Vegas, Honolulu are in the Southwestern region of the United States.

2.

```
studentsdb> db.students_usa.find({age:{$lt:20},hobbies:{$size:1},region:{$regex:/south/i}})
[ {
  _id: 90,
  name: 'Osvaldo Hirt',
  gender: 'male',
  class_id: 29,
  hobbies: [ 'Yoga' ],
  city: 'Los Angeles',
  region: 'Southwest',
  age: 18,
  scores: [
    { score: 67.44, type: 'exam' },
    { score: 41.77, type: 'quiz' },
    { score: 76.3, type: 'project' }
  ]
}
]
studentsdb> db.students_usa.find({age:{$lt:20},hobbies:{$size:1},region:{$regex:/south/i}}, {_id:1,age:1,hobbies:1,region:1})
[ { _id: 90, hobbies: [ 'Yoga' ], region: 'Southwest', age: 18 } ]
```

Finding adolescents aged under 20, lived in Southern regions of US(using case-insensitivity) and have only 1 hobby.

```
studentsdb> db.students_usa.find({age:{$lt:20},hobbies:{$size:1},region:{$regex:/south/i}}, {_id:1,age:1,hobbies:1,region:1}).explain('executionStats')
```

Used .explain('executionStats') to evaluate query performance

```
executionStats: {
  executionSuccess: true,
  nReturned: 1,
  executionTimeMillis: 0,
  totalKeysExamined: 0,
  totalDocsExamined: 200,
```

nReturned: 1 shows only 1 document was found to answer the query

totalDocsExamined: 200 - shows 200 documents were examined

```
studentsdb> db.students_usa.createIndex({age:1,region:1})
age_1_region_1
```

Indexing for age and region keys in ascending order

```
executionStats: {
  executionSuccess: true,
  nReturned: 1,
  executionTimeMillis: 23,
  totalKeysExamined: 31,
  totalDocsExamined: 15,
```

nReturned: 1 shows only 1 document was found to answer the query

totalDocsExamined: 15 - shows 15 documents were examined, significantly improving query performance with the significant drop in documents examined

3.

```
studentsdb> db.students_usa.aggregate([{$match:{gender: "Female"}},{$unwind: "scores"},{$group:{_id:{id: "$_id",region: "$_id.region"},scores:{$push:{k: "$_id.type",v: "score"}},$project:{region: "$_id.region",scoresObj:{$arrayToObject: "scores"},$match:{scoresObj.exam:{$gte:40}},scoresObj.quiz:{$gte:40},scoresObj.project:{$gte:40}}},{$group:{_id: "$region",femalesPassedAll:{$sum:1}}},{$sort:{femalesPassedAll:-1}}])
```

[{"_id": "Southwest", femalesPassedAll: 5}, {"_id": "Westsouth", femalesPassedAll: 5}, {"_id": "Southwestern", femalesPassedAll: 4}, {"_id": "Midwestern", femalesPassedAll: 4}, {"_id": "Southermost", femalesPassedAll: 4}, {"_id": "Northeastern", femalesPassedAll: 2}, {"_id": "Northernmost", femalesPassedAll: 1}, {"_id": "Southeastern", femalesPassedAll: 1}]

33 female students passed all 3 assessments in each region

4.

```
studentsdb> db.students_usa.aggregate([{$unwind: "scores"},{$group:{_id:{id: "$_id",region: "$_id.region",type: "scores.type"},score:{$first: "scores.score"},$sum:{$ifNull: ["$scores.passedAll", 0]},$min:{$min: "scores.score"},$max:{$max: "scores.score"}},$project:{class_id: "$_id.class_id",region: "$_id.region",score:{$arrayToObject: "scores"}},{$match:{scoresObj:exam:{$gte:40}},scoresObj.quiz:{$gte:40},scoresObj.project:{$gte:40}},{$group:{_id: "$region",class_id: "$_id.class_id",passedAll:{$sum:1}}},{$sort:{_id: 1,$_id.class_id: 1}}])
```

[{"_id": {"region": "Midwestern", class_id: 1}, passedAll: 5}, {"_id": {"region": "Midwestern", class_id: 20}, passedAll: 2}, {"_id": {"region": "Northeastern", class_id: 0}, passedAll: 2}, {"_id": {"region": "Northeastern", class_id: 25}, passedAll: 1}, {"_id": {"region": "Northeastern", class_id: 30}, passedAll: 1}, {"_id": {"region": "Northernmost", class_id: 17}, passedAll: 1}, {"_id": {"region": "Northernmost", class_id: 18}, passedAll: 1}, {"_id": {"region": "Pacific Northwest", class_id: 18}, passedAll: 1}, {"_id": {"region": "Pacific Northwest", class_id: 22}, passedAll: 2}, {"_id": {"region": "Pacific Northwest", class_id: 24}, passedAll: 2}, {"_id": {"region": "Southeastern", class_id: 11}, passedAll: 1}, {"_id": {"region": "Southeastern", class_id: 22}, passedAll: 1}, {"_id": {"region": "Southeastern", class_id: 25}, passedAll: 1}, {"_id": {"region": "Southern", class_id: 7}, passedAll: 2}, {"_id": {"region": "Southern", class_id: 11}, passedAll: 2}, {"_id": {"region": "Southernmost", class_id: 4}, passedAll: 1}, {"_id": {"region": "Southernmost", class_id: 12}, passedAll: 2}]

Type "it" for more
studentsdb> it

[{"_id": {"region": "Southernmost", class_id: 25}, passedAll: 1}, {"_id": {"region": "Southwest", class_id: 2}, passedAll: 1}, {"_id": {"region": "Southwest", class_id: 7}, passedAll: 1}, {"_id": {"region": "Southwest", class_id: 12}, passedAll: 1}, {"_id": {"region": "Southwest", class_id: 16}, passedAll: 1}, {"_id": {"region": "Southwest", class_id: 18}, passedAll: 1}, {"_id": {"region": "Southwest", class_id: 23}, passedAll: 1}, {"_id": {"region": "Southwestern", class_id: 16}, passedAll: 3}, {"_id": {"region": "Southwestern", class_id: 22}, passedAll: 2}, {"_id": {"region": "Southwestern", class_id: 30}, passedAll: 1}, {"_id": {"region": "Westsouth", class_id: 3}, passedAll: 2}, {"_id": {"region": "Westsouth", class_id: 6}, passedAll: 1}, {"_id": {"region": "Westsouth", class_id: 11}, passedAll: 1}, {"_id": {"region": "Westsouth", class_id: 28}, passedAll: 2}]

How many students passed all 3 assessments in each class of each region

5.

```
studentsdb> db.students_usa.aggregate([{$unwind: "hobbies"},{$group:{_id: "hobbies",count:{$sum:1},classes:{$addToSet: "$class_id"}},{$sort:{count:-1}},{$limit:1}])
```

[{"_id": "Martial Arts", count: 21, classes: [13, 29, 25, 11, 7, 6, 10, 27, 20, 16, 12, 3]}]

Most popular hobby is Martial Arts and classes that have it are displayed

Part 3:

1.

```
strokedb> db.createView("cardiovascularPatients", "stroke_risk", [{"$match": {"$or": [{"Chest_Pain": 1}, {"Irregular_Heartbeat": 1}, {"Swelling": 1}, {"Excessive_Sweating": 1}, {"Nausea/Vomiting": 1}, {"Fatigue_and_Weakness": 1}, {"Dizziness": 1}, {"Shortness_of_Breath": 1}, {"High_Blood_Pressure": 1}, {"Chest_Discomfort_(Activity)": 1}, {"Cold_Hands_Feet": 1}, {"Snoring/Sleep_Apnea": 1}, {"Anxiety/Feeling_of_Doom": 1}, {"Age": 74}, {"Stroke_Risk(%)": 68}, {"At_Risk(Binary)": 1}]}])
{ ok: 1 }
strokedb> show collections
cardiovascularPatients [view]
stroke_risk
system.views
strokedb> db.cardiovascularPatients.find()
[ {
    "_id": ObjectId('6833824180d3327f4d58a3d7'),
    "Chest_Pain": 0,
    "Shortness_of_Breath": 1,
    "Irregular_Heartbeat": 0,
    "Fatigue_and_Weakness": 1,
    "Dizziness": 1,
    "Swelling(Edema)": 0,
    "High_Blood_Pressure": 0,
    "Chest_Discomfort_(Activity)": 0,
    "Cold_Hands/Feet": 0,
    "Snoring/Sleep_Apnea": 0,
    "Anxiety/Feeling_of_Doom": 0,
    "Age": 74,
    "Stroke_Risk(%)": 68,
    "At_Risk(Binary)": 1
},
{
    "_id": ObjectId('6833824180d3327f4d58a3d8'),
    "Chest_Pain": 0,
    "Shortness_of_Breath": 1,
    "Irregular_Heartbeat": 0,
    "Fatigue_and_Weakness": 1,
    "Dizziness": 1,
```

View for cardiovascular patients

```
strokedb> db.createView("neurologicalPatients", "stroke_risk", [{"$match": {"$or": [{"Fatigue_and_Weakness": 1}, {"Dizziness": 1}, {"Persistent_Cough": 1}, {"Anxiety/Feeling_of_Doom": 1}, {"Chest_Pain": 0}, {"Shortness_of_Breath": 1}, {"Irregular_Heartbeat": 1}, {"Fatigue_and_Weakness": 1}, {"Dizziness": 1}, {"Swelling(Edema)": 0}, {"Pain_in_Neck/Jaw/Shoulder/Back": 1}, {"Excessive_Sweating": 0}, {"Persistent_Cough": 0}, {"Nausea/Vomiting": 0}, {"High_Blood_Pressure": 0}, {"Chest_Discomfort_(Activity)": 1}, {"Cold_Hands/Feet": 0}, {"Snoring/Sleep_Apnea": 1}, {"Anxiety/Feeling_of_Doom": 1}, {"Age": 74}, {"Stroke_Risk(%)": 68}, {"At_Risk(Binary)": 1}]}])
{ ok: 1 }
strokedb> show collections
cardiovascularPatients [view]
neurologicalPatients [view]
stroke_risk
system.views
strokedb> db.neurologicalPatients.find()
[ {
    "_id": ObjectId('6833824180d3327f4d58a3d7'),
    "Chest_Pain": 0,
    "Shortness_of_Breath": 1,
    "Irregular_Heartbeat": 1,
    "Fatigue_and_Weakness": 1,
    "Dizziness": 1,
    "Swelling(Edema)": 0,
    "Pain_in_Neck/Jaw/Shoulder/Back": 1,
    "Excessive_Sweating": 0,
    "Persistent_Cough": 0,
    "Nausea/Vomiting": 0,
    "High_Blood_Pressure": 0,
    "Chest_Discomfort_(Activity)": 1,
    "Cold_Hands/Feet": 0,
    "Snoring/Sleep_Apnea": 1,
    "Anxiety/Feeling_of_Doom": 1,
    "Age": 74,
    "Stroke_Risk(%)": 68,
    "At_Risk(Binary)": 1
},
{
    "_id": ObjectId('6833824180d3327f4d58a3d8'),
    "Chest_Pain": 0,
    "Shortness_of_Breath": 1,
    "Irregular_Heartbeat": 0,
    "Fatigue_and_Weakness": 1,
    "Dizziness": 1,
```

View for neurological patients

```
strokedb> db.createView("sleepPatients", "stroke_risk", [{"$match": {"$or": [{"Snoring/Sleep_Apnea": 1}]}]})
{ ok: 1 }
strokedb> show collections
cardiovascularPatients [view]
neurologicalPatients [view]
sleepPatients [view]
stroke_risk
system.views
strokedb> db.sleepPatients.find()
[ {
    "_id": ObjectId('6833824180d3327f4d58a3d7'),
    "Chest_Pain": 0,
    "Shortness_of_Breath": 1,
    "Irregular_Heartbeat": 1,
    "Fatigue_and_Weakness": 1,
    "Dizziness": 1,
    "Swelling(Edema)": 0,
    "Pain_in_Neck/Jaw/Shoulder/Back": 1,
    "Excessive_Sweating": 0,
    "Persistent_Cough": 0,
    "Nausea/Vomiting": 0,
    "High_Blood_Pressure": 0,
    "Chest_Discomfort_(Activity)": 1,
    "Cold_Hands/Feet": 0,
    "Snoring/Sleep_Apnea": 1,
    "Anxiety/Feeling_of_Doom": 1,
    "Age": 74,
    "Stroke_Risk(%)": 68,
    "At_Risk(Binary)": 1
},
{
    "_id": ObjectId('6833824180d3327f4d58a3d8'),
    "Chest_Pain": 0,
    "Shortness_of_Breath": 1,
    "Irregular_Heartbeat": 0,
```

View for patients with sleep-related issues

2.

```
strokedb> db.createRole({role:"Cardiologist",privileges:[{resource:{db:"strokedb",collection:"cardiovascularPatients"},actions:["find"]}],roles:[]})
{ ok: 1 }
strokedb> db.getRoles()
{
  roles: [
    {
      _id: 'strokedb.cardiologist',
      role: 'Cardiologist',
      db: 'strokedb',
      roles: [],
      isbuiltin: false,
      inheritedRoles: []
    }
  ]
}
ok: 1
```

Creating role: “Cardiologist”

```
strokedb> db.createUser({user:"DrJamesSmith",pwd:"pwd",roles:[{role:"Cardiologist",db:"strokedb",collection:"cardiovascularPatients"}]})
{ ok: 1 }
strokedb> db.getUsers()
{
  users: [
    {
      _id: 'strokedb.DrJamesSmith',
      userId: UUID('db882551-1005-4a9d-b254-4db124b22599'),
      user: 'DrJamesSmith',
      db: 'strokedb',
      roles: [ { role: 'Cardiologist', db: 'strokedb' } ],
      mechanisms: [ 'SCRAM-SHA-1', 'SCRAM-SHA-256' ]
    }
  ]
}
ok: 1
```

Creating user for Dr James Smith

```
strokedb> db.auth("DrJamesSmith","pwd")
{ ok: 1 }
strokedb> db.runCommand({connectionStatus:1})
{
  authInfo: {
    authenticatedUsers: [ { user: 'DrJamesSmith', db: 'strokedb' } ],
    authenticatedUserRoles: [ { role: 'Cardiologist', db: 'strokedb' } ]
  },
  ok: 1
}
```

Authenticating

```
strokedb> db.cardiovascularPatients.find({Age:{$lt:30},Chest_Pain:1,Irregular_Heartbeat:1,High_Blood_Pressure:1,"Stroke_Risk(%)":{$gt:40}}, {_id:0,Age:1,Chest_Pain:1,Irregular_Heartbeat:1,High_Blood_Pressure:1,"Stroke_Risk(%)":1})
{
  {
    Chest_Pain: 1,
    Irregular_Heartbeat: 1,
    High_Blood_Pressure: 1,
    Age: 27,
    'Stroke_Risk(%)': 49.5
  },
  {
    Chest_Pain: 1,
    Irregular_Heartbeat: 1,
    High_Blood_Pressure: 1,
    Age: 24,
    'Stroke_Risk(%)': 43
  },
  {
    Chest_Pain: 1,
    Irregular_Heartbeat: 1,
    High_Blood_Pressure: 1,
    Age: 19,
    'Stroke_Risk(%)': 65.5
  },
  {
    Chest_Pain: 1,
    Irregular_Heartbeat: 1,
    High_Blood_Pressure: 1,
    Age: 20,
    'Stroke_Risk(%)': 56
  },
  {
    Chest_Pain: 1,
    Irregular_Heartbeat: 1,
    High_Blood_Pressure: 1,
    Age: 20,
    'Stroke_Risk(%)': 51
  },
  {
    Chest_Pain: 1,
    Irregular_Heartbeat: 1,
    High_Blood_Pressure: 1,
    Age: 27,
    'Stroke_Risk(%)': 54.5
  }
}
```

Dr James Smith wants to know how many young patients have the following symptoms

3.

```
strokedb> db.runCommand({connectionStatus:1})
{
  authInfo: { authenticatedUsers: [], authenticatedUserRoles: [] },
  ok: 1
}
strokedb> show collections
cardiovascularPatients [view]
neurologicalPatients [view]
sleepPatients [view]
stroke_risk
system.views
```

Switched back to normal user

```
strokedb> db.stroke_risk.find({Age:{$gte:60,$lte:70}, "At_Risk(Binary)":0,Dizziness:1,"Fatigue_and_ Weakness":1,"Pain_in_Neck/Jaw/Shoulder/Back":1}).count()
23
```

23 pensioners between 60 and 70 not at risk of stroke even though they have symptoms

4.

```
strokedb> db.stroke_risk.find({Age:{$gte:40,$lte:50}, "Stroke_Risk(%)":{$lte:55}, "Snoring/Sleep_Apnea":1}).count()
3007
```

3007 patients aged between 40 and 50 have over 45% chance of NOT getting a stroke (55% chance of getting) when snoring heavily

5.

```
strokedb> db.stroke_risk.find({ "Anxiety/Feeling_of_Doom":0, "Chest_Discomfort_(Activity)":1,"Cold_Hands/Feet":1,Shortness_of_Breath:1,{_id:0,Age:1,"Stroke_Risk(%)":1})
[
  { Age: 40, "Stroke_Risk(%)": 66 },
  { Age: 47, "Stroke_Risk(%)": 44.5 },
  { Age: 79, "Stroke_Risk(%)": 60.5 },
  { Age: 54, "Stroke_Risk(%)": 58.7 },
  { Age: 41, "Stroke_Risk(%)": 53.7 },
  { Age: 43, "Stroke_Risk(%)": 52.5 },
  { Age: 35, "Stroke_Risk(%)": 63.5 },
  { Age: 82, "Stroke_Risk(%)": 82 },
  { Age: 23, "Stroke_Risk(%)": 32.5 },
  { Age: 81, "Stroke_Risk(%)": 71.5 },
  { Age: 49, "Stroke_Risk(%)": 60.5 },
  { Age: 38, "Stroke_Risk(%)": 60 },
  { Age: 74, "Stroke_Risk(%)": 58 },
  { Age: 20, "Stroke_Risk(%)": 51.5 },
  { Age: 89, "Stroke_Risk(%)": 80.5 },
  { Age: 77, "Stroke_Risk(%)": 79.5 },
  { Age: 61, "Stroke_Risk(%)": 89.5 },
  { Age: 21, "Stroke_Risk(%)": 65.5 },
  { Age: 84, "Stroke_Risk(%)": 73 }
]
Type "it" for more
```

The chances of getting a stroke and age of patient when no anxiety during activities, feeling discomfort in chest, cold hands/feet and shortness of breath