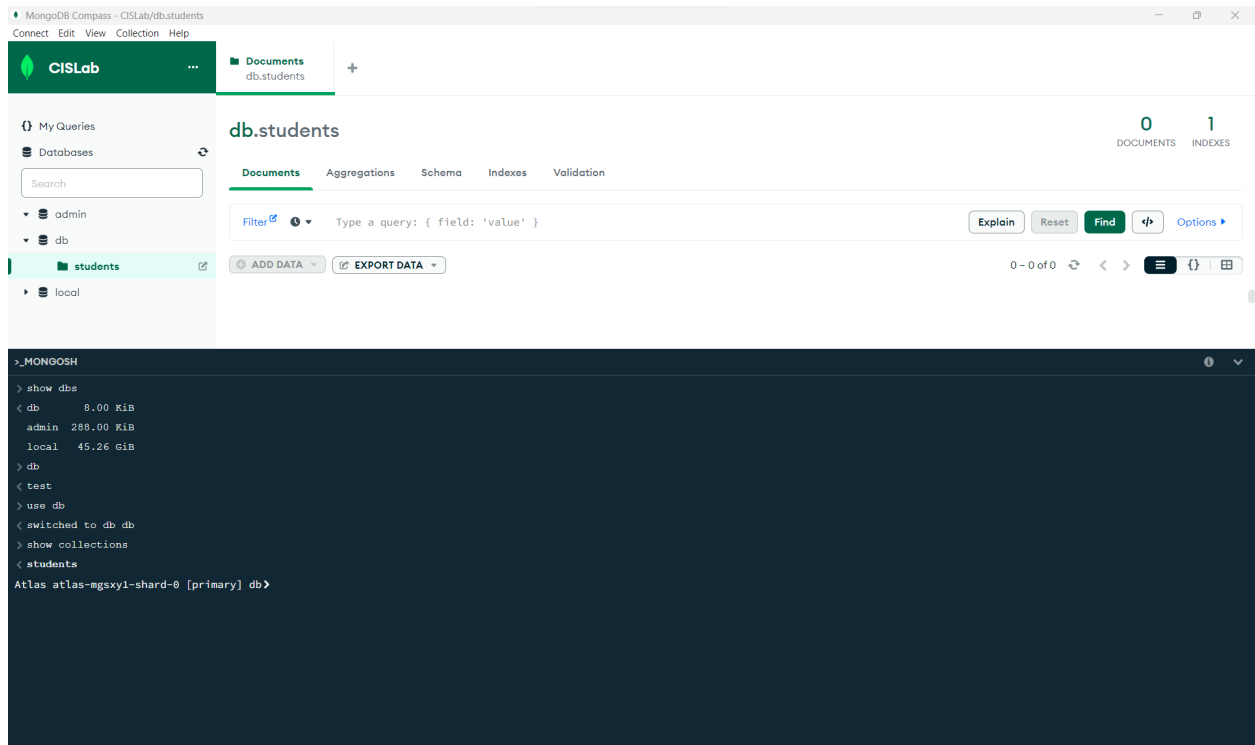
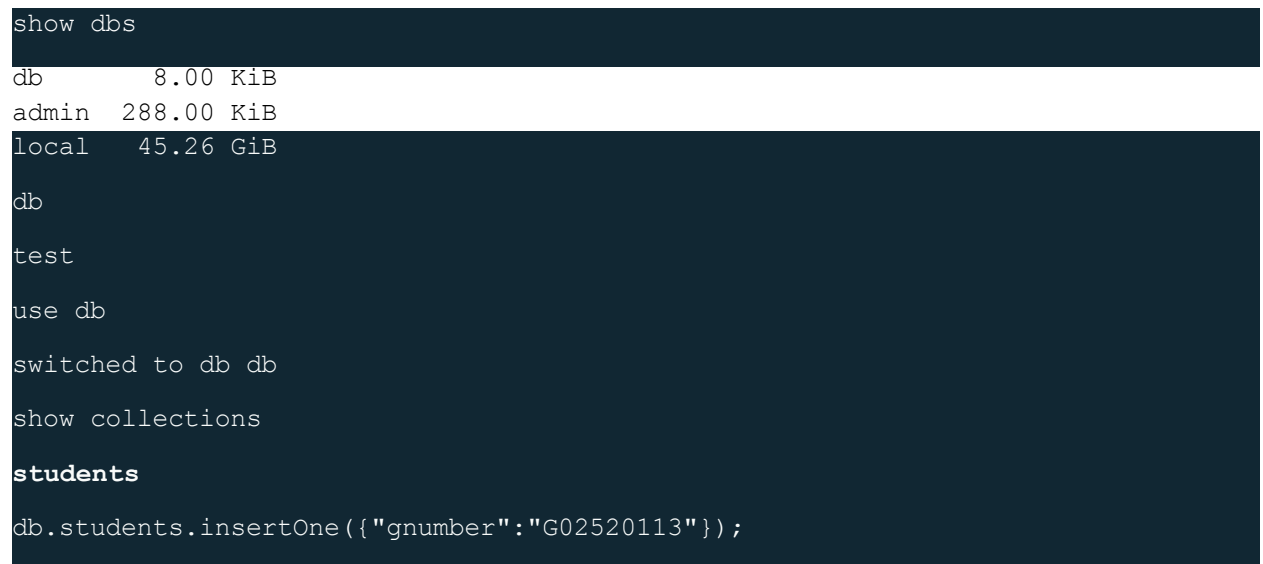


## MongoDB Lab-1

Environment setup and DB and collection creation.



Step 2.2:



**MongoServerError: 2242304:error:10000438:SSL routines:OPENSSL\_internal:TLSV1\_ALERT\_INTERNAL\_ERROR:..\..\third\_party\boringsssl\src\ssl\tls\_record.cc:592:SSL alert number 80**

```
db.students.insertOne({"gnumber":"G02520113"});

{
  acknowledged: true,
  insertedId: ObjectId("65035a6664ecb97b3bf288d1")
}

db.students.insertOne(
{
  "gnum":"G02520113",
  "firstname":"Mallikarjuna Rao",
  "middlename":null,
  "lastname":"Mannem",
  "hobbies":["Badminton", "8ball", "Outdoorsing",0,null],
  "age":27,
  "gpa":0.0,
  "gradstudent":true,
  "currentlyenrolled":true,
  "bikes":[
    { "make":"RE", "model":"Classic 350 (Grey)" },
    { "make":"Yamaha", "model":["R15 (Black)"] }
  ],
  "x":[{"key1":"val1"}, {"key2":"val2", "key3":"val3"}]
})

{
  acknowledged: true,
  insertedId: ObjectId("65035d8464ecb97b3bf288d2")
}

db.students.find(
{
  _id: ObjectId("65035a6664ecb97b3bf288d1"),
  gnumber: 'G02520113'
}
{
  _id: ObjectId("65035d8464ecb97b3bf288d2"),
  gnum: 'G02520113',
  firstname: 'Mallikarjuna Rao',
```

```
    middlename: null,
    lastname: 'Mannem',
    hobbies: [
      'Badminton',
      '8ball',
      'Outdoorsing',
    ],
    age: 27,
    gpa: 0,
    gradstudent: true,
    currentlyenrolled: true,
    bikes: [
      {
        make: 'RE',
        model: 'Classic 350 (Grey)'
      },
      {
        make: 'Yamaha',
        model: [
          'R15 (Black)'
        ]
      }
    ],
    x: [
      {
        key1: 'val1'
      },
      {
        key2: 'val2',
        key3: 'val3'
      }
    ]
  }
}
```

```

}
]
}

db.students.find({ "_id" : ObjectId("65035a6664ecb97b3bf288d1") })

{
  _id: ObjectId("65035a6664ecb97b3bf288d1"),
  gnumber: 'G02520113'
}

db.students.find({ "_id" : ObjectId("65035d8464ecb97b3bf288d2") },
{firstname:1,age:1})

{
  _id: ObjectId("65035d8464ecb97b3bf288d2"),
  firstname: 'Mallikarjuna Rao',
  age: 27
}

db.students.find(
{
  "gradstudent":true,
  age: {$gt: 26},
  lastname : {$in: ["Smith", "Brown", "Jones"]}
}
)

db.students.find(
{
  "gradstudent":true,
  age: {$gt: 26},
  lastname : {$in: ["Smith", "Mannem", "Jones"]}
}
)

{
  _id: ObjectId("65035d8464ecb97b3bf288d2"),
  gnum: 'G02520113',
  firstname: 'Mallikarjuna Rao',
  middlename: null,
  lastname: 'Mannem',
  hobbies: [

```

```
'Badminton',
'8ball',
'Outdoorsing',
0,
null
],
age: 27,
gpa: 0,
gradstudent: true,
currentlyenrolled: true,
bikes: [
{
make: 'RE',
model: 'Classic 350 (Grey)'
},
{
make: 'Yamaha',
model: [
'R15 (Black)'
]
}
],
x: [
{
key1: 'val1'
},
{
key2: 'val2',
key3: 'val3'
}
]
}
```

```
db.students.find(
{
"gradstudent":true,
age: {$lt: 26},
lastname : {$in: ["Smith", "Mannem", "Jones"]}
}
)
```

```
db.students.find(
{
$or: [
{"gradstudent":true},
{age: {$gt: 26}},
{lastname : {$in: ["Smith", "Brown", "Jones"]}}
]
}
)
```

```
{
  _id: ObjectId("65035d8464ecb97b3bf288d2"),
  gnum: 'G02520113',
  firstname: 'Mallikarjuna Rao',
  middlename: null,
  lastname: 'Mannem',
  hobbies: [
    'Badminton',
    '8ball',
    'Outdoorsing',
    0,
    null
  ],
  age: 27,
  gpa: 0,
  gradstudent: true,
  currentlyenrolled: true,
  bikes: [
    {
      make: 'RE',
      model: 'Classic 350 (Grey)'
```

```

    },
    {
      make: 'Yamaha',
      model: [
        'R15 (Black)'
      ]
    }
  ],
  x: [
    {
      key1: 'val1'
    },
    {
      key2: 'val2',
      key3: 'val3'
    }
  ]
}

```

```

db.students.find( { hobbies: [
  'Badminton',
  '8ball',
  'Outdoorsing',
  0,
  null
]

```

```

} )

{
  _id: ObjectId("65035d8464ecb97b3bf288d2"),
  gnum: 'G02520113',
  firstname: 'Mallikarjuna Rao',
  middlename: null,
  lastname: 'Mannem',
  hobbies: [
    'Badminton',

```

```
'8ball',
'Outdoorsing',
0,
null
],
age: 27,
gpa: 0,
gradstudent: true,
currentlyenrolled: true,
bikes: [
{
make: 'RE',
model: 'Classic 350 (Grey)'
},
{
make: 'Yamaha',
model: [
'R15 (Black)'
]
}
],
x: [
{
key1: 'val1'
},
{
key2: 'val2',
key3: 'val3'
}
]
}
db.students.find( { hobbies: "Badminton" } )
```



```
{
  _id: ObjectId("65035d8464ecb97b3bf288d2"),
  gnum: 'G02520113',
  firstname: 'Mallikarjuna Rao',
  middlename: null,
  lastname: 'Mannem',
  hobbies: [
    'Badminton',
    '8ball',
    'Outdoorsing',
    0,
    null
  ],
  age: 27,
  gpa: 0,
  gradstudent: true,
  currentlyenrolled: true,
  bikes: [
    {
      make: 'RE',
      model: 'Classic 350 (Grey)'
    },
    {
      make: 'Yamaha',
      model: [
        'R15 (Black)'
      ]
    }
  ],
  x: [
    {
      key1: 'vall'
```

```

},
{
  key2: 'val2',
  key3: 'val3'
}
]
}

db.students.find( { hobbies.2: "8ball" } )

```

**Error:** clone(t={}){const r=t.loc||{};return e({loc:new Position("line"in r?r.line:this.loc.line,"column"in r?r.column:...<omitted>...)} could not be cloned.

```

db.students.find( { "hobbies.2": "8ball" } )

db.students.find( { "hobbies.1": "8ball" } )

{
  _id: ObjectId("65035d8464ecb97b3bf288d2"),
  gnum: 'G02520113',
  firstname: 'Mallikarjuna Rao',
  middlename: null,
  lastname: 'Mannem',
  hobbies: [
    'Badminton',
    '8ball',
    'Outdoorsing',
  ],
  0,
  null
],
  age: 27,
  gpa: 0,
  gradstudent: true,
  currentlyenrolled: true,
  bikes: [
    {

```

```

make: 'RE',
model: 'Classic 350 (Grey)'
},
{
make: 'Yamaha',
model: [
'R15 (Black)'
]
},
],
x: [
{
key1: 'val1'
},
{
key2: 'val2',
key3: 'val3'
}
]
}

db.students.insertOne(
{
"gnum":"G0001234",
"firstname":"Jonathan",
"middlename":null,
"lastname":"Smith",
"hobbies":["SCUBA", "Falconry", "Gardening", "Outdoorsing",0,null],
"age":25,
"gpa":3.9,
"gradstudent":true,
"currentlyenrolled":false,
"cars":[
{ "make":"Ford", "model":["F150 (Red)", "F150 Crew Cab (Black)"] },
{ "make":"Chevy", "model":["Equinox (Silver)"] }
],
"x":[{"key1":"val1"}, {"key2":"val2", "key3":"val3"}]
})
{

```

```

acknowledged: true,

insertedId: ObjectId("650361b164ecb97b3bf288d3")
}

db.students.updateMany({
  _id :ObjectId("650361b164ecb97b3bf288d3") },
{$set:{"skills":
{db:"MongoDB", version:5, nosql:"document"}
}})

{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

db.students.find({ "_id" : ObjectId("650361b164ecb97b3bf288d3") })
{
  _id: ObjectId("650361b164ecb97b3bf288d3"),
  gnum: 'G0001234',
  firstname: 'Jonathan',
  middlename: 'B',
  lastname: 'Smith',
  hobbies: [
    'SCUBA',
    'Falconry',
    'Gardening',
    'Outdoorsing',
    0,
    null
  ],
  age: 25,
  gpa: 3.9,
  gradstudent: true,

```

```
currentlyenrolled: false,
cars: [
  {
    make: 'Ford',
    model: [
      'F150 (Red)',
      'F150 Crew Cab (Black)'
    ]
  },
  {
    make: 'Chevy',
    model: [
      'Equinox (Silver)'
    ]
  }
],
x: [
  {
    key1: 'val1'
  },
  {
    key2: 'val2',
    key3: 'val3'
  }
],
skills: {
  db: 'MongoDB',
  version: 5,
  nosql: 'document'
},
status: 'Cum Laude'
}
```

```
db.students.insertOne({hello:"world"})

{
  acknowledged: true,
  insertedId: ObjectId("65038fb364ecb97b3bf288d4")
}

db
db

db.students.find()

{
  _id: ObjectId("65035a6664ecb97b3bf288d1"),
  gnumber: 'G02520113'
}

{
  _id: ObjectId("65035d8464ecb97b3bf288d2"),
  gnum: 'G02520113',
  firstname: 'Mallikarjuna Rao',
  middlename: null,
  lastname: 'Mannem',
  hobbies: [
    'Badminton',
    '8ball',
    'Outdoorsing',
  ],
  age: 27,
  gpa: 0,
  gradstudent: true,
  currentlyenrolled: true,
  bikes: [
    {
      make: 'RE',
```

```
model: 'Classic 350 (Grey)'
},
{
make: 'Yamaha',
model: [
'R15 (Black)'
]
},
],
x: [
{
key1: 'val1'
},
{
key2: 'val2',
key3: 'val3'
}
]
}
{
_id: ObjectId("650361b164ecb97b3bf288d3"),
gnum: 'G0001234',
firstname: 'Jonathan',
middlename: 'B',
lastname: 'Smith',
hobbies: [
'SCUBA',
'Falconry',
'Gardening',
'Outdoorsing',
0,
null
]
```

```
],  
  age: 25,  
  gpa: 3.9,  
  gradstudent: true,  
  currentlyenrolled: false,  
  cars: [  
    {  
      make: 'Ford',  
      model: [  
        'F150 (Red)',  
        'F150 Crew Cab (Black)'  
      ]  
    },  
    {  
      make: 'Chevy',  
      model: [  
        'Equinox (Silver)'  
      ]  
    }  
  ],  
  x: [  
    {  
      key1: 'val1'  
    },  
    {  
      key2: 'val2',  
      key3: 'val3'  
    }  
  ],  
  skills: {  
    db: 'MongoDB',  
    version: 5,  
  },  
}
```



```
nosql: 'document'
},
status: 'Cum Laude'
}
{
  _id: ObjectId("65038fb364ecb97b3bf288d4"),
  hello: 'world'
}
db.students.deleteOne({hello:"world"})
{
  acknowledged: true,
  deletedCount: 1
}
db.students.find()
{
  _id: ObjectId("65035a6664ecb97b3bf288d1"),
  gnumber: 'G02520113'
}
{
  _id: ObjectId("65035d8464ecb97b3bf288d2"),
  gnum: 'G02520113',
  firstname: 'Mallikarjuna Rao',
  middlename: null,
  lastname: 'Mannem',
  hobbies: [
    'Badminton',
    '8ball',
    'Outdoorsing',
  ],
  age: 27,
```

```
gpa: 0,
gradstudent: true,
currentlyenrolled: true,
bikes: [
  {
    make: 'RE',
    model: 'Classic 350 (Grey)'
  },
  {
    make: 'Yamaha',
    model: [
      'R15 (Black)'
    ]
  }
],
x: [
  {
    key1: 'val1'
  },
  {
    key2: 'val2',
    key3: 'val3'
  }
]
}
{
  _id: ObjectId("650361b164ecb97b3bf288d3"),
  gnum: 'G0001234',
  firstname: 'Jonathan',
  middlename: 'B',
  lastname: 'Smith',
  hobbies: [
```

```
'SCUBA',
'Falconry',
'Gardening',
'Outdoorsing',
0,
null
],
age: 25,
gpa: 3.9,
gradstudent: true,
currentlyenrolled: false,
cars: [
{
make: 'Ford',
model: [
'F150 (Red)',
'F150 Crew Cab (Black)'
]
},
{
make: 'Chevy',
model: [
'Equinox (Silver)'
]
}
],
x: [
{
key1: 'val1'
},
{
key2: 'val2',
```

```
key3: 'val3'
}
],
skills: {
db: 'MongoDB',
version: 5,
nosql: 'document'
},
status: 'Cum Laude'
}
db.students.insertOne({hello:"world"})
{
acknowledged: true,
insertedId: ObjectId("6503909c64ecb97b3bf288d5")
}
db.students.deleteMany({hello:"world"})
{
acknowledged: true,
deletedCount: 1
}
Atlas atlas-mgsxyl-shard-0 [primary] db
```

**→ GUI interaction and website browsing are user friendly.  
Modifications are easy to perform.**

**Step #3:** Use MongoDB CRUD commands to store and manipulate a custom dataset Refer to these MongoDB examples. Keep a copy of all of your commands to turn in. Execute the commands and record the output from each step.

1. Download a simple, small dataset of JSON documents of your choice.

Energy Census and Economic Data US 2010-2014

2. Add at least five SEPARATE documents based on a subset of your dataset – 10% grade.

Note to yourself: how is this different than inserting one document that contains five sub-documents? Hint: DO NOT insert one big document that contains five nested sub-documents! Each document should get its own document ID.

Imported data using a csv file. 52 datasets.

3. Generate a query that retrieves a document based on its unique id – 20% grade.

```
db.students.find({ "_id" : ObjectId("6503bcc3355d9dc62be482b6") })
{
  GDP2010: 57673.75,
  GDP2011Q1: 58760,
  GDP2011Q2: 59581,
  GDP2011Q3: 60648,
  GDP2011Q4: 60064,
  GDP2011: 59763.25,
  GDP2012Q1: 61230,
  GDP2012Q2: 61132,
  GDP2012Q3: 60865,
  GDP2012Q4: 59870,
  GDP2012: 60774.25,
  GDP2013Q1: 60295,
  GDP2013Q2: 60868,
  GDP2013Q3: 61092,
  GDP2013Q4: 63440,
  GDP2013: 61423.75,
  GDP2014Q1: 63270,
  GDP2014Q2: 65382,
```

GDP2014Q3: 66893,  
GDP2014Q4: 66394,  
GDP2014: 65484.75,  
CENSUS2010POP: 897934,  
POPESTIMATE2010: 899731,  
POPESTIMATE2011: 907829,  
POPESTIMATE2012: 916881,  
POPESTIMATE2013: 925240,  
POPESTIMATE2014: 935614,  
RBIRTH2011: 12.50304278,  
RBIRTH2012: 12.22879252,  
RBIRTH2013: 11.95686928,  
RBIRTH2014: 11.96654869,  
RDEATH2011: 8.714510168,  
RDEATH2012: 8.438601202,  
RDEATH2013: 8.850667247,  
RDEATH2014: 8.896990307,  
RNATURALINC2011: 3.788532608,  
RNATURALINC2012: 3.790191318,  
RNATURALINC2013: 3.106202036,  
RNATURALINC2014: 3.069558386,  
RINTERNATIONALMIG2011: 2.436433645,  
RINTERNATIONALMIG2012: 2.62288254,  
RINTERNATIONALMIG2013: 2.608949141,  
RINTERNATIONALMIG2014: 2.565488749,  
RDOMESTICMIG2011: 2.866848127,  
RDOMESTICMIG2012: 3.598380017,  
RDOMESTICMIG2013: 3.397170979,  
RDOMESTICMIG2014: 5.148173903,  
RNETMIG2011: 5.303281772,  
RNETMIG2012: 6.221262557,  
RNETMIG2013: 6.006120119,

```
RNETMIG2014: 7.713662652
}
Atlas atlas-mgsxyl-shard-0 [primary] db
```

4. Generate a query that utilizes an 'and' condition – 20% grade.

```
db.students.find(
  {
    Region: {$lt: 4},
    State: {$in: ["Michigan", "Arizona"]}
  }
)
```

```
{
  GDP2010: 387167,
  GDP2011Q1: 395349,
  GDP2011Q2: 396078,
  GDP2011Q3: 400884,
  GDP2011Q4: 413128,
  GDP2011: 401359.75,
  GDP2012Q1: 412761,
  GDP2012Q2: 416887,
  GDP2012Q3: 419650,
  GDP2012Q4: 420015,
  GDP2012: 417328.25,
  GDP2013Q1: 428265,
  GDP2013Q2: 428285,
  GDP2013Q3: 431136,
  GDP2013Q4: 436760,
  GDP2013: 431111.5,
  GDP2014Q1: 438271,
  GDP2014Q2: 445584,
  GDP2014Q3: 450840,
```

GDP2014Q4: 454190,  
GDP2014: 447221.25,  
CENSUS2010POP: 9883640,  
POPESTIMATE2010: 9876498,  
POPESTIMATE2011: 9875736,  
POPESTIMATE2012: 9884781,  
POPESTIMATE2013: 9898193,  
POPESTIMATE2014: 9909877,  
RBIRTH2011: 11.54168182,  
RBIRTH2012: 11.44403256,  
RBIRTH2013: 11.41092335,  
RBIRTH2014: 11.38404701,  
RDEATH2011: 9.100641477,  
RDEATH2012: 9.015351167,  
RDEATH2013: 9.05687891,  
RDEATH2014: 9.124160002,  
RNATURALINC2011: 2.44104034,  
RNATURALINC2012: 2.428681395,  
RNATURALINC2013: 2.354044443,  
RNATURALINC2014: 2.259887006,  
RINTERNATIONALMIG2011: 1.825616282,  
RINTERNATIONALMIG2012: 1.87161095,  
RINTERNATIONALMIG2013: 2.029017477,  
RINTERNATIONALMIG2014: 2.028870051,  
RDOMESTICMIG2011: -4.371657403,  
RDOMESTICMIG2012: -3.394243177,  
RDOMESTICMIG2013: -2.98681078,  
RDOMESTICMIG2014: -2.895688474,  
RNETMIG2011: -2.546041121,  
RNETMIG2012: -1.522632227,  
RNETMIG2013: -0.957793302,  
RNETMIG2014: -0.866818423



```
}  
Atlas atlas-mgsxyl-shard-0 [primary] db  
Selection deleted
```

5. Generate a query that utilizes an 'or' condition – 20% grade.

```
db.students.find(  
  { $or: [  
    { Region: {$lt: 1}},  
    { State: {$in: ["Michigan"]}}  
  ]}  
)
```

LPGE2013: 1145.7

LPGE2014: 1273.4

LPGPrice2010: 22.72

LPGPrice2011: 23.91

LPGPrice2012: 22.5

LPGPrice2013: 24.33

LPGPrice2014: 27.78

GDP2010Q1: 373593

GDP2010Q2: 386054

GDP2010Q3: 394845

GDP2010Q4: 394176

GDP2010: 387167

GDP2011Q1: 395349

GDP2011Q2: 396078

GDP2011Q3: 400884

GDP2011Q4: 413128

GDP2011: 401359.75

GDP2012Q1: 412761

GDP2012Q2: 416887

GDP2012Q3: 419650

GDP2012Q4: 420015

GDP2012 : 417328.25  
GDP2013Q1 : 428265  
GDP2013Q2 : 428285  
GDP2013Q3 : 431136  
GDP2013Q4 : 436760  
GDP2013 : 431111.5  
GDP2014Q1 : 438271  
GDP2014Q2 : 445584  
GDP2014Q3 : 450840  
GDP2014Q4 : 454190  
GDP2014 : 447221.25  
CENSUS2010POP : 9883640  
POPESTIMATE2010 : 9876498  
POPESTIMATE2011 : 9875736  
POPESTIMATE2012 : 9884781  
POPESTIMATE2013 : 9898193  
POPESTIMATE2014 : 9909877  
RBIRTH2011 : 11.54168182  
RBIRTH2012 : 11.44403256  
RBIRTH2013 : 11.41092335  
RBIRTH2014 : 11.38404701  
RDEATH2011 : 9.100641477  
RDEATH2012 : 9.015351167  
RDEATH2013 : 9.05687891  
RDEATH2014 : 9.124160002  
RNATURALINC2011 : 2.44104034  
RNATURALINC2012 : 2.428681395  
RNATURALINC2013 : 2.354044443  
RNATURALINC2014 : 2.259887006  
RINTERNATIONALMIG2011 : 1.825616282  
RINTERNATIONALMIG2012 : 1.87161095  
RINTERNATIONALMIG2013 : 2.029017477

```

RINTERNATIONALMIG2014: 2.028870051,
RDOMESTICMIG2011: 4.371657403,
RDOMESTICMIG2012: 3.394243177,
RDOMESTICMIG2013: 2.98681078,
RDOMESTICMIG2014: 2.895688474,
RNETMIG2011: 2.546041121,
RNETMIG2012: 1.522632227,
RNETMIG2013: 0.957793302,
RNETMIG2014: 0.866818423
}

```

Atlas atlas-mgsxy1-shard-0 [primary] db

Selection deleted

6. Update one document by adding a new field based on an updateOne or updateMany statement – 10% grade.

```

db.students.updateMany(
  {State:"California"},
  {$set:{"StateCode":"CAL"}}
)

```

```

acknowledged: true
insertedId: null
matchedCount: 1
modifiedCount: 1
upsertedCount: 0
}

```

Atlas atlas-mgsxy1-shard-0 [primary] db

7. Delete one or more documents – 10% grade.

```

db.students.deleteMany(
  {State:"California"}
)

```

```
acknowledged: true
```

```
deletedCount: 1
```

Atlas atlas-mgsxy1-shard-0 [primary] db

Selection deleted

The screenshot shows the MongoDB Compass interface for the 'db.students' collection. The left sidebar displays the database structure with 'students' selected. The main panel shows the 'Aggregations' tab with an empty pipeline. Below the pipeline, there are buttons for 'SAVE', 'CREATE NEW', and 'EXPORT TO LANGUAGE'. A 'PREVIEW' button is also visible. The 'Preview of documents' section shows three document snippets. The first document has a 'gnumber' field. The second document has 'firstname', 'lastname', 'hobbies', 'age', and 'gpa' fields. The third document has 'firstname', 'lastname', 'hobbies', 'age', and 'gpa' fields. A '+ Add Stage' button is located at the bottom of the aggregation pipeline area, along with a link to 'Learn more about aggregation pipeline stages'.

Done for the day!