**Setting Up RPI for Networking Room(s) for Surveillance**

List of Hardware

* Raspberry Pi® 3B+ Starter Kit (*comes with Micro SD Card, Reader, Case, Power Supply, Heat Sinks*)

**CanaKit** / $69.99 + Free Shipping (Amazon)

* Raspberry Pi Camera Module (2 Options)

**Arducam** *5 Megapixels 1080p Sensor OV5647* / $12.99 + Free Shipping (Amazon)

**Raspberry** **Pi** *Camera Module V2-8 Megapixel, 1080p* / $24.80 + Free Shipping (Amazon)

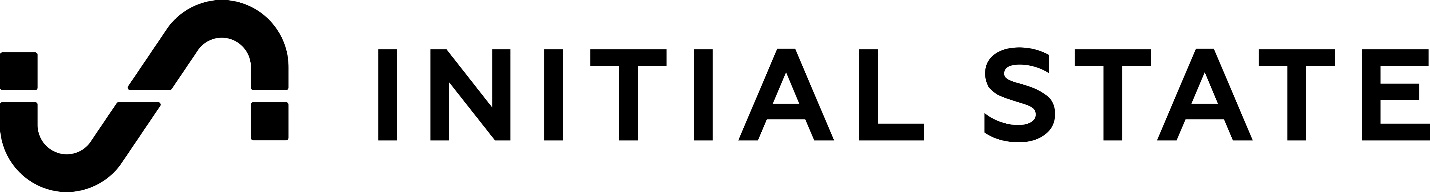
* DHT22 Temperature & Humidity Sensor

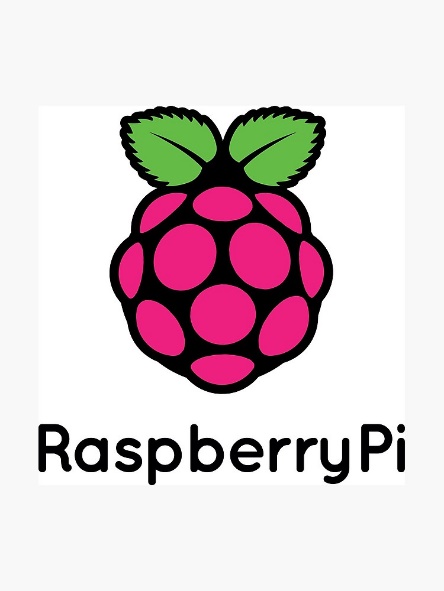
**Gowoops** 2 pk. DHT22 Temperature Humidity Sensor Module / $10.99

|  |  |
| --- | --- |
| **Item** | **Price** |
| CanaKit | $69.99 |
| Raspberry Pi V2 | $24.80 |
| Gowoops | $10.99 |
|  | **Total** |
| **Price Per Unit** | $105.78 |

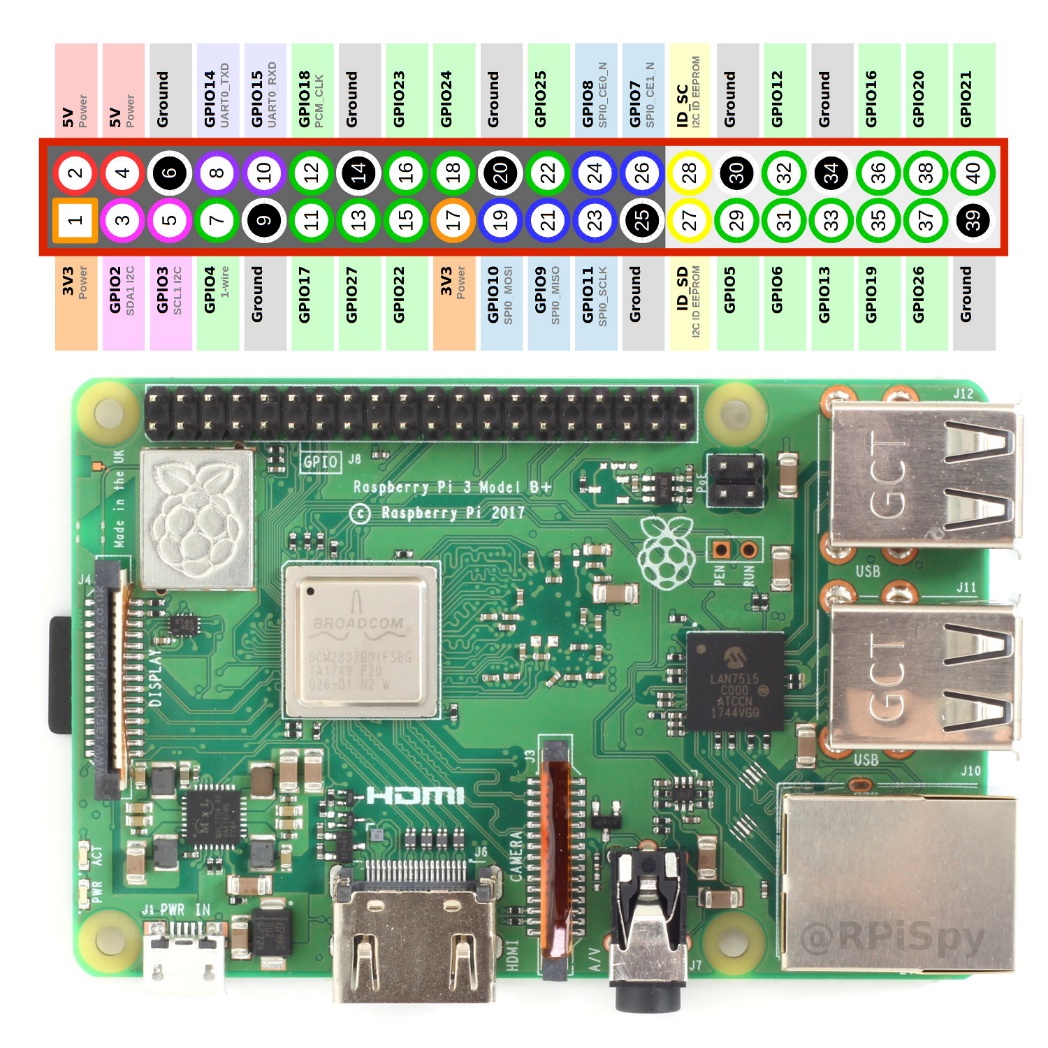
\*\*\****Included in this project will be a command sheet of nothing but commands and which component they refer to for installation. It is to benefit anyone who is familiar or fluent with Terminal™ commands and Python™ scripts. It will skip the explanations of services and be specifically commands and the script(s) for use****. (see at the bottom of document)\*\*\**

These Products will be everything you need to get started to set up the device in its entirety. The following will be a list of programs you will utilize to have GUI output for each component. The DHT22 Temperature and Humidity sensor will be using a free web-based GUI with Initial State™. The Camera Module you use will be utilizing the free web-based GUI MotionEye™.





**Raspberry Pi 3B+ / DHT22 / Initial State™**

****The Raspberry Pi 3B+ is a single board computer with:

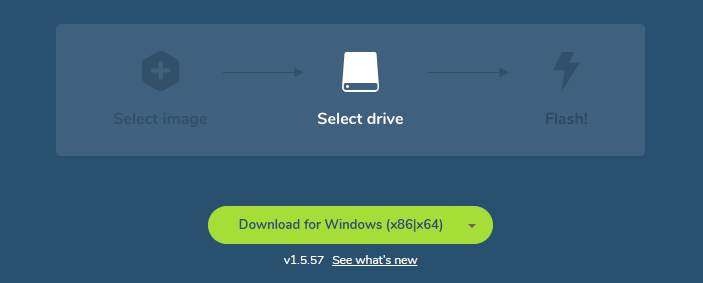
* 1 GB RAM
* 1 HDMI Slot
* 1 LAN Port
* 4 USB 2.0 Slots
* 1 Ribbon Cable Display Slot
* 1 Ribbon Cable Camera Slot
* 40 Header Pins
* 1 Type C USB Power Supply Slot

There is not a hard drive associated with the RPI (Raspberry Pi). The hard drive is the Micro SD card that is used to store the Raspbian image used as the OS for the RPI. Images for the RPI are available @ <https://www.raspberrypi.org/downloads/raspbian/>. You will need a Micro SD card reader to insert into your desktop to burn the image to. You will also need to download a program called Balena Etcher. You can get the free utility available @ <https://www.balena.io/etcher/>. In this project we will be using the latest Raspbian image as of September 2019 available which is Raspbian Buster; download the version “with desktop and recommended software”.

**Raspberry Pi 3B+ Computer with 40 Pin Header Reference Chart**

The steps to complete the project in its entirety will be as follows:

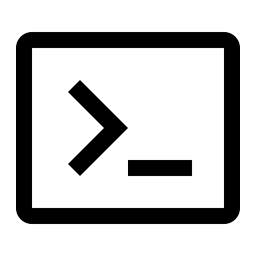
**Flash Image to Micro SD Card**

* Open a web browser window (Google Chrome, Firefox, Edge).
* Go to <https://www.balena.io/etcher/>
* Select Download for Windows (x86|x64) button to download the balenaEtcher™ utility.
* ****Select the download and install balenaEtcher™.

Once opened, you will select the Raspbian™ image that you downloaded, the drive you have inserted (Micro SD card), and select Flash. This will download and install the operating system image onto your Micro SD card that you will use to set up the RPI.

**balenaEtcher™ Download Button**

**Set up RPI** (Raspberry Pi)

* Connect the HDMI cable to TV or monitor and to the HDMI slot on the RPI.
* Connect a USB Mouse and Keyboard to the USB slots on the RPI.
* Connect a network cable to the LAN port on the RPI.
* Connect the power supply to the RPI.
* ****Power on the RPI.

**Updating RPI**

Open a **Terminal** window

Enter the following commands:

* sudo apt-get update

press **Enter** (*updates the RPI*)

* sudo apt-get upgrade

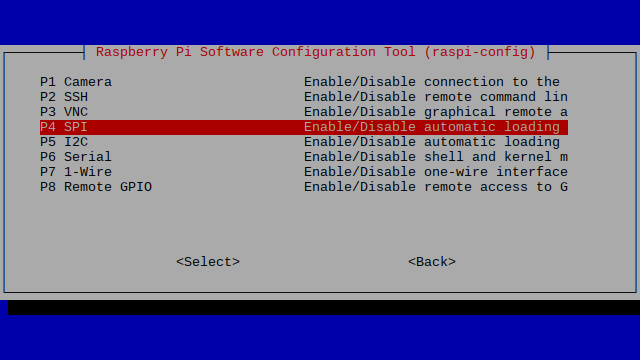
press **Enter** (*upgrades the RPI*)

Let the update and upgrade commands run 100%. Once they are completed, if necessary type in the following command:

**Linux Terminal Logo**

* sudo reboot
* press **Enter** (*reboots the RPI*)

(*Once the RPI has rebooted, you will need to obtain the IP Address if you want to access the RPI remotely from a desktop. Or you can use all of the commands directly on the RPI with the keyboard.*)

**Turning on the GPIO Header Pins**

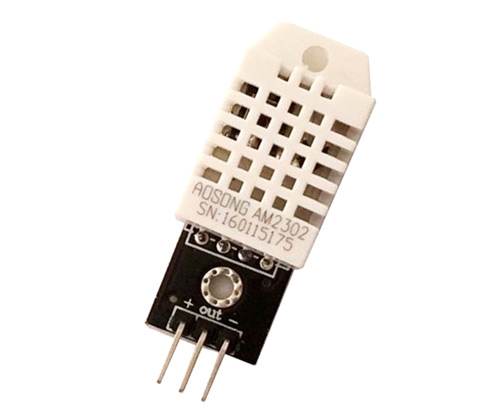
Open a **Terminal** window

Enter the following command:

* sudo raspi-config
* Press the down arrow until you highlight Interfacing Options in the menu.
* Once in the Interfacing Options Menu, which is in the image on the right, you will enable these options: Camera, VNC, SPI, I2C, Serial, 1-Wire. These options will enable all of the pins and features needed

**Raspi-Config GUI Interfacing Options users panel**

to complete the project. Once you have enabled all of the options, reboot the RPI with the reboot command listed above.

**Installing the DHT22**

Install the pins from the DHT22 Temperature and Humidity sensor onto the header of the RPI.

* The wire connected to the + on the sensor will connect to Pin 1 on the RPI.
* The wire connected to OUT on the sensor will connect to Pin 7 on the RPI.
* Lastly, the wire connected to the – on the sensor will connect to Pin 9 on the RPI.

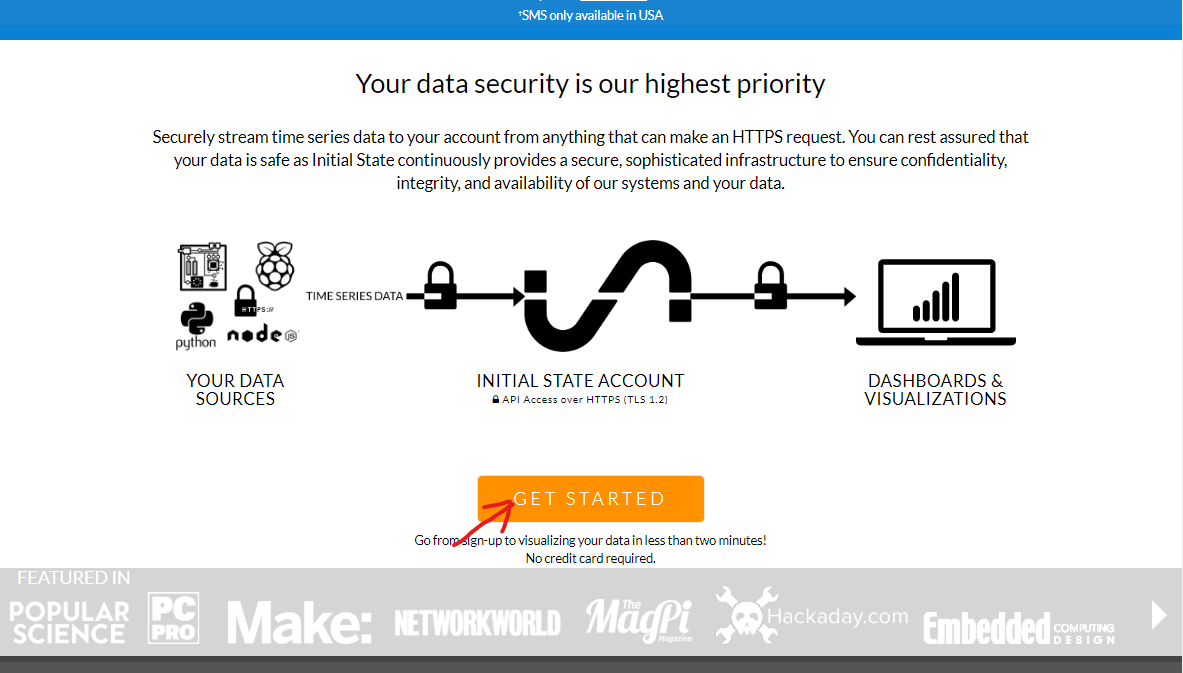
**DHT22 Temp and Humidity Sensor**

Once completed the installation of the DHT22 is completed. Next we will have to create an account with Initial State™ before we begin to run commands to install the Adafruit library and commands to make it functional.

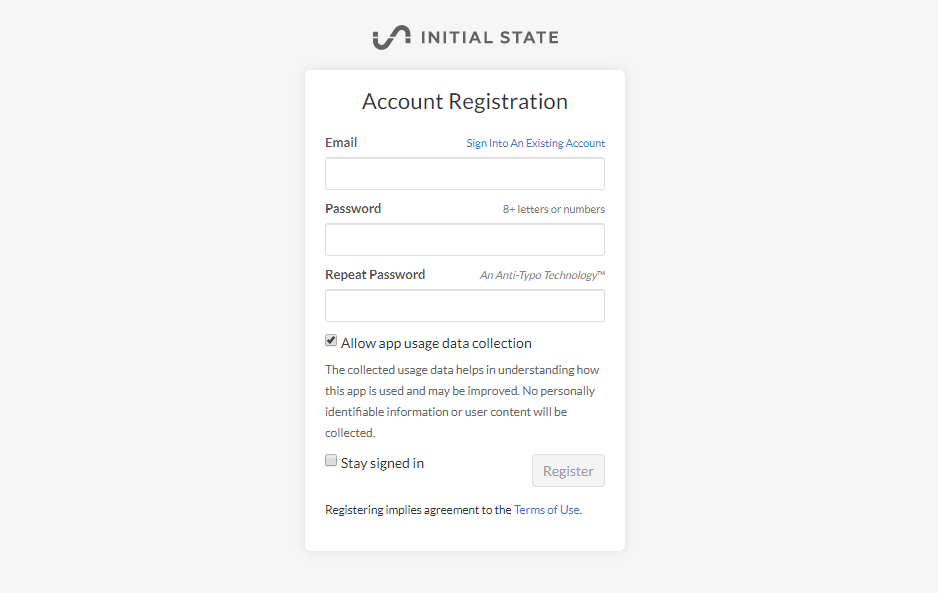
**Create Initial State™ account**

To create an account with **Initial State™,** go to the following link and view the steps below**:** <https://www.initialstate.com/>

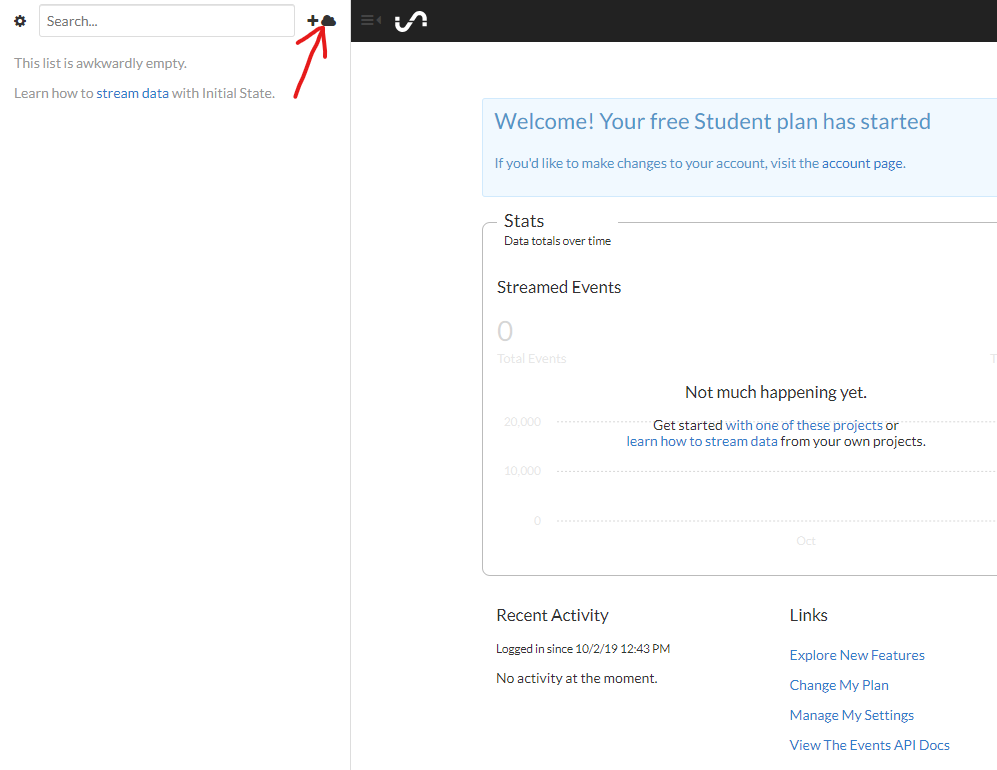
Sign up with a .edu email address for a free account, or choose the plan that best fits your needs. The free account is preferred and gives you the access you need for this project.

The next series of images will show and explain how to create the account with Initial State™ and locating the bucket key and access key needed to enter into the tempsensor.py file. 

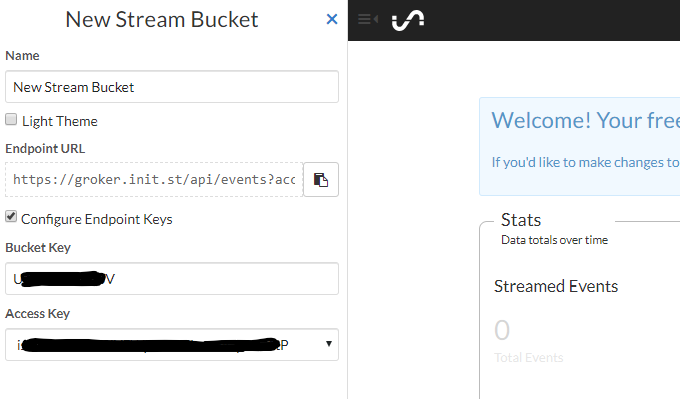
**When the website loads, scroll to the bottom of the screen and click the GET STARTED button.**



**Once you have selected GET STARTED, enter the information that you would like to create your account with. A .edu email address will give you the free student account which will be appropriate for this project. You will need to adjust and scale your plans accordingly once you have tested your device(s).**



**Next, you will click on the cloud image with the plus sign beside it. This will open new options and give you the ability to create a new Bucket Stream.**



**Once you click the +cloud button the menu in the image above will open and give you an Endpoint URL. To locate the Bucket Key and Access Key you must select the Configure Endpoint Keys checkbox and the list with the Bucket Key and Access key will Open. Document them both, they will be entered into the tempsensor.py script so the data from the DHT22 can be transferred to the GUI as output in real time. You can use this information to install more devices on the same Bucket Name. After you have them documented, scroll to the bottom of the page and select Create.**

The next steps we will do are going to give the DHT22 life and set it up for reading data and providing output of the data to the Initial State™ GUI.

**Commands for DHT22 Functionality**

Open a new **Terminal** window

Enter the following commands:

* sudo apt-get update

press **Enter**

* sudo pip install Adafruit\_DHT press **Enter**
* sudo pip install ISStreamer

press **Enter**

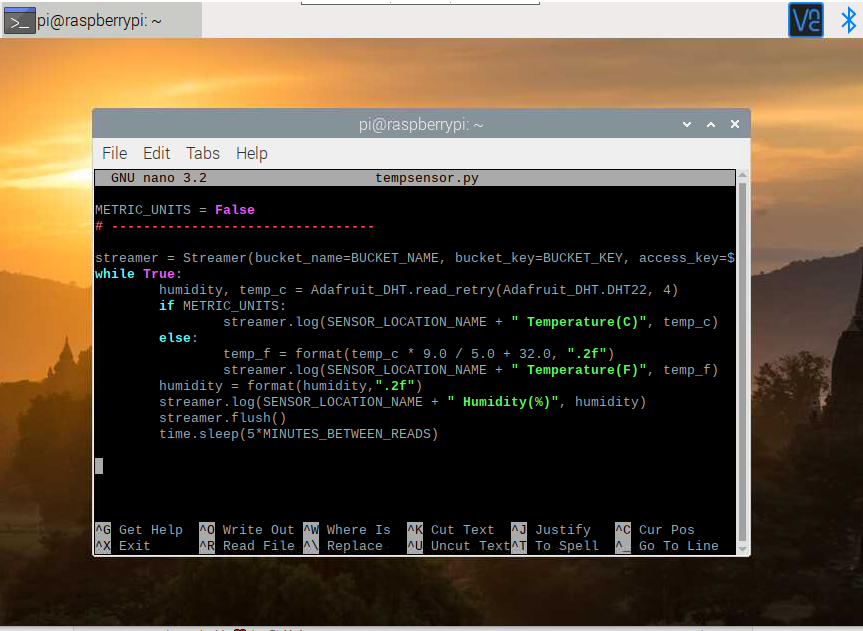
To create the python file named tempsensor.py, enter the following command:

* sudo touch tempsensor.py
* press **Enter**

Access the file by entering the following command:

* sudo nano tempsensor.py

press **Enter**

A window will open up to be able to insert the following Python™ script into the file as shown in the image above. The Python™ script will be needed to grab the data from the DHT22 and send it to the web-based GUI for viewing.

Once the file opens, it will be empty. If you are accessing the RPI remotely then you can simply copy and paste the following script. If not, then you will need to enter it manually. It must be entered as shown with no extra spaces, letter, symbols, etc. You will edit the document later, but this is the template script that is used.

**File editing window in Terminal** tempsensor.py

import Adafruit\_DHT

from ISStreamer.Streamer import Streamer

import time

# --------- User Settings ---------

SENSOR\_LOCATION\_NAME = "Office"

BUCKET\_NAME = ":partly\_sunny: Room Temperatures"

BUCKET\_KEY = "rt0129"

ACCESS\_KEY = "PLACE YOUR INITIAL STATE ACCESS KEY HERE"

MINUTES\_BETWEEN\_READS = 10

METRIC\_UNITS = False

# ---------------------------------

streamer = Streamer(bucket\_name=BUCKET\_NAME, bucket\_key=BUCKET\_KEY, access\_key=ACCESS\_KEY)

while True:

humidity, temp\_c = Adafruit\_DHT.read\_retry(Adafruit\_DHT.DHT22, 4)

if METRIC\_UNITS:

streamer.log(SENSOR\_LOCATION\_NAME + " Temperature(C)", temp\_c)

else:

temp\_f = format(temp\_c \* 9.0 / 5.0 + 32.0, ".2f")

streamer.log(SENSOR\_LOCATION\_NAME + " Temperature(F)", temp\_f)

humidity = format(humidity,".2f")

streamer.log(SENSOR\_LOCATION\_NAME + " Humidity(%)", humidity)

streamer.flush()

time.sleep(60\*MINUTES\_BETWEEN\_READS)

The values on lines 6 thru 11 will need to be edited. The SENSOR\_LOCATION\_NAME = “Office” begins line 6 in the Python™ script. You need to replace Office with the location the RPI will be placed, for example; “IT Office Building H 110”. The BUCKET\_NAME, BUCKET\_KEY, ACCESS\_KEY, will all be your own specific set of name, generated key, and access key from **Initial State**™. Be sure to enter all the information into the script between the double quotes “info goes here“.

For information on Bucket Names, Bucket Keys, Access Keys, visit the following links:

* <https://support.initialstate.com/hc/en-us/articles/360002911551-Renaming-a-Data-Bucket>
* <https://support.initialstate.com/hc/en-us/articles/360002911831-Finding-a-Bucket-Key>
* <https://support.initialstate.com/hc/en-us/articles/360002898512-Streamer-Concepts-Streaming-Access-Key>

Once you enter the information, you can change the read times on line 10 as well as specify metric or imperial units on line 11. When you have finished editing the file, press CTRL X / Y (*for yes*) / Enter, to save the edited file.

**Running the Python™ script**

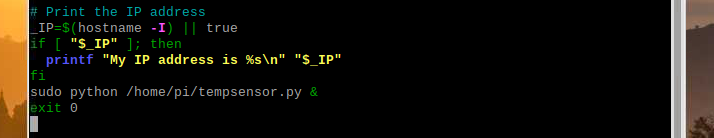
Once the file is saved you will return to the original Terminal™ window. There you will begin to run the Python™ script you have created called tempsensor.py. Enter the following command into your Terminal™ window:

* sudo python tempsensor.py

press **Enter**

This will initiate the script to run and begin to retrieve data and also send it to Initial State™ to show the output data on the GUI. The GUI can be altered and customized accordingly to your specifications for temperature thresholds by using temperature ranges and color schemes that you choose. You will be able to access the GUI on your desktop and mobile devices.

**Running the Python™ script on Startup**

The last step to this process will be to insert another command in your terminal window through the rc.local file. This command will allow the Python™ script to run when you reboot your RPI. Follow these steps to complete this task:

**File with the command in its correct position.**

Open a **Terminal**™ window and type the following commands:

1. cd /etc/
2. press **Enter** (*This will place your commands inside of the* etc *directory*.)
3. sudo nano rc.local
4. press **Enter** (*This will place you inside of the* rc.local *file where you will add a command to a specific area to autorun this program*.)
5. Press the down arrow to go to the bottom of the script. The last line will be exit 0.
6. Place your cursor beside exit 0 and press Enter once.
7. When you have done this, place your cursor in the empty space above exit 0 and insert the following command: sudo python /home/pi/tempsensor.py &
8. When you have entered the command, simply press CRTL X / Y / Enter to save the edited file.

This will complete the steps to set up the RPI with Initial State™ to be able to have the DHT22 reading and writing temperature and humidity data to your web based GUI. These steps give you the ability to monitor temperature and humidity with the Initial State™ GUI in any area that the RPI is placed, as long as the DHT22 is connected to it properly and the commands are entered correctly.

**Raspberry Pi® 3B+ / Camera Module / MotionEye™**

To install the camera module on the RPI, locate the connector on the RPI motherboard and lift the edges of the clip, and insert the ribbon cable with the blue side or the side opposite of the shiny connection points towards the LAN port. Once the cable is inserted and seated correctly, press the edges of the connector back down to lock it in

**RPI® w/ camera module installed**

place. That is all there is to installing the camera module onto the RPI. When it installed, you can follow these steps to set up MotionEye™.

**Setting up MotionEye™**

We will be using the same RPI in the previous section with the Raspbian OS and DHT22 installed to set up the camera module. This will give you the portability of an AIO (All-In-One) device. The entire process is combining two essential utilities for monitoring Network Closets, Head-End Rooms, etc. Today’s standards require this type of monitoring for Security purposes, and needed as an essential tool to maintain a favorable climate for hardware installed in critical areas.

To begin the installation of MotionEye™, you need to install the libraries, the program and directories. To do this follow these steps:

Open a **Terminal** window and enter the following commands:

(*When you enter these commands let them download and install 100% before proceeding to the next command*.)

* sudo apt-get install ffmpeg libmariadb3 libpq5 libmicrohttpd12

press **Enter**

* sudo wget https://github.com/Motion-Project/motion/releases/download/release-4.2.2/pi\_buster\_motion\_4.2.2-1\_armhf.deb

press **Enter**

* sudo dpkg -i pi\_buster\_motion\_4.2.2-1\_armhf.deb

press **Enter**

* sudo apt-get install python-pip python-dev libssl-dev libcurl4-openssl-dev libjpeg-dev libz-dev

press **Enter**

* sudo pip install motioneye

press **Enter**

* sudo apt-get install python-pillow

press Enter

* sudo mkdir -p /etc/motioneye

press **Enter**

* sudo cp /usr/local/share/motioneye/extra/motioneye.conf.sample /etc/motioneye/motioneye.conf

press **Enter**

* sudo mkdir –p /var/lib/motioneye

press **Enter**

* sudo cp /usr/local/share/motioneye/extra/motioneye.systemd-unit-local /etc/systemd/system/motioneye.service

press **Enter**

* sudo systemctl daemon-reload

press **Enter**

* sudo systemctl enable motioneye

press **Enter**

* sudo systemctl start motioneye

press **Enter**

Those commands will install the necessary libraries, directories, and auto run at Start Up. These commands need to be entered the same way as seen above in order. Failure to do so, could cause the program to not run at all. To upgrade to the newest version of MotionEye™, enter the following commands in a **Terminal** window:

* sudo pip install motioneye --upgrade

press **Enter**

* sudo systemctl restart motioneye

press **Enter**

* sudo reboot

press **Enter**

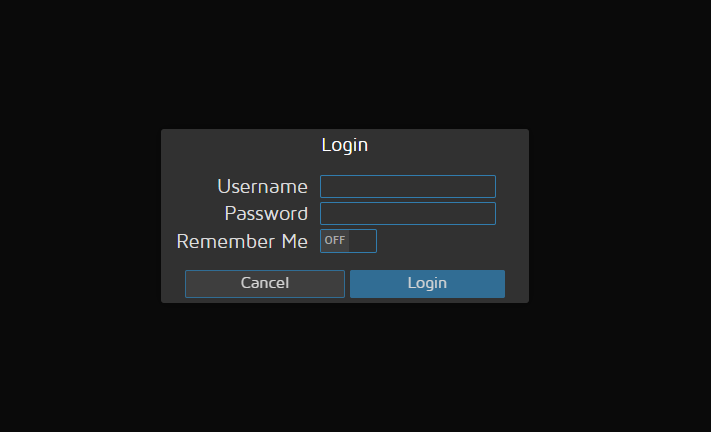
The next step will be to access the camera through Motioneye and configure the camera module’s settings. To do so follow the steps below.

* Obtain the IP Address of the RPI and document it for reference.
* Open a browser window, preferably Google Chrome.
* Enter the following URL:

<http://yourRPIipaddress:8765> (*replace “yourRPIipaddress” with the actual IP Address of the RPI.*)

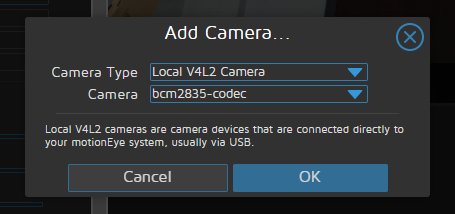
press **Enter**

* You will be pointed to the login page for MotionEye™
* Your initial login will be the username **admin**, no password required. It can be set up at a later time.

Once you have typed in admin and press Enter or click the Login button you will be taken to the MotionEye™ GUI where you will see an icon at the top left that are three, blue, horizontal lines. Click the lines and it will open an options panel. When the options panel opens, you will see the horizontal lines are now vertical at the top left. There will be a drop down menu/box one the right of these now three, vertical, blue bars. Select the drop down menu/box and it will have the option to add camera. Follow these steps to add a camera:



**Login screen for MotionEye™**



**Three, horizontal, blue lines turn vertical when selected. The drop down menu/box will give you the option to add a new camera to the GUI.**

**Select the appropriate Camera Type and the Camera from the drop down menu/boxes that appear in the add camera menu. These options will come from the camera that is installed into the RPI.**

Once you have completed the previous steps you will have a panel of settings when the three blue lines are vertical to set up your camera and fine-tune its options. You can set up the name of the camera, where the files are saved, as well as a timestamp and a lot more features. You only need the IP Address of your RPI and enter the URL as described above to access this camera for surveillance.

**Command & Script(S) Only Sheet**

*(All commands will be entered into* ***Terminal****. Each bullet point is one step. Press* ***Enter*** *after each bulleted command. Refer to each section of the tutorial for more info. )*

**Set up RPI (Raspberry Pi)**

Open a **Terminal** window

* sudo apt-get update
* sudo apt-get upgrade
* sudo reboot

**Turning on the GPIO Header Pins**

* sudo raspi-config

**Commands for DHT22 Functionality**

* sudo apt-get update

press **Enter**

* sudo pip install Adafruit\_DHT press **Enter**
* sudo pip install ISStreamer

press **Enter**

To create the python file named tempsensor.py, enter the following command:

* sudo touch tempsensor.py
* press **Enter**

Access the file by entering the following command:

* sudo nano tempsensor.py

press **Enter**

import Adafruit\_DHT

from ISStreamer.Streamer import Streamer

import time

# --------- User Settings ---------

SENSOR\_LOCATION\_NAME = "Office"

BUCKET\_NAME = ":partly\_sunny: Room Temperatures"

BUCKET\_KEY = "rt0129"

ACCESS\_KEY = "PLACE YOUR INITIAL STATE ACCESS KEY HERE"

MINUTES\_BETWEEN\_READS = 10

METRIC\_UNITS = False

# ---------------------------------

streamer = Streamer(bucket\_name=BUCKET\_NAME, bucket\_key=BUCKET\_KEY, access\_key=ACCESS\_KEY)

while True:

humidity, temp\_c = Adafruit\_DHT.read\_retry(Adafruit\_DHT.DHT22, 4)

if METRIC\_UNITS:

streamer.log(SENSOR\_LOCATION\_NAME + " Temperature(C)", temp\_c)

else:

temp\_f = format(temp\_c \* 9.0 / 5.0 + 32.0, ".2f")

streamer.log(SENSOR\_LOCATION\_NAME + " Temperature(F)", temp\_f)

humidity = format(humidity,".2f")

streamer.log(SENSOR\_LOCATION\_NAME + " Humidity(%)", humidity)

streamer.flush()

time.sleep(60\*MINUTES\_BETWEEN\_READS)

For information on Bucket Names, Bucket Keys, Access Keys, visit the following links:

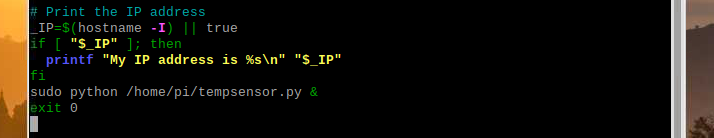
* <https://support.initialstate.com/hc/en-us/articles/360002911551-Renaming-a-Data-Bucket>
* <https://support.initialstate.com/hc/en-us/articles/360002911831-Finding-a-Bucket-Key>
* <https://support.initialstate.com/hc/en-us/articles/360002898512-Streamer-Concepts-Streaming-Access-Key>

**Running the Python™ script**

* sudo python tempsensor.py

press **Enter**

**Running the Python™ script on Startup**



* cd /etc/

press **Enter** (*This will place your commands inside of the* etc *directory*.)

* sudo nano rc.local

press **Enter** (*This will place you inside of the* rc.local *file where you will add a command to a specific area to autorun this program*.)

* sudo python /home/pi/tempsensor.py &

When you have entered the command simply press CRTL X / Y / Enter, to save the edited file.

**Setting up MotionEye™**

Open a **Terminal** window

* sudo apt-get install ffmpeg libmariadb3 libpq5 libmicrohttpd12

press **Enter**

* sudo wget https://github.com/Motion-Project/motion/releases/download/release-4.2.2/pi\_buster\_motion\_4.2.2-1\_armhf.deb

press **Enter**

* sudo dpkg -i pi\_buster\_motion\_4.2.2-1\_armhf.deb

press **Enter**

* sudo apt-get install python-pip python-dev libssl-dev libcurl4-openssl-dev libjpeg-dev libz-dev

press **Enter**

* sudo pip install motioneye

press **Enter**

* sudo apt-get install python-pillow

press **Enter**

* sudo mkdir -p /etc/motioneye

press **Enter**

* sudo cp /usr/local/share/motioneye/extra/motioneye.conf.sample /etc/motioneye/motioneye.conf

press **Enter**

* sudo mkdir –p /var/lib/motioneye

press **Enter**

* sudo cp /usr/local/share/motioneye/extra/motioneye.systemd-unit-local /etc/systemd/system/motioneye.service

press **Enter**

* sudo systemctl daemon-reload

press **Enter**

* sudo systemctl enable motioneye

press **Enter**

* sudo systemctl start motioneye

press **Enter**

* sudo pip install motioneye --upgrade

press **Enter**

* sudo systemctl restart motioneye

press **Enter**

* sudo reboot

press **Enter**

The next step will be to access the camera through Motioneye and configure the camera module’s settings. To do so follow the steps below.

* Obtain the IP Address of the RPI and document it for reference.
* Open a browser window, preferably Google Chrome.
* Enter the following URL:

<http://yourRPIipaddress:8765> (*replace “yourRPIipaddress” with the actual IP Address of the RPI.*)

press **Enter**

* You will be pointed to the login page for MotionEye™
* Your initial login will be the username **admin**, no password required. It can be set up at a later time.