

# Chapter 1: Introducing MongoDB

The screenshot shows the MongoDB download interface. At the top, there are four tabs: Atlas, Community Server (highlighted in green), Enterprise Server, and Ops Manager. Below the tabs, the main content area displays the following information:

- Current Stable Release (4.0.0)**
- 06/21/2018: [Release Notes](#) | [Changelog](#)
- Download Source: [tgz](#) | [zip](#)
- Version:** A dropdown menu is set to "Windows 64-bit x64".
- Installation Package:** A green button with a download icon and the text "DOWNLOAD (msi)".

On the right side of the page, there is a Windows logo icon and the text "Windows".

MongoDB 4.0.0 2008R2Plus SSL (64 bit) Service Custo... - □ ×

### Service Configuration

Specify optional settings to configure MongoDB as a service.

Install MongoDB as a Service

Run service as Network Service user

Run service as a local or domain user:

Account Domain:

Account Name:

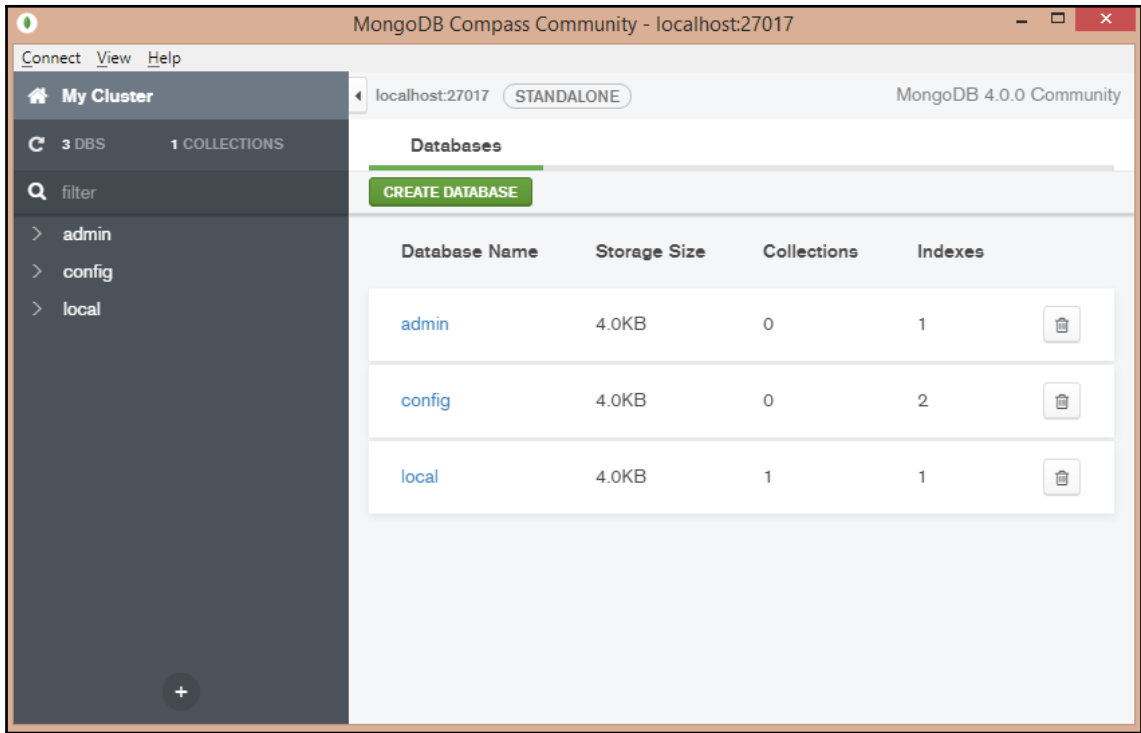
Account Password:

Service Name:

Data Directory:

Log Directory:

< Back   **Next >**   Cancel



```
MINGW64:/c/Users/george
george@asus-windoze MINGW64 ~
$ ls -l /c/Program\ Files/MongoDB/Server/4.0/
total 132
drwxr-xr-x 1 george 197121 0 Jul 5 15:56 bin/
drwxr-xr-x 1 george 197121 0 Jul 5 16:01 data/
-rw-r--r-- 1 george 197121 34520 Jun 21 20:53 GNU-AGPL-3.0
-rw-r--r-- 1 george 197121 2149 Jun 21 20:53 LICENSE-Community.txt
drwxr-xr-x 1 george 197121 0 Jul 5 15:59 log/
-rw-r--r-- 1 george 197121 16726 Jun 21 20:53 MPL-2
-rw-r--r-- 1 george 197121 2195 Jun 21 20:53 README
-rw-r--r-- 1 george 197121 57190 Jun 21 20:53 THIRD-PARTY-NOTICES
george@asus-windoze MINGW64 ~
$
```

```
mongod.cfg - C:\Program Files\MongoDB\Server\4.0\bin - Geany
File Edit Search View Document Project Build Tools Help
mongod.cfg
1 # mongod.conf
2
3 # for documentation of all options, see:
4 # http://docs.mongodb.org/manual/reference/configuration-options/
5
6 # Where and how to store data.
7 storage:
8   dbPath: C:\Program Files\MongoDB\Server\4.0\data
9   journal:
10     enabled: true
11 # engine:
12 # mmapv1:
13 # wiredTiger:
14
15 # where to write logging data.
16 systemLog:
17   destination: file
18   logAppend: true
19   path: C:\Program Files\MongoDB\Server\4.0\log\mongod.log
20
21 # network interfaces
22 net:
23   port: 27017
24   bindIp: 127.0.0.1
```

```
ed@ed: ~
File Edit View Search Terminal Help
ed@ed:~$ sudo apt-key adv \
> --keyserver hkp://keyserver.ubuntu.com:80 \
> --recv 9DA31620334BD75D9DCB49F368818C72E52529D4
[sudo] password for ed:
Executing: /tmp/apt-key-gpghome.cyD3ZMdcj/gpg.1.sh --keyserver hkp://keyserver.ubuntu.com:80
--recv 9DA31620334BD75D9DCB49F368818C72E52529D4
gpg: key 68818C72E52529D4: public key "MongoDB 4.0 Release Signing Key <packaging@mongodb.com
>" imported
gpg: Total number processed: 1
gpg: imported: 1
ed@ed:~$
```

```
ed@ed: ~
File Edit View Search Terminal Help
ed@ed:~$ echo \
> "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/4.0 multiverse" \
> | sudo tee /etc/apt/sources.list.d/mongodb-org-4.0.list
deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/4.0 multiverse
ed@ed:~$
```

```
Open ▾ /etc mongod.conf [Read-Only] Save ≡
# mongod.conf

# for documentation of all options, see:
# http://docs.mongodb.org/manual/reference/configuration-options/

# Where and how to store data.
storage:
  dbPath: /var/lib/mongodb
  journal:
    enabled: true
# engine:
# mmapv1:
# wiredTiger:

# where to write logging data.
systemLog:
  destination: file
  logAppend: true
  path: /var/log/mongodb/mongod.log

# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1

Plain Text ▾ Tab Width: 8 ▾ Ln 18, Col 18 ▾ INS
```

```
ed@ed: ~
File Edit View Search Terminal Help
ed@ed:~$ sudo service mongod start
ed@ed:~$ sudo service mongod status
● mongod.service - MongoDB Database Server
   Loaded: loaded (/lib/systemd/system/mongod.service; disabled; vendor preset: enabled)
   Active: active (running) since Thu 2018-07-05 13:50:31 BST; 4s ago
     Docs: https://docs.mongodb.org/manual
   Main PID: 19500 (mongod)
   CGroup: /system.slice/mongod.service
           └─19500 /usr/bin/mongod --config /etc/mongod.conf

Jul 05 13:50:31 ed systemd[1]: Started MongoDB Database Server.
Jul 05 13:50:33 ed mongod[19500]: 2018-07-05T13:50:33.829+0100 I CONTROL [main] Automatically disabling TLS 1.0, to force-enable T
```

```
Open [icon] mongodb-org-4.0.repo [Read-Only] Save [icon] - □ ×
/etc/yum.repos.d

[mongodb-org-4.0]
name=MongoDB Repository
baseurl=https://repo.mongodb.org/yum/redhat/$releasever/mongodb-org/4.0/x86_64/
gpgcheck=1
enabled=1
gpgkey=https://www.mongodb.org/static/pgp/server-4.0.asc

Plain Text ▾ Tab Width: 8 ▾ Ln 1, Col 1 ▾ INS
```

```
ned@ned:/home/ned
File Edit View Search Terminal Help

[root@ned ned]# sudo yum install -y mongodb-org
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: centos.ustc.edu.cn
 * extras: mirror.kku.ac.th
 * updates: mirror.kku.ac.th
base | 3.6 kB 00:00
extras | 3.4 kB 00:00
mongodb-org-4.0 | 2.4 kB 00:00
updates | 3.4 kB 00:00
(1/3): extras/7/x86_64/primary_db | 150 kB 00:00
(2/3): updates/7/x86_64/primary_db | 3.6 MB 00:00
(3/3): mongodb-org-4.0/7/primary_db | 8.7 kB 00:00
Resolving Dependencies
--> Running transaction check
---> Package mongodb-org.x86_64 0:4.0.0-1.el7 will be installed
--> Processing Dependency: mongodb-org-tools = 4.0.0 for package: mongodb-org-4.0.0-1.el7.x86_64
```

```
Open [icon] *mongod.conf [Read-Only] Save [icon] - [icon] x
/etc
# mongod.conf

# where to write logging data.
systemLog:
  destination: file
  logAppend: true
  path: /var/log/mongodb/mongod.log

# Where and how to store data.
storage:
  dbPath: /var/lib/mongo
  journal:
    enabled: true

# how the process runs
processManagement:
  fork: true # fork and run in background
  pidFilePath: /var/run/mongodb/mongod.pid # location of pidfile
  timeZoneInfo: /usr/share/zoneinfo

# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1 # Enter 0.0.0.0,:: to bind to all IPv4 and IPv6 addresses or,
alternatively, use the net.bindIpAll setting.

#security:

Plain Text Tab Width: 8 Ln 10, Col 9 INS
```

```
ned@ned:/home/ned
File Edit View Search Terminal Help
[root@ned ned]# /bin/systemctl start mongod.service
[root@ned ned]# /bin/systemctl status mongod.service
● mongod.service - MongoDB Database Server
   Loaded: loaded (/usr/lib/systemd/system/mongod.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2018-07-05 14:19:22 BST; 6s ago
     Docs: https://docs.mongodb.org/manual
    Process: 4475 ExecStart=/usr/bin/mongod $OPTIONS (code=exited, status=0/SUCCESS)
    Process: 4472 ExecStartPre=/usr/bin/chmod 0755 /var/run/mongodb (code=exited, status=0/SUCCESS)
    Process: 4470 ExecStartPre=/usr/bin/chown mongod:mongod /var/run/mongodb (code=exited, status=0/SUCCESS)
    Process: 4468 ExecStartPre=/usr/bin/mkdir -p /var/run/mongodb (code=exited, status=0/SUCCESS)
   Main PID: 4479 (mongod)
      Tasks: 26
     CGroup: /system.slice/mongod.service
             └─4479 /usr/bin/mongod -f /etc/mongod.conf

Jul 05 14:19:20 ned systemd[1]: Starting MongoDB Database Server...
```

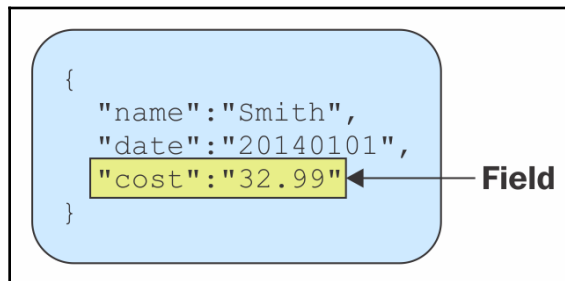
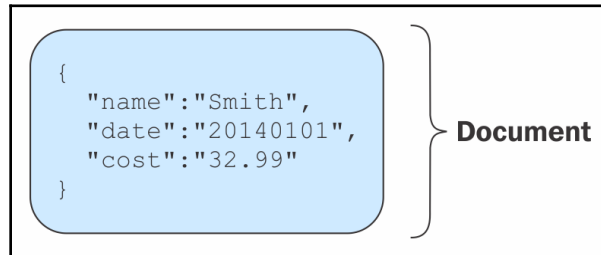
```
[root@ned ned]# git clone https://github.com/mongodb/mongo.git
Cloning into 'mongo'...
remote: Counting objects: 622827, done.
remote: Compressing objects: 100% (466/466), done.
remote: Total 622827 (delta 254), reused 245 (delta 134), pack-reused 622225
Receiving objects: 100% (622827/622827), 398.97 MiB | 10.54 MiB/s, done.
Resolving deltas: 100% (460128/460128), done.
Checking out files: 100% (19634/19634), done.
```

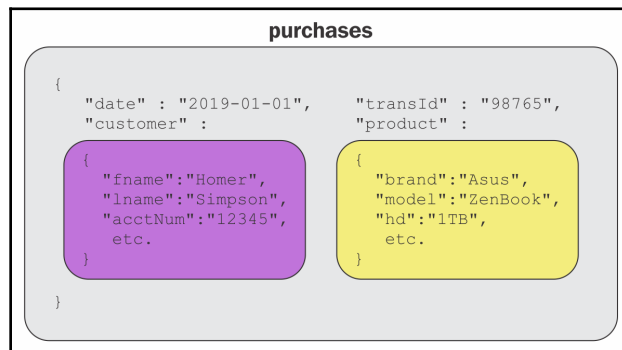
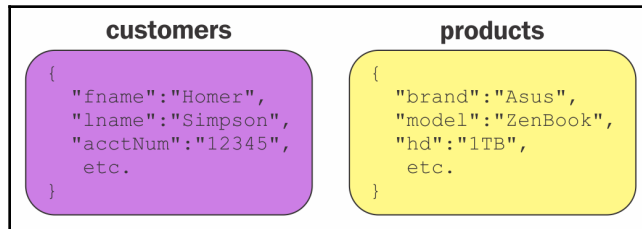
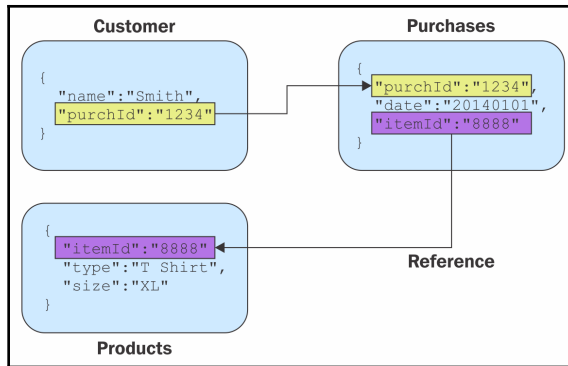
```
[root@ned mongo]# python -mpip install --user -r buildscripts/requirements.txt
Ignoring mypy: markers u'python_version > "3"' don't match your environment
Ignoring pypiwin32: markers u'sys_platform == "win32" and python_version < "3"' don't match your environment
Ignoring pypiwin32: markers u'sys_platform == "win32" and python_version > "3"' don't match your environment
Collecting cryptography==1.7.2 (from -r buildscripts/requirements.txt (line 2))
  Downloading https://files.pythonhosted.org/packages/99/df/71c7260003f5c469cec3db4c547115df39e9ce6c719a99e0
0e78fd8a/cryptography-1.7.2.tar.gz (420kB)
    100% |██████████████████████████████████████████████████████████████████████████████| 430kB 1.7MB/s
Collecting jira==1.0.10 (from -r buildscripts/requirements.txt (line 3))
  Downloading https://files.pythonhosted.org/packages/5e/6e/96a299deee19be84d1f6317f71dd86e736b4b3e660e7633c
```

```
[root@ned mongo]# buildscripts/scons.py all
scons: Reading SConscript files ...
scons version: 2.5.0
python version: 2 7 5 'final' 0
Checking whether the C compiler works... yes
Checking whether the C++ compiler works... yes
Checking that the C++ compiler can link a C++ program... yes
Checking if C++ compiler "g++" is GCC... yes
Checking if C compiler "gcc" is GCC... yes
Detected a x86_64 processor
Checking if target OS linux is supported by the toolchain... yes
```



# Chapter 2: Understanding MongoDB Data Structures





<input type="checkbox"/>	Field name	Type	Allow nulls?	Key	Default value	Extras
<input type="checkbox"/>	id	int(8)	No	Primary	NULL	auto_increment
<input type="checkbox"/>	type	varchar(64)	Yes	None	NULL	

<input type="checkbox"/>	Field name	Type	Allow nulls?	Key	Default value	Extras
<input type="checkbox"/>	id	int(8)	No	Primary	NULL	auto_increment
<input type="checkbox"/>	category	varchar(64)	Yes	None	NULL	

```

_id: ObjectId("5b2336da57b3f5706643ad8a")
type: "Social Media"
options: Array
  0: "Facebook"
  1: "Twitter"
  2: "Instagram"

_id: ObjectId("5b23382b57b3f5706643ad8b")
type: "Categories"
options: Array
  0: "Shirts"
  1: "Blouses"
  2: "Trousers"
  3: "etc."

```

My Cluster localhost:27017 STANDALONE MongoDB 4.0.2 Community

3 DBS 1 COLLECTIONS

filter

- admin
- config
- local

+ CREATE DATABASE

### Databases

CREATE DATABASE

Database Name	Storage Size	Collections	Indexes	
admin	4.0KB	0	1	
config	4.0KB	0	2	
local	4.0KB	1	1	

### Create Database

Database Name

Collection Name

Capped Collection ⓘ

Before MongoDB can save your new database, a collection name must also be specified at the time of creation. [More Information](#)

CANCEL CREATE DATABASE

The screenshot shows the MongoDB Compass interface. On the left, a sidebar lists databases: 'admin', 'config', 'local', 'sweetscomplete', and 'customers'. The 'customers' database is selected. The main area displays the 'Collections' tab for the 'customers' database. A table lists the collection 'customers' with 0 documents, an average document size of '-', a total document size of '0.0 B', 1 index, and a total index size of '4.0 KB'. A 'CREATE COLLECTION' button is visible at the top of the table.

Collection Name ^	Documents	Avg. Document Size	Total Document Size	Num. Indexes	Total Index Size
customers	0	-	0.0 B	1	4.0 KB

## Chapter 3: Using the MongoDB Shell

```
fred@fred-linux:~$ ls -l .mongorc.js
-rw----- 1 fred fred 42 Jun 26 11:03 .mongorc.js
fred@fred-linux:~$ ls -l /etc/mongorc.js
-rwxr-xr-x 1 mongodb root 42 Jun 26 11:04 /etc/mongorc.js
fred@fred-linux:~$
```

```
.mongorc.js ✕
host = db.serverStatus().host;
upTime = db.serverStatus().uptime;
prompt = function() {
  return db+"@"+host+" [up:"+upTime+"> ";
}
```

```

> help
db.help()          help on db methods
db.mycoll.help()  help on collection methods
sh.help()         sharding helpers
rs.help()        replica set helpers
help admin       administrative help
help connect     connecting to a db help
help keys        key shortcuts
help misc        misc things to know
help mr          mapreduce

show dbs         show database names
show collections show collections in current database
show users      show users in current database
show profile    show most recent system.profile entries with time >= 1ms
show logs       show the accessible logger names
show log [name] prints out the last segment of log in memory, 'global' is default
use <db_name>  set current database
db.foo.find()   list objects in collection foo
db.foo.find( { a : 1 } ) list objects in foo where a == 1
it              result of the last line evaluated; use to further iterate
DBQuery.shellBatchSize = x set default number of items to display on shell
exit           quit the mongo shell

```

```

sweetscomplete@fred-linux[up:174546]> db.products.find();
{ "_id" : ObjectId("5b34651536adf7141a398f73"), "sku" : "F1000", "title" : "Fudge", "description" : "Invenire percipitur eum ea, in saepe per", "price" : "0.10" }
{ "_id" : ObjectId("5b34651536adf7141a398f74"), "sku" : "S2000", "title" : "Sugar Cookies", "description" : "Et vix patrioque similique referrentur, ", "price" : "0.20" }
{ "_id" : ObjectId("5b34651536adf7141a398f75"), "sku" : "C3000", "title" : "Chocolate Angelfood Cupcakes", "description" : "Id ius detracto constituan, his possit p", "price" : "0.30" }
{ "_id" : ObjectId("5b34651536adf7141a398f76"), "sku" : "P4000", "title" : "Peanut Brittle", "description" : "Aliquam maestatis mea eu. Vel quot rebu", "price" : "0.40" }
{ "_id" : ObjectId("5b34651536adf7141a398f77"), "sku" : "T5000", "title" : "Toasted Marshmallows", "description" : "Ne mea velit tation euismod. Nec iudico ", "price" : "0.50" }
{ "_id" : ObjectId("5b34651536adf7141a398f78"), "sku" : "F6000", "title" : "Fruit Salad", "description" : "Vis quem voluptatum in. Id magna inintcu", "price" : "0.60" }
{ "_id" : ObjectId("5b34651536adf7141a398f79"), "sku" : "C7000", "title" : "Cheesecake", "description" : "Munere iriture tanquam no quo, sit et tam", "price" : "0.70" }
{ "_id" : ObjectId("5b34651536adf7141a398f7a"), "sku" : "G8000", "title" : "Glazed Doughnut", "description" : "Mel wist decore habemus ad, et his debit", "price" : "0.80" }
{ "_id" : ObjectId("5b34651536adf7141a398f7b"), "sku" : "F9000", "title" : "Fortune Cookies", "description" : "Oratio pericula sapientem te eum. Mei tr", "price" : "0.90" }
{ "_id" : ObjectId("5b34651536adf7141a398f7c"), "sku" : "D10000", "title" : "Devils Food Cake", "description" : "Ius nulla phaedrui ex, aliquip ullamcorp", "price" : "1.00" }
{ "_id" : ObjectId("5b34651536adf7141a398f7d"), "sku" : "P11000", "title" : "Peanut Butter Cups", "description" : "Vin et patrioque adolescens. Ludus ubiqu", "price" : "1.10" }
{ "_id" : ObjectId("5b34651536adf7141a398f7e"), "sku" : "C12000", "title" : "Crispy Rice Treats", "description" : "Cu inani nonumy vituperatoribus nec, ut ", "price" : "1.20" }
{ "_id" : ObjectId("5b34651536adf7141a398f7f"), "sku" : "C13000", "title" : "Cherry Pie", "description" : "Putant volumus reformidans in pri. Per e", "price" : "1.30" }
{ "_id" : ObjectId("5b34651536adf7141a398f80"), "sku" : "A14000", "title" : "Apple Turnover", "description" : "Vel id sumo assum, possim consequuntur s", "price" : "1.40" }
{ "_id" : ObjectId("5b34651536adf7141a398f81"), "sku" : "N15000", "title" : "Napoleon", "description" : "Sed an agam nominati, eam ubique ornatus", "price" : "1.50" }
{ "_id" : ObjectId("5b34651536adf7141a398f82"), "sku" : "C16000", "title" : "Chocolate chip Cookies", "description" : "Vel sonet audire ea, no latine scriptore", "price" : "1.60" }
{ "_id" : ObjectId("5b34651536adf7141a398f83"), "sku" : "C17000", "title" : "Chocolate Soufflé", "description" : "Quo oratio aperiti officilis te. Vel elit", "price" : "1.70" }
{ "_id" : ObjectId("5b34651536adf7141a398f84"), "sku" : "W18000", "title" : "Walnut Brownies", "description" : "Sanctus occurreret eloquentiam ad qui, ut", "price" : "1.80" }
{ "_id" : ObjectId("5b34651536adf7141a398f85"), "sku" : "A19000", "title" : "Ambrosia Salad", "description" : "Oblique vivendo legendos an vel. Has ne ", "price" : "1.90" }
{ "_id" : ObjectId("5b34651536adf7141a398f86"), "sku" : "P20000", "title" : "Peach Cobbler", "description" : "Et consul nostrum copiosae duo, usu cibo", "price" : "2.00" }
Type "it" for more

```

```
sweetscomplete@fred-linux[up:256884]> db.products.find(
...   {
...     price:{$gt:"2",$lt:"8"},
...     title:/chocolate/i
...   },
...   {
...     _id:0,description:0
...   }
... ).sort({sku:1});
```

Diagram labels for the MongoDB command above:

- Collection**: db.products
- Query**: price:{\$gt:"2",\$lt:"8"}, title:/chocolate/i
- Projection**: \_id:0,description:0
- Cursor Modifier**: .sort({sku:1})

```
{ "sku" : "C21000", "title" : "Chocolate Eclair", "price" : "2.10" }
{ "sku" : "C22000", "title" : "Chocolate Toaster Tarts", "price" : "2.20" }
{ "sku" : "C30000", "title" : "Chocolate Mousse", "price" : "3.00" }
{ "sku" : "C32000", "title" : "Chocolate Fondue", "price" : "3.20" }
{ "sku" : "C61000", "title" : "Chocolate Layer Cake", "price" : "6.10" }
{ "sku" : "M43000", "title" : "Mint Chocolate Milk Shake", "price" : "4.30" }
sweetscomplete@fred-linux[up:256884]> █
```

```
sweetscomplete@fred-linux[up:451235]> db.products.find({title:/Cookies/},{sku:1,title:1,price:1});
{ "_id" : ObjectId("5b34651536adf7141a398f74"), "sku" : "S2000", "title" : "Sugar Cookies", "price" : "0.20" }
{ "_id" : ObjectId("5b34651536adf7141a398f7b"), "sku" : "F9000", "title" : "Fortune Cookies", "price" : "0.90" }
{ "_id" : ObjectId("5b34651536adf7141a398f82"), "sku" : "C16000", "title" : "Chocolate Chip Cookies", "price" : "1.60" }
{ "_id" : ObjectId("5b34653d36adf7141a398f8e"), "sku" : "P28000", "title" : "Peanut Butter Cookies", "price" : "2.80" }
{ "_id" : ObjectId("5b34653d36adf7141a398f8f"), "sku" : "O29000", "title" : "Oatmeal Raisin Cookies", "price" : "2.90" }
{ "_id" : ObjectId("5b34653d36adf7141a398f97"), "sku" : "G37000", "title" : "Gingerbread Cookies", "price" : "3.70" }
{ "_id" : ObjectId("5b34654a36adf7141a398fb0"), "sku" : "S62000", "title" : "Shortbread Cookies", "price" : "6.20" }
{ "_id" : ObjectId("5b34654a36adf7141a398fb1"), "sku" : "O63000", "title" : "Oreo Cookies", "price" : "6.30" }
sweetscomplete@fred-linux[up:451235]> █
```

```
sweetscomplete@fred-linux[up:539431]> db.products.find({title:/Cookies/},{_id:0,sku:1,title:1,price:1});
{ "sku" : "S2000", "title" : "Sugar Cookies", "price" : "0.20" }
{ "sku" : "F9000", "title" : "Fortune Cookies", "price" : "0.90" }
{ "sku" : "C16000", "title" : "Chocolate Chip Cookies", "price" : "1.60" }
{ "sku" : "P28000", "title" : "Peanut Butter Cookies", "price" : "2.80" }
{ "sku" : "O29000", "title" : "Oatmeal Raisin Cookies", "price" : "2.90" }
{ "sku" : "G37000", "title" : "Gingerbread Cookies", "price" : "3.70" }
{ "sku" : "S62000", "title" : "Shortbread Cookies", "price" : "6.20" }
{ "sku" : "O63000", "title" : "Oreo Cookies", "price" : "6.30" }
sweetscomplete@fred-linux[up:539431]>
```

```
sweetscomplete@fred-linux[up:539431]> db.products.find({title:/Cookies/},{_id:0,description:0});
{ "sku" : "S2000", "title" : "Sugar Cookies", "price" : "0.20" }
{ "sku" : "F9000", "title" : "Fortune Cookies", "price" : "0.90" }
{ "sku" : "C16000", "title" : "Chocolate Chip Cookies", "price" : "1.60" }
{ "sku" : "P28000", "title" : "Peanut Butter Cookies", "price" : "2.80" }
{ "sku" : "O29000", "title" : "Oatmeal Raisin Cookies", "price" : "2.90" }
{ "sku" : "G37000", "title" : "Gingerbread Cookies", "price" : "3.70" }
{ "sku" : "S62000", "title" : "Shortbread Cookies", "price" : "6.20" }
{ "sku" : "O63000", "title" : "Oreo Cookies", "price" : "6.30" }
sweetscomplete@fred-linux[up:539431]> █
```

### Ascending Sort

```
>db.products.find({}, {_id:0,description:0}).sort({price:1}).limit(5);
{ "sku" : "F1000", "title" : "Fudge", "price" : "0.10" }
{ "sku" : "S2000", "title" : "Sugar Cookies", "price" : "0.20" }
{ "sku" : "C3000", "title" : "Chocolate Angelfood Cupcakes", "price" : "0.30" }
{ "sku" : "P4000", "title" : "Peanut Brittle", "price" : "0.40" }
{ "sku" : "T5000", "title" : "Toasted Marshmallows", "price" : "0.50" }
>
```

### Descending Sort

```
>db.products.find({}, {_id:0,description:0}).sort({price:-1}).limit(5);
{ "sku" : "P64000", "title" : "Pop Tarts", "price" : "6.40" }
{ "sku" : "O63000", "title" : "Oreo Cookies", "price" : "6.30" }
{ "sku" : "S62000", "title" : "Shortbread Cookies", "price" : "6.20" }
{ "sku" : "C61000", "title" : "Chocolate Layer Cake", "price" : "6.10" }
{ "sku" : "B60000", "title" : "Banana Bread", "price" : "6.00" }
>
```

```
local@fred-linux[up:85273]> use test;
switched to db test
test@fred-linux[up:85273]> show dbs;
admin                0.000GB
config               0.000GB
local                0.000GB
mongodb_quickstart  0.000GB
sweetscomplete      0.000GB
test@fred-linux[up:85273]>
```

```
test@fred-linux[up:85273]> db.users.insertOne({ firstName: "Fred",lastName: "Flintstone",status: "Cave Man"});
{
  "acknowledged" : true,
  "insertedId" : ObjectId("5b330bc44a8f38b62dcff69d")
}
test@fred-linux[up:85273]> show dbs;
admin                0.000GB
config               0.000GB
local                0.000GB
mongodb_quickstart  0.000GB
sweetscomplete      0.000GB
test                 0.000GB
test@fred-linux[up:85273]>
```

```
test@fred-linux[up:85273]> use test;
switched to db test
test@fred-linux[up:85273]> db.dropDatabase();
{ "dropped" : "test", "ok" : 1 }
test@fred-linux[up:85273]> show dbs;
admin                0.000GB
config               0.000GB
local                0.000GB
mongodb_quickstart  0.000GB
sweetscomplete      0.000GB
test@fred-linux[up:85273]>
```



```
test@fred-linux[up:85273]> use test;
switched to db test
test@fred-linux[up:85273]> db.products.insertOne({ title: "Hammer",price: 99.99});
{
  "acknowledged" : true,
  "insertedId" : ObjectId("5b3311e64a8f38b62dcff6a2")
}
test@fred-linux[up:85273]> db.createCollection("users");
{ "ok" : 1 }
test@fred-linux[up:85273]> show collections;
products
users
test@fred-linux[up:85273]> █
```

```
test@fred-linux[up:85273]> show collections;
products
users
test@fred-linux[up:85273]> db.users.drop();
true
test@fred-linux[up:85273]> show collections;
products
test@fred-linux[up:85273]> █
```

```
>db.customers.insertOne(
... {
...   "name": "Conrad Perry",
...   "address": "79 Amber Branch Falls",
...   "city": "Birdseye",
...   "state_province": "QC",
...   "postal_code": "G0U 0M5",
...   "country": "CA",
...   "phone": "484-181-9811",
...   "balance": "745.32",
...   "email": "conrad.perry@fastmedia.com",
...   "password": "$2y$10$mSzJ0Y9uvWbkaYLPgG6uEe0MnbZf3LdC25cTkqgFT6ueWCnqW.Y1a"
... }
... );
{
  "acknowledged" : true,
  "insertedId" : ObjectId("5b39a2d31539fefbb18b04d7")
}
>█
```

```
>db.customers.insertMany(
... [
...   {"name":"Lonnie Knapp","email":"lonnie.knapp@cablecom.com"},
...   {"name":"Darrel Roman","email":"darrel.roman@southcom.net"},
...   {"name":"Morgan Avila","email":"morgan.avila@northmedia.com"},
... ]
... );
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("5b39a53b1539fefbb18b04d8"),
    ObjectId("5b39a53b1539fefbb18b04d9"),
    ObjectId("5b39a53b1539fefbb18b04da")
  ]
}
>
```

```
>db.customers.find().pretty();
{
  "_id" : ObjectId("5b39a2d31539fefbb18b04d7"),
  "name" : "Conrad Perry",
  "address" : "79 Amber Branch Falls",
  "city" : "Birdseye",
  "state_province" : "QC",
  "postal_code" : "G0U 0M5",
  "country" : "CA",
  "phone" : "484-181-9811",
  "balance" : "745.32",
  "email" : "conrad.perry@fastmedia.com",
  "password" : "$2y$10$mSzJOY9uvWbkaYLPgG6uEe0MnbZf3LdC25cTkqpFT6ueWCnqW.Y1a"
}
{
  "_id" : ObjectId("5b39a53b1539fefbb18b04d8"),
  "name" : "Lonnie Knapp",
  "email" : "lonnie.knapp@cablecom.com"
}
{
  "_id" : ObjectId("5b39a53b1539fefbb18b04d9"),
  "name" : "Darrel Roman",
  "email" : "darrel.roman@southcom.net"
}
{
  "_id" : ObjectId("5b39a53b1539fefbb18b04da"),
  "name" : "Morgan Avila",
  "email" : "morgan.avila@northmedia.com"
}
>
```

```

>db.customers.updateOne(
...   { name:"Conrad Perry" },
...   { $set:
...     {
...       "phone" : "111-222-3333",
...       "balance" : 544.88
...     }
...   }
... );
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
>
>db.customers.find({name:"Conrad Perry"},{name:1,phone:1,balance:1}).pretty();
{
  "_id" : ObjectId("5b39a2d31539fefbb18b04d7"),
  "name" : "Conrad Perry",
  "phone" : "111-222-3333",
  "balance" : 544.88
}
>

```

```

>db.customers.updateMany(
...   { name: { $in: ["Lonnie Knapp", "Darrel Roman", "Morgan Avila"] }},
...   { $set: { "balance" : 0.0 } }
... );
{ "acknowledged" : true, "matchedCount" : 3, "modifiedCount" : 3 }
>
>db.customers.find({}, {name:1,phone:1,balance:1}).pretty();
{
  "_id" : ObjectId("5b39a2d31539fefbb18b04d7"),
  "name" : "Conrad Perry",
  "phone" : "111-222-3333",
  "balance" : 544.88
}
{
  "_id" : ObjectId("5b39a53b1539fefbb18b04d8"),
  "name" : "Lonnie Knapp",
  "balance" : 0
}
{
  "_id" : ObjectId("5b39a53b1539fefbb18b04d9"),
  "name" : "Darrel Roman",
  "balance" : 0
}
{
  "_id" : ObjectId("5b39a53b1539fefbb18b04da"),
  "name" : "Morgan Avila",
  "balance" : 0
}
>

```

```
>db.customers.replaceOne(
...   { name: "Lee Mccray" },
...   {
...     "name" : "Lee Mccray",
...     "address" : "195 Grand Dale Acres",
...     "city" : "Cake",
...     "state_province" : "QC",
...     "postal_code" : "DD65 82QD",
...     "country" : "UK",
...     "phone" : "265-666-2636",
...     "balance" : 539.35,
...     "email" : "lee.mccray@southnet.com",
...     "password" : "new5220and"
...   },
...   { upsert: true }
... );
{
  "acknowledged" : true,
  "matchedCount" : 0,
  "modifiedCount" : 0,
  "upsertedId" : ObjectId("5b39dbfa631cd39f250ea2bf")
}
>
```

```
>db.customers.updateOne(
...   { name: "Lee Mccray" },
...   { $unset: { "password" : 1 } }
... );
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
>
```

```
>db.customers.find({ name: "Lee Mccray" }).pretty();
{
  "_id" : ObjectId("5b39dbfa631cd39f250ea2bf"),
  "name" : "Lee Mccray",
  "address" : "195 Grand Dale Acres",
  "city" : "Cake",
  "state_province" : "QC",
  "postal_code" : "DD65 82QD",
  "country" : "UK",
  "phone" : "265-666-2636",
  "balance" : 539.35,
  "email" : "lee.mccray@southnet.com"
}
>
```

```
>db.customers.deleteOne({"name":"C.T. Russell"});
{ "acknowledged" : true, "deletedCount" : 1 }
>
```

```
>db.customers.find({password:{$exists:0}},{name:1,email:1});
{ "_id" : ObjectId("5b39a53b1539fefbb18b04d8"), "name" : "Lonnie Knapp", "email" : "lonnie.knapp@cablecom.com" }
{ "_id" : ObjectId("5b39a53b1539fefbb18b04d9"), "name" : "Darrel Roman", "email" : "darrel.roman@southcom.net" }
{ "_id" : ObjectId("5b39a53b1539fefbb18b04da"), "name" : "Morgan Avila", "email" : "morgan.avila@northmedia.com" }
{ "_id" : ObjectId("5b39dbfa631cd39f250ea2bf"), "name" : "Lee Mccray", "email" : "lee.mccray@southnet.com" }
{ "_id" : ObjectId("5b39efbf1bd9ed476f570286"), "name" : "Karyn Francis", "email" : "karyn.francis@fastnet.com" }
{ "_id" : ObjectId("5b39efbf1bd9ed476f570287"), "name" : "Blanca Le", "email" : "blanca.le@telecom.com" }
{ "_id" : ObjectId("5b39efbf1bd9ed476f570288"), "name" : "Renee Decker", "email" : "renee.decker@westcom.net" }
{ "_id" : ObjectId("5b39efbf1bd9ed476f570289"), "name" : "Obama", "email" : "obama@president.gov" }
{ "_id" : ObjectId("5b39efbf1bd9ed476f57028a"), "name" : "C.T. Russell", "email" : "ctrussell@jw.org" }
>
```

```
>db.customers.deleteMany({ "password" : { $exists : false } });
{ "acknowledged" : true, "deletedCount" : 9 }
>
```

```
fred@fred-linux:~$ mongo --eval "db.serverStatus();"
MongoDB shell version v4.0.0
connecting to: mongod://127.0.0.1:27017
MongoDB server version: 4.0.0
{
  "host" : "fred-linux",
  "version" : "4.0.0",
  "process" : "mongod",
  "pid" : NumberLong(15054),
  "uptime" : 828,
  "uptimeMillis" : NumberLong(828147),
  "uptimeEstimate" : NumberLong(828),
  "localTime" : ISODate("2018-07-04T03:58:28.386Z"),
```

```

fred@fred-linux:~$ mongo sweetscomplete --eval "db.customers.find
> (
>   {balance:0},
>   {name:1,email:1,balance:1}
> ).pretty();"
MongoDB shell version v4.0.0
connecting to: mongod://127.0.0.1:27017/sweetscomplete
MongoDB server version: 4.0.0
{
  "_id" : ObjectId("5b3c4734f5a48120315c7167"),
  "name" : "Lonnie Knapp",
  "balance" : 0,
  "email" : "lonnie.knapp@cablecom.com"
}
{
  "_id" : ObjectId("5b3c4734f5a48120315c7168"),
  "name" : "Darrel Roman",
  "balance" : 0,
  "email" : "darrel.roman@southcom.net"
}
{
  "_id" : ObjectId("5b3c4734f5a48120315c7169"),
  "name" : "Morgan Avila",
  "balance" : 0,
  "email" : "morgan.avila@northmedia.com"
}
fred@fred-linux:~$ █

```

```

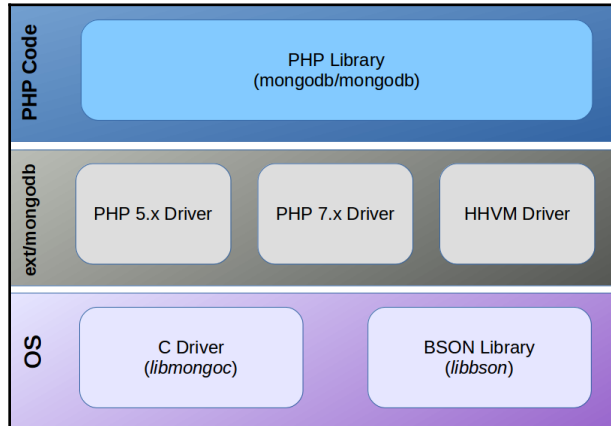
insert_three.js ✖
conn = new Mongo();
db = conn.getDB("sweetscomplete");
db.customers.insertOne({"name":"Karyn Francis","address":"871 Rocky Autumn Mews","city":"Gassaway",
  "state_province":"","postal_code":"RF21 26MI","country":"UK","phone":"385-836-7870",
  "balance":919.76,"email":"karyn.francis@fastnet.com","password":"the9642He"});
db.customers.updateOne({"email":"karyn.francis@fastnet.com"},{$unset:{"password":1}});
db.customers.insertOne({"name":"Blanca Le","address":"179 Noble Pine Place","city":"Diagonal",
  "state_province":"NS","postal_code":"B6R 2T3","country":"CA","phone":"185-787-5938",
  "balance":833.32,"email":"blanca.le@telecom.com","password":"and6425said"});
db.customers.updateOne({"email":"blanca.le@telecom.com"},{$unset:{"password":1}});
db.customers.insertOne({"name":"Renee Decker","address":"42 Robbers Way","city":"Nome",
  "state_province":"ACT","postal_code":"2900","country":"AU","phone":"660-333-4444",
  "balance":447.83,"email":"renee.decker@westcom.net","password":"Stephens6135that"});
db.customers.updateOne({"email":"renee.decker@westcom.net"},{$unset:{"password":1}});
db.customers.insertOne({"name":"C.T. Russell","address":"123 Main Street","city":"New York",
  "state_province":"NY","postal_code":"10001","country":"US","phone":"555-1212",
  "balance":0,"email":"ctrussell@jw.org","password":"password"});
db.customers.updateOne({"email":"ctrussell@jw.org"},{$unset:{"password":1}});

```

```
fred@fred-linux:~$ mongo --shell Desktop/Repos/MongoDB_Quick_Start/Source/insert_three.js
MongoDB shell version v4.0.0
connecting to: mongod://127.0.0.1:27017
MongoDB server version: 4.0.0
type "help" for help

sweetscomplete@fred-linux[up:1822]> db.customers.find({name:/Karyn|Blanca|Renee|C.T./},{name:1});
{ "_id" : ObjectId("5b3c49c67e8b700b9754d95b"), "name" : "Karyn Francis" }
{ "_id" : ObjectId("5b3c49c67e8b700b9754d95c"), "name" : "Blanca Le" }
{ "_id" : ObjectId("5b3c49c67e8b700b9754d95d"), "name" : "Renee Decker" }
{ "_id" : ObjectId("5b3c49c67e8b700b9754d95e"), "name" : "C.T. Russell" }
sweetscomplete@fred-linux[up:1822]> █
```

# Chapter 4: Developing with Program Language Drivers



```
Terminal
Spencer Sanford : PV9 74XN : UK : spencer.sanford@cablenet.net
Jose Carter : HH34 64HU : UK : jose.carter@westcom.net
Wilfredo Taylor : MF6 8WM : UK : wilfredo.taylor@telecom.net
Todd Lindsey : FW38 8RB : UK : todd.lindsey@fastnet.net
admin : 99999 : UK : admin@sweetscomplete.com

-----
(program exited with code: 0)
Press return to continue
```



```
Terminal
Processing Flag for Cuba
Processing Flag for Cyprus
Processing Flag for Czech Republic
Inserted 16 documents
object(MongoDB\Driver\WriteResult)#4 (9) {
  ["nInserted"]=>
  int(16)
  ["nMatched"]=>
  int(0)
  ["nModified"]=>
  int(0)
  ["nRemoved"]=>
  int(0)
  ["nUpserted"]=>
  int(0)
  ["upsertedIds"]=>
  array(0) {
  }
  ["writeErrors"]=>
  array(0) {
  }
  ["writeConcernError"]=>
  NULL
  ["writeConcern"]=>
  object(MongoDB\Driver\WriteConcern)#5 (0) {
  }
}

-----
(program exited with code: 0)
Press return to continue
```

```
fred@fred-linux: ~  
fred@fred-linux:~$ sudo pecl install mongodb  
[sudo] password for fred:  
WARNING: channel "pecl.php.net" has updated its protocols, use "pecl channel-update pecl.php.net"  
to update  
downloading mongodb-1.5.0.tgz ...  
Starting to download mongodb-1.5.0.tgz (1,054,840 bytes)  
....done: 1,054,840 bytes  
423 source files, building  
running: phpize  
Configuring for:  
PHP Api Version:          20160303  
Zend Module Api No:      20160303  
Zend Extension Api No:   320160303  
building in /tmp/pear/temp/pear-build-root1Uuvoi/mongodb-1.5.0  
running: /tmp/pear/temp/mongodb/configure --with-php-config=/usr/local/bin/php-config  
checking for grep that handles long lines and -e... /bin/grep  
checking for egrep... /bin/grep -E  
checking for a sed that does not truncate output... /bin/sed  
checking for cc... cc  
checking whether the C compiler works... yes  
checking for C compiler default output file name... a.out  
checking for suffix of executables...  
checking whether we are cross compiling... no  
checking for suffix of object files... o  
checking whether we are using the GNU C compiler... yes  
checking whether cc accepts -g... yes  
checking for cc option to accept ISO C89... none needed  
checking how to run the C preprocessor... cc -E  
checking for icc... no  
checking for suncc... no  
checking whether cc understands -c and -o together... yes  
checking for system library directory... lib  
checking if compiler supports -R... no
```

```
Build process completed successfully  
Installing '/usr/local/lib/php/extensions/no-debug-non-zts-20160303/mongodb.so'  
install ok: channel://pecl.php.net/mongodb-1.5.0  
configuration option "php_ini" is not set to php.ini location  
You should add "extension=mongodb.so" to php.ini  
fred@fred-linux:~$ █
```





- Home
- News

Documentation:

- Support

Downloads:

- Browse Packages
- Search Packages
- Download Statistics

[Top Level](#) :: [Database](#)

## mongodb

### Package Information

<b>Summary</b>	MongoDB driver for PHP
<b>Maintainers</b>	Derick Rethans < <a href="mailto:derick@php.net">derick@php.net</a> > (lead) [ <a href="#">wishlist</a> ] [ <a href="#">details</a> ] Jeremy Mikola < <a href="mailto:jmikola@gmail.com">jmikola@gmail.com</a> > (lead) [ <a href="#">details</a> ] Hannes Magnusson < <a href="mailto:bjori@php.net">bjori@php.net</a> > (lead) [ <a href="#">details</a> ] Katherine Walker (developer) [ <a href="#">details</a> ]
<b>License</b>	Apache License
<b>Description</b>	The purpose of this driver is to provide exceptionally thin glue between MongoDB and PHP, implementing only fundamental and performance-critical components necessary to build a fully-functional MongoDB driver.
<b>Homepage</b>	<a href="http://docs.mongodb.org/ecosystem/drivers/php/">http://docs.mongodb.org/ecosystem/drivers/php/</a>

[ [Latest Tarball](#) ]

[ [Changelog](#) ]

[ [Browse Source](#) ]

[ [Package Bugs](#) ]

### Available Releases

Version	State	Release Date	Downloads
<a href="#">1.5.1</a>	stable	2018-07-09	<a href="#">mongodb-1.5.1.tgz</a> (1028.5kB)
<a href="#">1.5.0</a>	stable	2018-06-26	<a href="#">mongodb-1.5.0.tgz</a> (1030.1kB)

```
C:\Windows\system32\cmd.exe
C:\Users\george>
C:\Users\george>php -i |more
phpinfo()
PHP Version => 7.2.0

System => Windows NT ASUS-WINDOZE 6.3 build 9600
on) AMD64
Build Date => Nov 28 2017 23:44:10
Compiler => MSVC15 (Visual C++ 2017)
Architecture => x64
Configure Command => cscript /nologo configure.js
enable-debug-pack" "--with-pdo-oci=c:\php-snap-bu
client_12_1\sdk,shared" "--with-oci8-12c=c:\php-s
nstantclient_12_1\sdk,shared" "--enable-object-ou
tnet=shared" "--without-analyzer" "--with-pgo"
Server API => Command Line Interface
Virtual Directory Support => enabled
Configuration File (php.ini) Path => C:\Windows
Loaded Configuration File => C:\PHP\php.ini
Scan this dir for additional .ini files => (none)
Additional .ini files parsed => (none)
PHP API => 20170718
PHP Extension => 20170718
Zend Extension => 320170718
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug/Source
fred@fred-linux:~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug/Source$
fred@fred-linux:~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug/Source$
php composer.phar require mongodb/mongodb
Using version ^1.4 for mongodb/mongodb
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
 - Installing mongodb/mongodb (1.4.0): Downloading (100%)
Writing lock file
Generating autoload files
fred@fred-linux:~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug/Source$
```

```
Terminal
82 documents inserted

-----
(program exited with code: 0)
Press return to continue
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:708003]> db.customers.find().count();
82
sweetscomplete@fred-linux[up:708003]> db.customers.find({}, {name:1});
{ "_id" : ObjectId("5b47108b533b8406ac227792"), "name" : "name" }
{ "_id" : ObjectId("5b47108b533b8406ac227793"), "name" : "Conrad Perry" }
{ "_id" : ObjectId("5b47108b533b8406ac227794"), "name" : "Lonnie Knapp" }
{ "_id" : ObjectId("5b47108b533b8406ac227795"), "name" : "Darrel Roman" }
{ "_id" : ObjectId("5b47108b533b8406ac227796"), "name" : "Morgan Avila" }
{ "_id" : ObjectId("5b47108b533b8406ac227797"), "name" : "Lee Mccray" }
{ "_id" : ObjectId("5b47108b533b8406ac227798"), "name" : "Spencer Sanford" }
{ "_id" : ObjectId("5b47108b533b8406ac227799"), "name" : "Thomas Kirby" }
{ "_id" : ObjectId("5b47108b533b8406ac22779a"), "name" : "Brian Crawford" }
{ "_id" : ObjectId("5b47108b533b8406ac22779b"), "name" : "Armando Barlow" }
{ "_id" : ObjectId("5b47108b533b8406ac22779c"), "name" : "Jess Rocha" }
{ "_id" : ObjectId("5b47108b533b8406ac22779d"), "name" : "Felix Blevins" }
{ "_id" : ObjectId("5b47108b533b8406ac22779e"), "name" : "Jose Carter" }
{ "_id" : ObjectId("5b47108b533b8406ac22779f"), "name" : "Orlando Fulton" }
{ "_id" : ObjectId("5b47108b533b8406ac2277a0"), "name" : "Mitchell Roth" }
{ "_id" : ObjectId("5b47108b533b8406ac2277a1"), "name" : "Eduardo Wright" }
{ "_id" : ObjectId("5b47108b533b8406ac2277a2"), "name" : "Marc Ellis" }
{ "_id" : ObjectId("5b47108b533b8406ac2277a3"), "name" : "Joaquin Moses" }
{ "_id" : ObjectId("5b47108b533b8406ac2277a4"), "name" : "Morris Vargas" }
{ "_id" : ObjectId("5b47108b533b8406ac2277a5"), "name" : "Gene Cruz" }
Type "it" for more
sweetscomplete@fred-linux[up:708003]>
```

```
Terminal
Spencer Sanford : spencer.sanford@cablenet.net : 451-815-7386 : -7.61
                  admin : admin@sweetscomplete.com : 000-000-000 : 0.00
Wilfredo Taylor : wilfredo.taylor@telecom.net : 126-579-1787 : 25.11
Todd Lindsey : todd.lindsey@fastnet.net : 565-309-4959 : 48.91
Jose Carter : jose.carter@westcom.net : 176-111-1052 : 56.22

-----
(program exited with code: 0)
Press return to continue
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug/Source/Chapter04
sweetscomplete@fred-linux[up:110]> db.customers.find({name:/Spencer/});
{ "_id" : ObjectId("5b47108b533b8406ac227798"), "name" : "Spencer Sanford", "address" : "620 Colonial Autumn Meadow", "city" : "Nooseneck", "state_province" : "", "postal_code" : "PV9 74XN", "country" : "UK", "phone" : "451-815-7386", "balance" : -7.61, "email" : "spencer.sanford@cablenet.net", "password" : "$2y$10$E9bGbfocN9GBJ2asveMI.E05Lb17Yw1B2Waw9r3pHj5KxWKCL7Fi" }
sweetscomplete@fred-linux[up:110]> db.customers.find({_id:"5b47108b533b8406ac227798"});
sweetscomplete@fred-linux[up:110]>
```

```
Terminal
Spencer Sanford
object(MongoDB\BSON\ObjectId)#18 (1) {
  ["oid"]=>
    string(24) "5b47108b533b8406ac227798"
}

-----
(program exited with code: 0)
Press return to continue
```

```
Terminal
Name: Spencer Sanford | Old Balance: -7.61
Matched 1 document(s)
Modified 1 document(s)
Name: Spencer Sanford | New Balance: 99.99

-----
(program exited with code: 0)
Press return to continue
```

**Delete These Customers?**

name : 0.00  
Spencer Sanford : -7.61  
Samuel Harding : -11.56  
Lauri Grimes : -37.95  
Obama : 0.00  
C.T. Russell : 0.00

Delete

Cancel

**Delete These Customers?**

Delete

Cancel

**Deleted 6 document(s)**

# Chapter 5: Building Complex Queries Using Aggregatio n

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug  
sweetscomplete@fred-linux[up:344837]> db.customers.find({country:/UK/}).count();  
22  
sweetscomplete@fred-linux[up:344837]>
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug  
sweetscomplete@fred-linux[up:344837]> db.customers.distinct("country");  
[ "CA", "US", "AU", "UK" ]  
sweetscomplete@fred-linux[up:344837]>
```

```
{  
  "_id" : ObjectId("5b500ad8533b844173064593"),  
  "customer" : {  
    "_id" : ObjectId("5b482b45533b843e7b6f70c5"),  
    "name" : "Darrel Roman",  
    "state_province" : "NT",  
    "country" : "AU",  
    "balance" : 357.51  
  },  
  "product" : {  
    "_id" : ObjectId("5b4c232accf2ea73a85ed2b4"),  
    "sku" : "C3000",  
    "title" : "Chocolate Angelfood Cupcakes",  
    "price" : 0.3  
  },  
  "date" : "2017-12-30",  
  "quantity" : 71,  
  "amount" : 21.3  
}
```



```
db.purchases.aggregate( [
```

```
{ $match : { "customer.country" : "/AU/ } },
```

*MATCH STAGE*

```
{ $group : { _id : "$customer.name",  
            total : { $sum : "$amount" } } }
```

*GROUP STAGE*

```
]);
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug  
sweetscomplete@fred-linux[up:965]> db.purchases.aggregate( [  
... {  
...   $bucket: {  
...     groupBy: "$customer.name",  
...     boundaries: [ "A", "G", "M", "S", "Y" ],  
...     default: "Y-Z",  
...     output: {  
...       "count": { $sum: 1 },  
...       "names" : { $push: "$customer.name" },  
...       "amounts" : { $push: "$amount" }  
...     }  
...   }  
... },  
... {  
...   $project: {  
...     _id : "$_id",  
...     count : "$count",  
...     amounts : { $sum: "$amounts" }  
...   }  
... }  
... ] );  
{ "_id" : "A", "count" : 263, "amounts" : 44894.1 }  
{ "_id" : "G", "count" : 325, "amounts" : 51267.2 }  
{ "_id" : "M", "count" : 173, "amounts" : 25485.5 }  
{ "_id" : "S", "count" : 51, "amounts" : 6113.2 }  
{ "_id" : "Y-Z", "count" : 18, "amounts" : 2726.5 }  
sweetscomplete@fred-linux[up:965]>
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:344837]> db.purchases.aggregate( [
...   { $group: {
...     id: "$customer.name",
...     total: { $sum: "$amount" },
...     avgQty: { $avg: "$quantity" },
...     count: { $sum: 1 }
...   }
... }
... ] );
{ "_id": "admin", "total": 963.1, "avgQty": 50.83333333333336, "count": 6 }
{ "_id": "Renee Decker", "total": 2254.2, "avgQty": 43.66666666666664, "count": 15 }
{ "_id": "Louella Allen", "total": 1964.9, "avgQty": 51.07692307692308, "count": 13 }
{ "_id": "Elba Mccall", "total": 2145.7, "avgQty": 48.06666666666667, "count": 15 }
{ "_id": "Socorro Jimenez", "total": 133, "avgQty": 95, "count": 1 }
{ "_id": "Fannie Moore", "total": 3210.4, "avgQty": 44.5, "count": 16 }
{ "_id": "Josefina Hampton", "total": 2046.8, "avgQty": 37.5625, "count": 16 }
{ "_id": "Lucille Bradford", "total": 95, "avgQty": 95, "count": 1 }
{ "_id": "Geneva Case", "total": 551.9, "avgQty": 67.33333333333333, "count": 3 }
{ "_id": "Jaime Noel", "total": 2692.4, "avgQty": 57.16666666666664, "count": 18 }
{ "_id": "Isabel Rodriguez", "total": 2953.9, "avgQty": 59.642857142857146, "count": 14 }
{ "_id": "Jami Ruiz", "total": 2393.8, "avgQty": 48.23529411764706, "count": 17 }
{ "_id": "Aileen Duncan", "total": 1098.3, "avgQty": 46.142857142857146, "count": 7 }
{ "_id": "Denis Snider", "total": 140, "avgQty": 43.66666666666664, "count": 3 }
{ "_id": "Todd Lindsey", "total": 466.5, "avgQty": 49.33333333333336, "count": 6 }
{ "_id": "Garrett Campos", "total": 3432.1, "avgQty": 51.294117647058826, "count": 17 }
{ "_id": "Hans Page", "total": 919.9, "avgQty": 45.77777777777778, "count": 9 }
{ "_id": "Ramiro Bentley", "total": 1475.2, "avgQty": 56.08333333333336, "count": 12 }
{ "_id": "Jesus Bright", "total": 1351.9, "avgQty": 43.81818181818182, "count": 11 }
{ "_id": "Cecelia Case", "total": 3690.7, "avgQty": 57.88235294117647, "count": 17 }
Type "it" for more
sweetscomplete@fred-linux[up:344837]>
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:344837]> db.purchases.aggregate([
...   { $lookup: {
...     from: "customers",
...     localField: "customer.name",
...     foreignField: "name",
...     as: "purch_plus_cust" }
...   },
...   { $limit: 1 },
...   { $project: { _id: 0, customer: 0, "purch_plus_cust.password": 0 } }
... ]).pretty();
{
  "product" : {
    "_id" : ObjectId("5b4c232accf2ea73a85ed2c7"),
    "sku" : "C22000",
    "title" : "Chocolate Toaster Tarts",
    "price" : 2.2
  },
  "date" : "2017-09-20",
  "quantity" : 15,
  "amount" : 33,
  "purch_plus_cust" : [
    {
      "_id" : ObjectId("5b482b45533b843e7b6f70c3"),
      "name" : "Conrad Perry",
      "address" : "79 Amber Branch Falls",
      "city" : "Birdseye",
      "state_province" : "QC",
      "postal_code" : "G0U 0M5",
      "country" : "CA",
      "phone" : "484-181-9811",
      "balance" : 745.32,
      "email" : "conrad.perry@fastmedia.com"
    }
  ]
}
sweetscomplete@fred-linux[up:344837]>
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:344837]> db.purchases.aggregate( [
...   { $match: {
...     "customer.country": "/UK/",
...     "product.title": "/chocolate/i
...   }
... },
... { $group: {
...   "id": "$product.title",
...   "total": { $sum: "$amount" }
... }
... }
... ] );
{ "_id": "Chocolate Angelfood Cupcakes", "total": 30.6 }
{ "_id": "Mint Chocolate Milk Shake", "total": 915.9 }
{ "_id": "Chocolate Toaster Tarts", "total": 286 }
{ "_id": "Chocolate Fondue", "total": 512 }
{ "_id": "Chocolate Layer Cake", "total": 695.4 }
{ "_id": "Chocolate Mousse", "total": 195 }
{ "_id": "Chocolate Eclair", "total": 562.8 }
{ "_id": "Chocolate Soufflé", "total": 561 }
{ "_id": "Chocolate Chip Cookies", "total": 160 }
sweetscomplete@fred-linux[up:344837]>
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:214862]>
days = ["Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"];
[ "Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat" ]
sweetscomplete@fred-linux[up:214862]> db.purchases.aggregate( [
... {
...   $bucket: {
...     groupBy: { $dayOfWeek: { $toDate: "$date" } },
...     boundaries: [ 1,2,3,4,5,6,7,8 ],
...     default: "other",
...     output: {
...       "dow": { $push: { $dayOfWeek: { $toDate: "$date" } } },
...       "amounts": { $push: "$amount" }
...     }
...   }
... },
... {
...   $project: {
...     dow: { $min: "$dow" },
...     day: { $arrayElemAt: [days, { $subtract: [{ $min: "$dow" }, 1] } ]},
...     amounts: { $sum: "$amounts" }
...   }
... }
... ] );
{ "_id": 1, "dow": 1, "day": "Sun", "amounts": 19364.5 }
{ "_id": 2, "dow": 2, "day": "Mon", "amounts": 14852.3 }
{ "_id": 3, "dow": 3, "day": "Tue", "amounts": 19859.5 }
{ "_id": 4, "dow": 4, "day": "Wed", "amounts": 21470.7 }
{ "_id": 5, "dow": 5, "day": "Thu", "amounts": 21182.7 }
{ "_id": 6, "dow": 6, "day": "Fri", "amounts": 16172.4 }
{ "_id": 7, "dow": 7, "day": "Sat", "amounts": 17584.4 }
sweetscomplete@fred-linux[up:214862]>
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:214862]> db.purchases.aggregate( [
... { $match: { "customer.country": /AU/ } },
... {
...   $bucket: {
...     groupBy: "$quantity",
...     boundaries: [ 0,10,50,100 ],
...     default: "other",
...     output: {
...       "qty" : { $push: "$quantity" }
...     }
...   }
... },
... { $project: { qty : { $avg: "$qty" } } }
... ] );
{ "_id" : 0, "qty" : 4.608695652173913 }
{ "_id" : 10, "qty" : 28.610526315789475 }
{ "_id" : 50, "qty" : 74.90677966101696 }
{ "_id" : "other", "qty" : 100 }
sweetscomplete@fred-linux[up:214862]>
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:214862]> db.purchases.aggregate( [
... { $match: { "customer.country": /AU/ } },
... {
...   $bucket: {
...     groupBy: "$quantity",
...     boundaries: [ 0,10,50,100 ],
...     default: "other",
...     output: {
...       "qty" : { $push: "$quantity" }
...     }
...   }
... },
... {
...   $project: {
...     _id : { $cond: {if: { $eq: ["other", "$_id"] },
...       then: "$REMOVE",
...       else: "$_id" }},
...     qty : { $cond: {if: { $eq: ["other", "$_id"] },
...       then: "$REMOVE",
...       else: { $avg: "$qty" }}}
...   }
... }
... ] );
{ "_id" : 0, "qty" : 4.608695652173913 }
{ "_id" : 10, "qty" : 28.610526315789475 }
{ "_id" : 50, "qty" : 74.90677966101696 }
{ }
sweetscomplete@fred-linux[up:214862]>
```

```
db.purchases.mapReduce(  
  function() {  
    if (this.customer.country == "US") {  
      emit( this.customer.state_province, this.amount );  
    }  
  },  
  function(key,values) {  
    return Array.sum(values);  
  },  
  {  
    query: { /* optional */ }  
    out: "totals_by_us_state_province"  
  }  
);
```

```
Fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug  
sweetscomplete@fred-linux[up:214862]> db.purchases.mapReduce(  
...   function() {  
...     if (this.customer.country == "US") {  
...       emit( this.customer.state_province, this.amount );  
...     }  
...   },  
...   function(key,values) { return Array.sum(values); },  
...   {  
...     out: "totals_by_us_state_province"  
...   }  
... );  
{  
  "result" : "totals_by_us_state_province",  
  "timeMillis" : 1616,  
  "counts" : {  
    "input" : 830,  
    "emit" : 219,  
    "reduce" : 16,  
    "output" : 17  
  },  
  "ok" : 1  
}  
sweetscomplete@fred-linux[up:214862]>
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:214862]> db.totals_by_us_state_province.find();
{ "_id" : "AK", "value" : 4849.4 }
{ "_id" : "AR", "value" : 3210.4 }
{ "_id" : "DE", "value" : 158.1 }
{ "_id" : "IN", "value" : 297 }
{ "_id" : "MA", "value" : 213.6 }
{ "_id" : "MT", "value" : 4740 }
{ "_id" : "ND", "value" : 988.6 }
{ "_id" : "NH", "value" : 2046.8000000000002 }
{ "_id" : "NM", "value" : 1351.9 }
{ "_id" : "OK", "value" : 1098.3 }
{ "_id" : "OR", "value" : 2353.8999999999996 }
{ "_id" : "RI", "value" : 956.9000000000001 }
{ "_id" : "SC", "value" : 4819.8 }
{ "_id" : "SD", "value" : 1475.2 }
{ "_id" : "TN", "value" : 3246.6 }
{ "_id" : "VA", "value" : 25.2 }
{ "_id" : "WA", "value" : 1964.9 }
sweetscomplete@fred-linux[up:214862]>
```

MongoDB Compass Community - localhost:27017/sweetscomplete.purchases

localhost:27017 (STANDALONE) MongoDB 4.0.0 Community

# sweetscomplete.purchases

DOCUMENTS 830 TOTAL SIZE 221.4KB AVG. SIZE 273B INDEXES 1 TOTAL SIZE 20.0KB AVG. SIZE 20.0KB

Documents Aggregations Explain Plan Indexes

Enter a pipeline name... SAVE PIPELINE ...

COMMENT MODE  SAMPLE MODE  AUTO PREVIEW

830 Documents in the Collection

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

Preview of Documents in the Collection

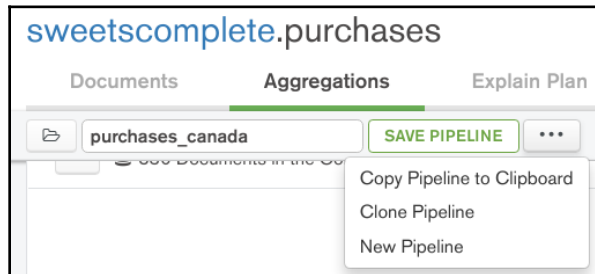
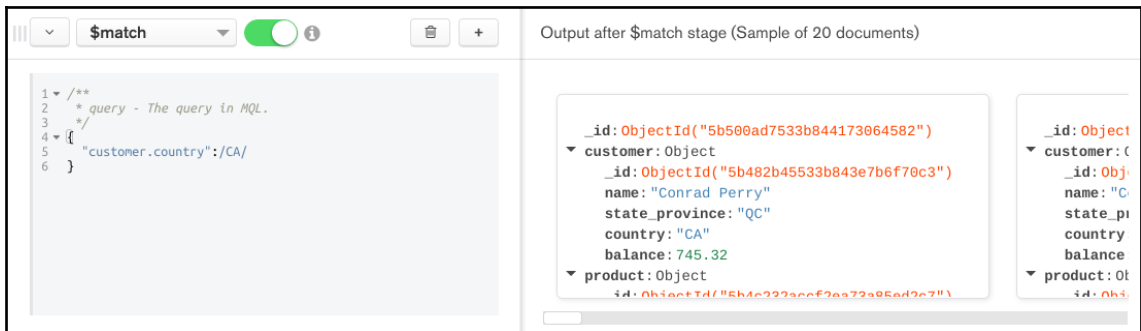
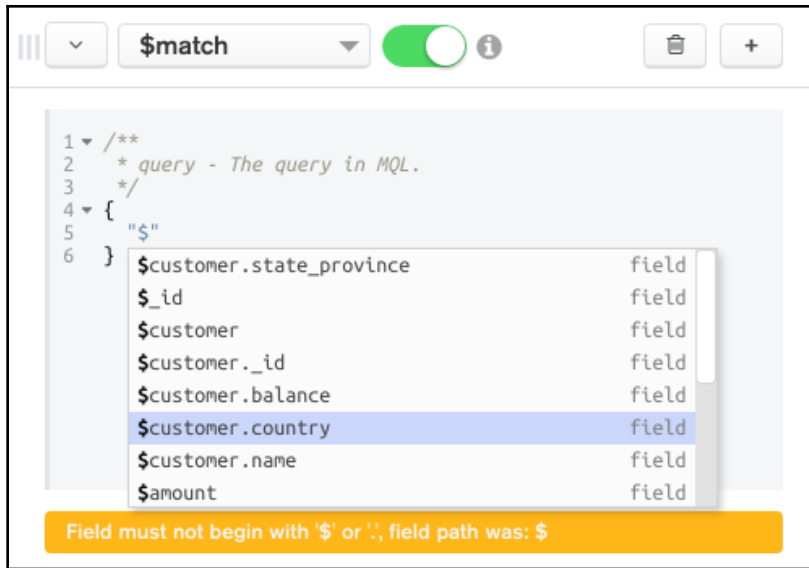
```
{ "_id": ObjectId("5b500ad7533b844173064582"),
  "customer": {
    "_id": ObjectId("5b482b45533b843e7b6f70c3"),
    "name": "Conrad Perry",
    "state_province": "QC",
    "country": "CA",
    "balance": 745.32
  },
  "product": {
    "_id": ObjectId("5b482b45533b843e7b6f70c3"),
    "name": "Conrad Perry",
    "state_province": "QC",
    "country": "CA",
    "balance": 745.32
  }
}
```

A sample of the aggregated results from this stage will be shown below

No Preview Documents

ADD STAGE





## Chapter 6:

# Maintaining MongoDB Performance

```
Terminal
> db.purchases.findOne();
{
  "_id" : ObjectId("5b500ad7533b844173064582"),
  "customer" : {
    "_id" : ObjectId("5b482b45533b843e7b6f70c3"),
    "name" : "Conrad Perry",
    "state_province" : "QC",
    "country" : "CA",
    "balance" : 745.32
  },
  "product" : {
    "_id" : ObjectId("5b4c232accf2ea73a85ed2c7"),
    "sku" : "C22000",
    "title" : "Chocolate Toaster Tarts",
    "price" : 2.2
  },
  "date" : "2017-09-20",
  "quantity" : 15,
  "amount" : 33
}
```

```
Terminal
> db.purchases.createIndex( { date: -1 } );
{
  "createdCollectionAutomatically" : false,
  "numIndexesBefore" : 1,
  "numIndexesAfter" : 2,
  "ok" : 1
}
> cls
```

```
Terminal
> db.purchases.createIndex( { "customer.country": 1 } );
{
  "createdCollectionAutomatically" : false,
  "numIndexesBefore" : 2,
  "numIndexesAfter" : 3,
  "ok" : 1
}
>
```

```
{
  "executionStats" : {
    "executionSuccess" : true,
    "nReturned" : 611,
    "executionTimeMillis" : 18,
    "totalKeysExamined" : 830,
    "totalDocsExamined" : 830,
  }
}
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:2861]>
db.purchases.createIndex(
...   {
...     "customer.country":1,
...     "customer.state_province":1,
...     "customer.name":1
...   }
... );
{
  "createdCollectionAutomatically" : false,
  "numIndexesBefore" : 1,
  "numIndexesAfter" : 2,
  "ok" : 1
}
sweetscomplete@fred-linux[up:2861]>
```

```
"executionStats" : {
  "executionSuccess" : true,
  "nReturned" : 611,
  "executionTimeMillis" : 12,
  "totalKeysExamined" : 830,
  "totalDocsExamined" : 830,
```

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
sweetscomplete@fred-linux[up:4532]> db.customers.createIndex( { purch_history: 1 } );
{
  "createdCollectionAutomatically" : false,
  "numIndexesBefore" : 1,
  "numIndexesAfter" : 2,
  "ok" : 1
}
sweetscomplete@fred-linux[up:4532]>
```

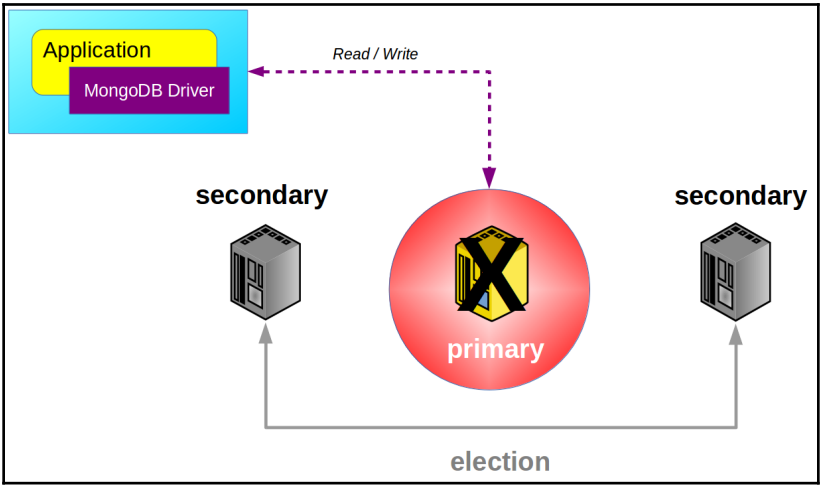
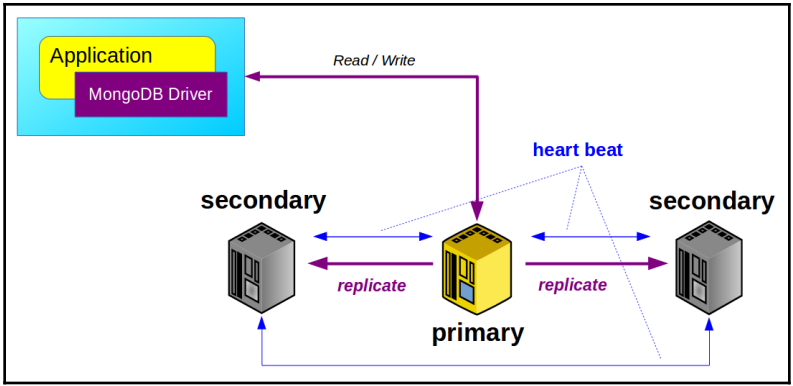
```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
fred@fred-linux:~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug$ mongodump
2018-08-05T20:29:39.040+0700   writing admin.system.version to
2018-08-05T20:29:39.041+0700   done dumping admin.system.version (1 document)
2018-08-05T20:29:39.041+0700   writing sweetscomplete.purchases to
2018-08-05T20:29:39.041+0700   writing sweetscomplete.products to
2018-08-05T20:29:39.042+0700   done dumping sweetscomplete.customers (76 documents)
2018-08-05T20:29:39.043+0700   done dumping sweetscomplete.products (64 documents)
2018-08-05T20:29:39.044+0700   done dumping sweetscomplete.purchases (830 documents)
fred@fred-linux:~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug$
```

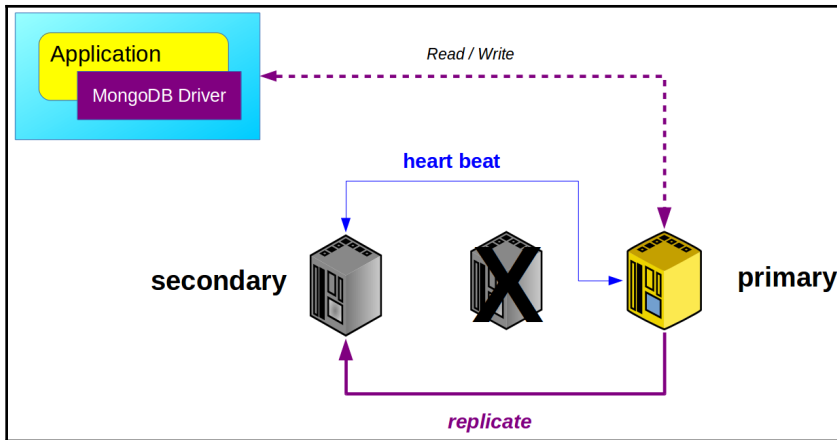
```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
fred@fred-linux:~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug$
mongodump -d sweetscomplete -c purchases
2018-07-28T14:39:54.031+0700   writing sweetscomplete.purchases to
2018-07-28T14:39:54.036+0700   done dumping sweetscomplete.purchases (830 documents)
fred@fred-linux:~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug$
```

```

fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
fred@fred-linux:~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug$
mongorestore -h sweets_33/ed:27017,fred:27017 -d sweetscomplete -c customers --drop -v -dryRun
./dump/sweetscomplete/customers.bson
2018-08-03T10:48:14.903+0700 using write concern: w='majority', j=false, fsync=false, wtimeout=0
2018-08-03T10:48:14.903+0700 checking for collection data in dump/sweetscomplete/customers.bson
2018-08-03T10:48:14.903+0700 found metadata for collection at dump/sweetscomplete/customers.metadata.json
2018-08-03T10:48:14.907+0700 dropping collection ryRun.customers before restoring
2018-08-03T10:48:14.952+0700 reading metadata for ryRun.customers from dump/sweetscomplete/customers.metadata.json
2018-08-03T10:48:14.952+0700 creating collection ryRun.customers using options from metadata
2018-08-03T10:48:14.985+0700 restoring ryRun.customers from dump/sweetscomplete/customers.bson
2018-08-03T10:48:15.375+0700 no indexes to restore
2018-08-03T10:48:15.375+0700 finished restoring ryRun.customers (76 documents)
2018-08-03T10:48:15.375+0700 done
fred@fred-linux:~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug$

```





```

fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
test@fred-linux[up:950]> rs.initiate( {
...   _id : "sweets_33",
...   members: [
...     { _id: 0, host: "ed" },
...     { _id: 1, host: "ted" },
...     { _id: 2, host: "fred" }
...   ]
... });
{
  "ok" : 1,
  "operationTime" : Timestamp(1532944427, 1),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1532944427, 1),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
test@fred-linux[up:950]>

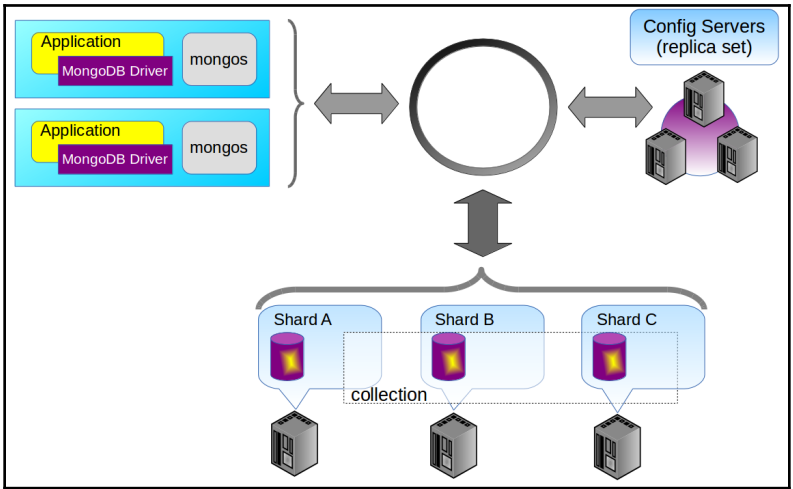
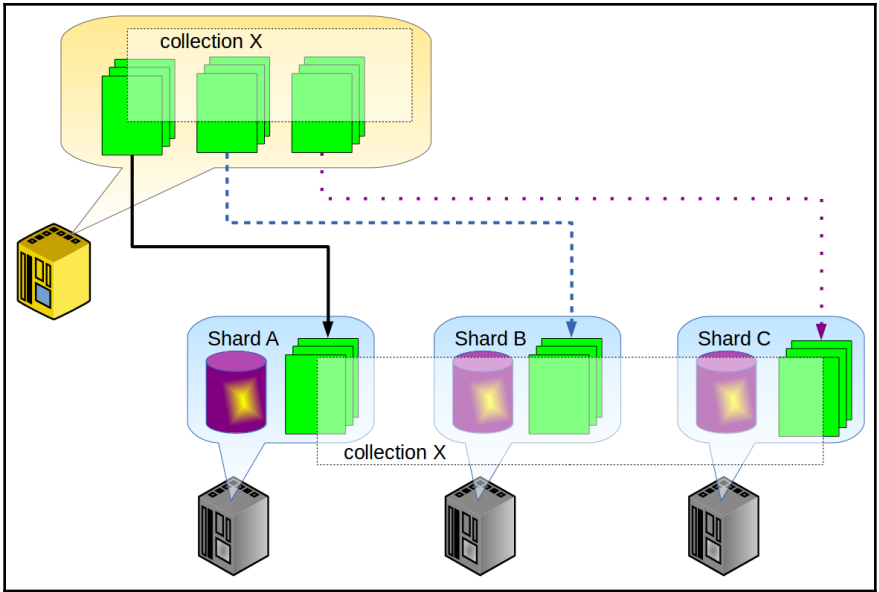
```

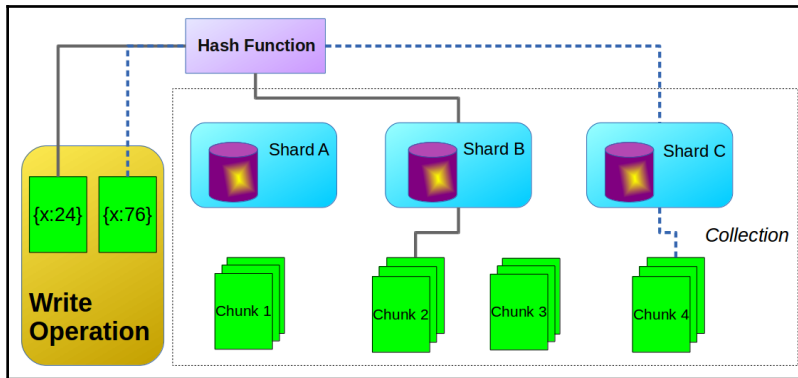
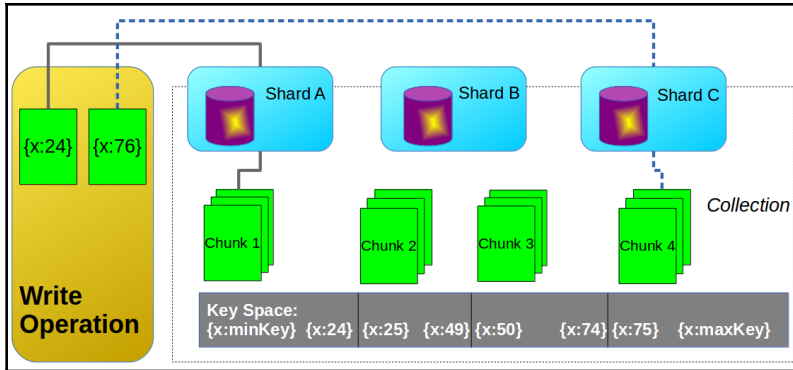
```
fred@fred-linux: ~/Desktop/www/dr_tom/dentalwellness4u.com
test@fred-linux[up:170640]> rs.conf();
{
  "_id" : "sweets_33",
  "version" : 1,
  "protocolVersion" : NumberLong(1),
  "writeConcernMajorityJournalDefault" : true,
  "members" : [
    {
      "_id" : 0,
      "host" : "ed:27017",
      "arbiterOnly" : false,
      "buildIndexes" : true,
      "hidden" : false,
      "priority" : 1,
      "tags" : {

      },
      "slaveDelay" : NumberLong(0),
      "votes" : 1
    },
    {
      "_id" : 1,
      "host" : "ted:27017",
      "arbiterOnly" : false,
      "buildIndexes" : true,
      "hidden" : false,
      "priority" : 1,
      "tags" : {

      }
    }
  ]
}
```

```
fred@fred-linux: ~/Desktop/www/dr_tom/dentalwellness4u.com
test@fred-linux[up:172410]> rs.add({ host: "ted:27017" });
{
  "ok" : 1,
  "operationTime" : Timestamp(1533116106, 1),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1533116106, 1),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
```





```

net:
  port: 27017
  bindIp: 0.0.0.0

replication:
  replSetName: "config1"

sharding:
  clusterRole: "configsvr"
  
```



```
ed@ed: ~
File Edit View Search Terminal Help
> rs.initiate(
... {
...   _id: "config1",
...   configsvr: true,
...   members: [ { _id:0, host:"ed" } ]
... }
... );
{
  "ok" : 1,
  "operationTime" : Timestamp(1533356344, 1),
  "$gleStats" : {
    "lastOpTime" : Timestamp(1533356344, 1),
    "electionId" : ObjectId("000000000000000000000000")
  },
  "lastCommittedOpTime" : Timestamp(0, 0),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1533356344, 1),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
config1:OTHER>
```

```
ed@ed: ~
File Edit View Search Terminal Help
# mongos.conf

sharding:
  configDB: "config1/ed:27017"

net:
  bindIp: 192.168.2.8:27019
_
```

```
fred@fred-linux: ~
fred@fred-linux:~$ mongo --host ed:27019
MongoDB shell version v4.0.0
connecting to: mongodb://ed:27019/
MongoDB server version: 4.0.0
Server has startup warnings:
2018-08-05T05:42:47.492+0100 I CONTROL [main]
2018-08-05T05:42:47.492+0100 I CONTROL [main] ** WARNING: Access control is
not enabled for the database.
2018-08-05T05:42:47.492+0100 I CONTROL [main] **          Read and write acc
ess to data and configuration is unrestricted.
2018-08-05T05:42:47.492+0100 I CONTROL [main] ** WARNING: You are running th
is process as the root user, which is not recommended.
2018-08-05T05:42:47.492+0100 I CONTROL [main]
test@ed:27019[up:1244]>
```

```
Fred@fred-linux: ~
test@ed:27019[up:1025]> sh.addShard("fred:27017");
{
  "shardAdded" : "shard0000",
  "ok" : 1,
  "operationTime" : Timestamp(1533445224, 6),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1533445224, 6),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
test@ed:27019[up:1025]>
```

```
Fred@fred-linux: ~
test@ed:27019[up:1244]> sh.enableSharding( "sweetscomplete" );
{
  "ok" : 1,
  "operationTime" : Timestamp(1533445840, 3),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1533445840, 3),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
test@ed:27019[up:1244]>
```

```
Fred@fred-linux: ~
sweetscomplete@ed:27019[up:1244]>
sh.shardCollection("sweetscomplete.purchases", { date: 1 } );
{
  "collectionsharded" : "sweetscomplete.purchases",
  "collectionUUID" : UUID("cfe9f701-ae26-46ef-ad0e-7b408901bf47"),
  "ok" : 1,
  "operationTime" : Timestamp(1533446408, 8),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1533446408, 8),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
sweetscomplete@ed:27019[up:1244]>
```

```
Fred@fred-linux: ~
sweetscomplete@ed:27019[up:1244]> sh.status();
--- Sharding Status ---
  sharding version: {
    "_id" : 1,
    "minCompatibleVersion" : 5,
    "currentVersion" : 6,
    "clusterId" : ObjectId("5b65293a7871cf9e665a7a5b")
  }
  shards:
    { "_id" : "shard0000", "host" : "fred:27017", "state" : 1 }
    { "_id" : "shard0001", "host" : "zed:27017", "state" : 1 }
  active mongoses:
    "4.0.0" : 1
  autosplit:
    Currently enabled: yes
  balancer:
    Currently enabled: yes
    Currently running: no
    Failed balancer rounds in last 5 attempts: 0
    Migration Results for the last 24 hours:
      No recent migrations
  databases:
    { "_id" : "config", "primary" : "config", "partitioned" : true }
      config.system.sessions
        shard key: { "_id" : 1 }
        unique: false
        balancing: true
        chunks:
          shard0000          1
          { "_id" : { "$minKey" : 1 } } --> { "_id" : { "$maxKey" : 1 } }
  } on : shard0000 Timestamp(1, 0)
    { "_id" : "sweetscomplete", "primary" : "shard0000", "partitioned" : true,
      "version" : { "uuid" : UUID("f0849cc0-a79c-4747-8215-0ae0515eef90"), "lastMod" : 1
    }
  }
```

## Chapter 7: Securing MongoDB

```
root@zed: /etc/ssl
File Edit View Search Terminal Help
root@zed:/etc/ssl# openssl req -new -x509 -days 365 -key ca.key -out ca.crt
Enter pass phrase for ca.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:TH
State or Province Name (full name) [Some-State]:Surin
Locality Name (eg, city) []:Surin
Organization Name (eg, company) [Internet Widgits Pty Ltd]:unlikelysource.com
Organizational Unit Name (eg, section) []:mongodb
Common Name (e.g. server FQDN or YOUR name) []:cn=zed,ou=mongodb,dc=unlikelysource,dc=com
Email Address []:doug@unlikelysource.com
root@zed:/etc/ssl#
```

```

net:
port: 27017
bindIp: 0.0.0.0
ssl:
mode: requireSSL
PEMKeyFile: /etc/ssl/mongod.pem
PEMKeyPassword: "password"
CAFile: /etc/ssl/ca.pem

```

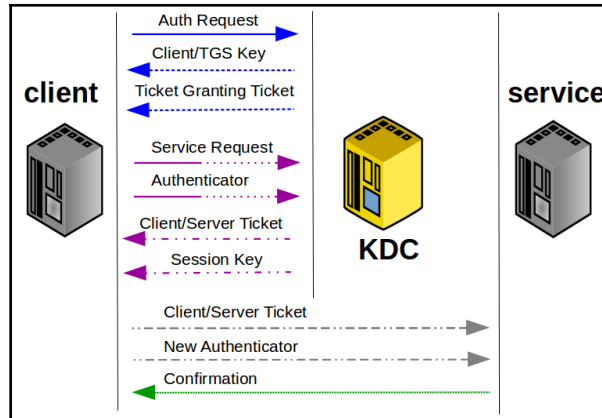
```

zed@zed: ~
File Edit View Search Terminal Help
zed@zed:~$ mongo --ssl --sslPEMKeyFile /etc/ssl/client.pem --sslPEMKeyPassword "password" --sslCAFile /etc/ssl/ca.pem --host mongod
MongoDB shell version v4.0.0
connecting to: mongod://mongod:27017/
MongoDB server version: 4.0.0
Server has startup warnings:
2018-08-19T14:29:10.614+0700 I STORAGE [initandlisten]
2018-08-19T14:29:10.614+0700 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine
2018-08-19T14:29:10.614+0700 I STORAGE [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem
2018-08-19T14:29:15.159+0700 I CONTROL [initandlisten]
2018-08-19T14:29:15.159+0700 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.
2018-08-19T14:29:15.159+0700 I CONTROL [initandlisten] ** Read and write access to data and configuration is unrestricted.
2018-08-19T14:29:15.159+0700 I CONTROL [initandlisten]
---
Enable MongoDB's free cloud-based monitoring service to collect 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 created for you. Anyone you share the URL with will also be able to view this page. MongoDB may use this information to make product improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command:
db.enableFreeMonitoring()
---
>

```



```
zed@zed: ~
File Edit View Search Terminal Help
> use admin;
switched to db admin
> db.createUser(
...   {
...     user: "superMan",
...     pwd: "up.up.and.away!",
...     roles: [ { role: "userAdminAnyDatabase", db: "admin" } ]
...   }
... );
Successfully added user: {
  "user" : "superMan",
  "roles" : [
    {
      "role" : "userAdminAnyDatabase",
      "db" : "admin"
    }
  ]
}
>
```

```
zed@zed: ~
File Edit View Search Terminal Help
> use admin;
switched to db admin
> db.createRole(
... {
...   role: "purchasing",
...   privileges: [
...     { resource: { db: "sweetscomplete", collection: "purchases" },
...       actions: [ "find", "update", "insert", "remove" ] }
...   ],
...   roles: [ { role: "read", db: "sweetscomplete" } ],
...   authenticationRestrictions: [
...     {
...       clientSource: ["192.168.2.107"],
...       serverAddress: ["192.168.2.107"]
...     }
...   ]
... }
... });
```

```
zed@zed: ~
File Edit View Search Terminal Help
> use sweetscomplete;
switched to db sweetscomplete
> db.createUser(
...   {
...     user: "sweetBasic",
...     pwd: "some.password",
...     roles: [ { role: "readWrite", db: "sweetscomplete" } ]
...   }
... );
Successfully added user: {
  "user" : "sweetBasic",
  "roles" : [
    {
      "role" : "readWrite",
      "db" : "sweetscomplete"
    }
  ]
}
>
```

## Chapter 8: Getting from a Web Form to MongoDB

```
fred@fred-linux: ~/Desktop/Repos/MongoDB-Quick-Start-Guide-Doug
test@fred-linux[up:2174]> use sweetscomplete;
switched to db sweetscomplete
sweetscomplete@fred-linux[up:2174]> db.purchases.findOne();
{
  "_id" : ObjectId("5b500ad7533b844173064582"),
  "customer" : {
    "_id" : ObjectId("5b482b45533b843e7b6f70c3"),
    "name" : "Conrad Perry",
    "state_province" : "QC",
    "country" : "CA",
    "balance" : 745.32
  },
  "product" : {
    "_id" : ObjectId("5b4c232accf2ea73a85ed2c7"),
    "sku" : "C22000",
    "title" : "Chocolate Toaster Tarts",
    "price" : 2.2
  },
  "date" : "2017-09-20",
  "quantity" : 15,
  "amount" : 33
}
sweetscomplete@fred-linux[up:2174]>
```

About Us

Products

Specials

Contact Us



**Purchases Admin**

Start Date:  End Date:  Limit:

Show  entries

Search:

Cust Name	Prod Title	Amount	Date
admin	Cheesecake	7	2018-01-01
Jaime Noel	Pumpkin Pie	176.4	2018-01-03
Lee Mccray	Pecan Praline Ice Cream	319.2	2018-01-01
Raymond Sanford	Apple Turnover	53.2	2018-01-01
Thomas Kirby	Peanut Butter Cups	16.5	2018-01-02

Showing 1 to 5 of 5 entries

Previous  Next



**Add Purchase**

Customer	<input type="text" value="Aileen Duncan"/>
Product	<input type="text" value="Ice"/>
Quantity	<input type="text" value=""/>
Address	<input type="text" value=""/>

Please enter product name, and quantity.

- Crispy Rice Treats [C12000]
- Ice Cream Cake [I53000]
- Ice Cream Cone [I42000]
- Ice Cream Sandwich [I55000]
- Neapolitan Ice Cream [N35000]
- Pecan Praline Ice Cream [P57000]
- Pumpkin Ice Cream [P50000]
- Spice Cake [S24000]

sweetscomplete.com. All rights reserved.  
[About Us](#) | [Products](#) | [Specials](#) | [Contact Us](#)  
 Tel: +44-1234567890 Fax: +44-1234567891 Email: sales@sweetscomplete.com

```
return [
  'uri' => [
    'host' => '127.0.0.1',
    'database' => 'sweetscomplete'
  ]
];
```

```
return [
  'uri' => [
    'host' => '192.168.2.107',
    'database' => 'sweetscomplete',
    'port' => 27017,
  ],
  'uriOpts' => [
    'replicaSet' => 'sweets_11',
  ],
];
```

```
replication:
  replSetName: "sweets_11"

net:
  port: 27017
  bindIp: 0.0.0.0
```

```
return [
  'uri' => [
    'host' => 'mongod',
    'database' => 'sweetscomplete',
    'username' => 'zed',
    'password' => 'password',
  ],
  'uriOpts' => [
    'ssl' => true,
    'replicaSet' => 'sweets_11',
    'authSource' => 'admin',
  ],
  'driverOpts' => [
    'ca_file' => '/etc/ssl/ca.pem',
    'pem_file' => '/etc/ssl/zed.pem',
    'pem_pwd' => 'password',
  ],
];
```

```
net:
  port: 27017
  bindIp: 0.0.0.0

net:
  ssl:
    mode: requireSSL
    PEMKeyFile: /etc/ssl/mongod.pem
    PEMKeyPassword: "password"
    CAFile: /etc/ssl/ca.pem

security:
  authorization: enabled

setParameter:
  authenticationMechanisms: PLAIN,SCRAM-SHA-256,MONGODB-X509
```