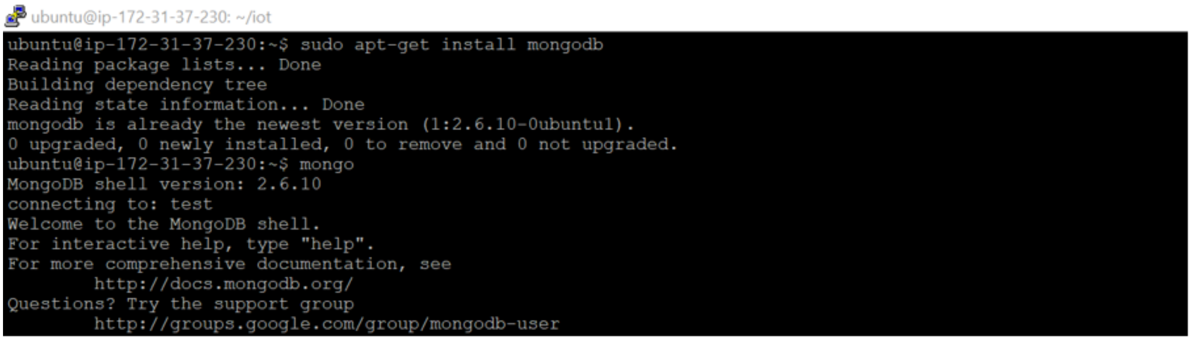
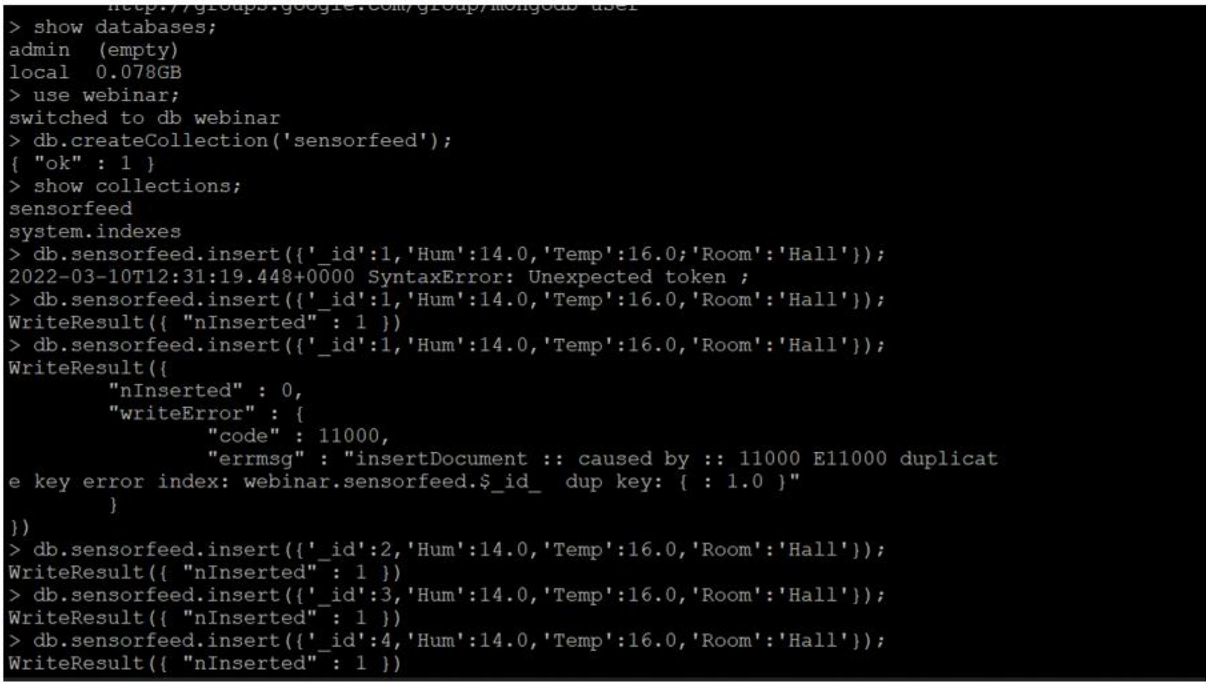
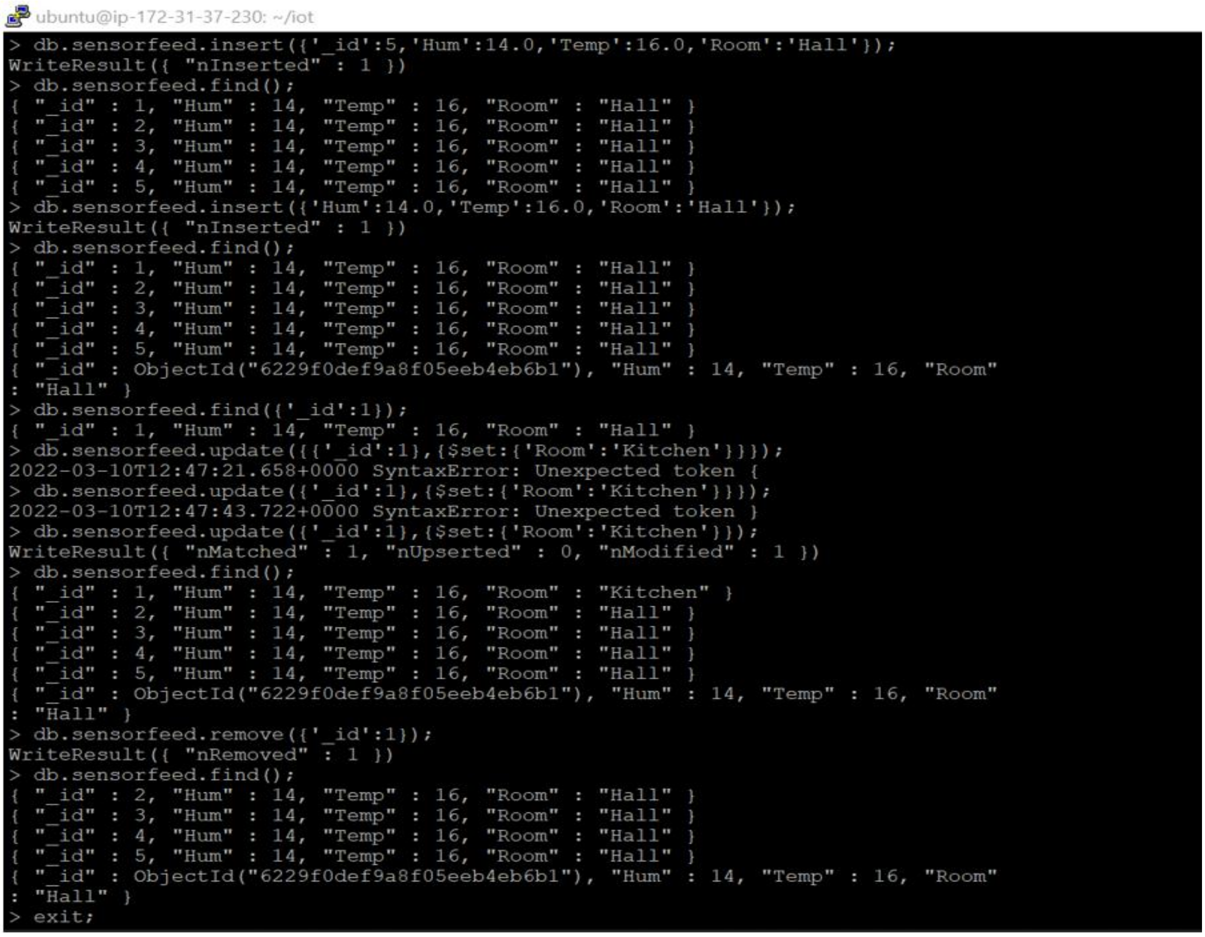
MongoDB Assignment



```
ubuntu@ip-172-31-37-230:~$ sudo apt-get install mongodb
Reading package lists... Done
Building dependency tree
Reading state information... Done
mongodb is already the newest version (1:2.6.10-0ubuntu1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-37-230:~$ mongo
MongoDB shell version: 2.6.10
connecting to: test
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
```

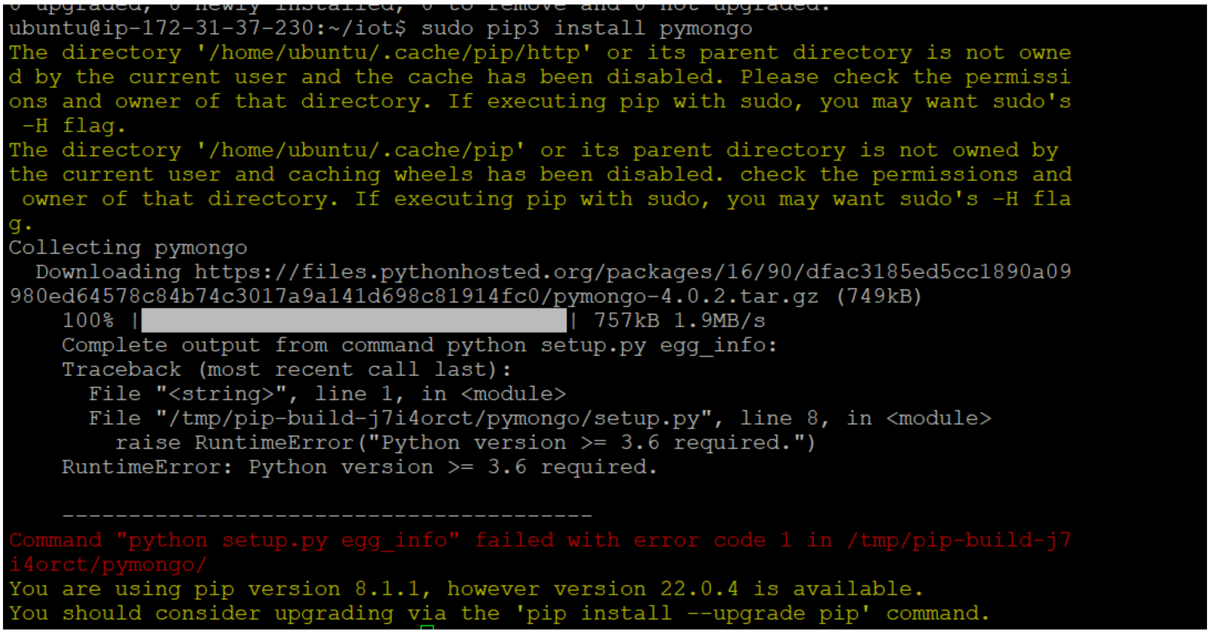


```
        http://groups.google.com/group/mongodb-user
> show databases;
admin  (empty)
local  0.078GB
> use webinar;
switched to db webinar
> db.createCollection('sensorfeed');
{ "ok" : 1 }
> show collections;
sensorfeed
system.indexes
> db.sensorfeed.insert({'_id':1,'Hum':14.0,'Temp':16.0;'Room':'Hall'});
2022-03-10T12:31:19.448+0000 SyntaxError: Unexpected token ;
> db.sensorfeed.insert({'_id':1,'Hum':14.0,'Temp':16.0,'Room':'Hall'});
WriteResult({ "nInserted" : 1 })
> db.sensorfeed.insert({'_id':1,'Hum':14.0,'Temp':16.0,'Room':'Hall'});
WriteResult({
        "nInserted" : 0,
        "writeError" : {
                "code" : 11000,
                "errmsg" : "insertDocument :: caused by :: 11000 E11000 duplicat
e key error index: webinar.sensorfeed.$_id_  dup key: { : 1.0 }"
        }
})
> db.sensorfeed.insert({'_id':2,'Hum':14.0,'Temp':16.0,'Room':'Hall'});
WriteResult({ "nInserted" : 1 })
> db.sensorfeed.insert({'_id':3,'Hum':14.0,'Temp':16.0,'Room':'Hall'});
WriteResult({ "nInserted" : 1 })
> db.sensorfeed.insert({'_id':4,'Hum':14.0,'Temp':16.0,'Room':'Hall'});
WriteResult({ "nInserted" : 1 })
```

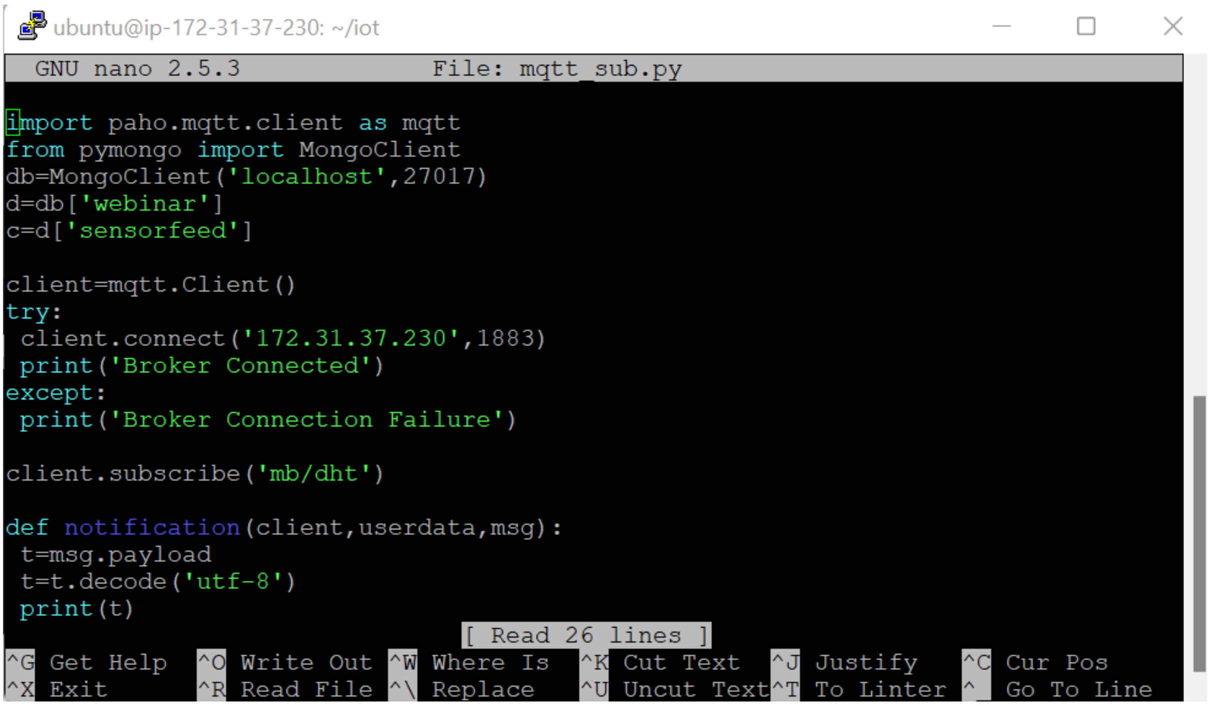


```
> db.sensorfeed.insert({'_id':5,'Hum':14.0,'Temp':16.0,'Room':'Hall'});
WriteResult({ "nInserted" : 1 })
> db.sensorfeed.find();
{ "_id" : 1, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 2, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 3, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 4, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 5, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
> db.sensorfeed.insert({'Hum':14.0,'Temp':16.0,'Room':'Hall'});
WriteResult({ "nInserted" : 1 })
> db.sensorfeed.find();
{ "_id" : 1, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 2, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 3, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 4, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 5, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : ObjectId("6229f0def9a8f05eeb4eb6b1"), "Hum" : 14, "Temp" : 16, "Room"
: "Hall" }
> db.sensorfeed.find({'_id':1});
{ "_id" : 1, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
> db.sensorfeed.update({{'_id':1},{$set:{'Room':'Kitchen'}}});
2022-03-10T12:47:21.658+0000 SyntaxError: Unexpected token {
> db.sensorfeed.update({'_id':1},{$set:{'Room':'Kitchen'}}});
2022-03-10T12:47:43.722+0000 SyntaxError: Unexpected token }
> db.sensorfeed.update({'_id':1},{$set:{'Room':'Kitchen'}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.sensorfeed.find();
{ "_id" : 1, "Hum" : 14, "Temp" : 16, "Room" : "Kitchen" }
{ "_id" : 2, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 3, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 4, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 5, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : ObjectId("6229f0def9a8f05eeb4eb6b1"), "Hum" : 14, "Temp" : 16, "Room"
: "Hall" }
> db.sensorfeed.remove({'_id':1});
WriteResult({ "nRemoved" : 1 })
> db.sensorfeed.find();
{ "_id" : 2, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 3, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 4, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : 5, "Hum" : 14, "Temp" : 16, "Room" : "Hall" }
{ "_id" : ObjectId("6229f0def9a8f05eeb4eb6b1"), "Hum" : 14, "Temp" : 16, "Room"
: "Hall" }
> exit;
```

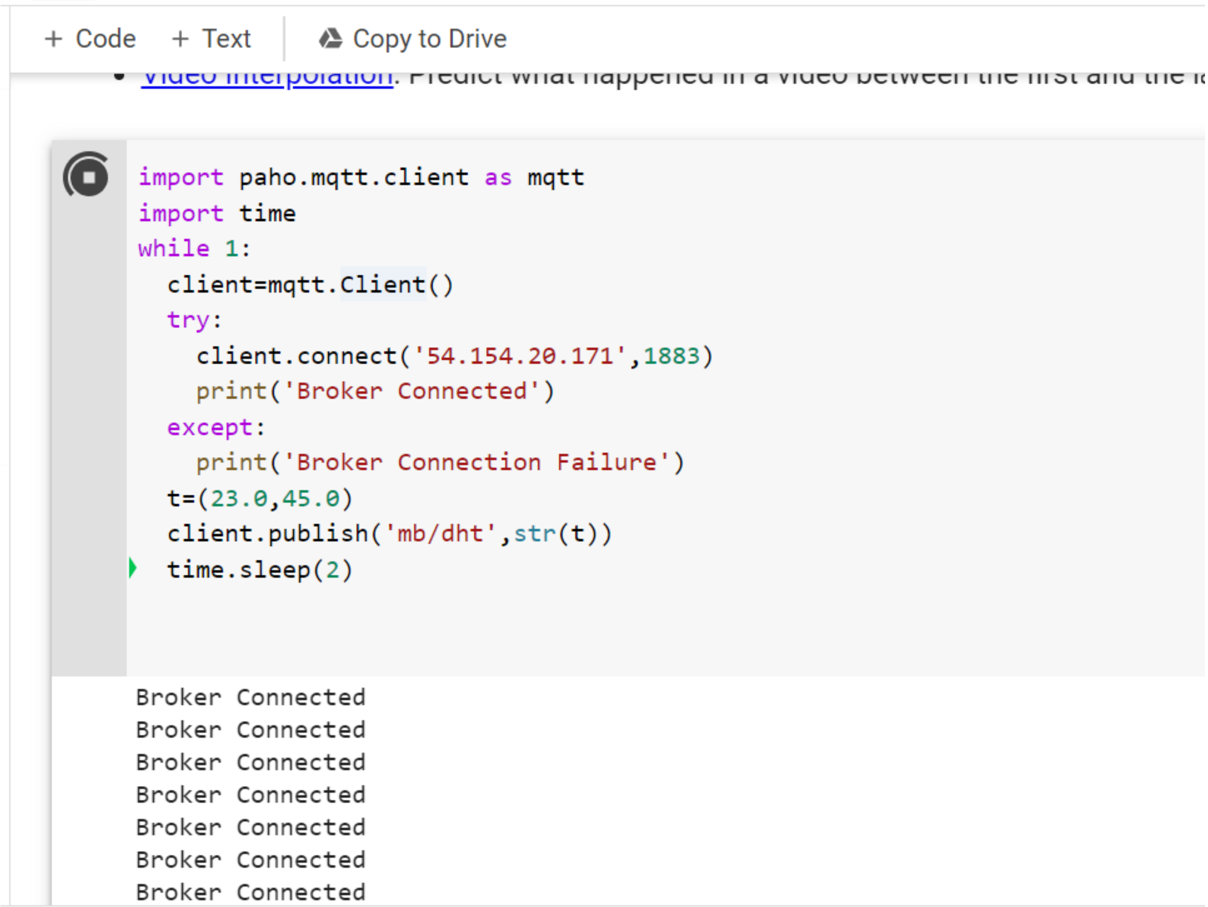
Installing pymongo



```
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-37-230:~/iot$ sudo pip3 install pymongo
The directory '/home/ubuntu/.cache/pip/http' or its parent directory is not owne
d by the current user and the cache has been disabled. Please check the permissi
ons and owner of that directory. If executing pip with sudo, you may want sudo's
 -H flag.
The directory '/home/ubuntu/.cache/pip' or its parent directory is not owned by
the current user and caching wheels has been disabled. check the permissions and
 owner of that directory. If executing pip with sudo, you may want sudo's -H fla
g.
Collecting pymongo
  Downloading https://files.pythonhosted.org/packages/16/90/dfac3185ed5cc1890a09
980ed64578c84b74c3017a9a141d698c81914fc0/pymongo-4.0.2.tar.gz (749kB)
    100% |████████████████████████████████| 757kB 1.9MB/s
    Complete output from command python setup.py egg_info:
    Traceback (most recent call last):
      File "<string>", line 1, in <module>
      File "/tmp/pip-build-j7i4orct/pymongo/setup.py", line 8, in <module>
        raise RuntimeError("Python version >= 3.6 required.")
    RuntimeError: Python version >= 3.6 required.

    ----------------------------------------
Command "python setup.py egg_info" failed with error code 1 in /tmp/pip-build-j7
i4orct/pymongo/
You are using pip version 8.1.1, however version 22.0.4 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
```

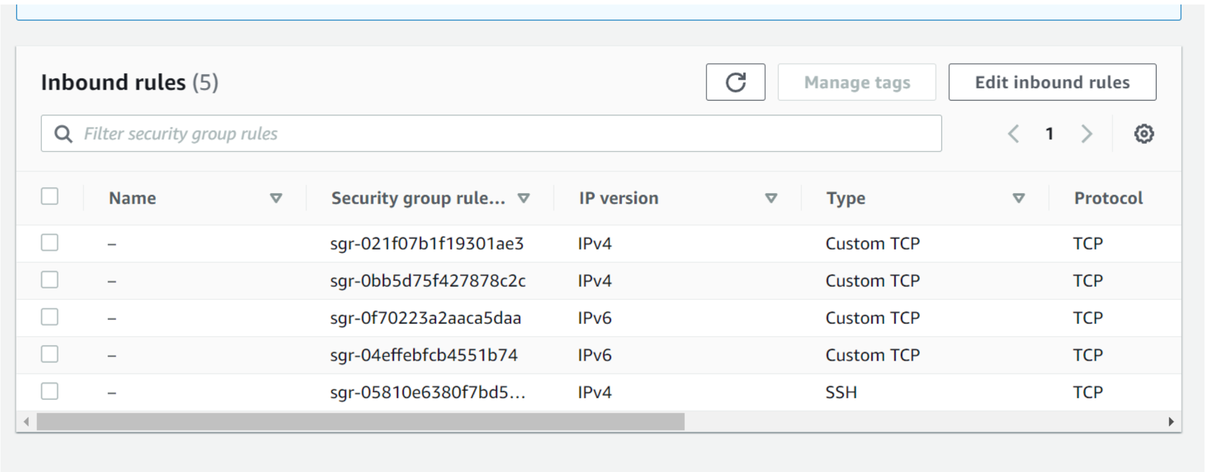


```
import paho.mqtt.client as mqtt
from pymongo import MongoClient
db=MongoClient('localhost',27017)
d=db['webinar']
c=d['sensorfeed']

client=mqtt.Client()
try:
 client.connect('172.31.37.230',1883)
 print('Broker Connected')
except:
 print('Broker Connection Failure')

client.subscribe('mb/dht')

def notification(client,userdata,msg):
 t=msg.payload
 t=t.decode('utf-8')
 print(t)
```

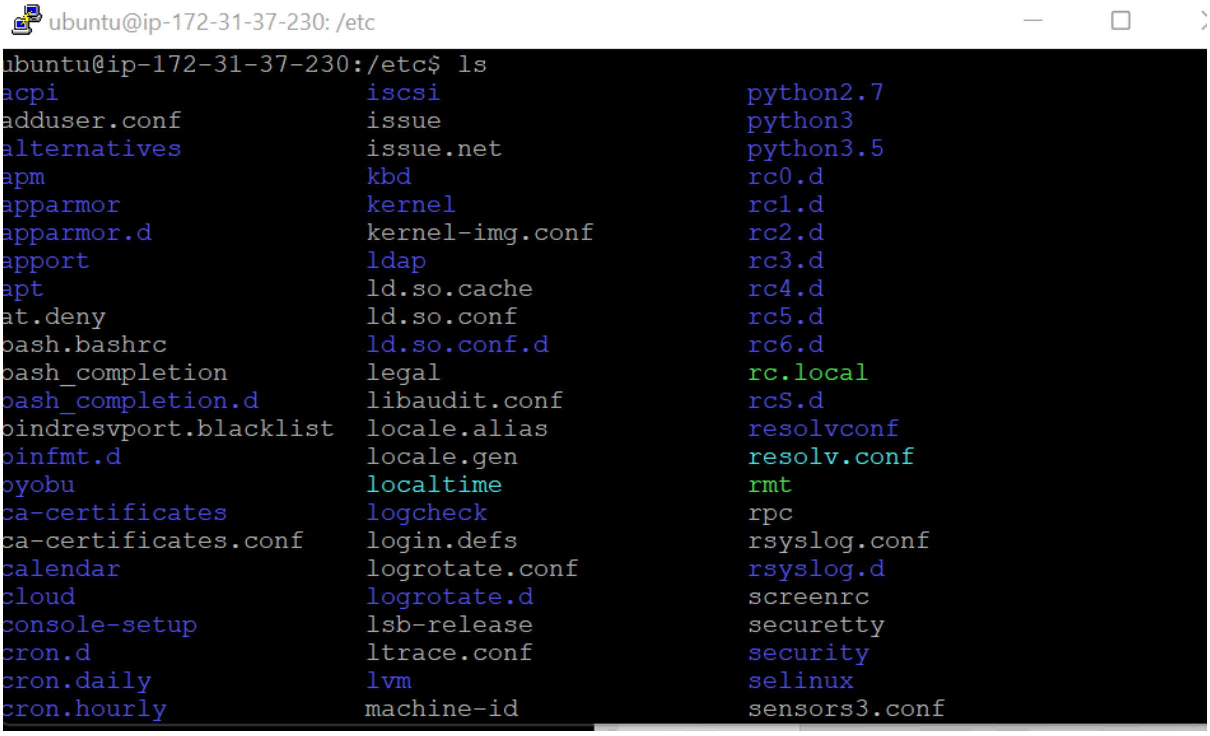

+ Code
+ Text
Copy to Drive
• Video interpolation: Predict what happened in a video between the first and the l

```
import paho.mqtt.client as mqtt
import time
while 1:
  client=mqtt.Client()
  try:
    client.connect('54.154.20.171',1883)
    print('Broker Connected')
  except:
    print('Broker Connection Failure')
  t=(23.0,45.0)
  client.publish('mb/dht',str(t))
  time.sleep(2)
```

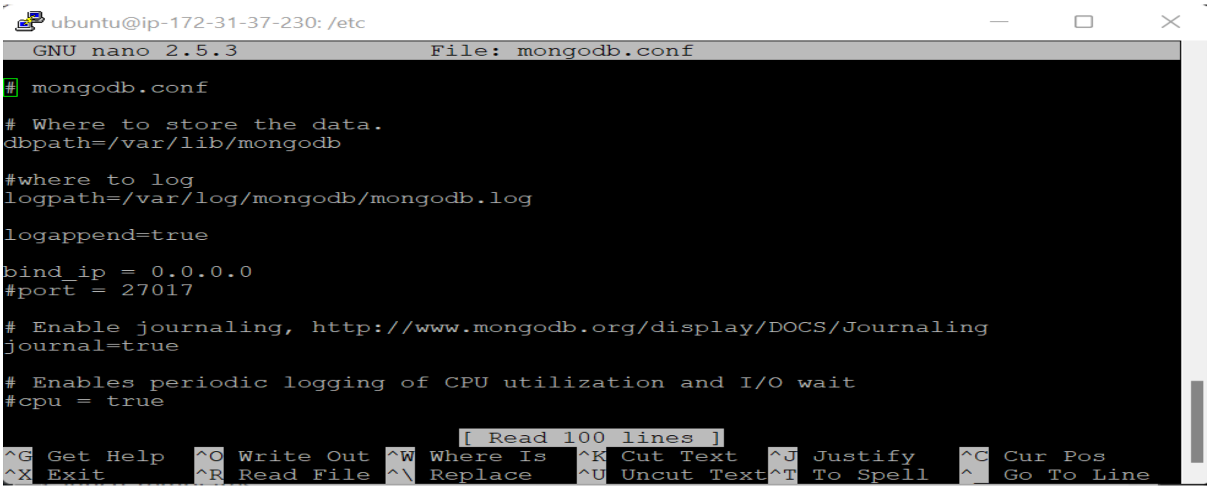
```
Broker Connected
Broker Connected
Broker Connected
Broker Connected
Broker Connected
Broker Connected
Broker Connected
```



□	Name	Security group rule...	IP version	Type	Protocol
□	–	sgr-021f07b1f19301ae3	IPv4	Custom TCP	TCP
□	–	sgr-0bb5d75f427878c2c	IPv4	Custom TCP	TCP
□	–	sgr-0f70223a2aaca5daa	IPv6	Custom TCP	TCP
□	–	sgr-04effebfcb4551b74	IPv6	Custom TCP	TCP
□	–	sgr-05810e6380f7bd5...	IPv4	SSH	TCP



```
ubuntu@ip-172-31-37-230:/etc$ ls
acpi                      iscsi                     python2.7
adduser.conf              issue                     python3
alternatives              issue.net                 python3.5
apm                       kbd                       rc0.d
apparmor                  kernel                    rc1.d
apparmor.d                kernel-img.conf           rc2.d
apport                    ldap                      rc3.d
apt                       ld.so.cache               rc4.d
at.deny                   ld.so.conf                rc5.d
bash.bashrc               ld.so.conf.d              rc6.d
bash_completion           legal                     rc.local
bash_completion.d         libaudit.conf             rcS.d
bindresvport.blacklist    locale.alias              resolvconf
binfmt.d                  locale.gen                resolv.conf
byobu                     localtime                 rmt
ca-certificates           logcheck                  rpc
ca-certificates.conf      login.defs                rsyslog.conf
calendar                  logrotate.conf            rsyslog.d
cloud                     logrotate.d               screenrc
console-setup             lsb-release               securetty
cron.d                    ltrace.conf               security
cron.daily                lvm                       selinux
cron.hourly               machine-id                sensors3.conf
```



```
  GNU nano 2.5.3                File: mongodb.conf

# mongodb.conf

# Where to store the data.
dbpath=/var/lib/mongodb

#where to log
logpath=/var/log/mongodb/mongodb.log

logappend=true

bind_ip = 0.0.0.0
#port = 27017

# Enable journaling, http://www.mongodb.org/display/DOCS/Journaling
journal=true

# Enables periodic logging of CPU utilization and I/O wait
#cpu = true
                               [ Read 100 lines ]
^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^\ Replace    ^U Uncut Text^T To Spell    ^_ Go To Line
```