

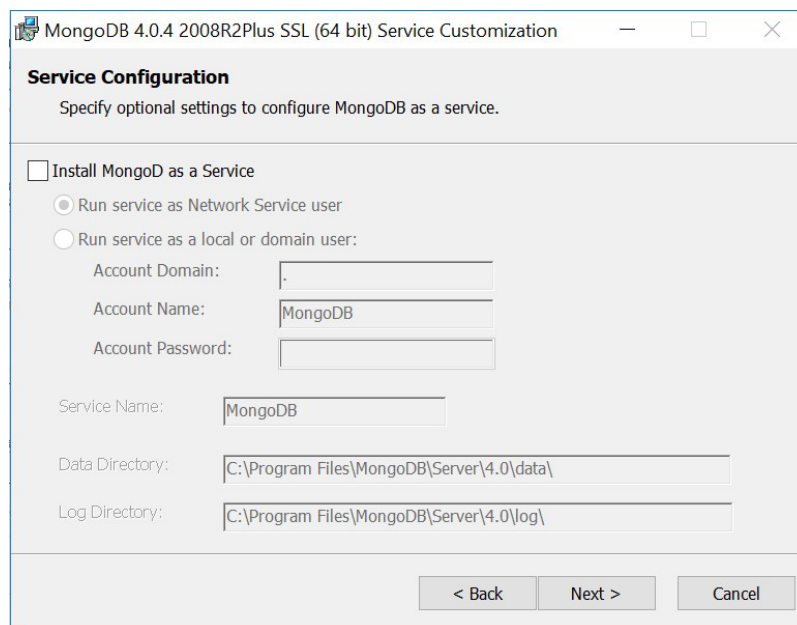
Single-node MongoDB

Overview

1. Setup MongoDB
2. Create a data directory
3. Start MongoDB
4. Import Dataset
5. Connect to MongoDB
6. Lab Assignment

1. Download and install MongoDB Community Server

- 1.1 Go to <https://www.mongodb.com/try/download/community>
- 1.2 Download MongoDB current version (or later than 4.X) for Windows 64-bit or for other OSs.
- 1.3 Double click the downloaded file (.msi) and proceed the installation.
- 1.4 MongoDB will be installed as a service by default. However, for this lab, we will choose NOT to install it as a service by UNCHECK this box.



- 1.5 Installation of Compass is NOT necessary.

Finally, MongoDB will be installed at **C:\Program Files\MongoDB**

2. Create a data directory with your student ID, for example, **C:\u63xxxxx**

3. Start MongoDB

- 3.1 Open command prompt (Start → Run → cmd)
- 3.2 Run mongod.exe in the command prompt with your ID as database path
(version may vary)

```
"C:\Program Files\MongoDB\Server\4.0\bin\mongod.exe" --dbpath "C:\u63xxxxx"
```

```

C:\Windows\system32\cmd.exe - "C:\Program Files\MongoDB\Server\4.0\bin\mongod.exe"
2018-11-21T02:20:44.343-0800 I FTDC [initandlisten] Initializing full-ti
diagnostic data capture with directory 'C:/ekasit/diagnostic.data'
2018-11-21T02:20:44.497-0800 I STORAGE [LogicalSessionCacheRefresh] createC
ction: config.system.sessions with generated UUID: 78c1db9e-536a-4e67-bfc4-4
a308ec7
2018-11-21T02:20:44.513-0800 I NETWORK [initandlisten] waiting for connecti
on port 27017
2018-11-21T02:20:44.878-0800 I INDEX [LogicalSessionCacheRefresh] build i
on: config.system.sessions properties: { v: 2, key: { lastUse: 1 }, name: "
TTLIndex", ns: "config.system.sessions", expireAfterSeconds: 1800 }
2018-11-21T02:20:44.879-0800 I INDEX [LogicalSessionCacheRefresh] bui
g index using bulk method; build may temporarily use up to 500 megabytes of
2018-11-21T02:20:44.882-0800 I INDEX [LogicalSessionCacheRefresh] build i
done. scanned 0 total records. 0 secs
2018-11-21T02:20:44.883-0800 I COMMAND [LogicalSessionCacheRefresh] command
fig.$cmd command: createIndexes { createIndexes: "system.sessions", indexes:
key: { lastUse: 1 }, name: "lsidTTLIndex", expireAfterSeconds: 1800 } 1, $d
config" } numYields:0 reslen:114 locks:< Global: { acquireCount: { r: 2, w:
}, Database: { acquireCount: { w: 2, W: 1 } }, Collection: { acquireCount: {
2 } } } protocol:op_msg 386ms

```

3.3 This will start MongoDB server that listens on port 27017

3.4 Leave this command-prompt window open

4. Import example dataset

4.1 Download the restaurant sample dataset from e-learning web site, or
<https://raw.githubusercontent.com/mongodb/docs-assets/primer-dataset/primerdataset.json>

The following is a sample document in the restaurants collection.

```

{
  "address": {
    "building": "1007",
    "coord": [ -73.856077, 40.848447 ],
    "street": "Morris Park Ave",
    "zipcode": "10462"
  },
  "borough": "Bronx",
  "cuisine": "Bakery",
  "grades": [
    { "date": { "$date": 1393804800000 }, "grade": "A", "score": 2 },
    { "date": { "$date": 1378857600000 }, "grade": "A", "score": 6 },
    { "date": { "$date": 1358985600000 }, "grade": "A", "score": 10 },
    { "date": { "$date": 1322006400000 }, "grade": "A", "score": 9 },
    { "date": { "$date": 1299715200000 }, "grade": "B", "score": 14 }
  ],
  "name": "Morris Park Bake Shop",
  "restaurant_id": "30075445" }

```

4.2 Open another command prompt

4.3 Run the following command to import data. Replace “%HOMEPATH%\Downloads\primer-dataset.json” with the path where you store your downloaded dataset file. There are 25,359 documents.

(This is a single-line command)

```
"c:\Program Files\MongoDB\Server\4.0\bin\mongoimport.exe" --db test --collection restaurants --drop --file "%HOMEPATH%\Downloads\primerdataset.json"
```

```
C:\Windows\system32\cmd.exe
C:\Users\ekasit>"c:\Program Files\MongoDB\Server\4.0\bin\mongoimport.exe" --db test --collection restaurants --drop --file "%HOMEPATH%\Downloads\primer-dataset.json"
2018-11-21T02:30:48.121-0800    connected to: localhost
2018-11-21T02:30:48.123-0800    dropping: test.restaurants
2018-11-21T02:30:50.168-0800    [#####.....] test.restaurants
9.18MB/11.3MB (72.3%)
2018-11-21T02:30:50.644-0800    [#####] test.restaurants
11.3MB/11.3MB (100.0%)
2018-11-21T02:30:50.644-0800    imported 25359 documents

C:\Users\ekasit>
```

5. Connect to MongoDB

5.1 Run mongo.exe in the command prompt

"C:\Program Files\MongoDB\Server\4.0\bin\mongo.exe"

5.2 The mongo shell is ready to receive command

```
C:\Windows\system32\cmd.exe - "C:\Program Files\MongoDB\Server\4.0\bin\...
ol is not enabled for the database.
2018-11-21T02:13:17.728-0800 I CONTROL [initandlisten] **      Read and write
access to data and configuration is unrestricted.
2018-11-21T02:13:17.728-0800 I CONTROL [initandlisten]
-----
Enable MongoDB's free cloud-based monitoring service, which will then receive and
display
metrics about your deployment (disk utilization, CPU, operation statistics, etc)
.

The monitoring data will be available on a MongoDB website with a unique URL acc
essible to you
and anyone you share the URL with. MongoDB may use this information to make prod
uct
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeM
onitoring()
-----
> _
```

5.3 Type `use test` command to switch to use database named “test”

5.4 Query the number of documents with the following query. There should be 25,359 documents in total.

`db.restaurants.count()`

```
> use test
switched to db test
> db.restaurants.count()
25359
>
```

Lab Assignment

Write the queries to:

1. Display all distinct cuisines

Ans: `db.restaurants.distinct("cuisine");`

```
test> db.restaurants.distinct("cuisine");
[
  'Afghan',
  'African',
  'American',
  'Armenian',
  'Asian',
  'Australian',
  'Bagels/Pretzels',
  'Bakery',
  'Bangladeshi',
  'Barbecue',
  'Bottled beverages, including water, sodas, juices, etc.',
  'Brazilian',
  'Café/Coffee/Tea',
  'Café/Coffee/Tea',
  'Cajun',
  'Californian',
  'Caribbean',
  'Chicken',
  'Chilean',
  'Chinese',
  'Chinese/Cuban',
  'Chinese/Japanese',
  'Continental',
  'Creole',
  'Creole/Cajun',
  'Czech',
  'Delicatessen',
  'Donuts',
  'Eastern European',
  'Egyptian',
  'English',
  'Ethiopian',
  'Filipino',
  'French',
  'Fruits/Vegetables',
  'German',
  'Greek',
  'Hamburgers',
  'Hawaiian',
  'Hotdogs',
  'Hotdogs/Pretzels',
  'Ice Cream, Gelato, Yogurt, Ices',
  'Indian',
  'Indonesian',
  'Iranian',
  'Irish',
  'Italian',
  'Japanese',
  'Jewish/Kosher',
  'Juice, Smoothies, Fruit Salads',
  'Korean',
  'Latin (Cuban, Dominican, Puerto Rican, South & Central American)',
  'Mediterranean',
  'Mexican',
  'Middle Eastern',
  'Moroccan',
  'Not Listed/Not Applicable',
  'Nuts/Confectionary',
  'Other',
  'Pakistani',
  'Pancakes/Waffles',
  'Peruvian',
  'Pizza',
  'Pizza/Italian',
  'Polish',
  'Polynesian',
  'Portuguese',
  'Russian',
  'Salads',
  'Sandwiches',
  'Sandwiches/Salads/Mixed Buffet',
  'Scandinavian',
  'Seafood',
  'Soul Food',
  'Soups',
  'Soups & Sandwiches',
  'Southwestern',
  'Spanish',
  'Steak',
  'Tapas',
  'Tex-Mex',
  'Thai',
  'Turkish',
  'Vegetarian',
  'Vietnamese/Cambodian/Malaysia'
```

2. Display only restaurant_id, name, and cuisine fields of all restaurants without the _id field

Ans: `db.restaurants.find({},{"restaurant_id" : 1,"name":1,"cuisine":1,"_id":0});`

```
test> db.restaurants.find({},{"restaurant_id" : 1,"name":1,"cuisine" :1,"_id":0});
[
  {
    cuisine: 'Bakery',
    name: 'Morris Park Bake Shop',
    restaurant_id: '30075445'
  },
  {
    cuisine: 'Irish',
    name: 'Dj Reynolds Pub And Restaurant',
    restaurant_id: '30191841'
  },
  {
    cuisine: 'Ice Cream, Gelato, Yogurt, Ices',
    name: 'Taste The Tropics Ice Cream',
    restaurant_id: '40356731'
  },
  {
    cuisine: 'Jewish/Kosher',
    name: 'Tov Kosher Kitchen',
    restaurant_id: '40356068'
  },
  {
    cuisine: 'Ice Cream, Gelato, Yogurt, Ices',
    name: 'Carvel Ice Cream',
    restaurant_id: '40360076'
  },
  {
    cuisine: 'Delicatessen',
    name: 'Nordic Delicacies',
    restaurant_id: '40361390'
  },
  {
    cuisine: 'Ice Cream, Gelato, Yogurt, Ices',
    name: 'Carvel Ice Cream',
    restaurant_id: '40361322'
  },
  {
    cuisine: 'American',
    name: 'Glorious Food',
    restaurant_id: '40361521'
  },
  {
    cuisine: 'American',
    name: 'The Movable Feast',
    restaurant_id: '40361606'
  },
  {
    cuisine: 'Delicatessen',
    name: 'Sal'S Deli',
    restaurant_id: '40361618'
  },
  {
    cuisine: 'Delicatessen',
    name: 'Bully'S Deli',
    restaurant_id: '40361708'
  },
  {
    cuisine: 'Delicatessen',
    name: 'Steve Chu'S Deli & Grocery',
    restaurant_id: '40361998'
  },
  {
    cuisine: 'Hamburgers',
    name: 'White Castle',
    restaurant_id: '40362344'
  },
  {
    cuisine: 'American',
    name: 'P & S Deli Grocery',
    restaurant_id: '40362264'
  },
  {
    cuisine: 'Jewish/Kosher',
    name: 'Kosher Island',
    restaurant_id: '40356442'
  },
  {
    cuisine: 'American',
    name: 'Angelika Film Center',
    restaurant_id: '40362274'
  },
]
```

3. Find restaurants (display only name) in Manhattan (borough field equal "Manhattan")

Ans: **db.restaurants.find({borough: "Manhattan"}, {name:1,_id:0});**

```
test> db.restaurants.find( {borough: "Manhattan"}, {name:1,_id:0} );
[
  { name: 'Dj Reynolds Pub And Restaurant' },
  { name: 'Glorious Food' },
  { name: 'Bully'S Deli' },
  { name: 'P & S Deli Grocery' },
  { name: 'Angelika Film Center' },
  { name: 'Harriet'S Kitchen' },
  { name: 'The Country Cafe' },
  { name: 'Downtown Deli' },
  { name: '1 East 66Th Street Kitchen' },
  { name: 'Olive'S' },
  { name: 'Lexler Deli' },
  { name: 'Domino'S Pizza' },
  { name: 'Lorenzo & Maria'S' },
  { name: 'Cafe Metro' },
  { name: 'Domino'S Pizza' },
  { name: 'Spoon Bread Catering' },
  { name: 'Berkely' },
  { name: 'Texas Rotisserie' },
  { name: 'Isle Of Capri Resturant' },
  { name: 'Metropolitan Club' }
]
```

4. Find restaurants (display only name) in Manhattan which has cuisine "Bakery"

Ans: **db.restaurants.find({borough: "Manhattan",cuisine:"Bakery"}, {name:1,_id:0});**

```
test> db.restaurants.find( {borough: "Manhattan",cuisine:"Bakery"}, {name:1,_id:0} );
[
  { name: 'Olive'S' },
  { name: 'De Robertis Pastry Shop' },
  { name: 'Little Pie Company' },
  { name: 'H & H Midtown Bagels East' },
  { name: 'Fay Da Bakery' },
  { name: 'Caffe Roma' },
  { name: 'Tai Pan Bakery' },
  { name: 'Glaser'S Bakery' },
  { name: 'La Delice Pastry Shop' },
  { name: 'Lung Moon Bakery' },
  { name: 'Capri Bakery' },
  { name: 'Amy'S Bread' },
  { name: 'Manna House' },
  { name: 'Zaro'S Bread Basket' },
  { name: 'Zaro'S Bread Basket' },
  { name: 'Amy'S Bread' },
  { name: 'Levain Bakery' },
  { name: 'Chiu Hong Bakery' },
  { name: 'Sweet Chef Southern Styles Bakery' },
  { name: 'Balthazar Bakery' }
]
```

5. Find restaurants (display name and borough) in Manhattan or in Bronx

Ans: `db.restaurants.find({$or: [{borough: "Manhattan"}, {borough: "Bronx"}]}, {name:1, borough:1, _id:0});`

```
test> db.restaurants.find({$or: [{borough: "Manhattan"}, {borough: "Bronx"}]}, {name:1, borough:1, _id:0});
[
  { borough: 'Bronx', name: 'Morris Park Bake Shop' },
  { borough: 'Manhattan', name: 'Dj Reynolds Pub And Restaurant' },
  { borough: 'Manhattan', name: 'Glorious Food' },
  { borough: 'Manhattan', name: 'Bully'S Deli' },
  { borough: 'Manhattan', name: 'P & S Deli Grocery' },
  { borough: 'Manhattan', name: 'Angelika Film Center' },
  { borough: 'Manhattan', name: 'Harriet'S Kitchen' },
  { borough: 'Manhattan', name: 'The Country Cafe' },
  { borough: 'Bronx', name: 'Carvel Ice Cream' },
  { borough: 'Manhattan', name: 'Downtown Deli' },
  { borough: 'Manhattan', name: '1 East 66Th Street Kitchen' },
  { borough: 'Bronx', name: 'Wild Asia' },
  { borough: 'Bronx', name: 'Happy Garden' },
  { borough: 'Manhattan', name: 'Olive'S' },
  { borough: 'Manhattan', name: 'Lexler Deli' },
  { borough: 'Manhattan', name: 'Domino'S Pizza' },
  { borough: 'Manhattan', name: 'Lorenzo & Maria'S' },
  { borough: 'Manhattan', name: 'Cafe Metro' },
  { borough: 'Manhattan', name: 'Domino'S Pizza' },
  { borough: 'Manhattan', name: 'Spoon Bread Catering' }
]
```

6. Find restaurants (display name, borough, and cuisine) in Manhattan or in Bronx which has cuisine “Bakery”

Ans: `db.restaurants.find({cuisine:"Bakery"}, $or: [{borough: "Manhattan"}, {borough: "Bronx"}]}, {name:1, borough:1, cuisine:1, _id:0});`

```
test> db.restaurants.find( {cuisine:"Bakery"}, $or: [{borough: "Manhattan"}, {borough: "Bronx"}]}, {name:1, borough:1, cuisine:1, _id:0});
[
  {
    borough: 'Bronx',
    cuisine: 'Bakery',
    name: 'Morris Park Bake Shop'
  },
  {
    borough: 'Manhattan', cuisine: 'Bakery', name: 'Olive'S' },
  {
    borough: 'Manhattan',
    cuisine: 'Bakery',
    name: 'De Robertis Pastry Shop'
  },
  {
    borough: 'Manhattan',
    cuisine: 'Bakery',
    name: 'Little Pie Company'
  },
  {
    borough: 'Manhattan',
    cuisine: 'Bakery',
    name: 'H & H Midtown Bagels East'
  },
  { borough: 'Manhattan', cuisine: 'Bakery', name: 'Fay Da Bakery' },
  {
    borough: 'Bronx',
    cuisine: 'Bakery',
    name: 'E & L Bakery & Coffee Shop'
  },
  {
    borough: 'Manhattan', cuisine: 'Bakery', name: 'Caffe Roma' },
  { borough: 'Bronx', cuisine: 'Bakery', name: 'Zaro'S Bread Basket' },
  { borough: 'Manhattan', cuisine: 'Bakery', name: 'Tai Pan Bakery' },
  { borough: 'Manhattan', cuisine: 'Bakery', name: 'Glaser'S Bakery' },
  {
    borough: 'Manhattan',
    cuisine: 'Bakery',
    name: 'La Delice Pastry Shop'
  },
  { borough: 'Bronx', cuisine: 'Bakery', name: 'Champion Bakery' },
  { borough: 'Manhattan', cuisine: 'Bakery', name: 'Lung Moon Bakery' },
  { borough: 'Manhattan', cuisine: 'Bakery', name: 'Capri Bakery' },
  { borough: 'Bronx', cuisine: 'Bakery', name: 'Franco Bakery' },
  { borough: 'Manhattan', cuisine: 'Bakery', name: 'Amy'S Bread' },
  { borough: 'Bronx', cuisine: 'Bakery', name: 'Biarritz Bakery' },
  { borough: 'Manhattan', cuisine: 'Bakery', name: 'Manna House' },
  {
    borough: 'Manhattan',
    cuisine: 'Bakery',
    name: 'Zaro'S Bread Basket'
  }
]
```

7. Find restaurants (display only name) in Bronx sorted by name in descending order

Ans: `db.restaurants.find({borough: "Bronx"}, {name:1, _id:0}).sort({"name":-1});`

```
test> db.restaurants.find( {borough: "Bronx"}, {name:1, _id:0}).sort({"name":-1});
[
  { name: 'Zymi Bar & Grill' },
  { name: 'Zoodo' },
  { name: 'Zone Culinaire' },
  { name: 'Zime Bistro' },
  { name: 'Zhang'S China Palace' },
  { name: 'Zero Otto Nove Trattoria' },
  { name: 'Zerega Avenue Deli' },
  { name: 'Zaro'S Bread Basket' },
  { name: 'Yung Hsin Restaurant' },
  { name: 'Yummy House' },
  { name: 'Your Daley Bread' },
  { name: 'Yolanda Pizzeria Restaurant' },
  { name: 'Yokohama Japanese Restaurant' },
  { name: 'Yo Angel Bakery' },
  { name: 'Yankees Clubhouse Kitchen' },
  { name: 'Yankee Tavern' },
  { name: 'Yankee Jz Pizza' },
  { name: 'Yankee Bar & Grill' },
  { name: 'Yang'S Happy Garden' },
  { name: 'Yang'S Good Taste' }
]
```

8. Find all restaurants in Bronx with zipcode 10462

Ans: `db.restaurants.find({$and:[{borough: "Bronx"}, {"address.zipcode":"10462"}]}).pretty();`

```
test> db.restaurants.find({$and:[{borough: "Bronx"}, {"address.zipcode":"10462"}]}).pretty();
{
  _id: ObjectId("636ca70038f5ed0aead839c4"),
  address: {
    building: '1007',
    coord: [ -73.856077, 40.848447 ],
    street: 'Morris Park Ave',
    zipcode: '10462'
  },
  borough: 'Bronx',
  cuisine: 'Bakery',
  grades: [
    {
      date: ISODate("2014-03-03T00:00:00.000Z"),
      grade: 'A',
      score: 2
    },
    {
      date: ISODate("2013-09-11T00:00:00.000Z"),
      grade: 'A',
      score: 6
    },
    {
      date: ISODate("2013-01-24T00:00:00.000Z"),
      grade: 'A',
      score: 10
    },
    {
      date: ISODate("2011-11-23T00:00:00.000Z"),
      grade: 'A',
      score: 9
    },
    {
      date: ISODate("2011-03-10T00:00:00.000Z"),
      grade: 'B',
      score: 14
    }
  ],
  name: 'Morris Park Bake Shop',
  restaurant_id: '30075445'
},
{
  _id: ObjectId("636ca70038f5ed0aead83a0f"),
  address: {
    building: '2222',
    coord: [ -73.84971759999999, 40.8304811 ],
    street: 'Haviland Avenue',
    zipcode: '10462'
  },
  borough: 'Bronx',
  cuisine: 'American',
  grades: [
    {
      date: ISODate("2014-12-18T00:00:00.000Z"),
      grade: 'A',
      score: 7
    },
    {
      date: ISODate("2014-05-01T00:00:00.000Z"),
      grade: 'B',
      score: 17
    },
    {
      date: ISODate("2013-03-14T00:00:00.000Z"),
      grade: 'A',
      score: 12
    }
  ]
}
```


9. Describe (in English sentences) what this query returns. Do not capture output screen.

```
db.restaurants.find( { "grades.score": { $gt: 30 } } )
```

Ans: Display all restaurants that have a grade greater than 30.

10. Describe what this query returns.

```
db.restaurants.aggregate([
  { $match: { "cuisine": "Pizza" } },
  { $group: {
    _id: "$borough", max_score: { $max: { $max: "$grades.score" } }
  } }
])
```

Ans: Display a group of all restaurants whose cuisine is pizza and group values with the borough and maximum score of this restaurant.

Save your answers into a PDF file, and submit to the mycourse website.

References:

- <http://www.w3resource.com/mongodb-exercises/>
- <https://docs.mongodb.org/getting-started/shell/query/>