

Firstly, Download the **MongoDB Command Line Database Tools** and setup path in Environment Variable and execute in the command Prompt.

```
Select Command Prompt
Microsoft Windows [Version 10.0.19045.2728]
(c) Microsoft Corporation. All rights reserved.

C:\Users\User45>mongoimport
2023-04-13T11:03:41.263+0530 no collection specified
2023-04-13T11:03:41.263+0530 using filename '' as collection
2023-04-13T11:03:41.265+0530 error validating settings: invalid collection name: collection name cannot be an empty string

C:\Users\User45>
```

1. Import vendorsJson file in database with vendors collection

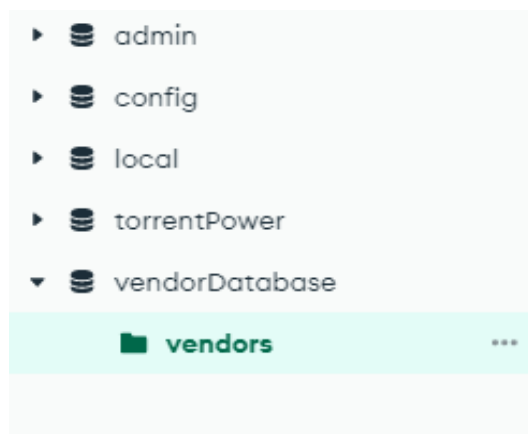
- mongoimport
"D:/Sakshi Chauhan/MongoDb/Mongoimport/vendorsJson.json"
-d vendorDatabase
-c vendors
--jsonArray

```
C:\Users\User45>mongoimport "D:/Sakshi Chauhan/MongoDb/Mongoimport/vendorsJson.json" -d vendorDatabase -c vendors --jsonArray
2023-04-13T11:33:35.009+0530 connected to: mongodb://localhost/
2023-04-13T11:33:35.269+0530 3772 document(s) imported successfully. 0 document(s) failed to import.

C:\Users\User45>
```

Here is my imported Database:-

databaseName:- vendorDatabase **collectionName:-** vendors



2. Display all the documents.

- `db.vendors.find();`

vendorDatabase.vendors

3.8k 1
DOCUMENTS INDEXES

Documents Aggregations Schema Explain Plan Indexes Validation

Filter ⓘ ⓘ Type a query: { field: 'value' } Reset Find </> More Options ▶

⬇️ ADD DATA EXPORT COLLECTION 1 - 20 of 3772 ⌂ < > ⋮ {} E

```
{
  "_id": ObjectId("64379b371b29837f3d01f031"),
  "street_address": {
    "district": "Brooklyn",
    "characteristic": "Hamburgers"
  },
  "vendor_grades": Array,
  "vendor_name": "Wendy'S",
  "vendor_id": "30112340"
}
```

3. display the fields vendor_id, vendor_name, district for all the documents in the collection.

- `db.vendors.find({},{'vendor_id':1,'vendor_name':1,'district':1});`

```
> _MONGOSH
> use vendorDatabase
< 'switched to db vendorDatabase'
> db.vendors.find({},{'vendor_id':1,'vendor_name':1,'district':1});
< {
  _id: ObjectId("64379b371b29837f3d01f031"),
  district: 'Brooklyn',
  vendor_name: "Wendy'S",
  vendor_id: '30112340'
}
{
  _id: ObjectId("64379b371b29837f3d01f032"),
  district: 'Brooklyn',
  vendor_name: 'Riviera Caterer',
  vendor_id: '40356018'
}
```

- `db.vendors.find({},{'vendor_id':0,'vendor_name':0,'district':0});`

```
> db.vendors.find({},{'vendor_id':0,'vendor_name':0,'district':0});
< {
  _id: ObjectId("64379b371b29837f3d01f031"),
  street_address: {
    building: '469',
    coord: [
      -73.961704,
      40.662942
    ],
    street: 'Flatbush Avenue',
    zipcode: '11225'
  },
  characteristic: 'Hamburgers',
  vendor_grades: [
    {
      date: 2014-12-30T00:00:00.000Z,
      grade: 'A',
      score: 8
    }
  ]
}
```

4. Display the fields `vendor_id`, `vendor_name`, `street_address`. Building but removing the field `_id` for all the documents in output.

- `db.vendors.find({},{'vendor_id':1,'vendor_name':1,'street_address.building':1,'_id':0});`

```
> db.vendors.find({},{'vendor_id':1,'vendor_name':1,'street_address.building':1,'_id':0});
< {
  street_address: {
    building: '469'
  },
  vendor_name: "Wendy'S",
  vendor_id: '30112340'
}
{
  street_address: {
    building: '2780'
  },
  vendor_name: 'Riviera Caterer',
  vendor_id: '40356018'
}
```

5. display all the vendors which is in the district 'Brooklyn'

- `db.vendors.find({'district':'Brooklyn'});`

```
> db.vendors.find({'district':'Brooklyn'});
< {
  _id: ObjectId("64379b371b29837f3d01f031"),
  street_address: {
    building: '469',
    coord: [
      -73.961704,
      40.662942
    ],
    street: 'Flatbush Avenue',
    zipcode: '11225'
  },
  district: 'Brooklyn',
  characteristic: 'Hamburgers',
  vendor_grades: [
    {
```

- Total count:-

```
Type "it" for more
> db.vendors.find({'district':'Brooklyn'}).count()
< 684
vendorDatabase> |
```

6. display the first 5 vendors which is in the district 'Brooklyn'

- `db.vendors.find({'district':'Brooklyn'}).limit(5)`

```
> db.vendors.find({'district':'Brooklyn'}).limit(5)
< {
  _id: ObjectId("64379b371b29837f3d01f031"),
  street_address: {
    building: '469',
    coord: [
      -73.961704,
      40.662942
    ],
    street: 'Flatbush Avenue',
    zipcode: '11225'
  },
  district: 'Brooklyn',
  characteristic: 'Hamburgers',
  vendor_grades: [
    {
      date: 2014-12-30T00:00:00.000Z,
```

- Total count:-

```
> db.vendors.find({'district':'Brooklyn'}).limit(5).count()
< 5
vendorDatabase> |
```

7. display the next 5 vendors after skipping first 5 which is in the district 'Brooklyn'

- `db.vendors.find({'district':'Brooklyn'}).skip(5).limit(5)`

```
> db.vendors.find({'district':'Brooklyn'}).skip(5).limit(5)
< {
  _id: ObjectId("64379b371b29837f3d01f03b"),
  street_address: {
    building: '1269',
    coord: [
      -73.871194,
      40.6730975
    ],
    street: 'Sutter Avenue',
    zipcode: '11208'
  },
  district: 'Brooklyn',
  characteristic: 'Chinese',
  vendor_grades: [
    {
      date: 2014-09-16T00:00:00.000Z,
      grade: 'B',
      score: 21
    },
  ],
}
```

8. find the vendors who achieved a score more than 90 in vendor_grades

- `db.vendors.find({vendor_grades : { $elemMatch:{"score":{$gt : 90}}}});`
- `db.vendors.find({"vendor_grades.score" : {$gt : 90}});`

```
{
  _id: ObjectId("64379b371b29837f3d01f391"),
  street_address: {
    building: '130',
    coord: [
      -73.984758,
      40.7457939
    ],
    street: 'Madison Avenue',
    zipcode: '10016'
  },
  district: 'Manhattan',
  characteristic: 'Pizza/Italian',
  vendor_grades: [
    {
      date: 2014-12-24T00:00:00.000Z,
      grade: 'Z',
      score: 31
    },
    {
      date: 2014-06-17T00:00:00.000Z,
      grade: 'C',
      score: 98
    }
  ]
}
```

- Total count:-

```
}
> db.vendors.find({vendor_grades : { $elemMatch:{"score":{$gt : 90}}}}).count()
< 3
vendorDatabase>
```

9. find the vendors that achieved a score, more than 80 but less than 100.

- `db.vendors.find({vendor_grades :{ $elemMatch:{"score":{$gt : 80 , $lt :100}}}});`

```
{
  _id: ObjectId("64379b371b29837f3d01f391"),
  street_address: {
    building: '130',
    coord: [
      -73.984758,
      40.7457939
    ],
    street: 'Madison Avenue',
    zipcode: '10016'
  },
  district: 'Manhattan',
  characteristic: 'Pizza/Italian',
  vendor_grades: [
    {
      date: 2014-12-24T00:00:00.000Z,
      grade: 'Z',
      score: 31
    },
    {
      date: 2014-06-17T00:00:00.000Z,
      grade: 'C',
      score: 98
    }
  ]
}
```

```
{
  date: 2014-06-27T00:00:00.000Z,
  grade: 'C',
  score: 89
},
{
  date: 2013-06-06T00:00:00.000Z,
  grade: 'A',
  score: 6
}
```

- Total count:-

```
> db.vendors.find({vendor_grades : { $elemMatch:{"score":{$gt : 80 , $lt :100}}}}).count()
< 3
vendorDatabase> |
```


10. find the vendors which are located in latitude(coord) value less than -92.754168.

- `db.vendors.find({"street_address.coord" : {$lt : -92.754168}});`

```
> db.vendors.find({"street_address.coord" : {$lt : -92.754168}});
<  {
  _id: ObjectId("64379b371b29837f3d01f678"),
  street_address: {
    building: '3707',
    coord: [
      -101.8945214,
      33.5197474
    ],
    street: '82 Street',
    zipcode: '11372'
  },
  district: 'Queens',
  characteristic: 'American ',
  vendor_grades: [
    {
      date: 2014-06-04T00:00:00.000Z,
      grade: 'A',
      score: 12
    }
  ]
}
```

```
{
  _id: ObjectId("64379b371b29837f3d01fe89"),
  street_address: {
    building: '60',
    coord: [
      -111.9975205,
      42.0970258
    ],
    street: 'West Side Highway',
    zipcode: '10006'
  },
}
```

- Total count:-

```
> db.vendors.find({"street_address.coord" : {$lt : -92.754168}}).count()
< 3
vendorDatabase> |
```

11. find the vendors that characteristic is not 'Italian' and their grade score more than 70 and latitude less than -65.754168

- `db.vendors.find({
 "characteristics" : { $ne : "Italian "},
 "vendor_grades.score" : { $gt: 70},
 "street_address.coord" : { $lt : -65.754168}
});` **OR**
- `db.vendors.find({
 "characteristics" : { $not: { $eq: "Italian" } },
 "vendor_grades.score" : { $gt: 70},
 "street_address.coord" : { $lt : -65.754168}
});`

```
>_MONGOSH
{
  "_id": ObjectId("64379b371b29837f3d01f391"),
  "street_address": {
    "building": '130',
    "coord": [
      -73.984758,
      40.7457939
    ],
    "street": 'Madison Avenue',
    "zipcode": '10016'
  },
  "district": 'Manhattan',
  "characteristic": 'Pizza/Italian',
  "vendor_grades": [
    {
      "date": 2014-12-24T00:00:00.000Z,
      "grade": 'Z',
      "score": 31
    },
    {
      "date": 2014-06-17T00:00:00.000Z,
      "grade": 'C',
      "score": 98
    }
  ]
}
```

- Total count:-

```
> db.vendors.find(
    {
        "characteristics" : { $not: { $eq: "Italian" } },
        "vendor_grades.score" :{$gt: 70},
        "street_address.coord" : {$lt : -65.754168}
    }
).count();

< 9
```

12. find the vendors that characteristic is not 'Italian' and their grade score more than 70 and latitude less than -65.754168. The document must be displayed according to the characteristic in descending order.

- `db.vendors.find({"characteristic": { $ne: "Italian" }, "vendor_grades.score": { $gt: 70 }, "street_address.coord": { $lt: -65.754168 }}).sort({ characteristic: -1 })`

```
_id: ObjectId("64379b371b29837f3d01f391"),
street_address: {
  building: '130',
  coord: [
    -73.984758,
    40.7457939
  ],
  street: 'Madison Avenue',
  zipcode: '10016'
},
district: 'Manhattan',
characteristic: 'Pizza/Italian',
vendor_grades: [
  {
    date: 2014-12-24T00:00:00.000Z,
    grade: 'Z',
    score: 31
  },
  {
    date: 2014-06-17T00:00:00.000Z,
    grade: 'C',
    score: 98
  }
]
```

```

_id: ObjectId("64379b371b29837f3d01f3a1"),
street_address: {
  building: '101',
  coord: [
    -73.9243061,
    40.8276297
  ],
  street: 'East 161 Street',
  zipcode: '10451'
},
district: 'Bronx',
characteristic: 'Latin (Cuban, Dominican, Puerto Rican, South & Central American)',

```

```

_id: ObjectId("64379b371b29837f3d01f4f2"),
street_address: {
  building: '289',
  coord: [
    -73.94610279999999,
    40.7137587
  ],
  street: 'Manhattan Avenue',
  zipcode: '11211'
},
district: 'Brooklyn',
characteristic: 'Bakery',

```

- Total count:-

```

}
> db.vendors.find({
  "characteristic": { $ne: "Italian" },
  "vendor_grades.score": { $gt: 70 },
  "street_address.coord": { $lt: -65.754168 }
}).sort({ characteristic: -1 }).count();

< 8
vendorDatabase>

```

13. find the vendor_id, vendor_name, for those vendors which contain 'Gio' as first three letters for its name.

- `db.vendors.find(`
 `{vendor_name:{regex:/^Gio/ }},`
 `{ vendor_id: 1, vendor_name: 1, _id: 0 })`

```
> db.vendors.find({ vendor_name: { $regex: /^Gio/ } },{ vendor_id: 1, vendor_name: 1, _id: 0 })
<  {
  vendor_name: 'Giovanni 25',
  vendor_id: '40535297'
}
{
  vendor_name: "Giovanni'S Restaurant & Cafe",
  vendor_id: '40641408'
}
{
  vendor_name: "Giorgio'S Of Gramercy",
  vendor_id: '40741900'
}
{
  vendor_name: 'Gioiosa Caffè Espresso',
  vendor_id: '40869841'
}
vendorDatabase>
```

- Total count:-

```
db.vendors.find(
  { vendor_name: { $regex: /^Gio/ } },
  { vendor_id: 1, vendor_name: 1, _id: 0 }
).count();

< 4
```

14. find the vendor_id, vendor_name, for those vendors which contain 'cas' as last three letters for its name.

- `db.vendors.find({ vendor_name: { $regex: /cas$/ } },{ vendor_id: 1, vendor_name: 1, _id: 0 })`

```
> db.vendors.find({ vendor_name: { $regex: /cas$/ } },{ vendor_id: 1, vendor_name: 1, _id: 0 })  
  
< {  
  vendor_name: 'Cositas Ricas',  
  vendor_id: '40729439'  
}  
vendorDatabase>
```

15. find the vendor_id, vendor_name, for those vendors which contain 'Rpg' as three letters somewhere in its name

- `db.vendors.find({vendor_name:{ $regex:/rpg/i}},
{vendor_id: 1,vendor_name: 1, _id: 0 })`

```
}  
> db.vendors.find({ vendor_name: { $regex: /rpg/i } }, { vendor_id: 1, vendor_name: 1, _id: 0 })  
<  
vendorDatabase>
```

→ 0 records in this Query.

16. arrange the name of the vendors in ascending order along with all the columns.

- `db.vendors.find().sort({"vendor_name":1});`

```
}  
],  
  vendor_name: '1 East 66Th Street Kitchen',  
  vendor_id: '40359480'  
}  
{
```

```
  ],  
  vendor_name: '101 Deli',  
  vendor_id: '40591271'  
}
```

```
  ],  
  vendor_name: '101 Restaurant And Bar',  
  vendor_id: '40560108'  
}
```

```
  ],  
  vendor_name: '15 East Restaurant',  
  vendor_id: '40698892'  
}  
{  
  _id: ObjectId("64379b371b29837f3d01f284"),
```

```
  ],  
  vendor_name: '200 Fifth Avenue Restaurant & Sports Bar',  
  vendor_id: '40385555'  
}
```

17. arrange the name of the vendors in descending order along with all the columns.

- `db.vendors.find().sort({"vendor_name":-1});`

```
  }  
  ],  
  vendor_name: 'Zum Stammtisch',  
  vendor_id: '40367377'  
}
```

```
  ],  
  vendor_name: 'Zum Schneider',  
  vendor_id: '40763382'  
}
```

```

    }
  ],
  vendor_name: "Zorba'S",
  vendor_id: '40877247'
},
  ],
  vendor_name: 'Yvonne Yvonne Restaurant',
  vendor_id: '40875849'
}

```

18. query to know whether all the street_address contains the street or not.

- `db.vendors.find({"street_address.street" : { $exists : true } });`

This Query returns to if in the street_address (street or not), \$exists: true then Check if street_address street is there or not.

19. query which will select all documents in the vendors collection where the 'coord' field value is Double.

- `db.vendors.find({"street_address.coord" : {$type : 1}});`

This query will select all documents in the vendors collection where the 'coord' field value is Double.

20. find the vendors that have at least one grade with a score of less than 5 and that are located in the district of Manhattan or Brooklyn, and their characteristics are not Italian.

- `db.vendors.find({
 $and: [
 { "vendor_grades.score": { $lt: 5 } },
 { $or: [{"district": "Manhattan" }, {"district": "Brooklyn" }] },
 { "characteristic": { $ne: "Italian" } }
]
})`

```
> db.vendors.find({  
  $and: [  
    { "vendor_grades.score": { $lt: 5 } },  
    { $or: [{"district": "Manhattan" }, {"district": "Brooklyn" }] },  
    { "characteristic": { $ne: "Italian" } }  
  ]  
}).count()  
< 887  
vendorDatabase> |
```