Terminal de mac es Linux.   
y profesionales usan Linux.   
Raspberry es Linux.

WSL –Windows subsystem for Linux: (Subsistema de Windows para Linux)  
a partir de la powershell:  
>**wsl --install –d Debian**  
Pero molts hem tingut problemes, reiniciant, errors, entrar com administrador…  
Quan al escriptori fas click dret amb control i shift s obre la terminal de Linux.

WSL no tiene subsistema gráfico. Xcode es necessary per fer apps Linux.

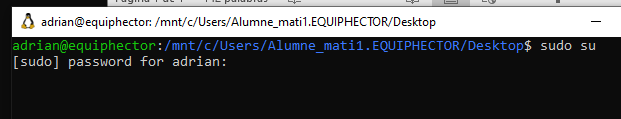
Githubs per fer projectes i publicitar-se.

MAC tiene OS (sistema operativo para bebes) para no perder tiempo en sistema operativo, es más para producción. Electrònica és millor en mac.

Programacio amb fils s’utilitza molt, un programa que fa 2 o 3 coses alhora, varios procesos alhora. Es fa servir dintre la robotica pero tambe es pot fer servir fora de la robotica.

Marenostrum farà IA i ordenadors quantics.

Linux commands:  
>**sudo su**   
superusuario



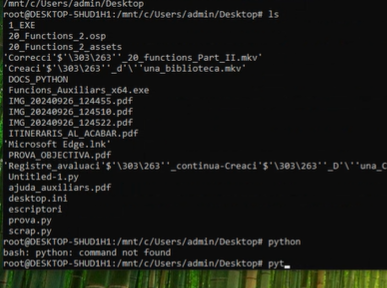
Ara estem com a root.



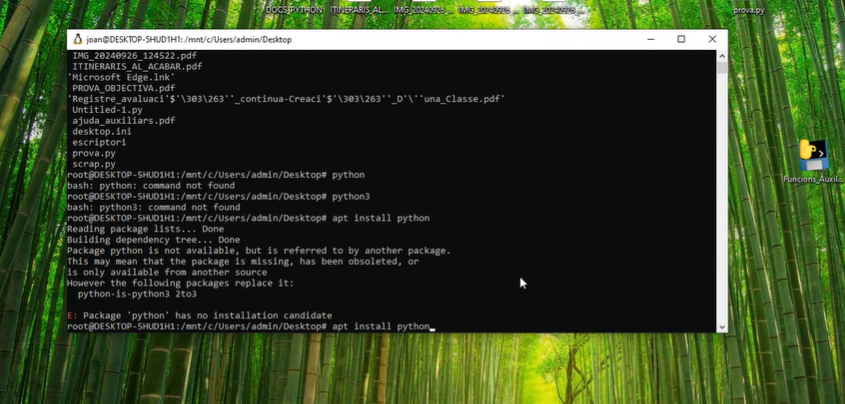
(Reset WSL password Windows10:   
>**wsl –u root**  
entrem a la terminal com a root)

>**pwd  
(print working directory  
estem al escriptori)**

**>ls  
(lista tot el que hi ha al escriptori)**

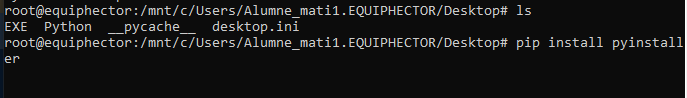


Instalar Python

  
>**apt update**  
(actualitza catàleg de programes disponibles desde’ls repositoris)

>**apt install python3**  
(instala el Python3 i dependències a Linux Debian (WSL))

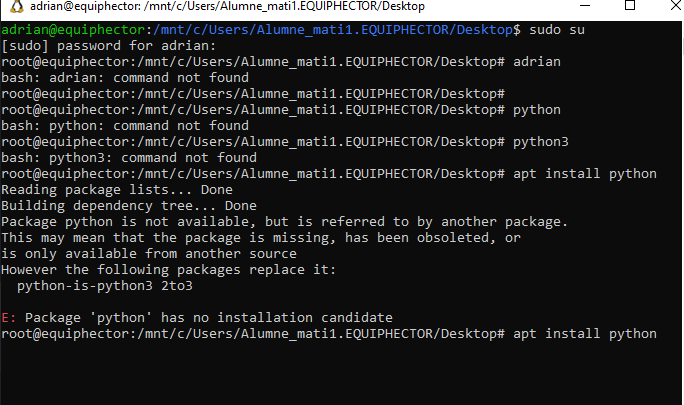
Podem agafar programa i executar-lo desde linux.



apt install python3-pip  
(installa l instalador de Python)

Ambos son instaladores:  
**apt** es instalador de Linux  
**pip** es de Python  
pip install pyinstaller

Ara podem executar codis de Python desde la terminal de Linux posant al escriptori els codis per exemple (el working directory del wsl és l escriptori).   
Podem per tant posar:  
>**python3 nom\_de\_arxiu.py**



Aquest Linux n te entorn grafic de momento.

>**ls**  
(per llistar tot el que hi ha a l escriptori)

>**ping 8.8.8.8**   
(per saber si tens conexio a internet si et dona resposta 64 bytes o 32 bytes).

>**^C**   
Tanca el proces que s estigui fent, per exemple ping fins l infinit. ^C l’atura.

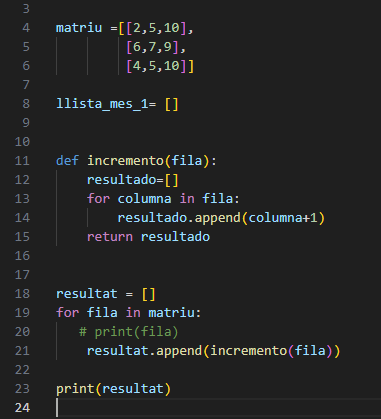
Linux de forma natural acepta millor les dependencies que Windows.

Sapiguer recorrer matrius: llistes de llistes, cadenes de cadenes…   
normalment requereixen un for dintre un bucle for. Millor fer una funcio i despres fer un bucle for amb la funcio.

Expercici Python llista de lllistas:

'''Despres de passar pel codi, tindrem la llista amb tots els elements incrementats amb '''

MODIFICAR UNA FILA NO SE PUEDE A NO SER QUE SE HAGA CON RANGE.



def incremento(fila):

    resultado=[]

    resultado = [element +1 for element in 1] #equivale al bucle for

    return resultado

(**list comprehension**)

resultat = []

for fila in matriu:

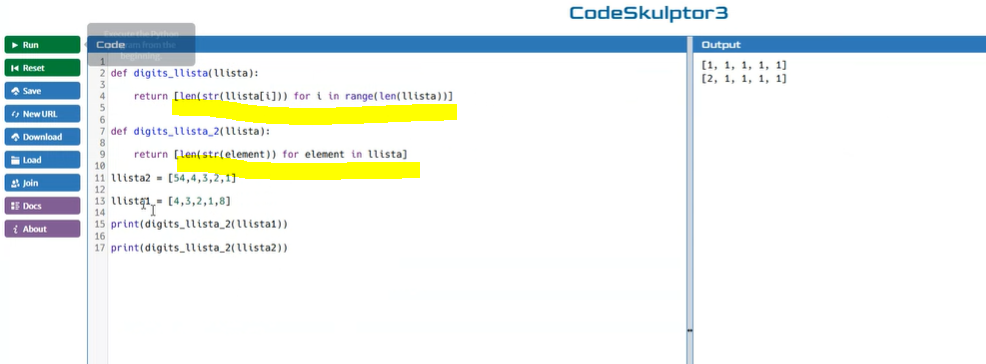
   # print(fila)

    resultat.append(incremento(fila))

resultat= [incremento(fila) for fila in matriu]

incremento(fila) dice como quieres modificar la matriz.

**LIST COMPREHENSION**: A) es lo mismo que B) (sólo sirve con listas y sólo retorna lista)  
A)  
resultat =[]  
for element in llista:  
 resultat.append(element\*\*2)

B)  
**resultat=[element\*\*2 for element in llista]  
aleatorios = [random.randint(1,1000) for contador in range(100)]  
return [len(str(llista[i])) for i in range(len(llista))]  
return [len(str(element)) for element in llista]**aquí RECORREM elements de la llista no les posicions.

A google.com:   
“**List comprehension site:Python.org**” (perque dins de Python.org es busqui el terme list comprehension).  
\_-------

HEM PARLAT DE:  
LLISTES (**LIST**)  
STRINGS(**STR**)  
**SET** (colleccio de objectes no repetits)

Llista =[1,1,3,4,4,5]

**set(llista)**   
(surten en curly brackets {1,3,4,5})

{1,3,4,5}   
(és desordenat (Unordered) pero podem fer -🡪  
**list(set(llista))** ho pots ordenar pero cada cop será un ordre diferent, perque set t’ordena com vol els elements (diferents maneres de ordenar una mateixa llista)