Pi Camera

Tushar B. Kute,
http://tusharkute.com

Raspberry Pi Camera Module

- The Raspberry Pi Camera Module v2 replaced the original Camera Module in April 2016.
- The v2 Camera Module has a Sony IMX219 8-megapixel sensor (compared to the 5-megapixel OmniVision OV5647 sensor of the original camera).
- The Camera Module can be used to take high-definition video, as well as stills photographs. It’s easy to use for beginners, but has plenty to offer advanced users if you’re looking to expand your knowledge.
- You can also use the libraries we bundle with the camera to create effects.

Raspberry Pi Camera Module

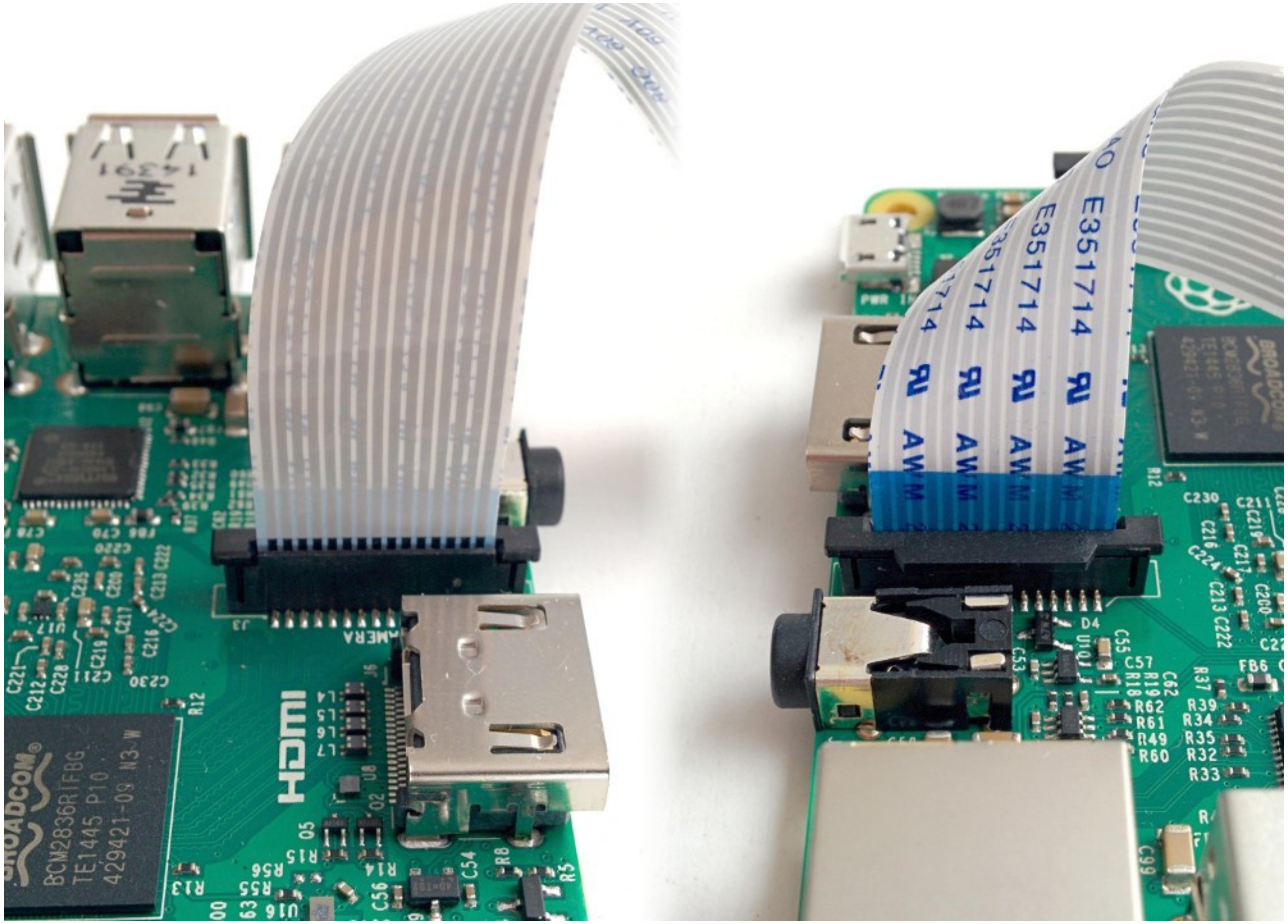
- You can read all the gory details about IMX219 and the Exmor R back-illuminated sensor architecture on Sony’s website, but suffice to say this is more than just a resolution upgrade: it’s a leap forward in image quality, colour fidelity, and low-light performance.
- It supports 1080p30, 720p60 and VGA90 video modes, as well as still capture. It attaches via a 15cm ribbon cable to the CSI port on the Raspberry Pi.

Raspberry Pi Camera Module

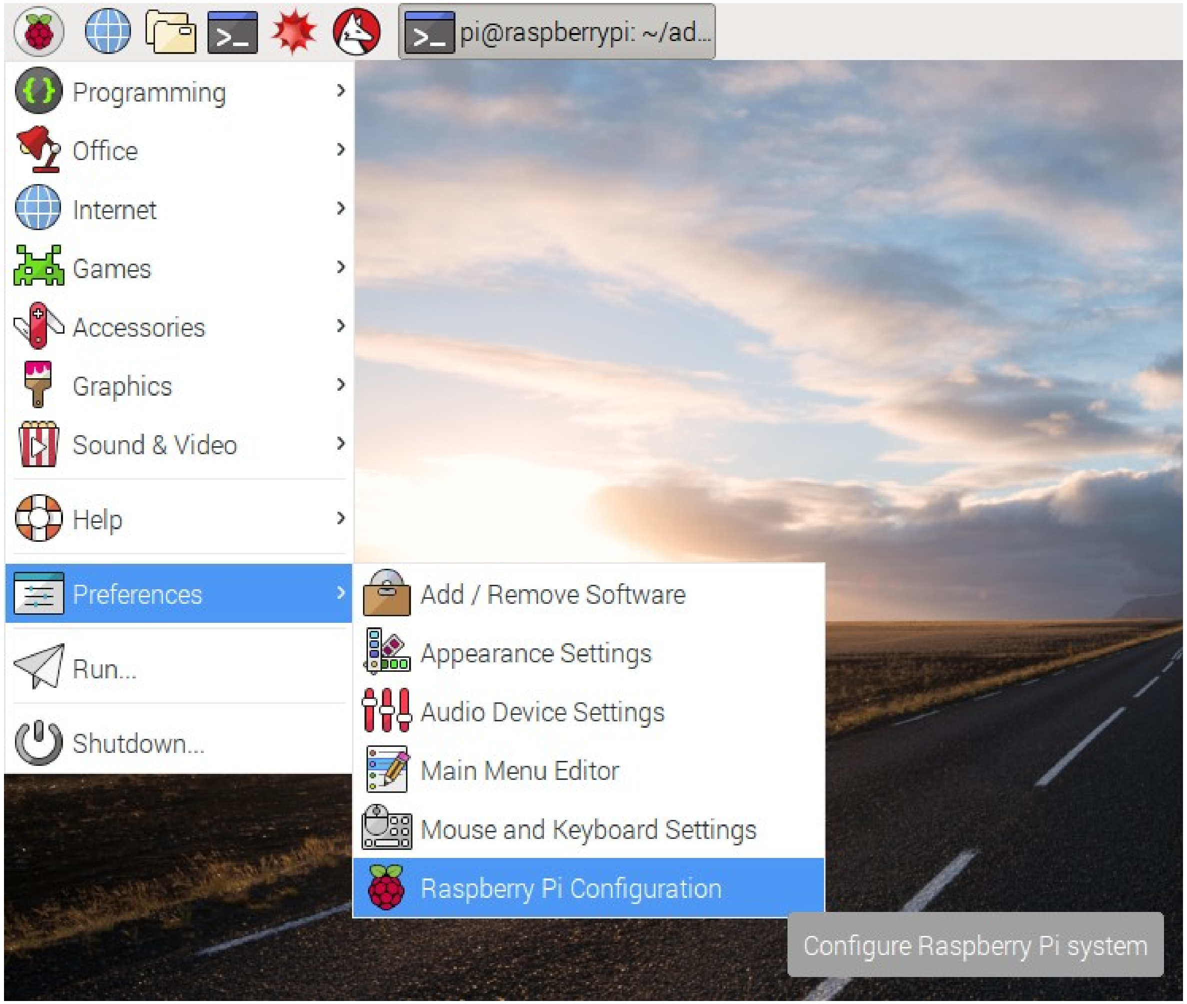
- The camera works with all models of Raspberry Pi 1, 2, and 3.
- It can be accessed through the MMAL and V4L APIs, and there are numerous third-party libraries built for it, including the Picamera Python library.
- The camera module is very popular in home security applications, and in wildlife camera traps.

Pi Camera

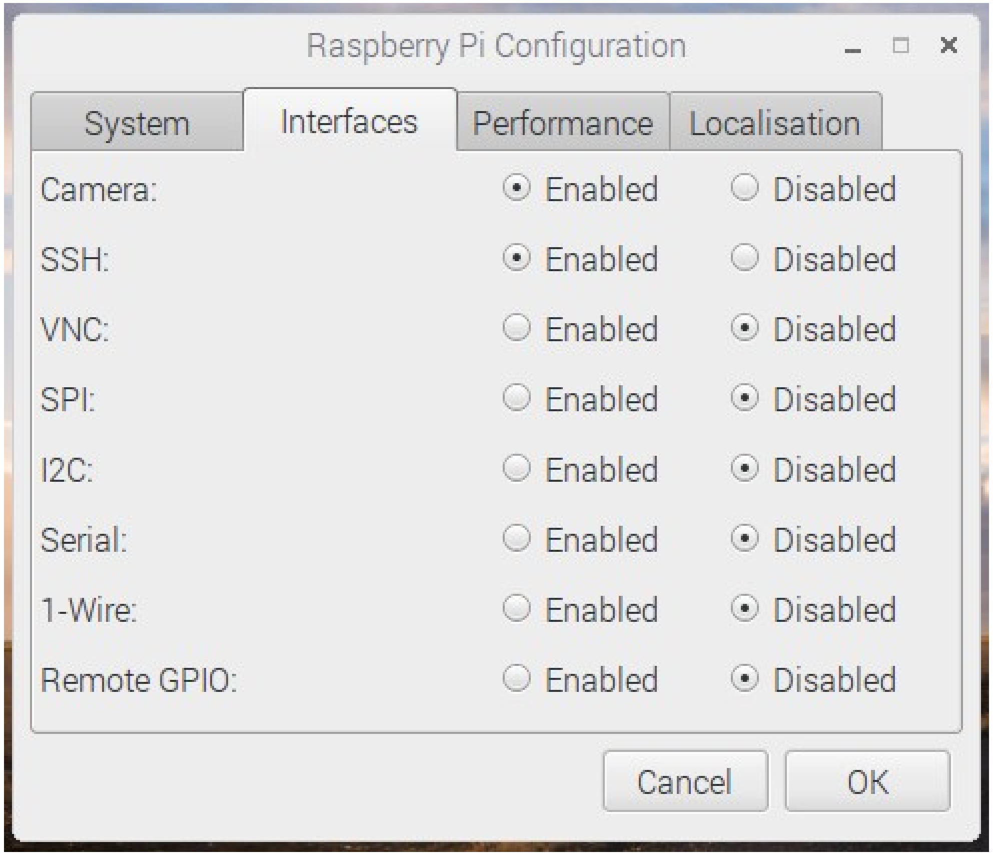
Connecting the Pi Camera



Open Raspberry Pi Configuration



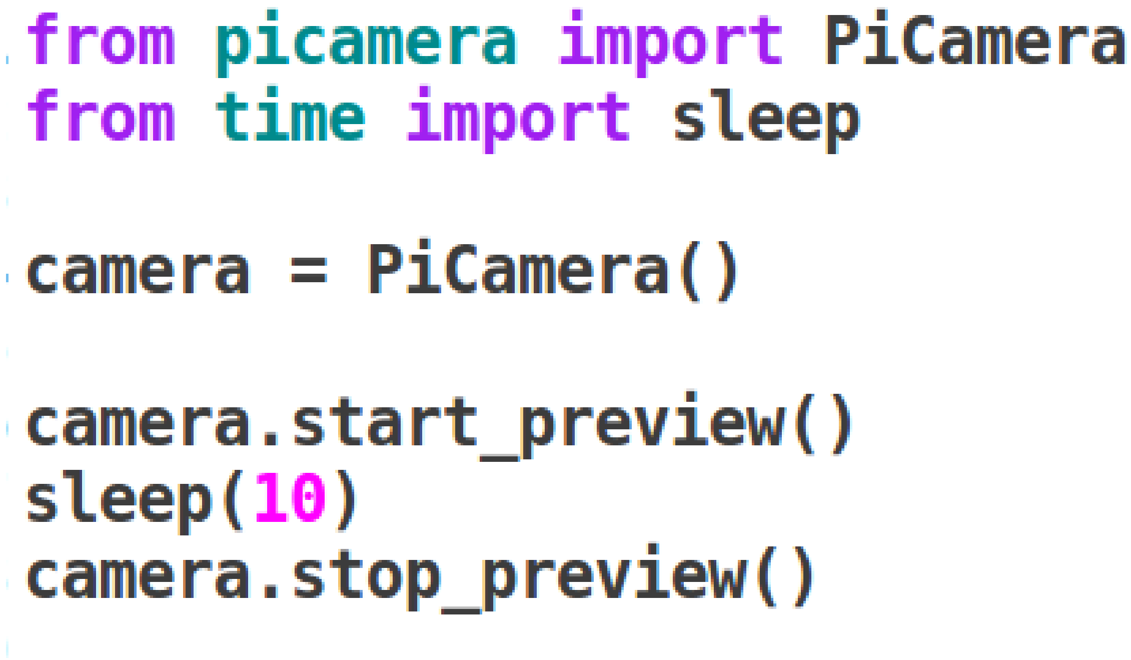
Enable the Camera interface



Raspberry Pi Configuration

Camera:	● Enabled	○ Disabled
SSH:	● Enabled	○ Disabled
VNC:	○ Enabled	● Disabled
SPI:	○ Enabled	● Disabled
I2C:	○ Enabled	● Disabled
Serial:	○ Enabled	● Disabled
1-Wire:	○ Enabled	● Disabled
Remote GPIO:	○ Enabled	● Disabled

Camera preview

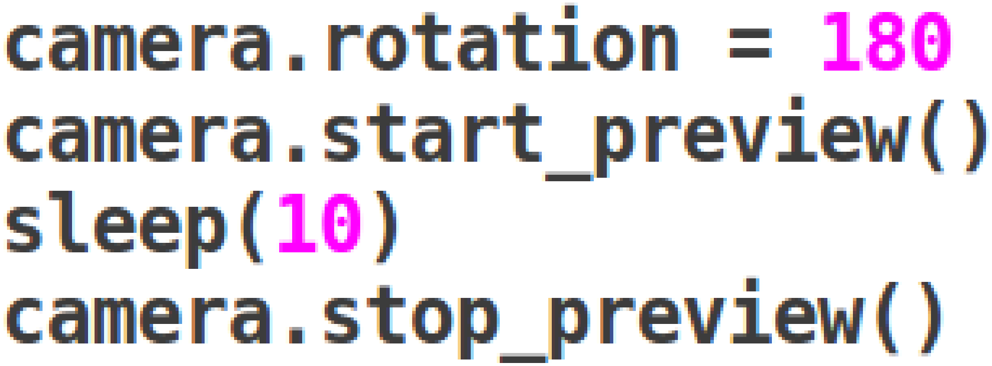


```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview()
sleep(10)
camera.stop_preview()
```

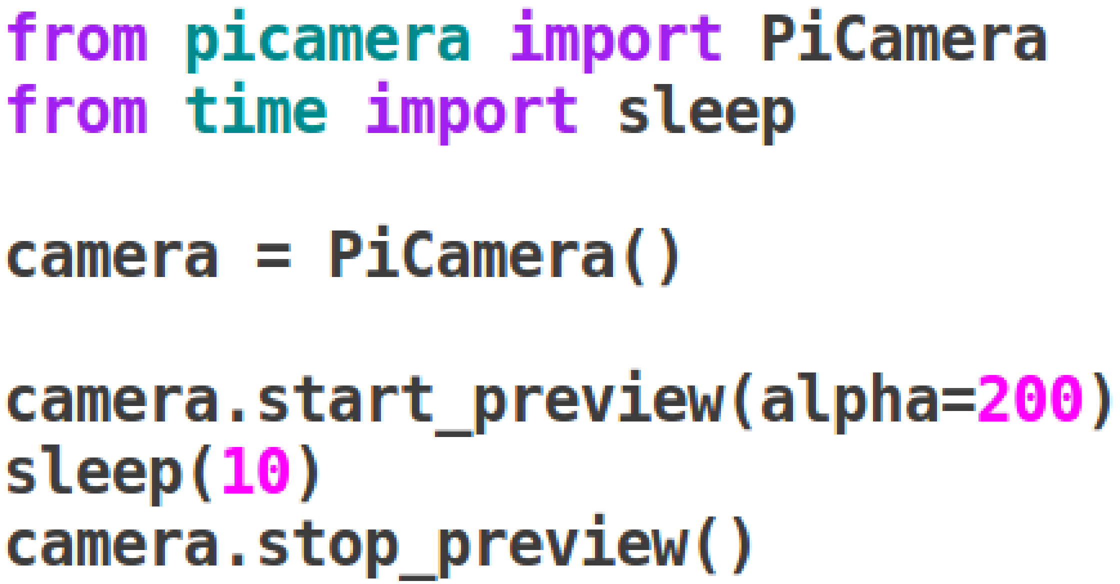
Rotating the Camera



```
camera.rotation = 180
camera.start_preview()
sleep(10)
camera.stop_preview()
```

You can rotate the image by 90, 180, or 270 degrees, or you can set it to 0 to reset.

Change transperancy

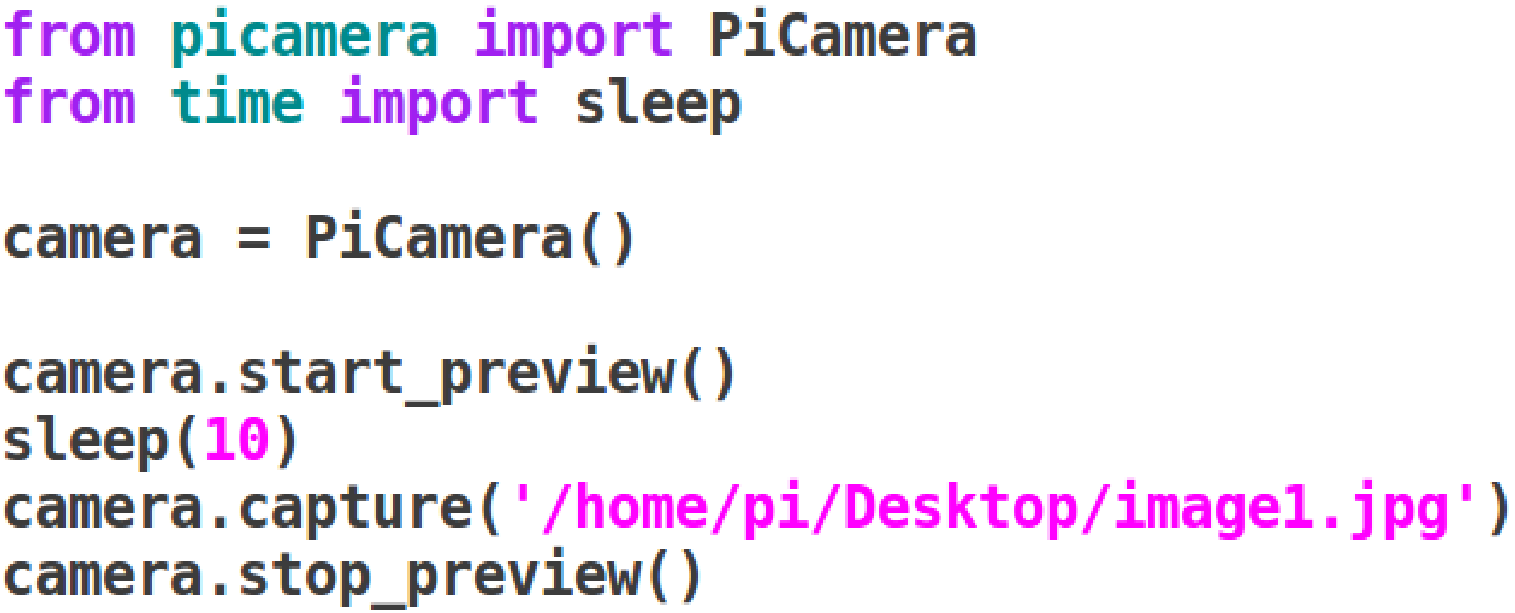


```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview(alpha=200)
sleep(10)
camera.stop_preview()
```

Storing the image

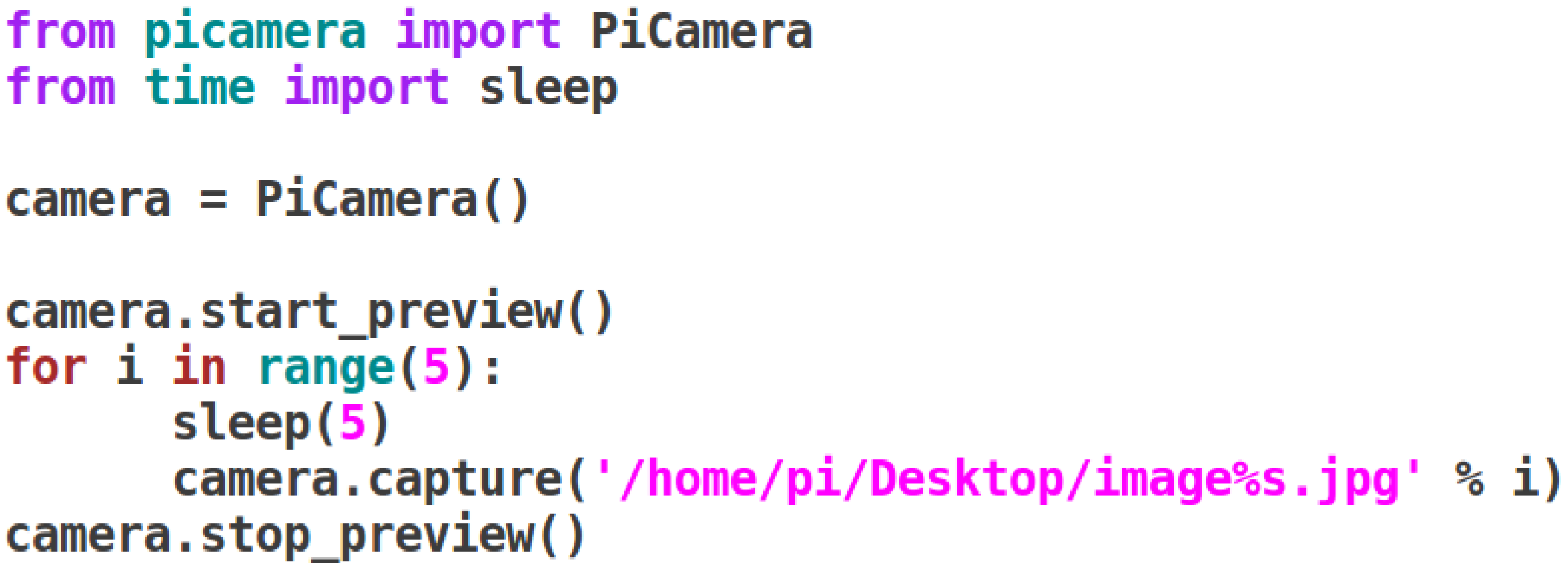


```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview()
sleep(10)
camera.capture('/home/pi/Desktop/image1.jpg')
camera.stop_preview()
```

Taking five pictures in a row

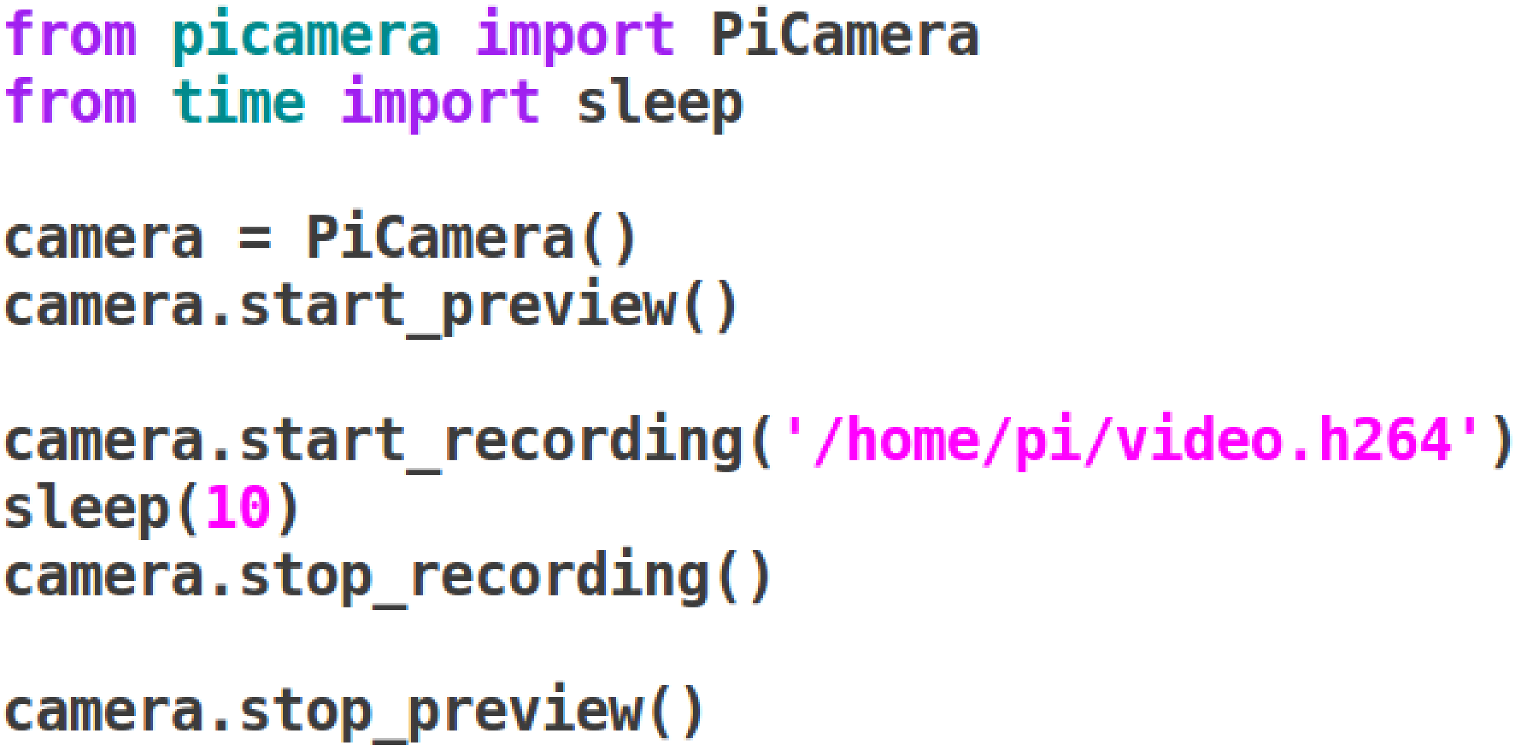


```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview()
for i in range(5):
      sleep(5)
      camera.capture('/home/pi/Desktop/image%s.jpg' % i)
camera.stop_preview()
```

Recording a Video



```
from picamera import PiCamera
from time import sleep

camera = PiCamera()
camera.start_preview()

camera.start_recording('/home/pi/video.h264')
sleep(10)
camera.stop_recording()

camera.stop_preview()
```

Converting and Playing Video

- The video format need to get converted to MP4. So install gpac.

  ```
  sudo apt-get install gpac
  ```

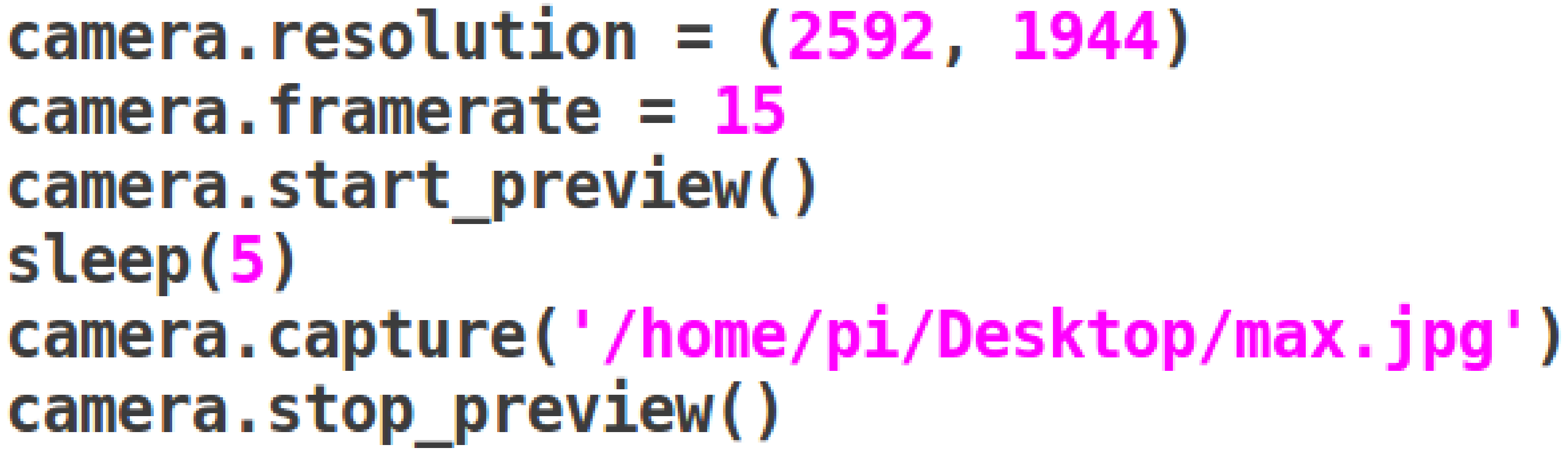
- Now convert the video to MP4:

  ```
  MP4Box  -fps  30  -add  video.h264  video.mp4
  ```

Effects

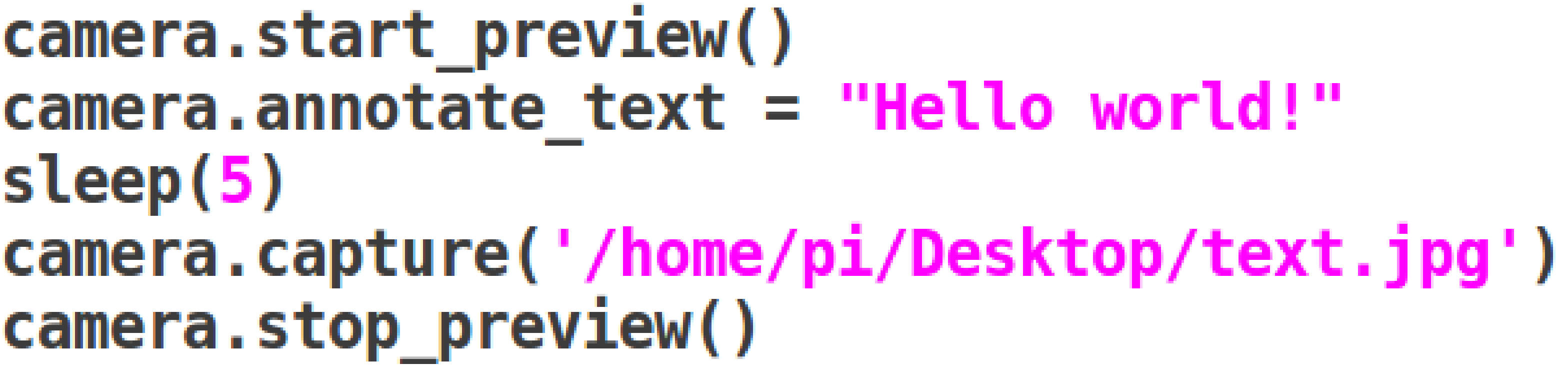
- The resolution of the capture is configurable.
- By default it's set to the resolution of your monitor, but the maximum resolution is 2592 x 1944 for still photos and 1920 x 1080 for video recording.
- Try the following example to set the resolution to max.
- Note that you'll also need to set the frame rate to 15 to enable this maximum resolution:
- The minimum resolution allowed is 64 x 64. Try taking one at that resolution.

Effects: Frame rate and Resolution



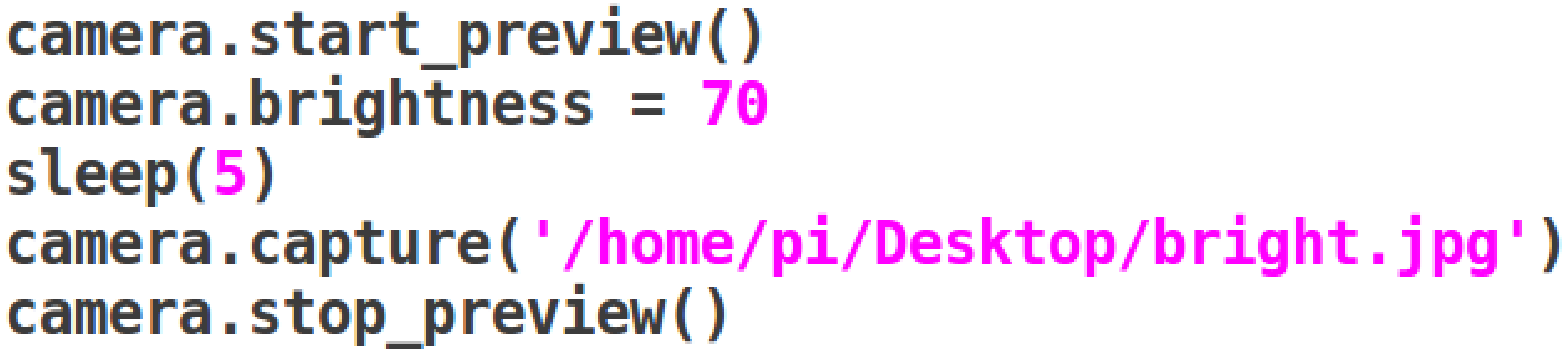
```
camera.resolution = (2592, 1944)
camera.framerate = 15
camera.start_preview()
sleep(5)
camera.capture('/home/pi/Desktop/max.jpg')
camera.stop_preview()
```

Effects: Text Annotation



```
camera.start_preview()
camera.annotate_text = "Hello world!"
sleep(5)
camera.capture('/home/pi/Desktop/text.jpg')
camera.stop_preview()
```

Effects: Brightness



```
camera.start_preview()
camera.brightness = 70
sleep(5)
camera.capture('/home/pi/Desktop/bright.jpg')
camera.stop_preview()
```

Annotation Text Sizes

- You can set the annotation text size with the following code:

```
camera.annotate_text_size = 50
```

- Valid sizes are 6 to 160. The default is 32.

Annotation Colors

- You can also alter the annotation colours. First of all, ensure that Color is imported by amending your import line at the top:

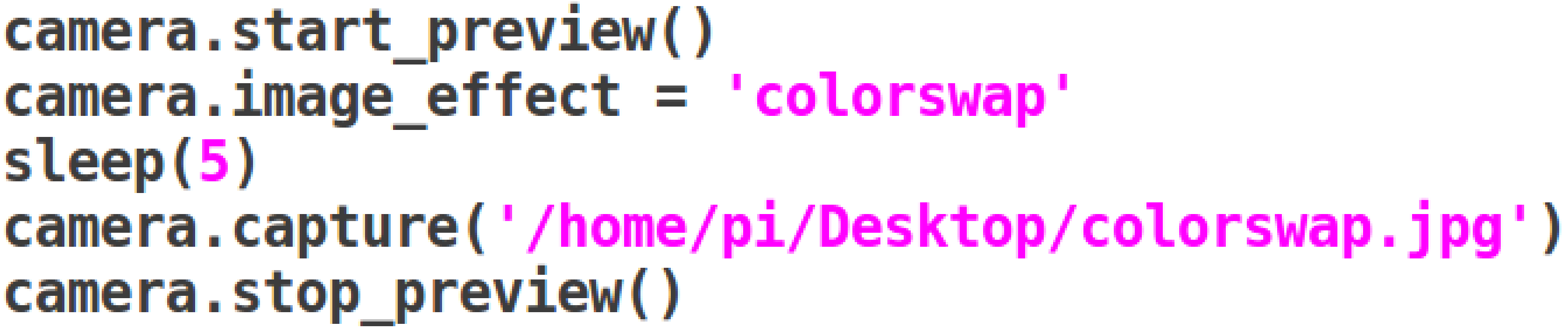
```
from picamera import PiCamera, Color
```

- Then amend the rest of your code as follows:

Image Effects

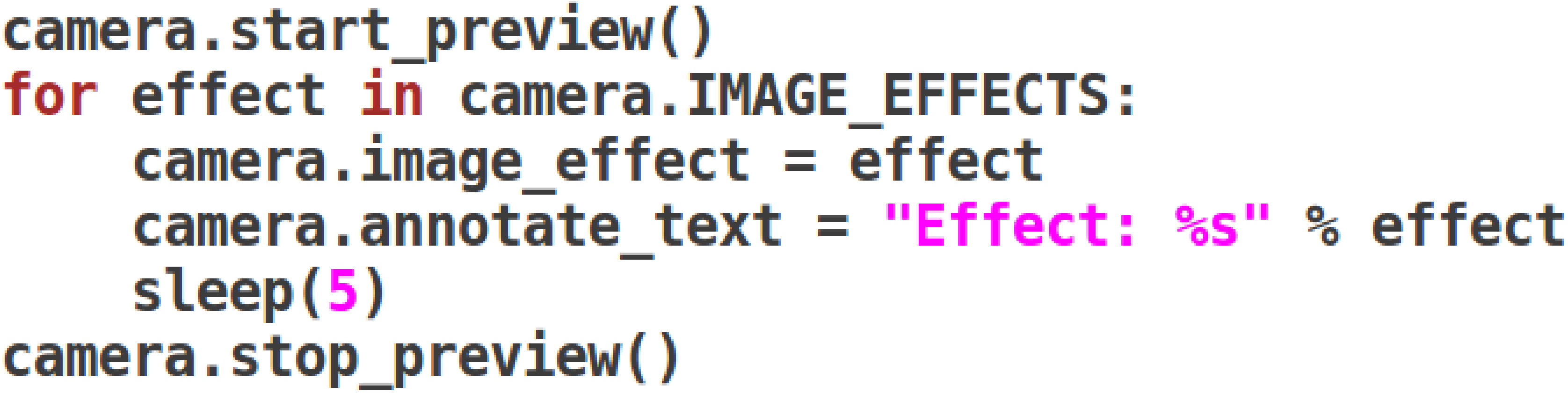
- You can use camera.image_effect to apply a particular image effect.
- The options are: none, negative, solarize, sketch, denoise, emboss, oilpaint, hatch, gpen, pastel, watercolor, film, blur, saturation, colorswap, washedout, posterise, colorpoint, colorbalance, cartoon, deinterlace1, and deinterlace2.
- The default is **none**.

Image Effects



```
camera.start_preview()
camera.image_effect = 'colorswap'
sleep(5)
camera.capture('/home/pi/Desktop/colorswap.jpg')
camera.stop_preview()
```

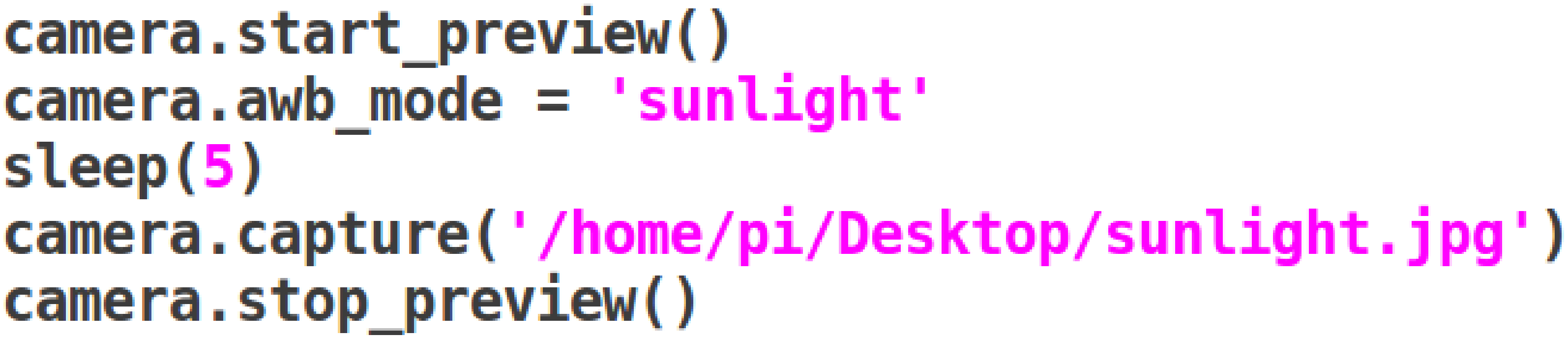
Looping various Effects



```
camera.start_preview()
for effect in camera.IMAGE_EFFECTS:
    camera.image_effect = effect
    camera.annotate_text = "Effect: %s" % effect
    sleep(5)
camera.stop_preview()
```

AWB mode

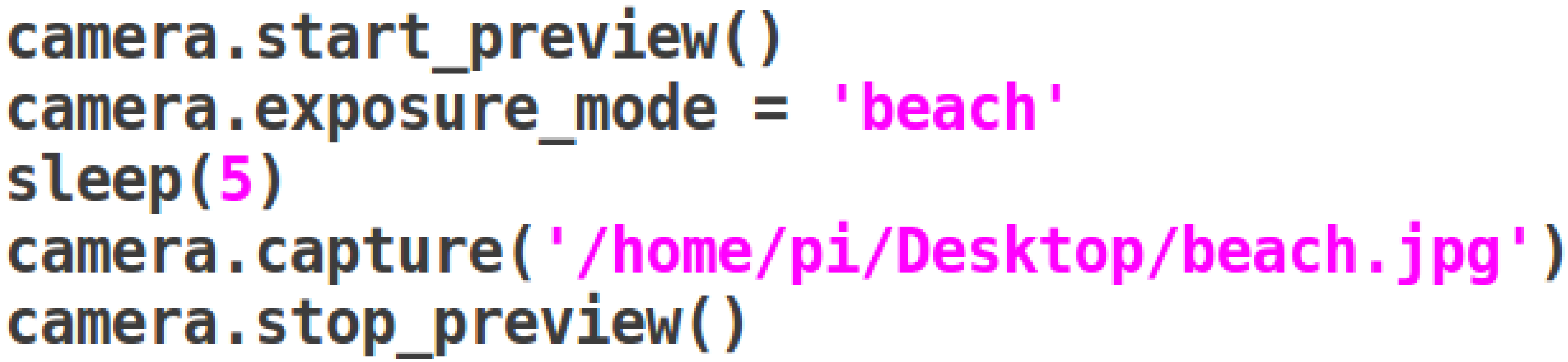
- You can use camera.awb_mode to set the auto white balance to a preset mode to apply a particular effect.
- The options are: off, auto, sunlight, cloudy, shade, tungsten, fluorescent, incandescent, flash, and horizon. The default is auto.



```
camera.start_preview()
camera.awb_mode = 'sunlight'
sleep(5)
camera.capture('/home/pi/Desktop/sunlight.jpg')
camera.stop_preview()
```

Exposure mode

- You can use camera.exposure_mode to set the exposure to a preset mode to apply a particular effect.
- The options are: off, auto, night, nightpreview, backlight, spotlight, sports, snow, beach, verylong, fixedfps, antishake, and fireworks. The default is **auto**.



```
camera.start_preview()
camera.exposure_mode = 'beach'
sleep(5)
camera.capture('/home/pi/Desktop/beach.jpg')
camera.stop_preview()
```

Live streaming

On the pi:

```
raspivid -t 0 -o - | nc -l -p 2222
(-p: choose any port higher than 1023)
```

In VLC: open network stream:

```
tcp://192.168.1.20:2222
add in options: :demux=h264
```

Thank you

@mitu_skillologies

/mITuSkillologies

@mitu_group

Web Resources
http://mitu.co.in
http://tusharkute.com

Blogs
http://digitallocha.blogspot.in
http://kyamputar.blogspot.in

tushar@tusharkute.com