Universitatea Politehnica Timișoara
Facultatea de Automatică și Calculatoare
Secția Tehnologii Informatice, anul 1
Mădălina Andreea Turlea

PROIECT ABD

Proiect "Administrarea bazelor de date"

1. Overview

The goal of the project is to assess the MongoDB database skills of the student. The project has a medium difficulty level and is relevant to industry employers of today.

2. Project description

You are required to make several statistical computations on some US Zips dataset using MongoDB as the database platform.

Prerequisites:

- Download the latest US Zips dataset from https://simplemaps.com/data/us-zips (choose the free tier). The dataset has approximately 33k entries.
- Create a MongoDB instance. You may use your own MongoDB Atlas instance in cloud or use a local instance. For local instances Docker is preferred, but you may also choose to install MongoDB as a standalone server on your OS.
- Import the dataset into the MongoDB instance.

Requirements:

a) Get the states with a total population of over 10 million.

b) Get the average city population by state.

c) Get the largest and the smallest city in each state.

d) Get the largest and the smallest counties in each state.

e) Get the nearest 10 zips from one of Chicago's landmarks, the Willis Tower situated at coordinates 41.878876, -87.635918.

f) Get the total population situated between 50 and 200 kms around New York's landmark, the Statue of Liberty at coordinates 40.689247, -74.044502.

Notes:

- Create the indexes you deem relevant for your collection. You will be asked on the performance of your indexes so be prepared to defend your choice, preferably by analyzing the execution statistics.
- For requirements e) and f), you may add a geo field to your collection in order to leverage geospatial query operators.
- Your solution must be original so please don't rely on cheating.

3. Scoring:

- Requirement a) - 1 point

- Requirement b) - 1.5 points

- Requirement c) - 1.5 points

- Requirement d) - 1.5 points

- Requirement e) - 2 points

- Requirement f) - 2.5 points

In order to pass, you must earn 5 points or more.

4. Solution Delivery

Upload your final project solution to the designated area in the virtual campus.

PROIECT ABD

1. Open PowerShell
   Change the directory to where the project is.
   Ex: cd E:\Master\ABD

2. Start a local MongoDB instance running inside a Docker container
   docker run -d --name mongo-project-abd -p 27017:27017 -e MONGO_INITDB_ROOT_USERNAME=madaUser -e MONGO_INITDB_ROOT_PASSWORD=madaUser mongo

3. Use mongoimport tool to import the sample dataset
   mongoimport --db=abd_project --collection=states --file=simplemaps/uszips.csv --type=csv --headerline mongodb://madaUser:madaUser@localhost:27017/?authSource=admin

4. Connect to the local MongoDB instance using Mongo Shell
   mongosh mongodb://madaUser:madaUser@localhost:27017/?authSource=admin

5. Change the database to the one from your project
   Ex: use abd_project

6. Create index for state_name and population:
   db.states.createIndex({state_name : 1})
   db.states.createIndex({population : 1})
7. Get the states with a total population of over 10 million.
   db.states.aggregate([
   {

```
    $group: {
      _id: "$state_name",
      total_population_over_10million: { $sum: "$population" }
    }
  },
  {
    $match: {
      total_population_over_10million: { $gt: 10000000 }
    }
  }
])
```

8. Get the average city population by state.

```
db.states.aggregate([
 {
  $group: {
    _id: "$state_name",
    average_city_population: { $avg: "$population" }
  }
 }
])
```

9. Get the largest and the smallest city in each state.

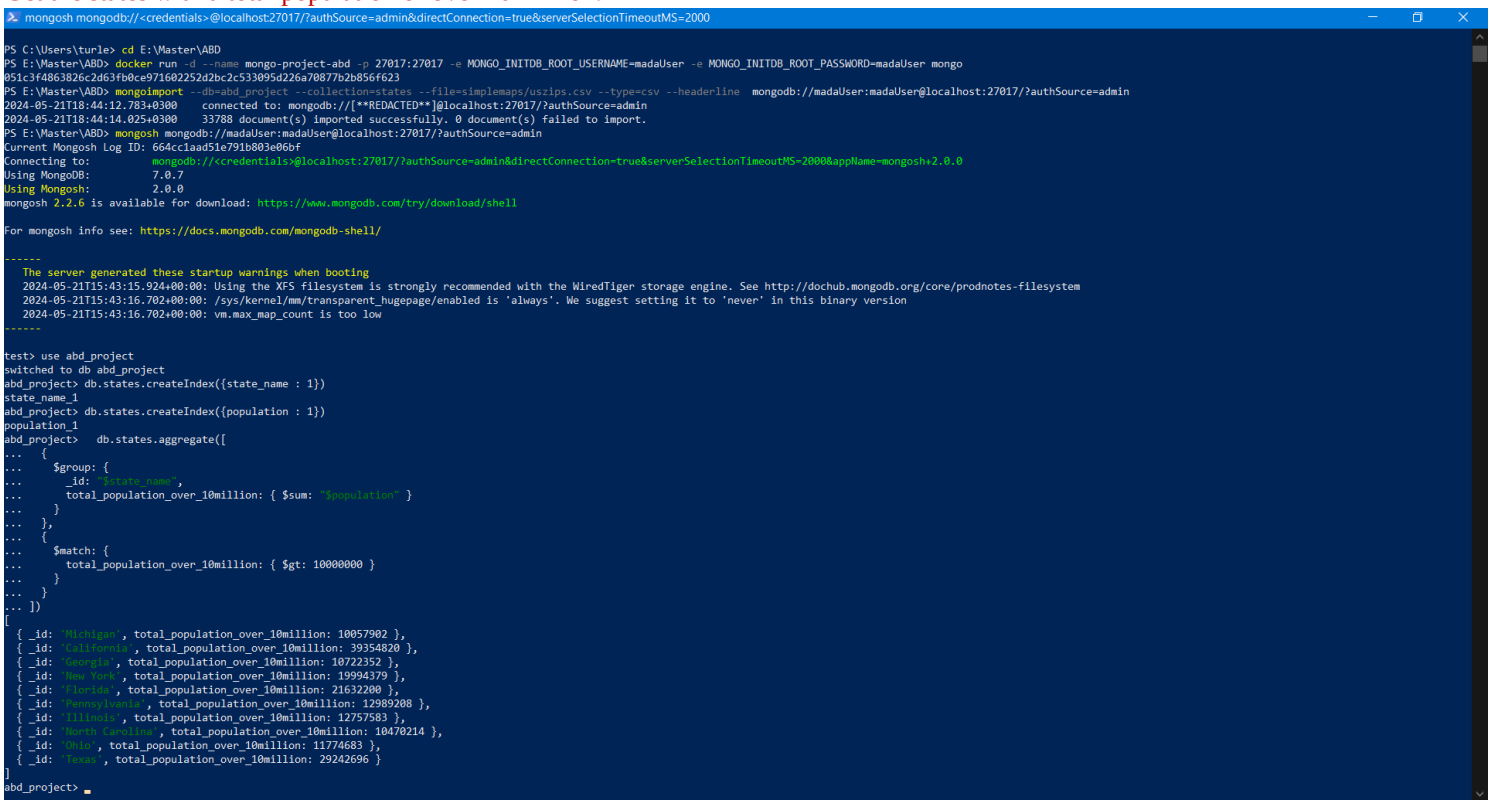
```
db.states.aggregate([
 {
  $sort: {
    state_name: 1,
    population: -1
  }
 },
 {
  $group: {
    _id: "$state_name",
    LARGEST_CITY: { $first: { city: "$city", population: "$population" } },
    SMALLEST_CITY: { $last: { city: "$city", population: "$population" } }
  }
 },
 {
  $project: {
    state_name: "$_id",
    LARGEST_CITY: "$LARGEST_CITY.city",
    LARGEST_CITY_POPULATION: "$LARGEST_CITY.population",
    SMALLEST_CITY: "$SMALLEST_CITY.city",
    SMALLEST_CITY_POPULATION: "$SMALLEST_CITY.population"
  }
 }
])
```

10. Get the largest and the smallest counties in each state.

```
db.states.aggregate([
 {
  $sort: {
    state_name: 1,
    population: -1
  }
 },
 {
  $group: {
    _id: "$state_name",
    LARGEST_COUNTY: { $first: { county_name: "$county_name", population: "$population" } },
    SMALLEST_COUNTY: { $last: { county_name: "$county_name", population: "$population" } }
```

```
      }
    },
    {
      $project: {
        state_name: "$_id",
        LARGEST_COUNTY: "$LARGEST_COUNTY.county_name",
        LARGEST_COUNTY_POPULATION: "$LARGEST_COUNTY.population",
        SMALLEST_COUNTY: "$SMALLEST_COUNTY.county_name",
        SMALLEST_COUNTY_POPULATION: "$SMALLEST_COUNTY.population"
      }
    }
  ])
```

Get the states with a total population of over 10 million.



Get the average city population by state.

Get the largest and the smallest city in each state.

```
abd_project> db.states.aggregate([
...   {
...     $sort: {
...       state_name: 1,
...       population: -1
...     }
...   },
...   {
...     $group: {
...       _id: "$state_name",
...       LARGEST_CITY: { $first: { city: "$city", population: "$population" } },
...       SMALLEST_CITY: { $last: { city: "$city", population: "$population" } }
...     }
...   },
...   {
...     $project: {
...       state_name: "$_id",
...       LARGEST_CITY: "$LARGEST_CITY.city",
...       LARGEST_CITY_POPULATION: "$LARGEST_CITY.population",
...       SMALLEST_CITY: "$SMALLEST_CITY.city",
...       SMALLEST_CITY_POPULATION: "$SMALLEST_CITY.population"
...     }
...   }
... ])
[
  {
    _id: 'American Samoa',
    state_name: 'American Samoa',
    LARGEST_CITY: 'Pago Pago',
    LARGEST_CITY_POPULATION: '',
    SMALLEST_CITY: 'Pago Pago',
    SMALLEST_CITY_POPULATION: ''
  },
  {
    _id: 'New Hampshire',
    state_name: 'New Hampshire',
    LARGEST_CITY: 'Manchester',
    LARGEST_CITY_POPULATION: 37093,
    SMALLEST_CITY: 'Dalton',
    SMALLEST_CITY_POPULATION: 0
  },
  {
    _id: 'Iowa',
    state_name: 'Iowa',
    LARGEST_CITY: 'Ankeny',
    LARGEST_CITY_POPULATION: 45195,
    SMALLEST_CITY: 'Des Moines',
    SMALLEST_CITY_POPULATION: 0
  },
  {
    _id: 'New Jersey',
    state_name: 'New Jersey',
    LARGEST_CITY: 'Lakewood',
```

Get the largest and the smallest counties in each state.

```
abd_project>    db.states.aggregate([
...   {
...     $sort: {
...       state_name: 1,
...       population: -1
...     }
...   },
...   {
...     $group: {
...       _id: "$state_name",
...       LARGEST_COUNTY: { $first: { county_name: "$county_name", population: "$population" } },
...       SMALLEST_COUNTY: { $last: { county_name: "$county_name", population: "$population" } }
...     }
...   },
...   {
...     $project: {
...       state_name: "$_id",
...       LARGEST_COUNTY: "$LARGEST_COUNTY.county_name",
...       LARGEST_COUNTY_POPULATION: "$LARGEST_COUNTY.population",
...       SMALLEST_COUNTY: "$SMALLEST_COUNTY.county_name",
...       SMALLEST_COUNTY_POPULATION: "$SMALLEST_COUNTY.population"
...     }
...   }
... ])
[
  {
    _id: 'Oklahoma',
    state_name: 'Oklahoma',
    LARGEST_COUNTY: 'Canadian',
    LARGEST_COUNTY_POPULATION: 83451,
    SMALLEST_COUNTY: 'Bryan',
    SMALLEST_COUNTY_POPULATION: 0
  },
  {
    _id: 'New Hampshire',
    state_name: 'New Hampshire',
    LARGEST_COUNTY: 'Hillsborough',
    LARGEST_COUNTY_POPULATION: 37093,
    SMALLEST_COUNTY: 'Coos',
    SMALLEST_COUNTY_POPULATION: 0
  },
  {
    _id: 'New Jersey',
    state_name: 'New Jersey',
    LARGEST_COUNTY: 'Ocean',
    LARGEST_COUNTY_POPULATION: 134008,
    SMALLEST_COUNTY: 'Hunterdon',
```