

Códigos en Python para raspberry Pi (IoT al alcance de todos)

Telegram

```
import telebot
TOKEN='123456789101112131415161718192021222324252627282930313233343536373839404142434445464748495051525354555657585960616263646566676869707172737475767778798081828384858687888990919293949596979899100'
bot = telebot.TeleBot(TOKEN)
@bot.message_handler(func=lambda message: True)
def echo_all(message):
    chat_id=message.chat.id
    print 'ID del chat: ' + str(chat_id)
    print 'Mensaje: ' + message.text
    bot.send_message(chat_id, 'Hola!!')
    img=open('4.jpg','rb')
    bot.send_photo(chat_id, img)
bot.polling()
```

Código que realiza una conexión con Telegram y despliega todos los mensajes recibidos.

Código para tomar una fotografía mediante Webcam.

```
from cv2 import *
import time
cam = VideoCapture(0)
time.sleep(1)
s, img = cam.read()
time.sleep(1)
imwrite('foto.jpg',img)
del cam
```

Programa que toma una fotografía y la manda por telegram si recibe el mensaje “foto”.


```

#!/usr/bin/env python
import smtplib
import getpass
dir_origen = 'origen@gmail.com'
dir_destino = 'destino@gmail.com'
msg = 'Cuerpo del mensaje'

msg="Subject: Asunto_del_correo\nCuerpo del correo"

# Claves
username = 'origen@gmail.com'
password = getpass.getpass("Ingresa tu password ")

# Proceso de envio
server = smtplib.SMTP('smtp.gmail.com:587')
server.starttls()
server.login(username,password)
server.sendmail(dir_origen, dir_destino, msg)
server.quit()

```

GPIO Raspberry

Código para prender un led con la raspberry.

```

#!/usr/bin/env python
# -*- coding: utf-8 -*-import RPi.GPIO as GPIO
#Importa la libreria para poder pausar posteriormente el programa
import RPi.GPIO as GPIO
import time
#Define que convenci n de nombre se utiliza en el programa
GPIO.setmode(GPIO.BCM)
#Le dice a Python que no imprima los mensajes de alerta del GPIO en
#pantalla
GPIO.setwarnings(False)
#Indica a Python que el pin 18 se utiliza para la salida de informaci n
#es decir poner el pin en 'on' y 'oof
GPIO.setup(17, GPIO.OUT)
#Imprimimos un mensaje en terminal
while 1:
    print "LED on"
    #Coloca el pin del GPIO 'encendido'. Lo que en realidad
    # es que el pin provee 3.3 volts. Lo suficiente para encender el LED
    GPIO.output(17, GPIO.HIGH)
    #Pausa el programa de Python por un segundo
    time.sleep(1)
    #Coloca el pin del GPIO 'apagado'. Con lo que se tiene el LED apagado
    print "LED off"
    GPIO.output(17, GPIO.LOW)
    time.sleep(1)

```