

Guia de trabajo para deteccion con yolo en raspberry pi 4

1 Sistema operativo

1.1 Requisitos

- Conexión a internet
- Memoria micro SD de 16GB, recomendado 32GB o más
- Computador con lector micro SD o adaptador SD

1.2 Procedimiento instalación

1. Instalación del raspberry pi imager
2. Selección del sistema operativo a instalar
3. Selección de la unidad donde se desea instalar

1.2.1 Intalación del Raspberry Pi Imager

Para el uso de la raspberry pi debemos obtener un sistema operativo y para ello debemos descargar el Raspberry Pi Imager(instalador); para eso nos dirigimos al siguiente link: <https://www.raspberrypi.com/software/>; una vez descargado e instalado el programa podemos ver una interfaz sencilla



Figure 1: Interfaz raspberry pi imager

1.2.2 Selección del sistema operativo

Ahora bien debemos seleccionar el sistema operativo con el que deseemos trabajar: Raspberry pi OS (Raspbian), Ubuntu, Manjaro. Se opto por usar Raspberry pi OS con entorno de escritorio

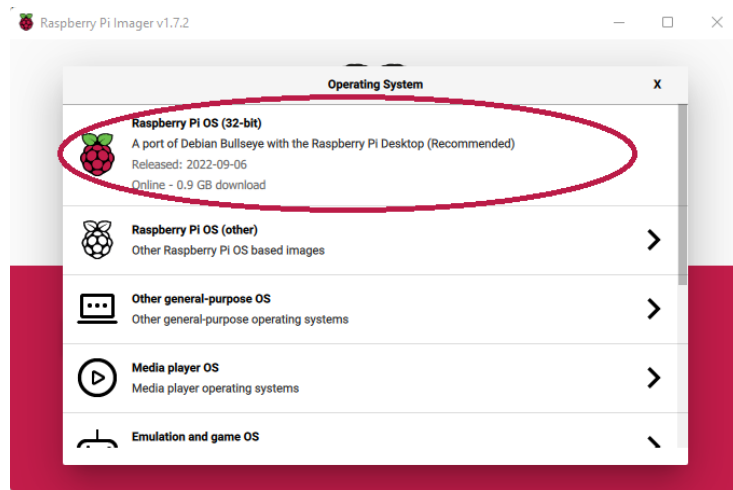


Figure 2: Interfaz raspberry pi imager

1.2.3 Selección de la unidad

Luego de seleccionar el sistema operativo con el cual deseamos trabajar, se debe seleccionar el lugar donde queremos subirlo, para eso debemos insertar la memoria micro SD al PC y en la interfaz del instalador seleccionarla

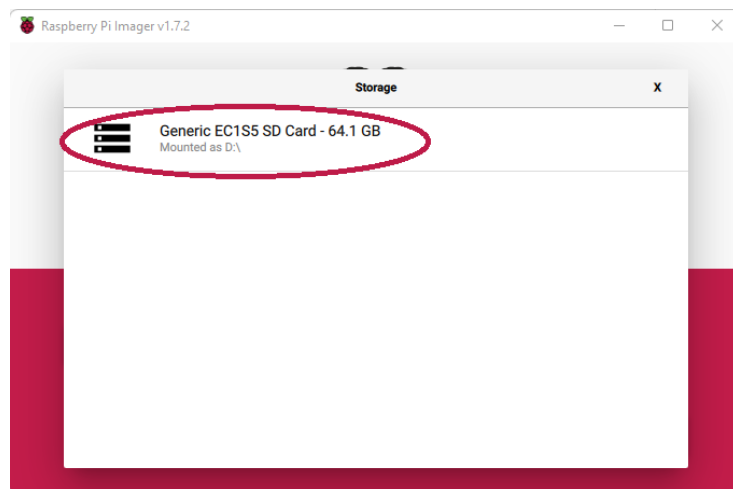


Figure 3: Interfaz raspberry pi imager

Por último se debe dar click en "WRITE" para poder montar el sistema operativo en la micro SD; el proceso de instalación comenzará, se espera un momento sin retirar la SD del PC; ya al terminar el proceso la memoria estará lista para insertar en la Raspberry pi y comenzar el proceso de configuración del sistema.

2 Python

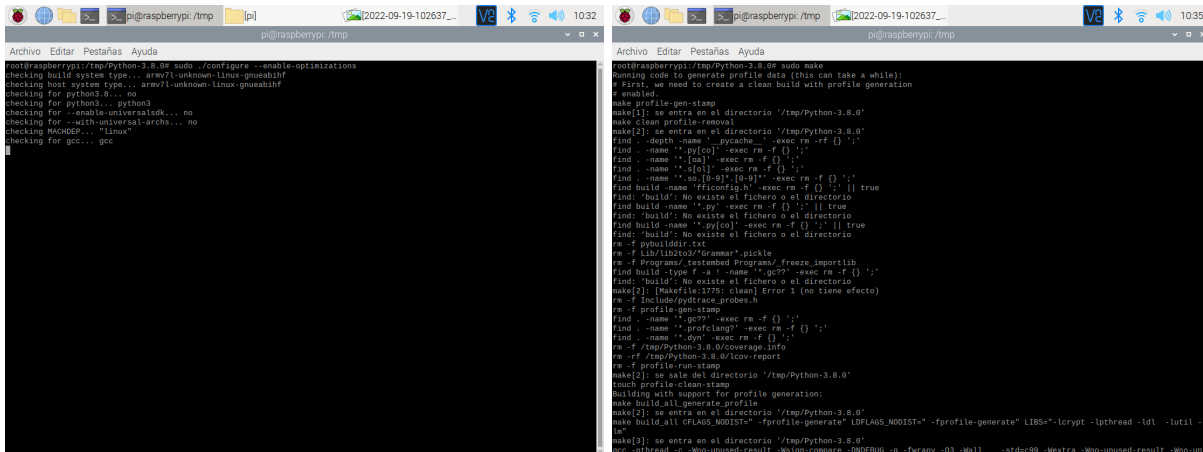
Para la instalación o actualización de python en la raspberry se debe abrir el terminal y seguir los siguientes pasos:

- Instalar los pre-requisitos

```
1 sudo apt-get update
2 sudo apt-get install build-essential tk-dev libncurses5-dev libncursesw5-dev libreadline6-dev libdb5
  .3-dev libgdbm-dev libsqlite3-dev libssl-dev libbz2-dev libexpat1-dev liblzma-dev zlib1g-dev
```

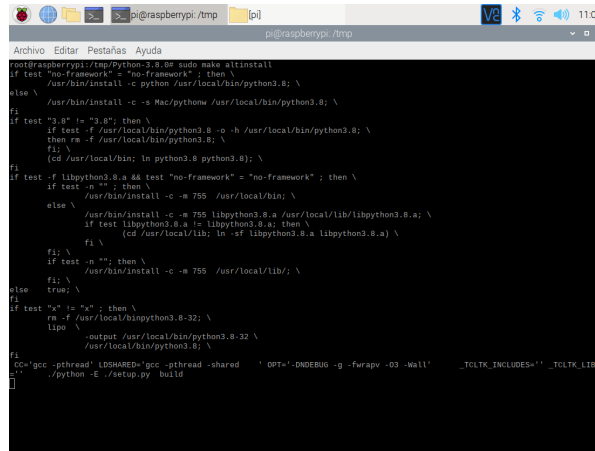


```
3 sudo make altinstall
```



(a) Interfaz raspberry pi imager

(b) Interfaz raspberry pi imager



(c) Interfaz raspberry pi imager

- Una vez terminadas las compilaciones e instalaciones podemos comprobar que tenemos la última versión de Python ejecutando:

```
1 python3.8 --version
```

```

changing mode of /usr/local/lib/python3.8/lib-dynload/grp.cpython-38-arm-linux-gnueabihf.so to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/_struct.cpython-38-arm-linux-gnueabihf.so to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/_bisect.cpython-38-arm-linux-gnueabihf.so to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/_sha3.cpython-38-arm-linux-gnueabihf.so to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/resource.cpython-38-arm-linux-gnueabihf.so to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/_lib.cpython-38-arm-linux-gnueabihf.so to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/_decimal.cpython-38-arm-linux-gnueabihf.so to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/_parser.cpython-38-arm-linux-gnueabihf.so to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/_xtest fuzz.cpython-38-arm-linux-gnueabihf.so to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/_ to 755
changing mode of /usr/local/lib/python3.8/lib-dynload/_pycache_ to 755
running install scripts
copying build/scripts-3.8/pydoc3.8 -> /usr/local/bin
copying build/scripts-3.8/2to3-3.8 -> /usr/local/bin
copying build/scripts-3.8/idle3.8 -> /usr/local/bin
changing mode of /usr/local/bin/pydoc3.8 to 755
changing mode of /usr/local/bin/2to3-3.8 to 755
changing mode of /usr/local/bin/idle3.8 to 755
rm -rf /usr/local/lib/python3.8/lib-dynload/_sysconfigdata__linux_arm-linux-gnueabihf.py
creating directory /usr/local/share/man/man1
/usr/bin/install -c m 644 /usr/local/share/man/man1/python3.8.1
if test "xupgrade" = "xno" ; then \
    case upgrade in \
        upgrade) ensurepip="--altinstall --upgrade" ;; \
        install") ensurepip="--altinstall" ;; \
    esac; \
    python -E -m ensurepip \
        --root=/ ; \
fi
Looking in links: /tmp/pip17u4kn1
Collecting setuptools
Collecting pip
Installing collected packages: setuptools, pip
Successfully installed pip-19.2.3 setuptools-41.2.0
root@raspberrypi:/tmp/Python-3.8.0# python3.8 --version
python 3.8.0
root@raspberrypi:/tmp/Python-3.8.0# pip3 --version
pip 19.2.3 from /usr/local/lib/python3.8/site-packages/pip (python 3.8)
root@raspberrypi:/tmp/Python-3.8.0#

```

Figure 7: Caption

- Ahora queremos que nuestro python 3.8 quede por defecto en python3 entonces debemos ejecutar:

```

1  which python3.8
2  sudo update-alternatives --install /usr/bin/python3 python3 /usr/local/bin/python3.8 1
3  sudo update-alternatives --install /usr/bin/python3 python3 /usr/local/bin/python3.8 2
4  sudo update-alternatives --config python3
5  ls -alith /usr/bin/python3
6  python3 -V
7  ls -alith /usr/bin/pip3
8  pip3 -V

```

```

root@raspberrypi:/tmp/Python-3.8.0# alias python3=python3.8
root@raspberrypi:/tmp/Python-3.8.0# sudo update-alternatives --install /usr/bin/python3 python3 /usr/local/bin/python3.8 1
update-alternatives: error: la ruta alternativa /usr/bin/python3.8 no existe
root@raspberrypi:/tmp/Python-3.8.0# python3 -V
python 3.8.0
root@raspberrypi:/tmp/Python-3.8.0# which python3.8
/usr/local/bin/python3.8
root@raspberrypi:/tmp/Python-3.8.0# sudo update-alternatives --install /usr/bin/python3 python3 /usr/local/bin/python3.8 1
update-alternatives: utilizando /usr/local/bin/python3.8 para proveer /usr/bin/python3 (python3) en modo automatico
root@raspberrypi:/tmp/Python-3.8.0# sudo update-alternatives --install /usr/bin/python3 python3 /usr/local/bin/python3.8 2
root@raspberrypi:/tmp/Python-3.8.0# sudo update-alternatives --config python3
Sólo hay una alternativa en el grupo de enlaces python3 (provee /usr/bin/python3): /usr/local/bin/python3.8
Nada que configurar.
root@raspberrypi:/tmp/Python-3.8.0# ls -alith /usr/bin/python3
99105 lrwxrwxr-x 1 root root 25 sep 19 11:29 /usr/bin/python3 -> /etc/alternatives/python3
root@raspberrypi:/tmp/Python-3.8.0# python3 -V
Python 3.8.0
root@raspberrypi:/tmp/Python-3.8.0# ls -alith /usr/bin/pip3
9113 -rwxr-xr-x 1 root root 303 abr 11 2019 /usr/bin/pip3
root@raspberrypi:/tmp/Python-3.8.0# pip3 -V
pip 19.2.3 from /usr/local/lib/python3.8/site-packages/pip (python 3.8)
root@raspberrypi:/tmp/Python-3.8.0#

```

Figure 8: Caption

Y listo, ya tendríamos instalada o actualizada la version de python3 a la version python 3.8