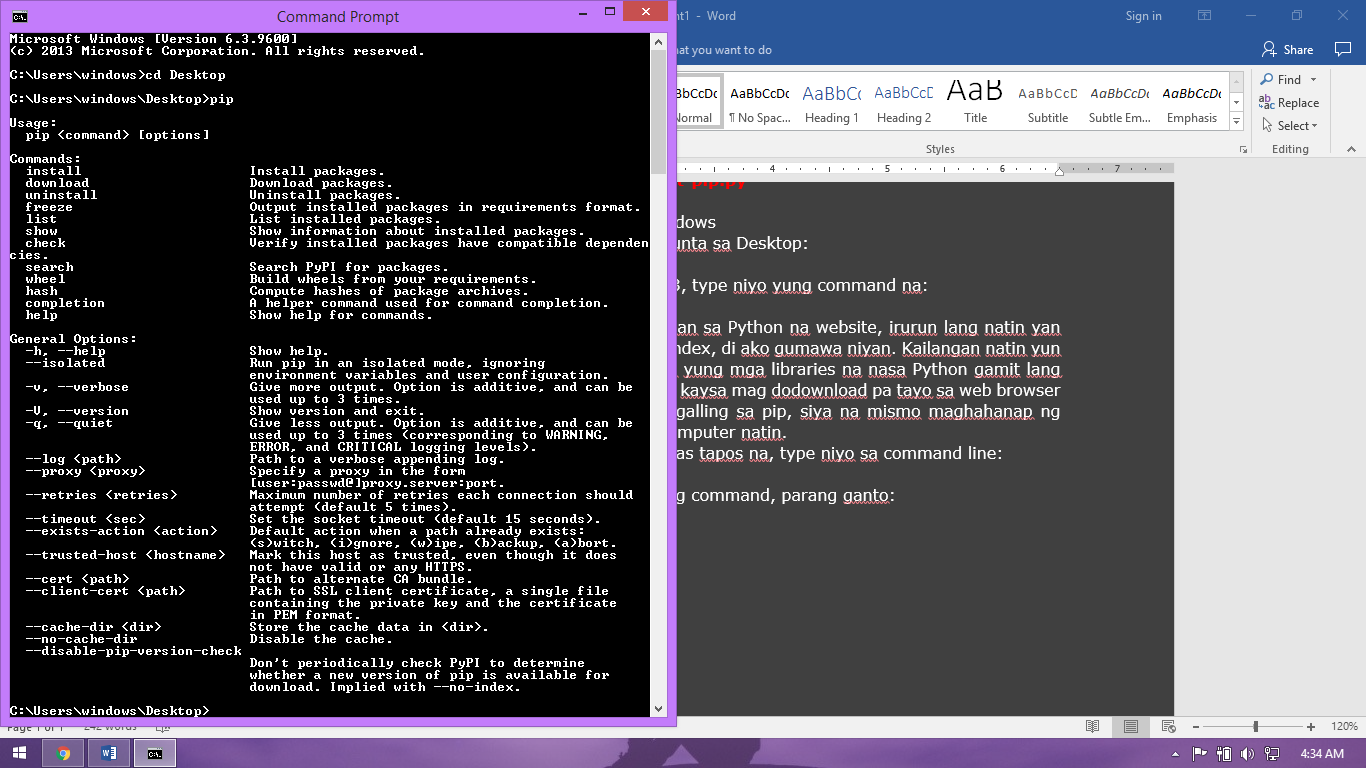
Taglish na ko para mabilis mag-isip ng words haha. Natest niyo na yung led strips diba, so ang gagawin natin ngayon is yung pag send ng data from Raspberry Pi to Arduino. Yung RPi to Arduino parang Laptop to Arduino lang yan kasi computer din ang RPi. So sa laptop niyo muna tayo magtetest using Windows OS.

Dapat meron kayong Python 3, pip saka Arduino IDE sa laptop niyo.

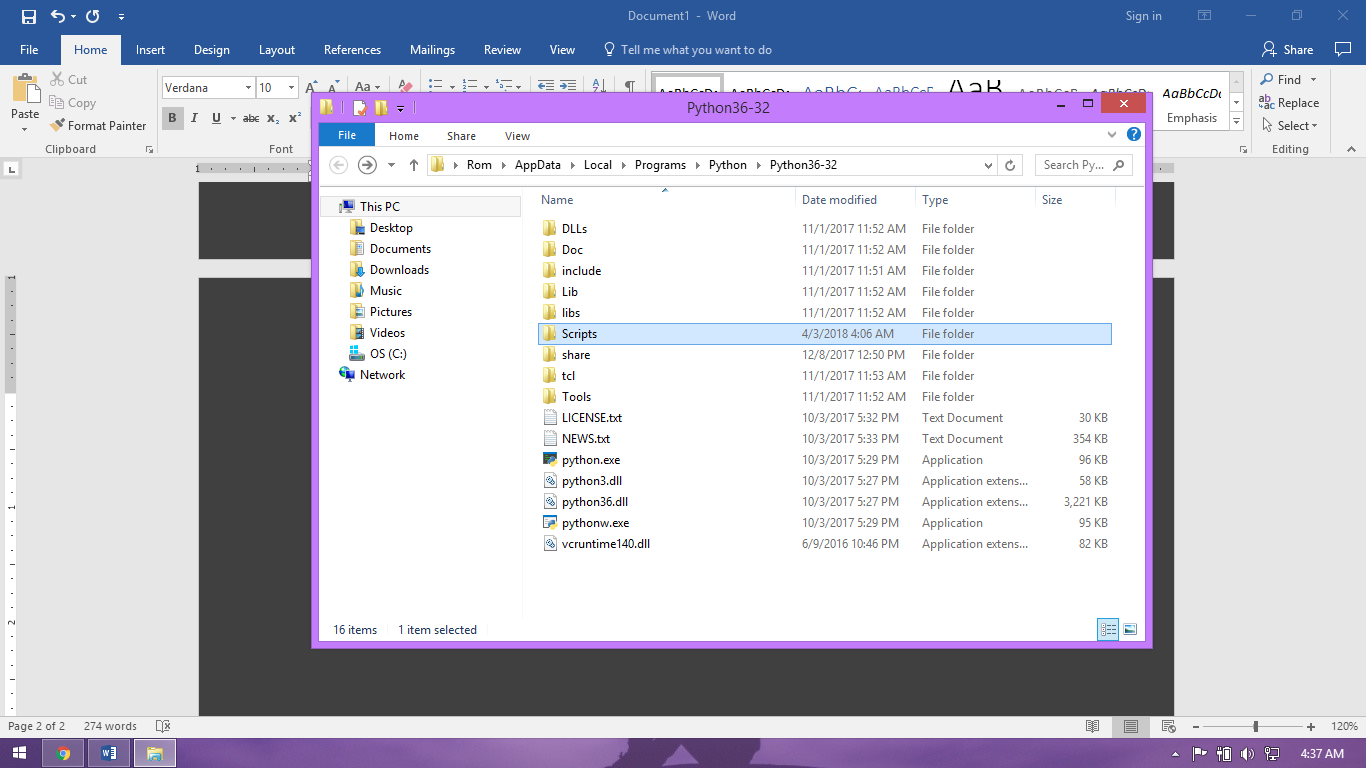
Install niyo Python 3 sa official site: <https://www.python.org/downloads/>

Install niyo yung pip by this instruction:

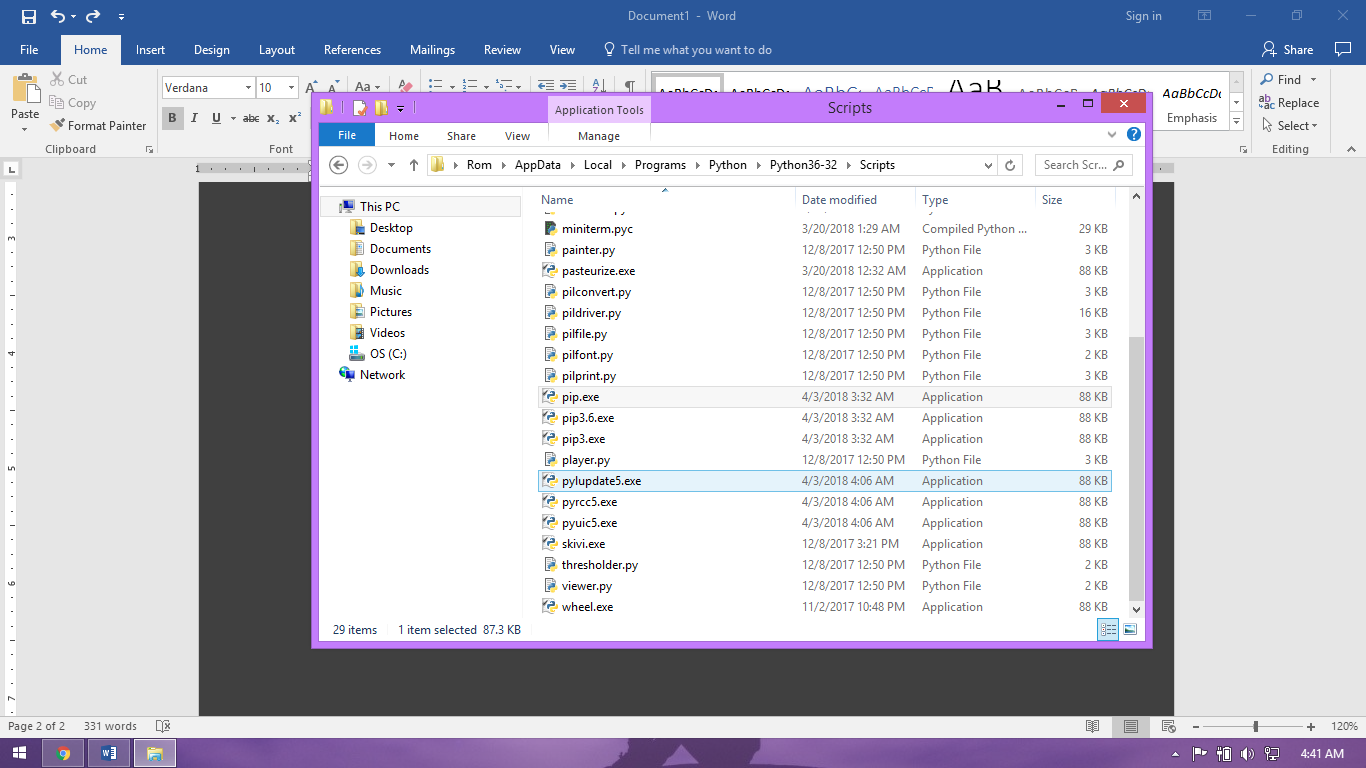
* Download niyo sa drive natin yung **get-pip.py**
* Lagay niyo sa desktop niyo.
* Open niyo yung command line sa Windows
* Change directory command kayo papunta sa Desktop:
* **cd Desktop**
* Given na nakainstall na yung Python 3, type niyo yung command na:
* **python3 get-pip.py**
* Yung file na **get-pip.py**, galling din yan sa Python na website, irurun lang natin yan para mainstall yung Python Package Index. Kailangan natin yun para sa pip commands para mainstall yung mga libraries na nasa Python gamit lang yung pip commands. Mas madali yun kaysa mag dodownload pa tayo sa web browser tas iinstall natin manually. Kasi pag galling sa pip, siya na mismo maghahanap ng version ng library na compatible sa computer natin tas siya na maglalagay kung san folder man siya dapat nakalagay.
* Mag-iinstall lang yan, wait niyo. Tas pas tapos na, type niyo sa command line:
* **pip**
* Dapat may lalabas na parang usage ng command, parang ganto:



* Pag walang lumabas na ganyan pero nakainstall naman yung pip, ganto gagawin niyo. Punta kayo sa directory kung san nakainstall yung Python 3 tapos hanapin niyo yung folder na **Scripts**. Example sa baba. Pero pag may lumabas na, skip niyo na to, next page na kayo dun sa Installing Required Libraries.



* Open niyo yung Scripts na folder tas copy niyo yung path ng directory na yan, click niyo lang yung nasa may taas na list ng directory path para makita niyo yung complete directory path. Parang ganto:



* Tas open niyo uli command line tas change directory kayo dun gamit yung path kung asan yung file niyo. Yung sakin ganto kaya yung command ko ganto:
* cd C:\Users\windows\AppData\Local\Programs\Python\Python36-32\Scripts
* Tas dyan natin tatawagin yung pip na command.
* Type niyo uli **pip**. Dapat may lalabas na command usage.
* Dyan kayo mag ttype ng pip na command pag gagamitin niyo.

**Installing Required Libraries**

* Open command line tas type niyo command **pip3 pyserial**.
* Remember may 3 dun sa pip kasi Python 3 gamit natin.
* Pag ayaw ng pip3, pip na lang gamitin niyo, automatic na pang Python 3 ang iinstall ng pip kasi wala namang Python 2 dyan sa laptop niyo diba? Pag meron, dapat pip3 lalagay niyo. Sa Raspberry Pi kasi dalawang version ng Python nakainstall kaya magkaiba yung pip saka pip3. Pero pag isang version ng Python lang nakainstall sa inyo, automatic pang version na yun na yung iinstall ng pip.
* Take in mind also na pag ayaw nung pip gumana, dun kayo sa directory kung san nakalagay yung pip. Backread na lang sa taas.

Ngayon my pyserial na kayo. Ang gamit ng library na to is magsend and receive ng data from serial port. Every port sa computer natin may pangalan, yung mga USB port natin, COMn pangalan niyan, where n is a positive integer. Pag sa Raspberry Pi, iba yung pangalan ng USB port kasi Linux OS yan, yung COM pang Windows lang. Mamaya tayo sa RPi.

Windows muna tayo, download niyo yung dalawang program sa drive natin, yung **Receive-from-RPi.ino** saka **RPi-to-Arduino.py**. Sa Windows muna natin to tatry, wag kayo malito sa pangalan ng file na may RPi. Heavily commented yung mga code dun sa dalawang program, basahin iyo explanation, magegets niyo yan.

Una niyong gawin, magconnect muna kayo ng LED sa pin3 ng Arduino. Pin3->Resistor->GND ng Arduino. Pang test lang to, ang gagawin kasi natin, pag nagsend ng data yung Python program sa Arduino, babasahin ng Arduino yung sinned niyang data tapos gagamit tayo ng if-else statements para icompare yung value na sinend tapos gagawa tayo ng certain action pag tama yung narecive niyang data, in this case papablink muna tayo led.

Pag nawire niyo na yung led, upload niyo muna yung **Receive-from-RPi.ino** sa Arduino niyo. Pro Mini meron kayo diba. Tas run niyo yung **RPi-to-Arduino.py.** Pag naginstall kayo ng Python dun sa link na sinend ko, automatic kasama na dun yung Python IDLE, editor yun ng Python source code. Yun na lang din gamitin niyo sa RPi, nakainstall na din, wag na yung NinjaIDE. Pwede niyo irun via command line uli, change directory tas command **python3** **RPi-to-Arduino.py**. O kaya open niyo yung file by right clicking it and choose open with IDLE. Sa laptop niyo na Windows to ah.

Ang ginagawa lang ng Python program is nagsesend lang ng string tas babasahin ng Arduino tas icocompare niya kung dapat ganto ba yung nareceive niya. If tama yung nareceive niya, edi gagawa lang siya ng certain action which is blink led para sa example natin.

At this point, gumana na yun, try natin sa RPi. Ganun lang din, icoconnect lang natin yung USB cable ng Arduino papuntang RPi. Pero eedit natin yung program kasi hindi COM ang pangalan ng USB port ng Raspberry Pi. Wag niyo muna connect, sundan niyo tong steps:

* Open niyo muna RPi niyo tas buksan niyo command line.
* Type niyo command:
* **ls dev/tty/\***
* Ang ginagawa lang ng command na yan is ililist yung ports na active sa RPi. Marami lalabas dyan, di ko matry wala akong RPi ngayon.
* Pag katapos niyo itype yung command na yan, iconnect yung Arduino USB cable sa RPi. Tas ulitin niyo uli yung command.
* Pansinin niyo dapat may madadagdag dun sa current list ng devices, ayun yung name ng port na dapat iedit niyo sa Python program.
* Example kung ang pangalan ng port is **/dev/ttyACM0**. Edi yung line sa program na **port=serial.Serial(‘COM8’,9600)** is babaguhin natin. Gagawin nating:
* **port=serial.Serial(‘/dev/ttyACM0’,9600)**.
* Yun dat nagbiblink na uli yung led every 5 seconds pag nirun natin as indicated dun sa program natin.

At this point:

* Nakakapagsend na kayo ng data from RPi to Arduino.
* Nakakapagcompare nakayo ng data sa Arduino na nareceive from RPi tas magpeperform ng certain action pag tama yung nareceive.
* Nababsa niyo na yung dalawang sensor niyo.
* Napagana niyo na yung led strips niyo.

Ngayon, ang gusto ko gawin niyo is pagsamahin lahat ng program saka circuit na alam niyo na ngayon.

* Yung **WorkingCombinedV2.py** sa Step 1 sa drive natin, saka yung pagsend ng data from RPi to Arduino na **RPi-to-Arduino.py**, pagsamahin niyo na, pH value yung isend niyo sa Arduino. Remember na may /n sa dulo. Eh yung pHValue dun sa **WorkingCombinedV2.py**is float yung datatype kasi cinonvert natin, balik niyo sa string ulit tapos mag add kayo ng /n bago niyo isend sa Arduino. Tas yung led strips na sample code na meron kayo, icombine niyo sa **Receive-from-RPi.ino**. Ang gusto ko gawin niyo is red and kulay ng strips pag 1-5 ang range ng pH, green pag 6-10 tapos blue pag 7-14. Basahin niyo yung **Receive-from-RPi.ino**, yung variable dun na pinangcocompare ko sa pangalan ni Jezza  xD, is naka String yung format. Gawa kayo ng bagong variable bago kayo magcompare na:
* **float phReading=dataIn.subString(0,-1).toFloat**
* Tas ayun gamitin niyo sa if-else statements.
* Ang ginawa lang niyan is may bago kayong variable na float ang data type. Iseset natin siya sa nareceive na data pero icoconvert natin yung data sa float din. Ang ibig sabihin ng substring dyan is kukunin ko lang yung certain part ng data, yung unahan hanggang sa dulo minus 1. Kasi yung dulo is /n, gusto ko tanggalin yung dulo. Considered na one character lang ang /n kasi special character yan na meaning new line. Kahit two character yan “/” saka “n”, one lang siya pag pinagsama.
* Yun lang good luck, need niyo mahirapan hehehehe . Assist ko na lang kayo bukas via FB, di pa ako makakapasok eh.