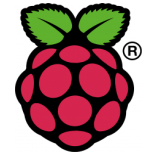


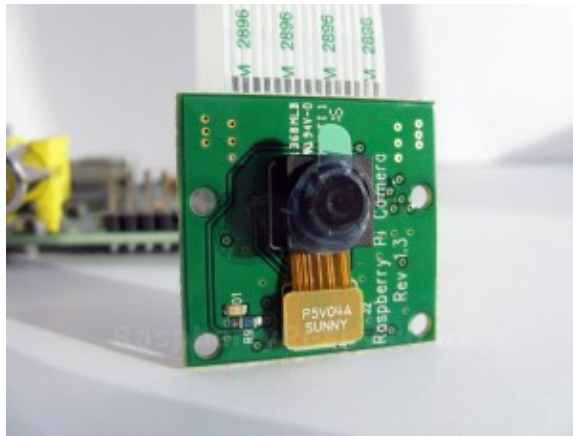


# PiCamera & Python



## What is it?

The PiCamera is an add-on for the Raspberry Pi that provides HD quality (1080p for those of you who care) still and video footage. Basically it's the same type of camera you have in a smartphone. There is also a version called the Pi Noir which is sensitive to infra-red light, so can be used for low light or IR lit shots like monitoring animals at night.



*Images from <http://raspberrypi.org>*

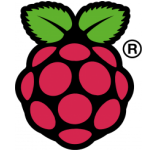
## Table of Contents

Getting Started  
Task 1 - Say Cheese  
Task 2 - Let's get Artistic  
Task 3 - Hollywood Calling

Project - Burglar Alarm  
Project - Tweet a Pic



# PiCamera & Python



Oooh, selfies! Let's get started

Step 1 - Power up and log on (probably done already)

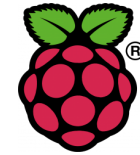
1. If it's not already running plug the micro-USB (phone charger) lead into the connector on the board
2. If prompted for a login use **pi** as the user and **raspberrypi** as the password, don't panic; you won't see the password as you type, just press the Enter key when you're done
3. Type **startx** and press Enter
4. The "magic" will start to happen and eventually you'll get to a desktop with a giant raspberry on it. Congratulations, Step 1 complete!

Step 2 - Starting IDLE (aka getting your Geek on)

1. Double click the **LXTerminal** icon - a small black console window will appear
2. Create a folder for your code - we'll use the current time so we don't clash with others:
  - a. At the prompt type **mkdir** a space and the time, so **mkdir 1030** or **mkdir 1451**. Check the clock in the bottom right if you need to. Don't forget to press Enter
  - b. Now type **cd** and the time you just entered (i.e. **cd 1030**) and press Enter
  - c. You should now be in your own folder ready to code
3. At the prompt type **sudo idle &** and press Enter; the **&** is important - a line of numbers will appear and another window will open
4. From the *File* menu select *New Window* - a blank white window will appear
5. Aaaand breathe - you're ready to code



# PiCamera & Python



## Task 1 - Say Cheese

First things first, let's make sure the camera is working and use its preview function. Down the left will be the code you need to enter, on the right some explanation of what you are doing. One thing to remember when coding in Python; indentation matters! So if a line of code is further to the left than the line above use the Tab key to put in the extra space. Let's get coding

<pre>import picamera import time</pre>	The PiCamera comes with it's own set of commands, this tells Python we want to use them. You'll see what <code>time</code> is for in a bit
<pre>camera = picamera.PiCamera()</pre>	This creates us an object called <code>camera</code> we can then perform actions on
<pre>try:     camera.start_preview()     time.sleep(5)     camera.stop_preview()  finally:     camera.close()</pre>	This section shows the preview window for 5 seconds then closes it again. <i>try...finally</i> This is some error handling. Due to the way PiCamra works, if there's an error you might get stuck with the preview up and no way out! This makes sure we close the <code>camera</code> object when we're done.

Before we get too far let's check it works. *File - Save* (or CTRL+S) and give it a filename of **camera**. Now it's saved press F5; if you can see on screen whatever the camera can then we're in business. Otherwise, have a look at the code above and check for typos. If it's upside down rotate the camera!



# PiCamera & Python



OK, so let's save that image of your beautiful, grubby face (you are on camp!). Only one extra line is needed to achieve this.

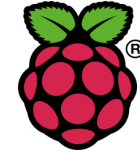
<pre>import picamera import time  camera = picamera.PiCamera()</pre>	
<pre>try:     camera.start_preview()     time.sleep(5)      camera.capture('image01.jpg')      camera.stop_preview()  finally:     camera.close()</pre>	<p>This now saves a file called <i>image01.jpg</i> to your folder</p>

Save and run to see what happens.... not much else? (I'm assuming it didn't error) Ah, but the magic is hidden from view. Go back to the LXTerminal window (the black one from the start) and at the prompt type **gpicview image01.jpg**. Ta-dah!, your selfie, or whatever you took a picture of.

Close the image window when you're done.



# PiCamera & Python



## Task 2 - Let's get Artistic

We can take a picture, now lets have some fun with effects. Open a new code window and enter the following:

<pre>import picamera import time  camera = picamera.PiCamera()</pre>	
<pre>camera.start_preview()  camera.image_effect = 'sketch'  time.sleep(5) camera.capture('sketch.jpg') camera.stop_preview() camera.close()</pre>	<p><code>image_effect</code> is the property we are using to change how this looks. We've also changed the filename.</p>

Save and run .... oh yes, very you! Remember, to view the image go back to the LXTerminal window (the black one from the start) and at the prompt type ***gpicview sketch.jpg***.

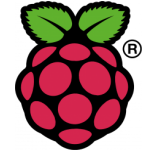
Assuming that all worked here's a list of all the effects, give them a try:

sketch, posterise, gpen, colorbalance, film, pastel, emboss, denoise, negative, hatch, colorswap, colorpoint, saturation, blur, watercolor, cartoon, none, washedout, solarize, oilpaint

Oh, and if you're getting a sore arm from holding the camera up the right way use `camera.hflip() = True` and `camera.vflip() = True` to rotate the image as needed.



# PiCamera & Python



## Task 3 - Hollywood Calling

Photos are all well and good, but what about a video clip. So we might not be making our YouTube debut with this next bit but you get the idea.

<pre>import picamera import time  camera = picamera.PiCamera()</pre>	
<pre>try:     camera.resolution = (640, 480)     camera.start_preview()     camera.start_recording('video01.h264')     camera.wait_recording(10)     camera.stop_recording()  finally:     camera.close()</pre>	<p>Just in case you're not sure what this does;</p> <ul style="list-style-type: none"><li>• Set image size to 640x480</li><li>• Start to record to <i>video01.h264</i></li><li>• Wait for 10 seconds</li><li>• Stop recording</li></ul>

Save and run .... are those your best moves?! This time, to view the video go back to the LXTerminal window (the black one from the start) and at the prompt type **omxplayer video01.h264**. To exit the recording press the escape key (usually **Esc** in the top left of your keyboard).

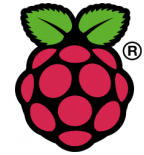
Again, go nuts with the effects if you like.

## Achievement Unlocked!

Assuming everything up to this point has gone according to plan good job, you've learnt a bit of Python and how to use the PiCamera. If you fancy a go at a couple of projects read on. Equally, hack your own ideas.



# PiCamera & Python



## Project - Burglar Alarm

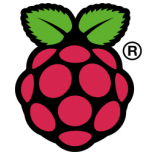
For this we will make use of the Pibrella board as well. For a more detailed look at the Pibrella there's a worksheet on that too!

<pre>import picamera import time import pibrella  camera = picamera.PiCamera()</pre>	
<pre>try:     while True:         if pibrella.button.read() == 1:             camera.start_preview()             time.sleep(2)             camera.capture('naughtyperson.jpg')             camera.stop_preview()  finally:     camera.close()</pre>	<p>This saves a file called <i>naughtyperson01.jpg</i> to your folder</p>

- Extensions:
- How about making the numbering of the picture unique? (hint: look at the next project)
  - Can you make it capture a video of them in the act?
  - How about some noise and lights? (you'll need to have done the Pibrella sheet for this one)



# PiCamera & Python



## Project - Tweet a Pic

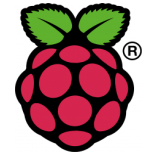
As the name suggests this will capture a picture (and a message) and tweet it! A friendly reminder; Twitter is public, play nice! Don't forget you can include your own handle if you like e.g. '@<your twitter> coded this @Gilwell24 #GeeksRule'

<pre>import tweepy import picamera from datetime import datetime from time import sleep</pre>	tweepy is a Python library used to communicate with Twitter
<pre>i = datetime.now() now = i.strftime('%Y%m%d-%H%M%S') photo_name = now + '.jpg'</pre>	Create a filename with the current date and time
<pre>with picamera.PiCamera() as camera:     camera.resolution = (1024, 768)     camera.start_preview()     sleep(5)     camera.capture(photo_name)     camera.stop_preview()</pre>	Configure the camera and show the preview (you could add an effect here....)
<pre>status= raw_input('Your message (max 140 chars): ') if status != '':</pre>	Get the message, if the message is blank (i.e. you just pressed Enter) then don't start the twitter code
<pre>consumer_key = &lt;from file&gt; consumer_secret = &lt;from file&gt; access_token = &lt;from file&gt; access_token_secret = &lt;from file&gt;</pre>	This sets up our authentication. Please open the file <b>tweepy.txt</b> on the desktop and copy/paste the contents where it says <from file>





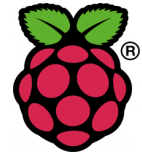
# PiCamera & Python



<pre>auth = tweepy.OAuthHandler(consumer_key, consumer_secret) auth.set_access_token(access_token, access_token_secret)  api = tweepy.API(auth)</pre>	Create our Tweepy object ready to use
<pre>photo_path = photo_name api.update_with_media(photo_path, status=status) print 'Tweeted to @G24_Pi' else: print 'Tweet cancelled'</pre>	Send the message! (or not as the case may be)



# PiCamera & Python



## Acknowledgements

Raspberry Pi and the Raspberry Pi logo are trademarks owned by the Raspberry Pi Foundation (<http://www.raspberrypi.org>)

PiCamera was created by Dave Hughes - resources inspired by his examples <http://picamera.readthedocs.org/>

Tweet a Pic was adapted from an article by Alex Eames <http://raspi.tv/?p=5918>

Pibrella is a joint venture by Cyntech (<http://www.cynotech.co.uk>) and Pimoroni (<http://www.pimoroni.com>)