**TO INSTALL PYTHON 3 ON RASPBERRY PI**

## PREREQUISITES:

Before installing python 3.7 there are some dependencies that we need to install. Use the following command to install the required dependencies.

sudo apt-get install -y 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 libffi-dev

If one of the packages cannot be found, try a newer version number (e.g. libdb5.4-dev instead of libdb5.3-dev).

### 1. Download Python:

You can download Python from the official **website** or use the following command.

wget https://www.python.org/ftp/python/3.7.0/Python-3.7.0.tgz

### 2. Install Python 3.7 On Raspberry Pi

Now we will extract and install Python from source.

sudo tar zxf Python-3.7.0.tgz  
cd Python-3.7.0  
sudo ./configure  
sudo make -j 4  
sudo make altinstall

You can use the newly installed python version by calling pip3.7 install –e.  and python3.7 example.py respectively.

**3. Convert the version of python 2 to python 3**

First The Pi (Raspbian) comes pre-installed with 2 versions of Python.

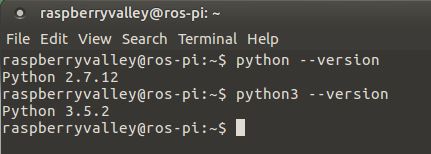
First of all, check which versions of Python your system has. Just type the following:

python --version

python3 --version

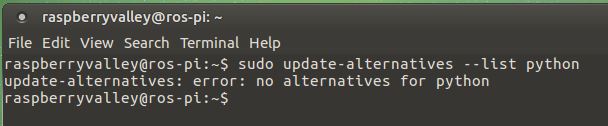
You'll get something similar to this. Note the versions available (first 2 digits only)

You'll get something similar to this. Note the versions available (first 2 digits only)



* We will use the **update-alternatives** command. First, we will list the available alternatives (most probably, you will get an empty list as seen in the screenshot below)

sudo update-alternatives --list python

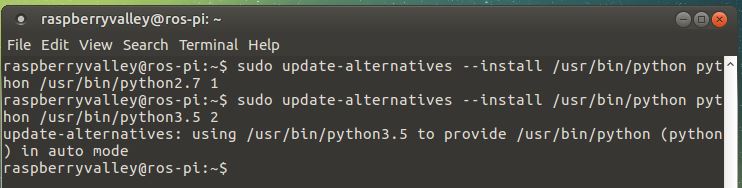


* Update the alternatives table to include both Python 2.7.x and Python 3.5.x. Make sure to substitute the versions of Python in the command below with the versions of your system, which you have noted in the first step. Notice we use only the first 2 digits of the version (i.e. we ignore the 3rd version number part)

sudo update-alternatives --install /usr/bin/python python /usr/bin/python2.7 1

sudo update-alternatives --install /usr/bin/python python /usr/bin/python3.5 2

If successful, you will end up with output similar to this.



The **--install** option take multiple arguments from which it will be able to create a symbolic link. The last argument specified is priority. That means, if no manual alternative selection is made the alternative with the highest priority number will be set.

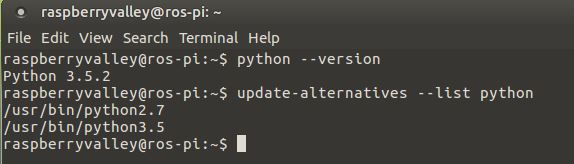
* Check the result. Type the following:

python --version

You should see Python 3 as preferred version (due to the priority setting in the priorities table)

Similarly to above, you can also list the priorities table to confirm:

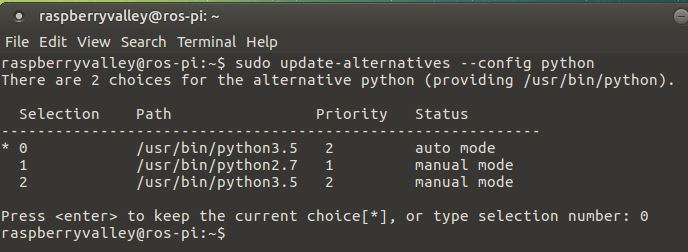
update-alternatives --list python



## Switching default Python versions

If you have followed the guide above, you can from now on switch preferred Python versions anytime. Just invoke the command below and make your choice.

sudo update-alternatives --config python



## Cleanup

One day, your Python version(s) will change. To remove an entry from the alternatives table, simply type something like this (we will remove the table entry for Python 3.5 in this example):

sudo update-alternatives --remove python /usr/bin/python3.5

**INSTALLING PYTHON LIBRARIES**

Not all Python packages are available in the Raspbian archives, and those that are can sometimes be out-of-date. If you can't find a suitable version in the Raspbian archives, you can install packages from the [Python Package Index](http://pypi.python.org/) (PyPI). To do so, use the pip tool.

pip is installed by default in Raspbian Desktop images (but not Raspbian Lite). You can install it with apt:

sudo apt install python3-pip

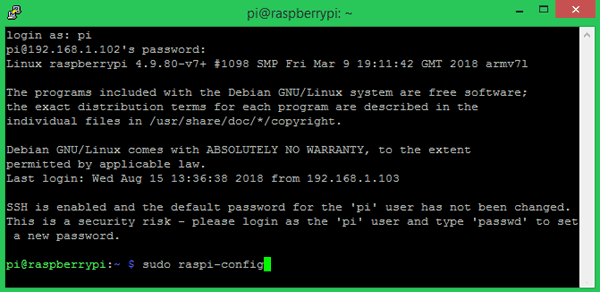
pip3 installs modules for Python 3

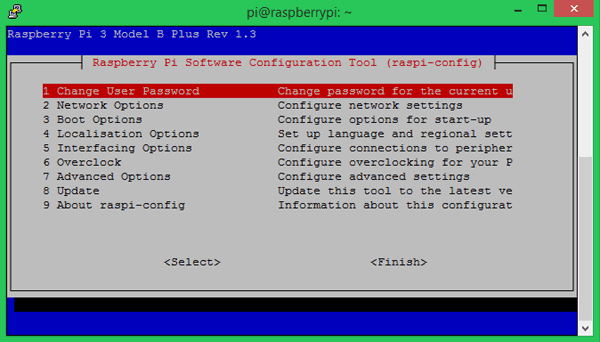
### **Configuring the Raspberry Pi for LoRa module**

As told earlier the LoRa module works with SPI communication, so we have to enable SPI on Pi and then install the ***pylora*package**. Follow the below steps to do the same, after opening the terminal window of Pi. Again, I am using putty to connect to my Pi you can use your convenient method.

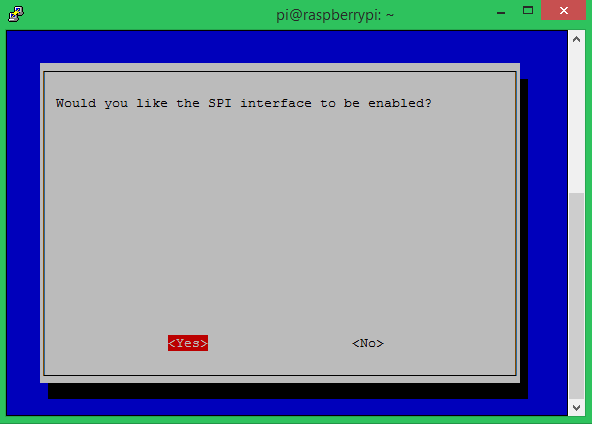
**Step 1:** Get into the **configuration window** using the following command. To get the below window

**sudo raspi-config**





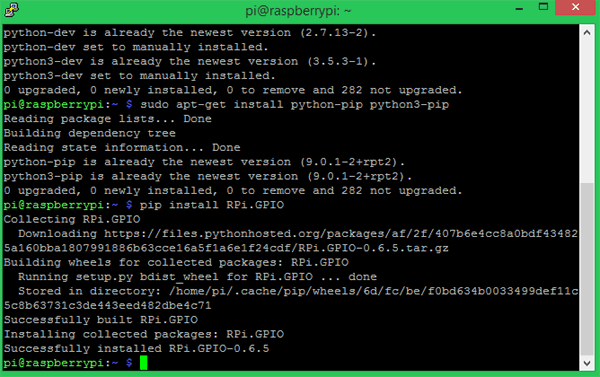
**Step 2:**  Navigate to interfacing options and enable SPI as shown in the image below. We have to **enable the SPI interface**because as we discussed the LCD and PI communicates through SPI protocol



**Step 3:**Save the changes and get back to the terminal window. Make sure pip and python is updated and then **install the *RPi.GPIO* package** using the following command.

**Pip3 install RPi.GPIO**

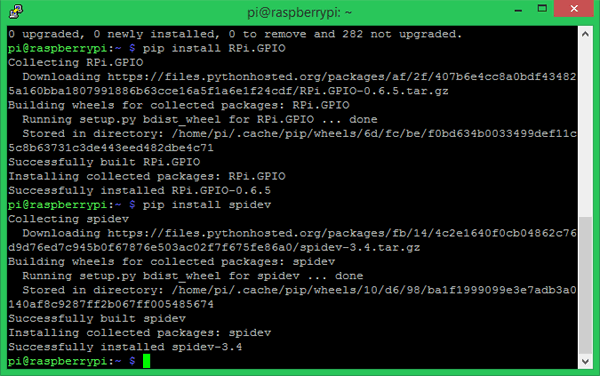
This package class will help us control the GPIO pin on the Pi. If successfully installed your screen will look like this



**Step 4:**Similarly proceed with installing the ***spidev* package** using the following command. Spidev is a python binding for Linux which can be used to perform SPI communication on Raspberry Pi.

**Pip3 install spidev**

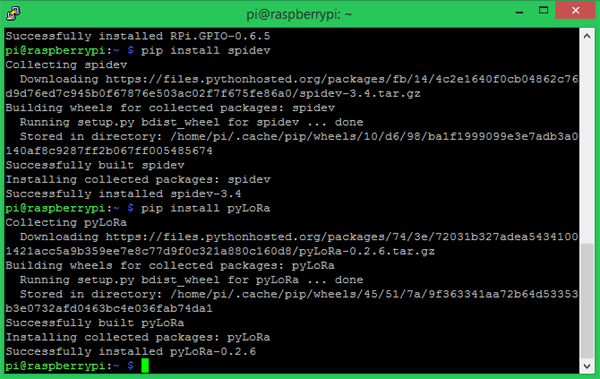
If the installation is successful the terminal should look something like this below.



**Step 5:**Next lets **install the pyLoRa package** using the following pip command. This package installs the Radio models associated with LoRa.

**Pip3 install pyLoRa**

If the installation is successful you will see the following screen.



**Step 6:**Download and **install the python-rpi.gpio package and spidev package** using the below command.

**sudo apt-get install python3-rpi.gpio**

**sudo apt-get install python3-spidev**

