

```
#!/usr/bin/python

# hat.py
#
# Script to grab one set of serial data and log it to a sql database
# Requires a database and table to be set up already for the data
#
# Sources:
#   pyserial.readthedoc.io/en/latest/pyserial_api.html
#   pyserial.readthedoc.io/en/latest/shortintro.html
#   https://docs.python.org/3/library/sqlite3.html
#
# Written by Josh Andrews 4/27/18

import serial
import time
import datetime
import sqlite3 as sq

# list to hold all data going to sql database
sqldat = []

# Get date and time in YYYY-MM-DD HH:MM:SS
timestamp = datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
sqldat.append(timestamp)

# connect to the expansion hat and write to it
# (any write returns all sensor values).
ser = serial.Serial('/dev/ttyAMA0', 9600, timeout=0.5)
ser.write(b'1')

# while we have data to read, read it and store in list as a float.
# currently uses the timeout to breakout of loop if no errors.
# need to look more at documentation on EOL with readline()
while True:
    try:
        ser_bytes = ser.readline()
        info = float(ser_bytes[0:len(ser_bytes)-2].decode("utf-8"))
        sqldat.append(info)
    except:
        break

# use tuple form to insert data into sql
sqldat = tuple(sqldat,)

#open/create sql database, use full path to make crontab happy.
conn = sq.connect('/var/www/html/sensor_data.db')
c = conn.cursor()

# insert our sensor information to database
c.execute('''INSERT INTO sens_data VALUES (?, (?*1.8)+32,?,?,?,((?/100)*1.8)+32,?/10000,?/1000)''',sqldat)

# delete data over 24 hours old
c.execute('''DELETE FROM sens_data WHERE timestamp <= datetime('now','-24 hours','localtime')''')

# commit the changes and close the database
conn.commit()
conn.close()
```