Installing Python 3.6 on Raspberry Pi 4

Objective

This document provides step-by-step instructions for installing Python 3.6 on a Raspberry Pi 4 and setting it as the default Python version. As python 3.11 which will be already installed on the rasp-pi OS, project requirements will be completed with python 3.6.

Prerequisites

- Raspberry Pi 4 running Raspbian OS (or a compatible distribution)
- Internet connection
- It is a sequential installation process so by chance you get any error then **do not** proceed ahead without solving it. You might face an issue while making your installation setup ready for python source, for this you can just find an alternate source for installation.

Steps

1. Check Available Python Versions

Run the following command to check the available Python versions installed on your Raspberry Pi:

```
ls /usr/bin/python*
```

This will list all installed Python versions.

2.Install Python 3.6

```
sudo apt-get update
```

```
sudo apt-get install -y build-essential zlib1g-dev libncurses5-dev libgdbm-dev libnss3-dev libssl-dev libreadline-dev libffi-dev libsqlite3-dev wget libbz2-dev
```

```
wget https://www.python.org/ftp/python/3.6.15/Python-3.6.15.tgz
```

```
tar xzf Python-3.6.15.tgz
```

cd Python-3.6.15

./configure

Make

sudo make install

cd ..

rm -r Python-3.6.15

rm Python-3.6.15.tgz

3. Update Alternatives

Use the `update-alternatives` command to set Python 3.6 as the default. Replace `<python3.6-path>` with the path to your Python 3.6 interpreter.

sudo update-alternatives --install /usr/bin/python3 python3 /usr/bin/python3.6 1

4. Configure Default Version

Run the following command to configure the default Python version and select Python 3.6:

sudo update-alternatives --config python3

This will display a list of available alternatives. Choose the number corresponding to Python 3.6.

5. Verify Default Version

After making the changes, verify that Python 3.6 is set as the default:

python3 --version

This should display the version information for Python 3.6. (Output will be>>python3.6)

If we get this output, then we are good to go ahead with installation of all the required python packages for the project.

Mavproxy Installation

The following instructions for installing dependencies for MAVProxy. Follow sequentially.

```
sudo apt-get install python3-dev python3-opencv python3-wxgtk4.0 python3-pip

 sudo apt-get install python3-matplotlib python3-lxml python3-pygame

pip3 install PyYAML mavproxy --user

echo 'export PATH="$PATH:$HOME/.local/bin"' >> ~/.bashrc
```

Drone-kit Installation

Linux requires you to prefix the command with sudo.

```
sudo apt-get install python3-pip python3-dev
sudo pip3 install dronekit
```

Upon completion now we need to verify the installation, so we will just run a simple python code to check the autopilot state. For this you can connect any Pixhawk autopilot to raspberry pi with USB cable

Code