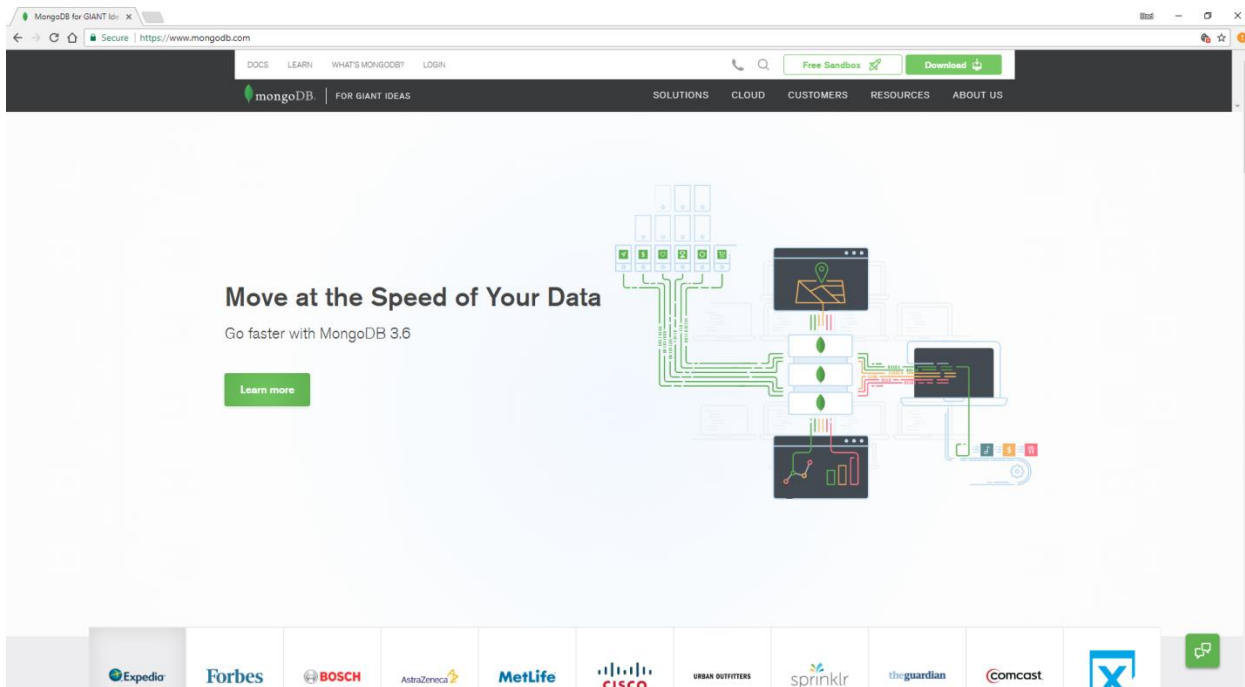
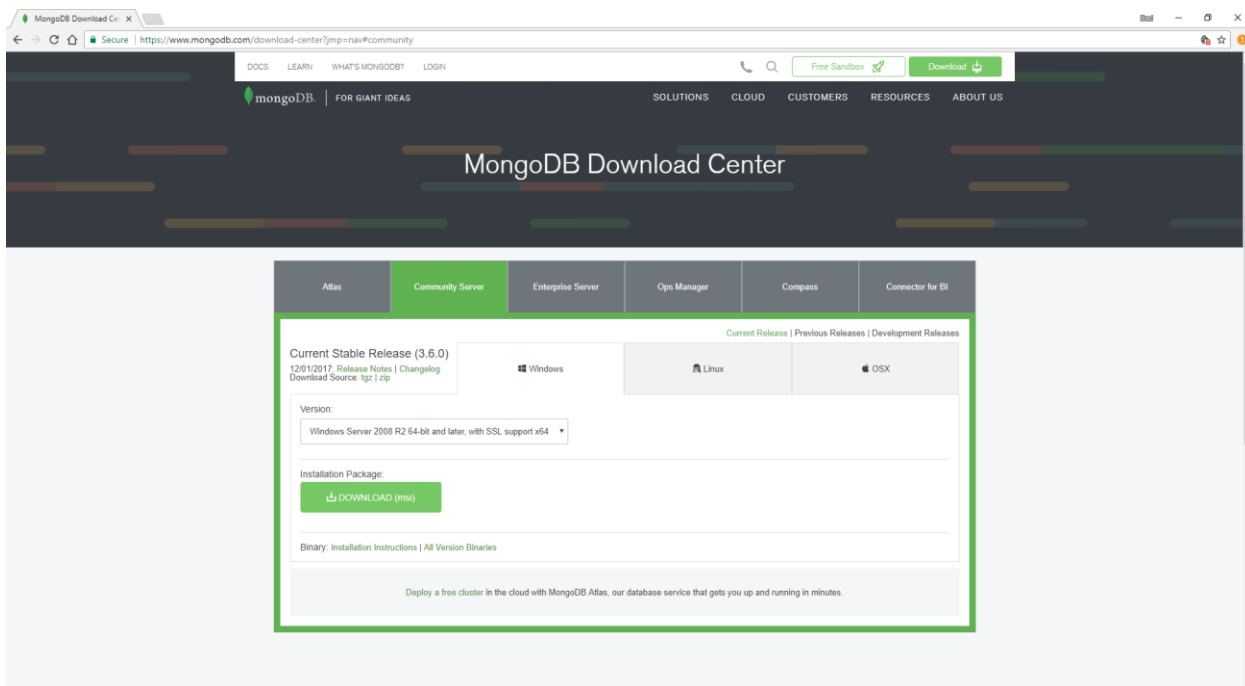


Analytics Take-Home Practicum – Partial Directions (Final)



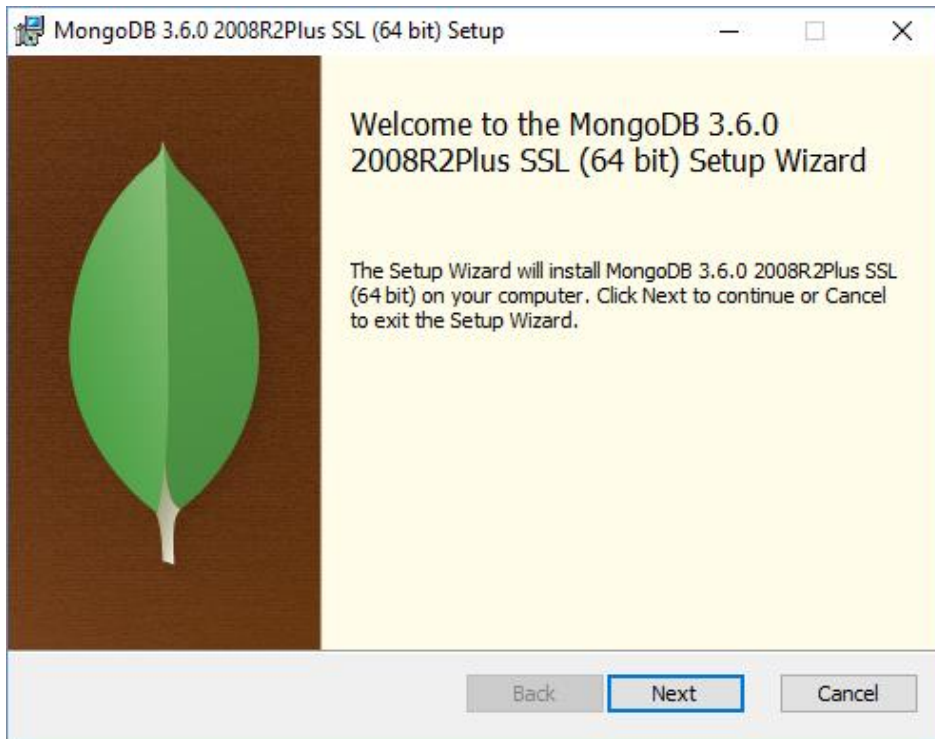
1. Start here and click the green download button.



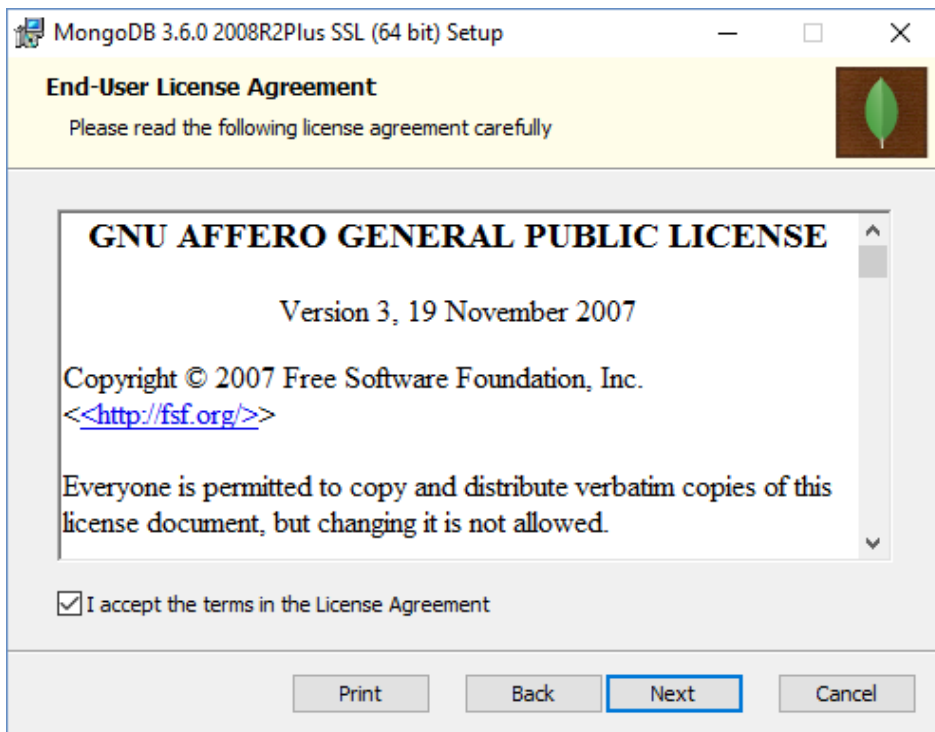
2. Select the Community Server tab and download the MSI install file. The one shown above works fine with Windows 10 (and should also work with Windows 7 & 8 but I have not tested these). If you are using Linux or a Mac then select the appropriate tab above and follow the sites directions. Most of the rest of the steps should be similar but may need to be adapted to the specific OS.

mongodb-win32-x86_64-2008plus-ssl-3.6.0-signed.msi 12/8/2017 2:07 PM Windows Installer ... 185,235 KB Installer

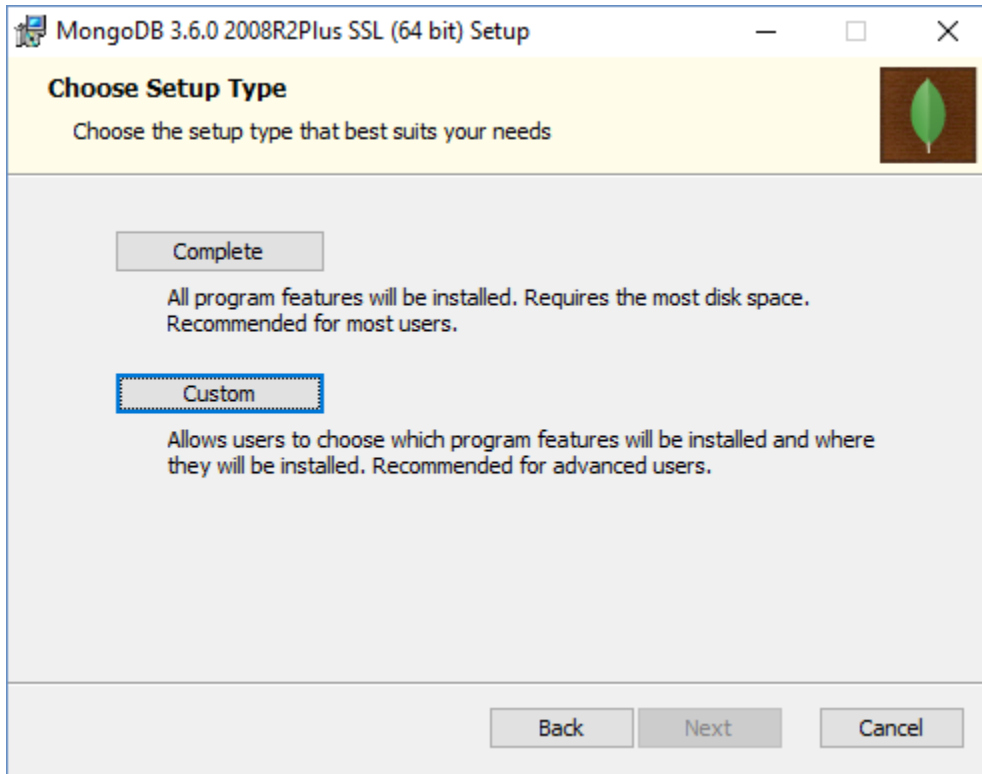
3. Right-click on downloaded file and select 'Install'.



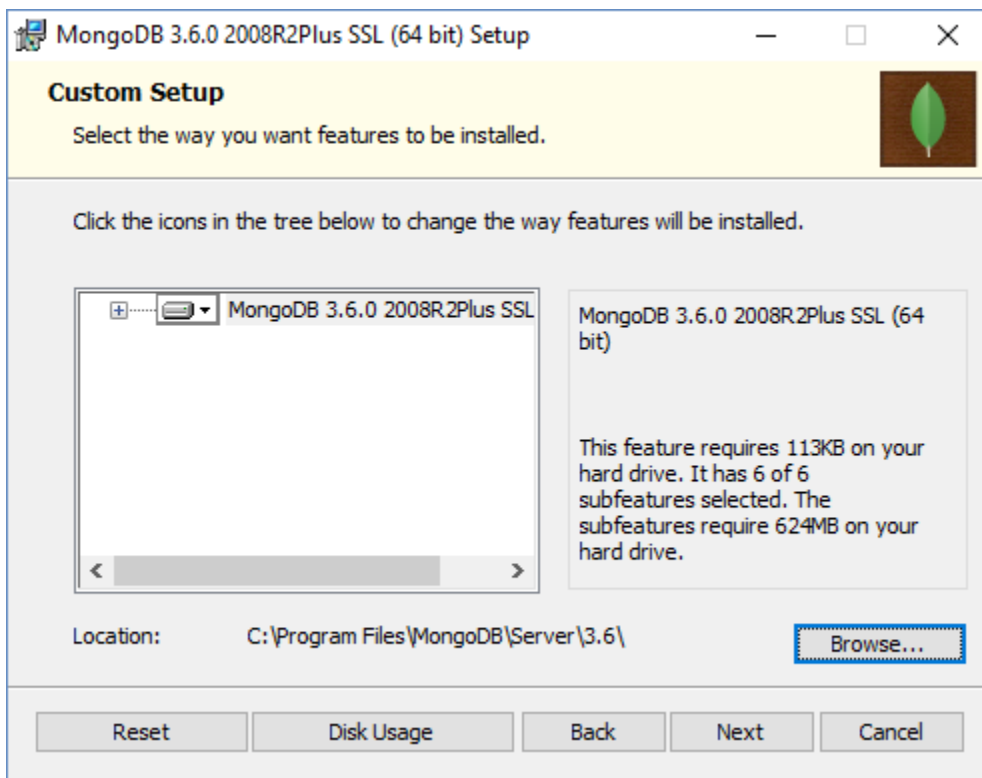
4. Click Next.



5. Read and accept the License Agreement and click Next.

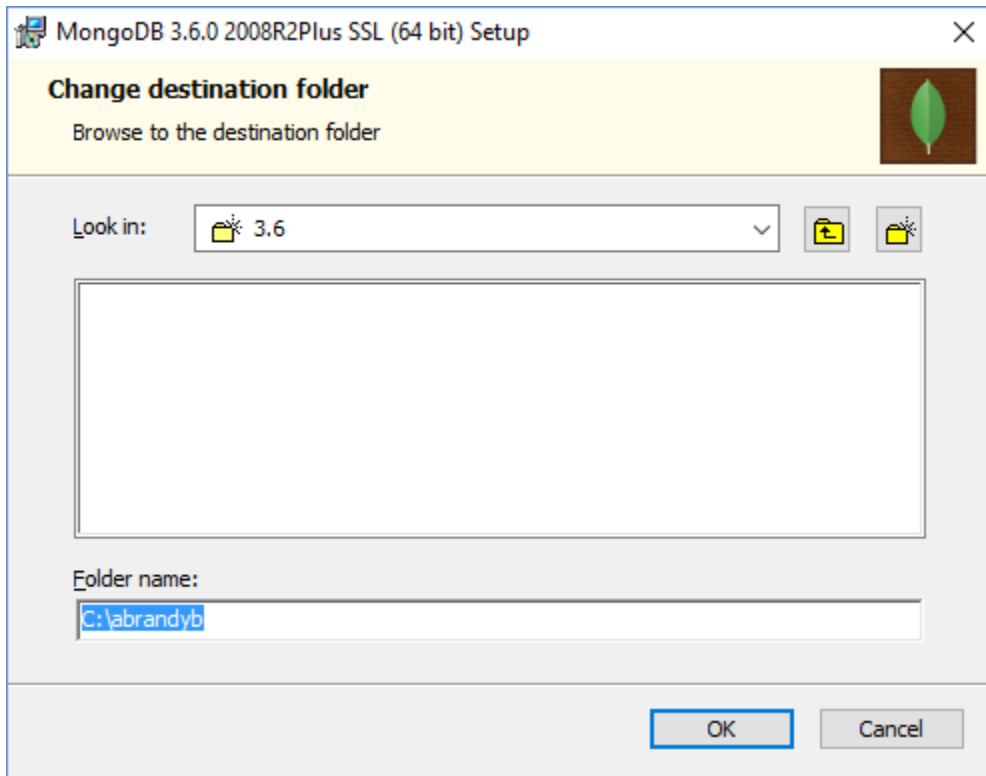


6. Select "Custom" Install (click on Custom Button).

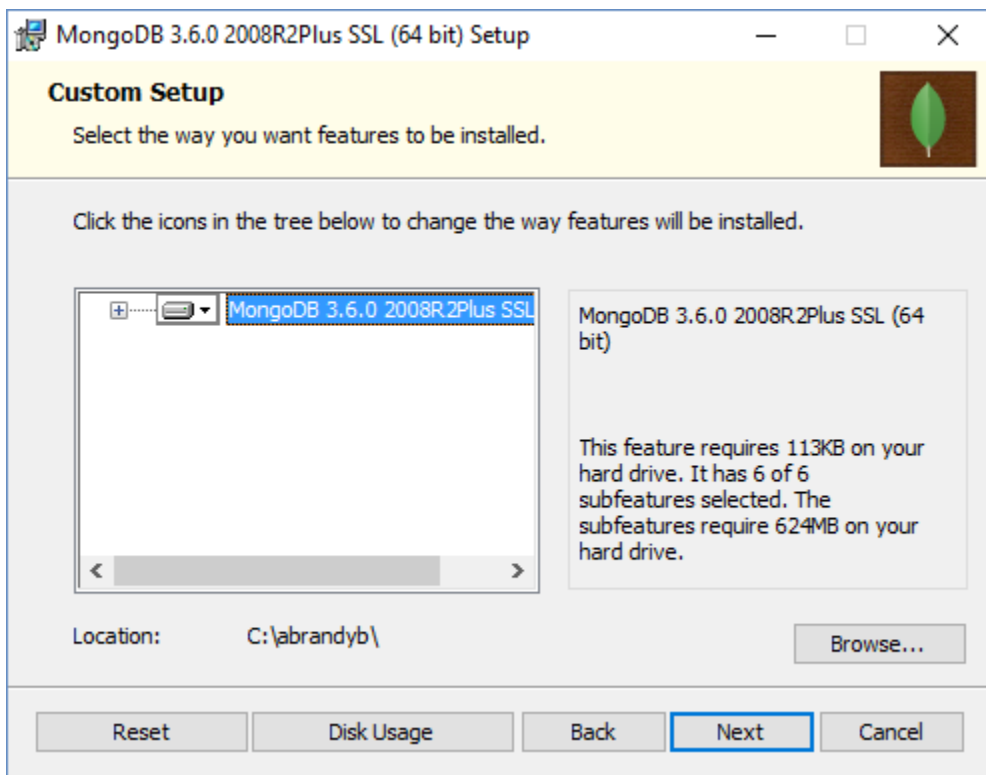


7. Here you will change the install directory. Read these directions closely, they differ slightly from the video. Click Browse above and enter your email username (the part before "@kent.edu"). When you finish it should look like the screen below...except it should be your username, not mine!

IMPORTANT! MUST DO to receive credit: Use your email username as the directory!

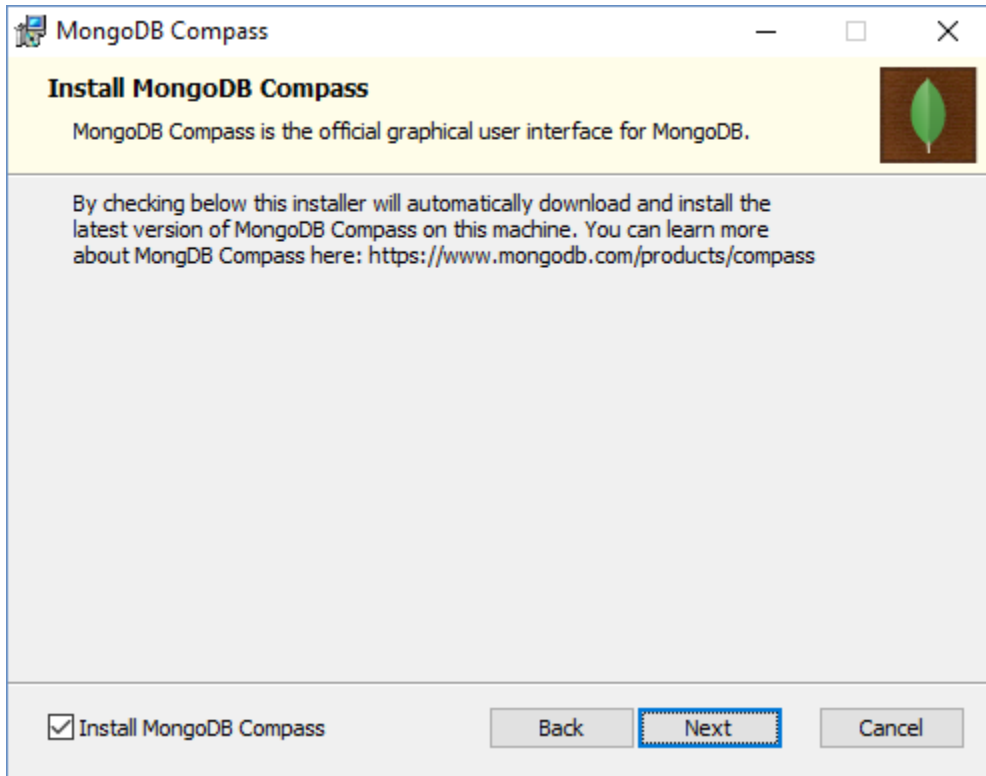


8. Confirm the folder name is as above except with your username and click OK. Note that whenever the video uses c:\mongodb, you should substitute this directory name for 'mongodb'.

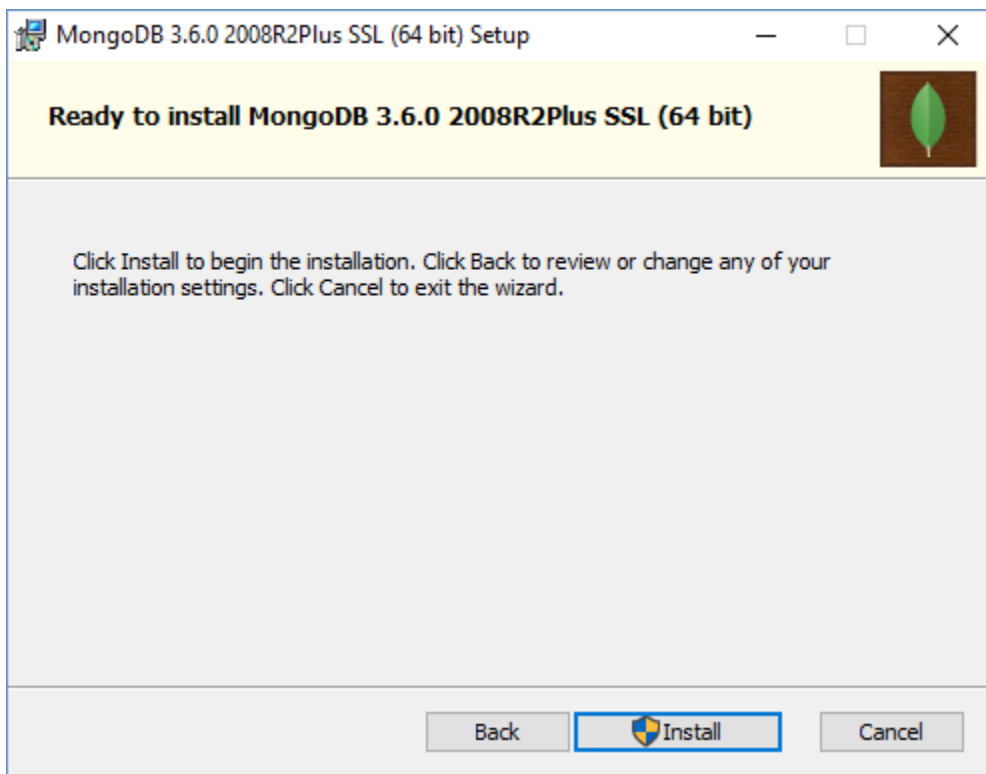


9. Click Next.

This Document is NOT designed for economical printing. It should be used digitally only for reasons of sustainability.
PLEASE DO NOT PRINT!

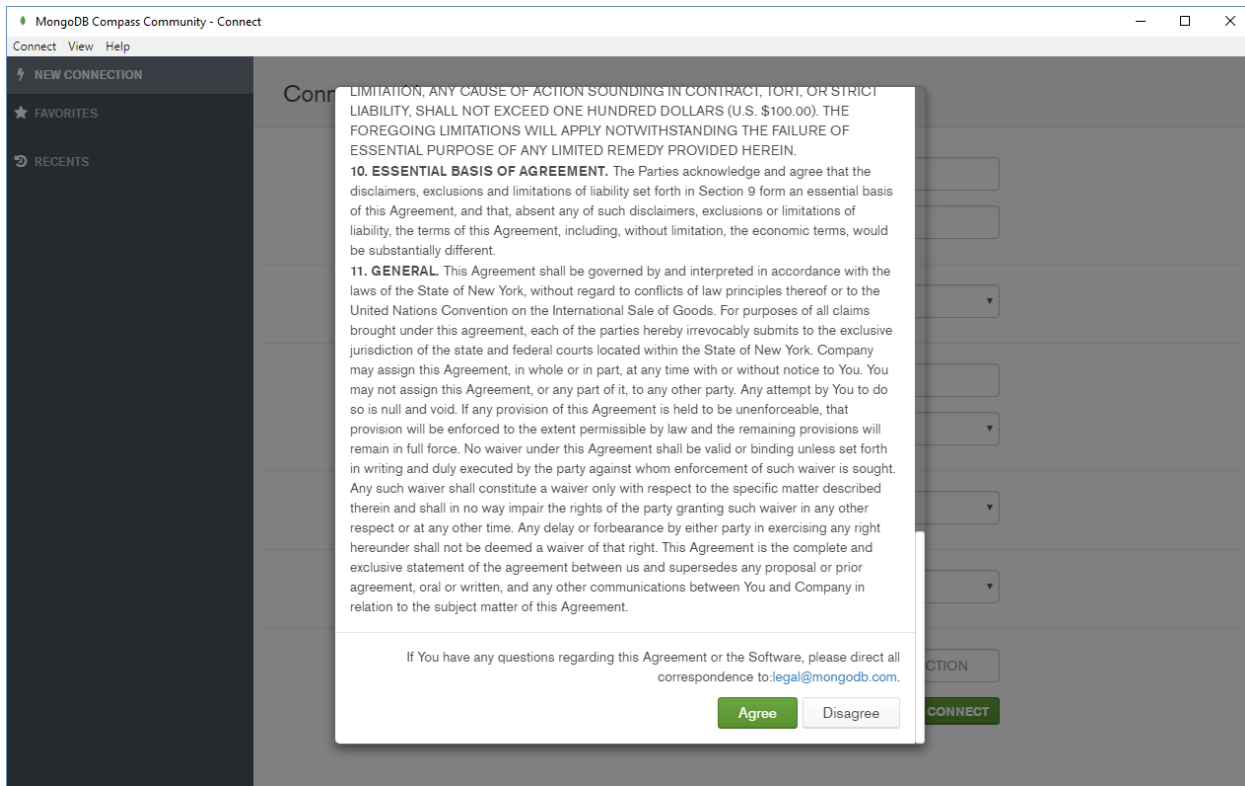


10. I suggest you install Compass so you can check it out later. Compass is an official GUI interface tool. We will not be using compass in this project practicum. If you plan to uninstall immediately after completing this – don't bother to install Compass.

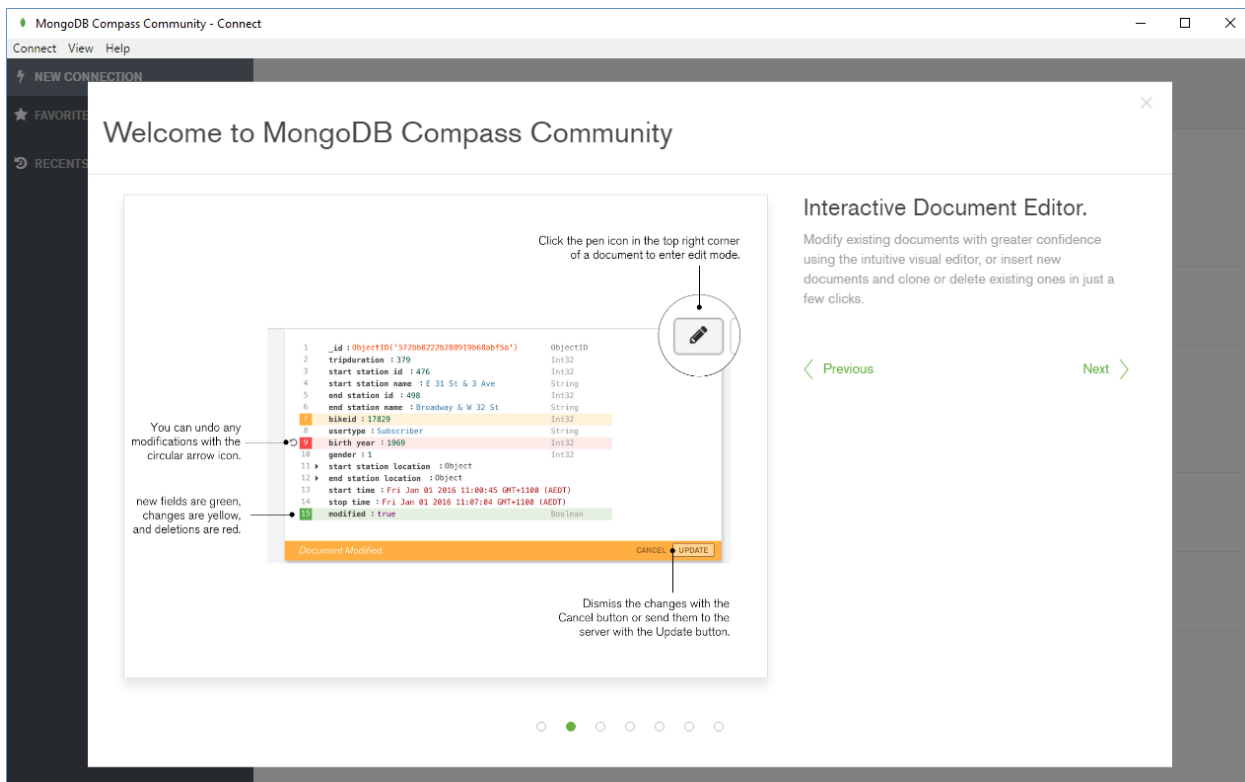


11. Click Install.

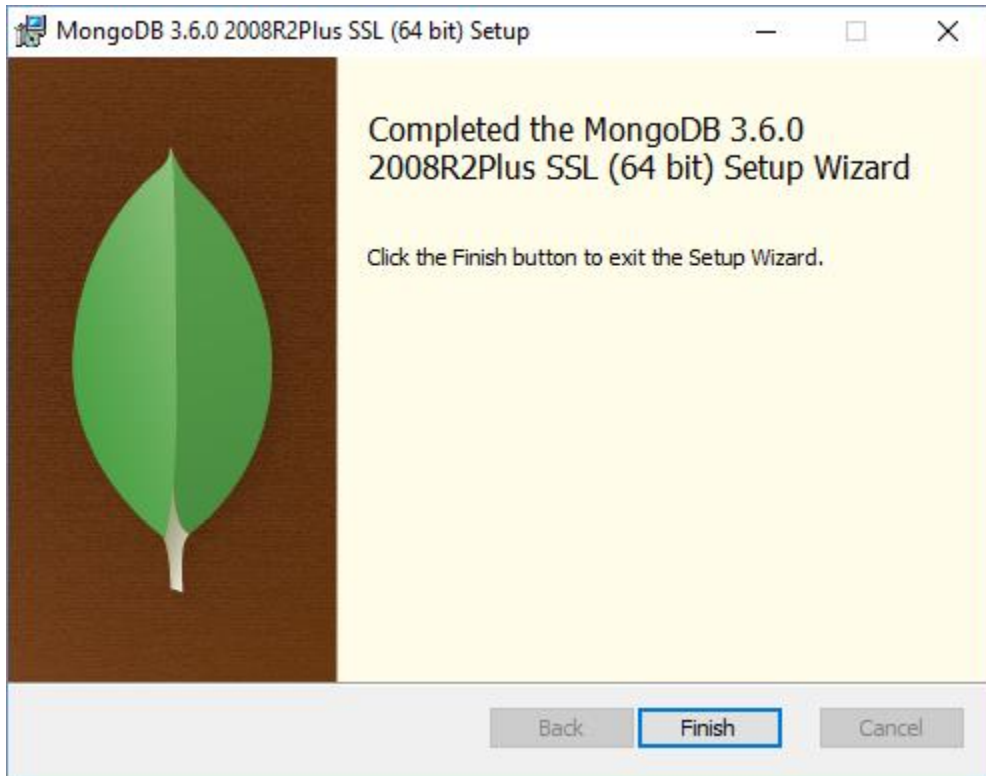
This Document is NOT designed for economical printing. It should be used digitally only for reasons of sustainability.
PLEASE DO NOT PRINT!



12. Read and click agree.



13. Just close this window or leave it in the background. We will not use Compass here.



14. Now that it is properly installed, begin following the directions from the video at time 4:00. You should make the directories as described. Remember that your home directory is your email username... not MongoDB. Every time he uses the directory MongoDB you should use your ID instead.



15. Enter 'cmd' into Windows Search then right click on the Command app it retrieves and click 'Run as Administrator'. If you do not run as administrator, you will get "Access Denied" errors.

Beginning at this point you should screenshot everything done in the command prompt window. It should be similar to the following but it does not need to use the same separation points between screenshots. Additionally, if you encounter an error, make sure to fix the problem and rerun but you do not have to remove the errors from the screen shots. Finally, keep the screen shots in the order of the video!

Deliverables (2 documents only):

1. Screen shots of all actions performed from the command prompt including the install and net start commands etc. before starting the mongo shell. Do NOT submit each screen shot as an individual file! Place them in a document as I have done here and save as a .pdf. 60/100 points.
2. A 500 word essay (minimum words = 400, maximum words = 600). 40/100 points.

```
Administrator: Command Prompt - mongo
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd c:\

c:\>cd abrandyb

c:\abrandyb>cd bin

c:\abrandyb\bin>mongod --directoryperdb --dbpath c:\abrandyb\data\db --logpath c:\abrandyb\log\mongo.log --logapp
end --rest --install
Error parsing command line: unrecognised option '--rest'
try 'mongod --help' for more information

c:\abrandyb\bin>mongod --directoryperdb --dbpath c:\abrandyb\data\db --logpath c:\abrandyb\log\mongo.log --logapp
end --install

c:\abrandyb\bin>net start mongod
The MongoDB service is starting.
The MongoDB service was started successfully.

c:\abrandyb\bin>mongo
MongoDB shell version v3.6.0
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 3.6.0
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
  http://docs.mongodb.org/
Questions? Try the support group
  http://groups.google.com/group/mongodb-user
Server has startup warnings:
2017-12-08T15:47:51.343-0500 I CONTROL [initandlisten]
2017-12-08T15:47:51.343-0500 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the databas
e.
2017-12-08T15:47:51.343-0500 I CONTROL [initandlisten] **      Read and write access to data and configurati
on is unrestricted.
2017-12-08T15:47:51.343-0500 I CONTROL [initandlisten]
2017-12-08T15:47:51.343-0500 I CONTROL [initandlisten] ** WARNING: This server is bound to localhost.
2017-12-08T15:47:51.343-0500 I CONTROL [initandlisten] **      Remote systems will be unable to connect to t
his server.
2017-12-08T15:47:51.343-0500 I CONTROL [initandlisten] **      Start the server with --bind_ip <address> to
specify which IP
2017-12-08T15:47:51.344-0500 I CONTROL [initandlisten] **      addresses it should serve responses from, or
with --bind_ip_all to
2017-12-08T15:47:51.344-0500 I CONTROL [initandlisten] **      bind to all interfaces. If this behavior is d
esired, start the
2017-12-08T15:47:51.344-0500 I CONTROL [initandlisten] **      server with --bind_ip 127.0.0.1 to disable th
is warning.
2017-12-08T15:47:51.344-0500 I CONTROL [initandlisten]
2017-12-08T15:47:51.344-0500 I CONTROL [initandlisten]
2017-12-08T15:47:51.344-0500 I CONTROL [initandlisten] ** WARNING: The file system cache of this machine is conf
igured to be greater than 40% of the total memory. This can lead to increased memory pressure and poor performanc
e.
2017-12-08T15:47:51.344-0500 I CONTROL [initandlisten] See http://dochub.mongodb.org/core/wt-windows-system-file
-cache
2017-12-08T15:47:51.344-0500 I CONTROL [initandlisten]
2017-12-08T15:48:42.188-0500 E - [main] Error loading history file: FileOpenFailed: Unable to fopen() file
C:\Users\AAB\.dbshell: The system cannot find the file specified.
> cls
```

This first attempt at the command failed. Need to drop “--rest” from the arguments. It isn’t recognized here and it is not needed here anyway. Just remove it before attempting.

Do not enter 'cls' (clear screen!) until you have screen captured this information.

Ignore all the warnings generated starting the mongo shell. None of it will interfere with this project and much of it is actually desired behavior (e.g. remote systems will be unable to connect – we want that – local only).

```
Administrator: Command Prompt - mongo
> show dbs
admin    0.000GB
config  0.000GB
local    0.000GB
> use mycustomers
switched to db mycustomers
> db
mycustomers
> db.createUser({
... user: "brad",
... pwd: "1234",
... roles: [ "readWrite", "dbAdmin" ]
... });
Successfully added user: { "user" : "brad", "roles" : [ "readWrite", "dbAdmin" ] }
>
>
> db.createCollection('customers');
{ "ok" : 1 }
> show collections
customers
> db.customers.insert({first_name:"John", lastname: "Doe"});
WriteResult({ "nInserted" : 1 })
> db.customers.find();
{ "_id" : ObjectId("5a2b01611301ad075fb3d3f0"), "first_name" : "John", "lastname" : "Doe" }
>
```

```
Administrator: Command Prompt - mongo
> db.customers.insert([{first_name:"Steven", lastname: "Smith"}, {first_name:"Joan",
lastname:"Johnson", gender:"female"}]);
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 2,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
> db.customers.find();
{ "_id" : ObjectId("5a2b01611301ad075fb3d3f0"), "first_name" : "John", "lastname" : "Doe" }
{ "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"), "first_name" : "Steven", "lastname" : "Smith" }
{ "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"), "first_name" : "Joan", "lastname" : "Johnson", "gender" : "female" }
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "lastname" : "Doe"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"),
  "first_name" : "Steven",
  "lastname" : "Smith"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
```

I inserted the original data with the wrong attribute name. These should use "last_name" instead of "lastname". It made no meaningful difference until the sort operations. Only one of these records still existed by that time and was corrected with:

```
db.customers.update({first_name:"Joan"},
{$rename:{"lastname":"last_name"}});
```

```
Administrator: Command Prompt - mongo
> db.customers.update({first_name:"John"},{first_name:"John", last_name:"Doe", gender:"male"});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"),
  "first_name" : "Steven",
  "lastname" : "Smith"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
> db.customers.update({first_name:"Steven"},{$set:{gender:"male"}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"),
  "first_name" : "Steven",
  "lastname" : "Smith",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
> ■
```

```
Administrator: Command Prompt - mongo
> db.customers.update({first_name:"Steven"},{$set:{age:45}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"),
  "first_name" : "Steven",
  "lastname" : "Smith",
  "gender" : "male",
  "age" : 45
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
> db.customers.update({first_name:"Steven"},{$inc:{age:5}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"),
  "first_name" : "Steven",
  "lastname" : "Smith",
  "gender" : "male",
  "age" : 50
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
> █
```

```
Administrator: Command Prompt - mongo
> db.customers.update({first_name:"Steven"},{$unset:age});
2017-12-10T13:00:23.778-0500 E QUERY [thread1] ReferenceError: age is not defined
:
@(shell):1:44
> db.customers.update({first_name:"Steven"},{$unset:{age:1}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"),
  "first_name" : "Steven",
  "lastname" : "Smith",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
> db.customers.update({first_name:"Mary"},{first_name:"Mary", last_name:"Sampson"});
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"),
  "first_name" : "Steven",
  "lastname" : "Smith",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
> █
```

```
Administrator: Command Prompt - mongo
> db.customers.update({first_name:"Mary"},{first_name:"Mary", last_name:"Sampson"},{upsert: true});
WriteResult({
  "nMatched" : 0,
  "nUpserted" : 1,
  "nModified" : 0,
  "_id" : ObjectId("5a2d8dfe96c0412b6746e596")
})
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"),
  "first_name" : "Steven",
  "lastname" : "Smith",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
{
  "_id" : ObjectId("5a2d8dfe96c0412b6746e596"),
  "first_name" : "Mary",
  "last_name" : "Sampson"
}
> db.customers.update({first_name:"Steven"},{$rename:{"gender":"sex"}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f1"),
  "first_name" : "Steven",
  "lastname" : "Smith",
  "sex" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
{
  "_id" : ObjectId("5a2d8dfe96c0412b6746e596"),
  "first_name" : "Mary",
  "last_name" : "Sampson"
}
> ■
```



```
Administrator: Command Prompt - mongo
> db.customers.remove({first_name:"Steven"});
WriteResult({ "nRemoved" : 1 })
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "last_name" : "Johnson",
  "gender" : "female"
}
{
  "_id" : ObjectId("5a2d8dfe96c0412b6746e596"),
  "first_name" : "Mary",
  "last_name" : "Sampson"
}
```

Data file used in the video – cut and paste from here:

```
db.customers.insert([
  {
    first_name:"Troy",
    last_name:"Makons",
    gender:"male",
    age:33,
    address:{
      street:"432 Essex St",
      city: "Lawrence",
      state:"MA"
    },
    memberships:["mem1", "mem2"],
    balance:125.32
  },
  {
    first_name:"Beth",
    last_name:"Jenkins",
    gender:"female",
    age:23,
    address:{
      street:"411 Blue St",
      city: "Boston",
      state:"MA"
    },
    memberships:["mem2", "mem3"],
    balance:505.33
  },
  {
    first_name:"Timothy",
    last_name:"Wilkins",
    gender:"male",
    age:53,
    address:{
      street:"22 School St",
      city: "Amesbury",
      state:"MA"
    },
    memberships:["mem3", "mem4"],
    balance:22.25
  },
  {
    first_name:"William",
    last_name:"Jackson",
    gender:"male",
    age:43,
    address:{
      street:"11 Albany St",
      city: "Boston",
      state:"MA"
    },
    memberships:["mem1"],
    balance:333.23
  },
  {
    first_name:"Sharon",
    last_name:"Thompson",
    gender:"female",
    age:35,
    address:{
      street:"19 Willis St",
      city: "Worcester",
      state:"MA"
    },
    memberships:["mem1", "mem2"],
    balance:99.99
  }
]);
```

Administrator: Command Prompt - mongo

```
> db.customers.insert([
... {
...   first_name:"Troy",
...   last_name:"Makons",
...   gender:"male",
...   age:33,
...   address:{
...     street:"432 Essex St",
...     city: "Lawrence",
...     state:"MA"
...   },
...   memberships:["mem1", "mem2"],
...   balance:125.32
... },
... {
...   first_name:"Beth",
...   last_name:"Jenkins",
...   gender:"female",
...   age:23,
...   address:{
...     street:"411 Blue St",
...     city: "Boston",
...     state:"MA"
...   },
...   memberships:["mem2", "mem3"],
...   balance:505.33
... },
... {
...   first_name:"Timothy",
...   last_name:"Wilkins",
...   gender:"male",
...   age:53,
...   address:{
...     street:"22 School St",
...     city: "Amesbury",
...     state:"MA"
...   },
...   memberships:["mem3", "mem4"],
...   balance:22.25
... },
... {
...   first_name:"William",
...   last_name:"Jackson",
...   gender:"male",
...   age:43,
...   address:{
...     street:"11 Albany St",
...     city: "Boston",
...     state:"MA"
...   },
...   memberships:["mem1"],
...   balance:333.23
... },
... ])
```

```
... {
...   first_name:"Sharon",
...   last_name:"Thompson",
...   gender:"female",
...   age:35,
...   address:{
...     street:"19 Willis St",
...     city: "Worcester",
...     state:"MA"
...   },
...   memberships:["mem1", "mem2"],
...   balance:99.99
... }
... });
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 5,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
> ■
```

```
Administrator: Command Prompt - mongo
> db.customers.find().pretty();
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "lastname" : "Johnson",
  "gender" : "female"
}
{
  "_id" : ObjectId("5a2d8dfe96c0412b6746e596"),
  "first_name" : "Mary",
  "last_name" : "Sampson"
}
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"),
  "first_name" : "Troy",
  "last_name" : "Makons",
  "gender" : "male",
  "age" : 33,
  "address" : {
    "street" : "432 Essex St",
    "city" : "Lawrence",
    "state" : "MA"
  },
  "memberships" : [
    "mem1",
    "mem2"
  ],
  "balance" : 125.32
}
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f4"),
  "first_name" : "Beth",
  "last_name" : "Jenkins",
  "gender" : "female",
  "age" : 23,
  "address" : {
    "street" : "411 Blue St",
    "city" : "Boston",
    "state" : "MA"
  },
  "memberships" : [
    "mem2",
    "mem3"
  ],
  "balance" : 505.33
}
```

```
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f5"),
  "first_name" : "Timothy",
  "last_name" : "Wilkins",
  "gender" : "male",
  "age" : 53,
  "address" : {
    "street" : "22 School St",
    "city" : "Amesbury",
    "state" : "MA"
  },
  "memberships" : [
    "mem3",
    "mem4"
  ],
  "balance" : 22.25
},
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f6"),
  "first_name" : "William",
  "last_name" : "Jackson",
  "gender" : "male",
  "age" : 43,
  "address" : {
    "street" : "11 Albany St",
    "city" : "Boston",
    "state" : "MA"
  },
  "memberships" : [
    "mem1"
  ],
  "balance" : 333.23
},
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f7"),
  "first_name" : "Sharon",
  "last_name" : "Thompson",
  "gender" : "female",
  "age" : 35,
  "address" : {
    "street" : "19 Willis St",
    "city" : "Worcester",
    "state" : "MA"
  },
  "memberships" : [
    "mem1",
    "mem2"
  ],
  "balance" : 99.99
}
> ■
```

```
Administrator: Command Prompt - mongo
> db.customers.find({first_name:"Sharon"});
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f7"), "first_name" : "Sharon", "last_name" : "Thompson", "gender" : "female", "age" : 35, "address" : { "street" : "19 Willis St", "city" : "Worcester", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 99.99 }
> db.customers.find({$or:[{first_name:"Sharon"},{first_name:"Troy"}]});
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"), "first_name" : "Troy", "last_name" : "Makons", "gender" : "male", "age" : 33, "address" : { "street" : "432 Essex St", "city" : "Lawrence", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 125.32 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f7"), "first_name" : "Sharon", "last_name" : "Thompson", "gender" : "female", "age" : 35, "address" : { "street" : "19 Willis St", "city" : "Worcester", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 99.99 }
> db.customers.find({gender:"male"});
{ "_id" : ObjectId("5a2b01611301ad075fb3d3f0"), "first_name" : "John", "last_name" : "Doe", "gender" : "male" }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"), "first_name" : "Troy", "last_name" : "Makons", "gender" : "male", "age" : 33, "address" : { "street" : "432 Essex St", "city" : "Lawrence", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 125.32 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f5"), "first_name" : "Timothy", "last_name" : "Wilkins", "gender" : "male", "age" : 53, "address" : { "street" : "22 School St", "city" : "Amesbury", "state" : "MA" }, "memberships" : [ "mem3", "mem4" ], "balance" : 22.25 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f6"), "first_name" : "William", "last_name" : "Jackson", "gender" : "male", "age" : 43, "address" : { "street" : "11 Albany St", "city" : "Boston", "state" : "MA" }, "memberships" : [ "mem1" ], "balance" : 333.23 }
> db.customers.find({age:{$lt:40}});
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"), "first_name" : "Troy", "last_name" : "Makons", "gender" : "male", "age" : 33, "address" : { "street" : "432 Essex St", "city" : "Lawrence", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 125.32 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f4"), "first_name" : "Beth", "last_name" : "Jenkins", "gender" : "female", "age" : 23, "address" : { "street" : "411 Blue St", "city" : "Boston", "state" : "MA" }, "memberships" : [ "mem2", "mem3" ], "balance" : 505.33 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f7"), "first_name" : "Sharon", "last_name" : "Thompson", "gender" : "female", "age" : 35, "address" : { "street" : "19 Willis St", "city" : "Worcester", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 99.99 }
>
```

```
Administrator: Command Prompt - mongo
> db.customers.find({age:{$lt:40}}).pretty();
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"),
  "first_name" : "Troy",
  "last_name" : "Makons",
  "gender" : "male",
  "age" : 33,
  "address" : {
    "street" : "432 Essex St",
    "city" : "Lawrence",
    "state" : "MA"
  },
  "memberships" : [
    "mem1",
    "mem2"
  ],
  "balance" : 125.32
},
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f4"),
  "first_name" : "Beth",
  "last_name" : "Jenkins",
  "gender" : "female",
  "age" : 23,
  "address" : {
    "street" : "411 Blue St",
    "city" : "Boston",
    "state" : "MA"
  },
  "memberships" : [
    "mem2",
    "mem3"
  ],
  "balance" : 505.33
},
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f7"),
  "first_name" : "Sharon",
  "last_name" : "Thompson",
  "gender" : "female",
  "age" : 35,
  "address" : {
    "street" : "19 Willis St",
    "city" : "Worcester",
    "state" : "MA"
  },
  "memberships" : [
    "mem1",
    "mem2"
  ],
  "balance" : 99.99
}
> █
```



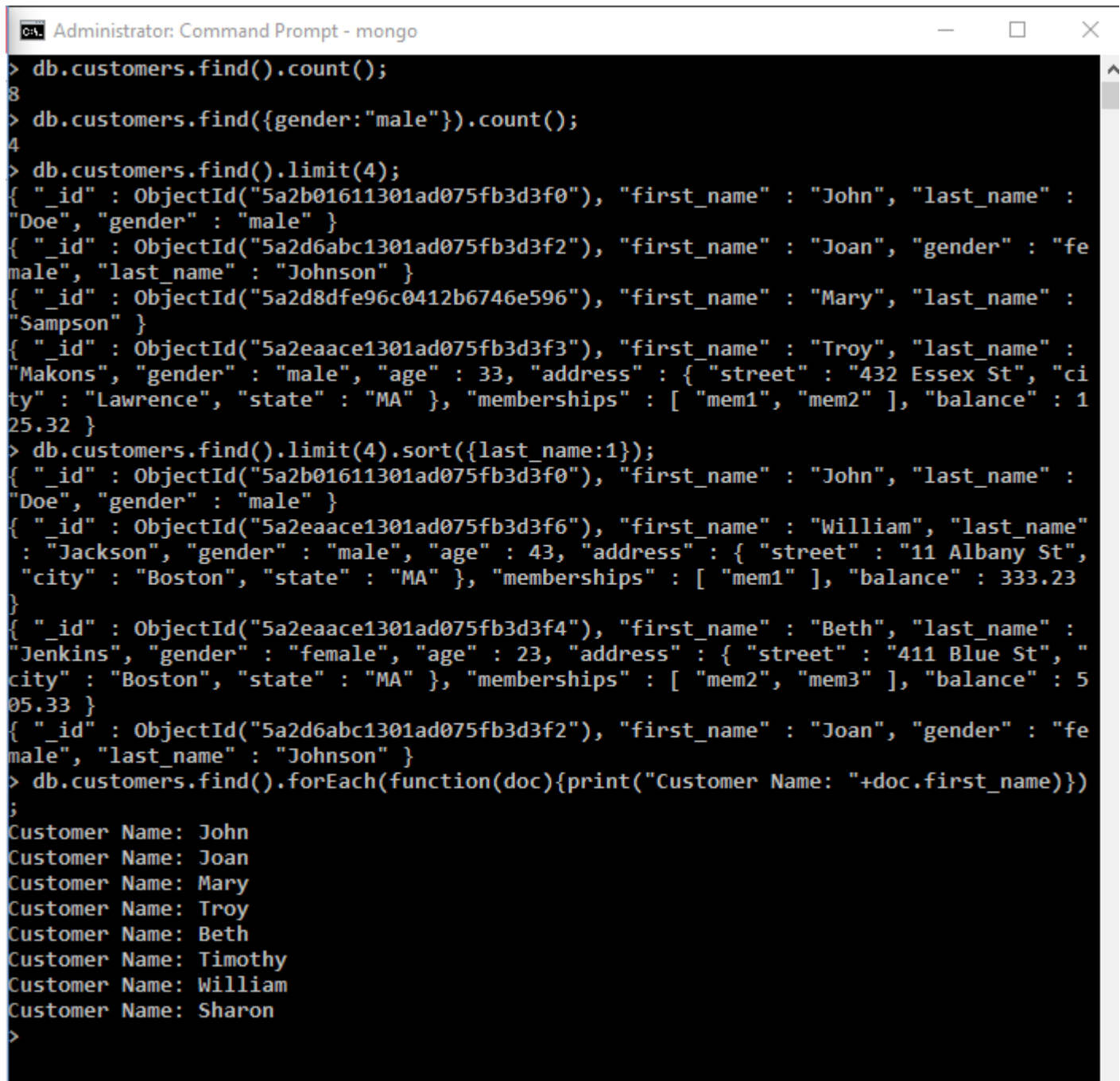
```
CA Select Administrator: Command Prompt - mongo
> db.customers.find({"address.city":"Boston"});
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f4"), "first_name" : "Beth", "last_name" :
"Jenkins", "gender" : "female", "age" : 23, "address" : { "street" : "411 Blue St", "
city" : "Boston", "state" : "MA" }, "memberships" : [ "mem2", "mem3" ], "balance" : 5
05.33 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f6"), "first_name" : "William", "last_name"
: "Jackson", "gender" : "male", "age" : 43, "address" : { "street" : "11 Albany St",
"city" : "Boston", "state" : "MA" }, "memberships" : [ "mem1" ], "balance" : 333.23
}
> db.customers.find({address.city:"Boston"});
2017-12-11T11:40:57.298-0500 E QUERY [thread1] SyntaxError: missing : after proper
ty id @(shell):1:26
>
> db.customers.find({memberships:"mem1"});
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"), "first_name" : "Troy", "last_name" :
"Makons", "gender" : "male", "age" : 33, "address" : { "street" : "432 Essex St", "ci
ty" : "Lawrence", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 1
25.32 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f6"), "first_name" : "William", "last_name"
: "Jackson", "gender" : "male", "age" : 43, "address" : { "street" : "11 Albany St",
"city" : "Boston", "state" : "MA" }, "memberships" : [ "mem1" ], "balance" : 333.23
}
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f7"), "first_name" : "Sharon", "last_name"
: "Thompson", "gender" : "female", "age" : 35, "address" : { "street" : "19 Willis St
", "city" : "Worcester", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "bala
nce" : 99.99 }
>
```

Administrator: Command Prompt - mongo

```
> db.customers.find().sort({last_name:1});
{ "_id" : ObjectId("5a2b01611301ad075fb3d3f0"), "first_name" : "John", "last_name" :
"Doe", "gender" : "male" }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f6"), "first_name" : "William", "last_name"
: "Jackson", "gender" : "male", "age" : 43, "address" : { "street" : "11 Albany St",
"city" : "Boston", "state" : "MA" }, "memberships" : [ "mem1" ], "balance" : 333.23
}
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f4"), "first_name" : "Beth", "last_name" :
"Jenkins", "gender" : "female", "age" : 23, "address" : { "street" : "411 Blue St", "
city" : "Boston", "state" : "MA" }, "memberships" : [ "mem2", "mem3" ], "balance" : 5
05.33 }
{ "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"), "first_name" : "Joan", "gender" : "fe
male", "last_name" : "Johnson" }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"), "first_name" : "Troy", "last_name" :
"Makons", "gender" : "male", "age" : 33, "address" : { "street" : "432 Essex St", "ci
ty" : "Lawrence", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 1
25.32 }
{ "_id" : ObjectId("5a2d8dfe96c0412b6746e596"), "first_name" : "Mary", "last_name" :
"Sampson" }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f7"), "first_name" : "Sharon", "last_name"
: "Thompson", "gender" : "female", "age" : 35, "address" : { "street" : "19 Willis St
", "city" : "Worcester", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "bala
nce" : 99.99 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f5"), "first_name" : "Timothy", "last_name"
: "Wilkins", "gender" : "male", "age" : 53, "address" : { "street" : "22 School St",
"city" : "Amesbury", "state" : "MA" }, "memberships" : [ "mem3", "mem4" ], "balance"
: 22.25 }
> db.customers.find().sort({last_name:-1});
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f5"), "first_name" : "Timothy", "last_name"
: "Wilkins", "gender" : "male", "age" : 53, "address" : { "street" : "22 School St",
"city" : "Amesbury", "state" : "MA" }, "memberships" : [ "mem3", "mem4" ], "balance"
: 22.25 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f7"), "first_name" : "Sharon", "last_name"
: "Thompson", "gender" : "female", "age" : 35, "address" : { "street" : "19 Willis St
", "city" : "Worcester", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "bala
nce" : 99.99 }
{ "_id" : ObjectId("5a2d8dfe96c0412b6746e596"), "first_name" : "Mary", "last_name" :
"Sampson" }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"), "first_name" : "Troy", "last_name" :
"Makons", "gender" : "male", "age" : 33, "address" : { "street" : "432 Essex St", "ci
ty" : "Lawrence", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 1
25.32 }
{ "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"), "first_name" : "Joan", "gender" : "fe
male", "last_name" : "Johnson" }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f4"), "first_name" : "Beth", "last_name" :
"Jenkins", "gender" : "female", "age" : 23, "address" : { "street" : "411 Blue St", "
city" : "Boston", "state" : "MA" }, "memberships" : [ "mem2", "mem3" ], "balance" : 5
05.33 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f6"), "first_name" : "William", "last_name"
: "Jackson", "gender" : "male", "age" : 43, "address" : { "street" : "11 Albany St",
"city" : "Boston", "state" : "MA" }, "memberships" : [ "mem1" ], "balance" : 333.23
}
{ "_id" : ObjectId("5a2b01611301ad075fb3d3f0"), "first_name" : "John", "last_name" :
"Doe", "gender" : "male" }
> ■
```

```
Administrator: Command Prompt - mongo
> db.customers.find().sort({last_name:-1}).pretty();
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f5"),
  "first_name" : "Timothy",
  "last_name" : "Wilkins",
  "gender" : "male",
  "age" : 53,
  "address" : {
    "street" : "22 School St",
    "city" : "Amesbury",
    "state" : "MA"
  },
  "memberships" : [
    "mem3",
    "mem4"
  ],
  "balance" : 22.25
},
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f7"),
  "first_name" : "Sharon",
  "last_name" : "Thompson",
  "gender" : "female",
  "age" : 35,
  "address" : {
    "street" : "19 Willis St",
    "city" : "Worcester",
    "state" : "MA"
  },
  "memberships" : [
    "mem1",
    "mem2"
  ],
  "balance" : 99.99
},
{
  "_id" : ObjectId("5a2d8dfe96c0412b6746e596"),
  "first_name" : "Mary",
  "last_name" : "Sampson"
},
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"),
  "first_name" : "Troy",
  "last_name" : "Makons",
  "gender" : "male",
  "age" : 33,
  "address" : {
    "street" : "432 Essex St",
    "city" : "Lawrence",
    "state" : "MA"
  },
  "memberships" : [
    "mem1",
    "mem2"
  ],
  "balance" : 125.32
}
```

```
Administrator: Command Prompt - mongo
{
  "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"),
  "first_name" : "Joan",
  "gender" : "female",
  "last_name" : "Johnson"
}
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f4"),
  "first_name" : "Beth",
  "last_name" : "Jenkins",
  "gender" : "female",
  "age" : 23,
  "address" : {
    "street" : "411 Blue St",
    "city" : "Boston",
    "state" : "MA"
  },
  "memberships" : [
    "mem2",
    "mem3"
  ],
  "balance" : 505.33
}
{
  "_id" : ObjectId("5a2eaace1301ad075fb3d3f6"),
  "first_name" : "William",
  "last_name" : "Jackson",
  "gender" : "male",
  "age" : 43,
  "address" : {
    "street" : "11 Albany St",
    "city" : "Boston",
    "state" : "MA"
  },
  "memberships" : [
    "mem1"
  ],
  "balance" : 333.23
}
{
  "_id" : ObjectId("5a2b01611301ad075fb3d3f0"),
  "first_name" : "John",
  "last_name" : "Doe",
  "gender" : "male"
}
> ■
```



```
> db.customers.find().count();
8
> db.customers.find({gender:"male"}).count();
4
> db.customers.find().limit(4);
{ "_id" : ObjectId("5a2b01611301ad075fb3d3f0"), "first_name" : "John", "last_name" : "Doe", "gender" : "male" }
{ "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"), "first_name" : "Joan", "gender" : "female", "last_name" : "Johnson" }
{ "_id" : ObjectId("5a2d8dfe96c0412b6746e596"), "first_name" : "Mary", "last_name" : "Sampson" }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f3"), "first_name" : "Troy", "last_name" : "Makons", "gender" : "male", "age" : 33, "address" : { "street" : "432 Essex St", "city" : "Lawrence", "state" : "MA" }, "memberships" : [ "mem1", "mem2" ], "balance" : 125.32 }
> db.customers.find().limit(4).sort({last_name:1});
{ "_id" : ObjectId("5a2b01611301ad075fb3d3f0"), "first_name" : "John", "last_name" : "Doe", "gender" : "male" }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f6"), "first_name" : "William", "last_name" : "Jackson", "gender" : "male", "age" : 43, "address" : { "street" : "11 Albany St", "city" : "Boston", "state" : "MA" }, "memberships" : [ "mem1" ], "balance" : 333.23 }
{ "_id" : ObjectId("5a2eaace1301ad075fb3d3f4"), "first_name" : "Beth", "last_name" : "Jenkins", "gender" : "female", "age" : 23, "address" : { "street" : "411 Blue St", "city" : "Boston", "state" : "MA" }, "memberships" : [ "mem2", "mem3" ], "balance" : 505.33 }
{ "_id" : ObjectId("5a2d6abc1301ad075fb3d3f2"), "first_name" : "Joan", "gender" : "female", "last_name" : "Johnson" }
> db.customers.find().forEach(function(doc){print("Customer Name: "+doc.first_name)})
;
Customer Name: John
Customer Name: Joan
Customer Name: Mary
Customer Name: Troy
Customer Name: Beth
Customer Name: Timothy
Customer Name: William
Customer Name: Sharon
>
```

FINISHED!

Deliverables (2 documents only):

3. Screen shots of all actions performed from the command prompt including the install and net start commands etc. before starting the mongo shell. Do NOT submit each screen shot as an individual file! Place them in a document as I have done here and save as a .pdf. 60/100 points.
4. A 500 word essay (minimum words = 400, maximum words = 600). 40/100 points.

Due: Upload documents to Blackboard no later than midnight, the end of day, on Friday, December 15.