

Python : working with NMEA data sent to URL/IP

<https://stackoverflow.com/questions/28160010/working-with-nmea-data-sent-to-url-ip>

i have a bunch of devices that send NMEA sentences to a URL/ip. that look like this

```
"$GPGGA,200130.0,3447.854659,N,11014.636735,W,1,11,0.8,41.4,M,-24.0,M,,*53"
```

i want to read this data in, parse it and upload the key parts to a database. i know how to parse it and upload it to the DB but i am at a complete loss on how to "read"/accept/get the data into a python program so that i can parse and upload.

my first thought was to point it at a Django page and then have Django parse it and upload to the database (data will be accessed from Django site) but its a NMEA sentence not a HTTP request so Django rejects it as "message Bad request syntax"

what is the best (python) way to read NMEA sentences sent to a url/IP?

I assume you have some hardware that has an ethernet connection, and it pipes out the NMEA string over its ethernet connection. this probably defaults to having some random 192.168.0.x ip address and spitting out data over port 12002 or something

you would typically create a socket to listen for this incoming data

server.py

```
import socket
host = "" #Localhost
port = 12002
PACKET_SIZE=1024 # how many characters to read at a time
sock = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
sock.bind((host,port))
sock.listen(5) # we should never have more than one client
def work_thread(client):
    while True: #continuously read and handle data
        data = client.recv(PACKET_SIZE)
        if not data:
            break # done with this client
        processData(data)
```

```
while True:
    client,addr = sock.accept() # your script will block here waiting for
a connection
    t = threading.Thread(target=work_thread,args=(client,))
    t.start()
```

sometimes however you need to ping the device to get the data

client.py

```
import socket
host = "192.168.0.204" #Device IP
port = 12002
PACKET_SIZE=1024 # how many characters to read at a time
sock = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
sock.connect((host,port)) #connect to the device
while True: #continuously read and handle data
    data = sock.recv(PACKET_SIZE)
    processData(data)
```