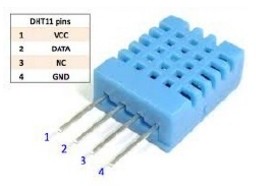
# DHT11(SICAKLIK/NEM) SENSÖR UYGULAMASI

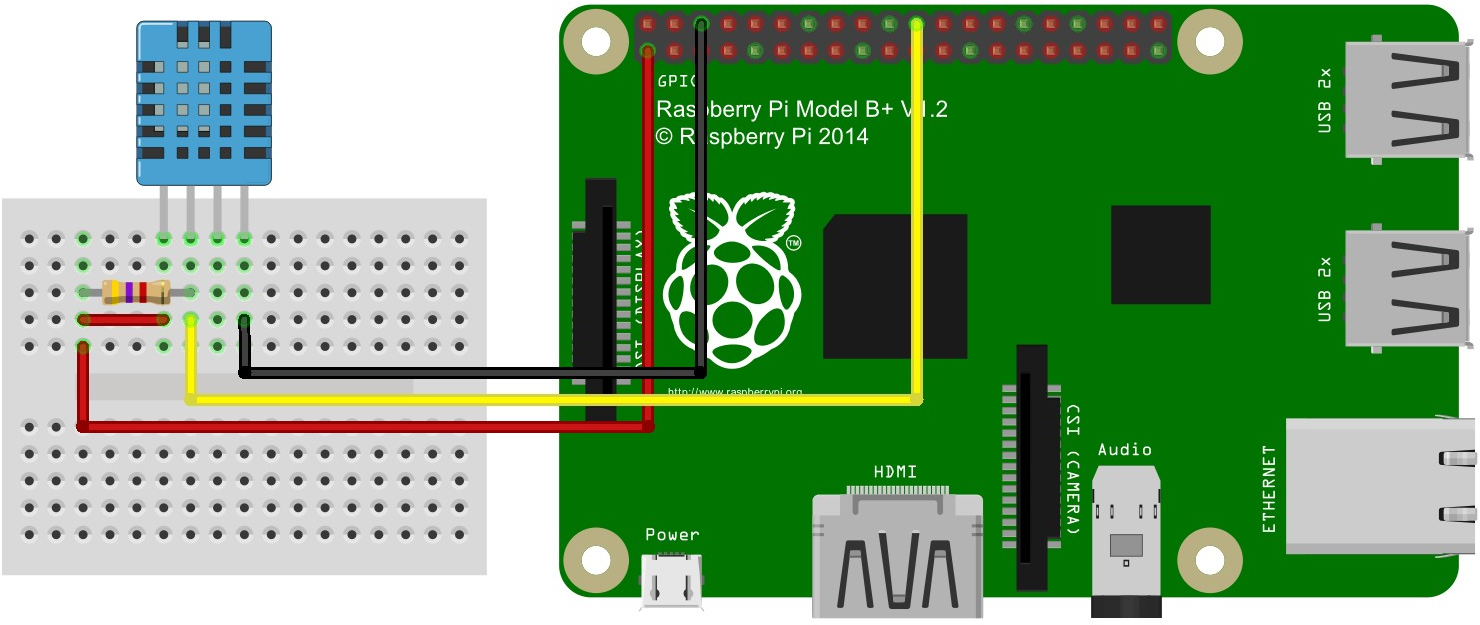
* + 1. **Sıcaklık Nem(DHT11) Sensörü**

Şekil 5.3'de gösterilen DHT 11 sıcaklık ve nem algılayıcı kalibre edilmiş dijital sinyal çıkışı veren gelişmiş bir algılayıcı birimidir. Yüksek güvenilirliktedir ve uzun dönem çalışmalarda dengelidir. 8 bit mikroişlemci içerir, hızlı ve kaliteli tepki verir. 0 ile 50 arasında 20 hata payı ile sıcaklık ölçen birim, 20-90 % RH arasında 5% RH hata payı ile nem ölçer. Bu sensör ev içerisindeki sıcaklık ve nemi ölçmek için kullanılır. Sıcaklık bilgileri kullanıcıya bildirilir ve programlandığı sıcaklık değerini ayarlamak için gerekli modülleri ( Fan,klima) aktif eder.



Şekil 5.3. Sıcaklık ve nem sensörü

# Board Üzerinde Kurulumu



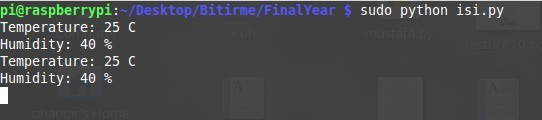
Şekil 5.4 Sıcaklık/Nem Algılama Sensör Devresi

Şekil 5.4’deki gibi Raspberry pi ve Sıcaklık/nem sönsörü board üzerinde gerekli baglantılar yapılarak kuruldu.

# Python Programlama ile Kodlanması

**Sekil 5.5: Sıcaklık/Nem Algılama Sensör Kodu**

Burada ısı ve nem ölçümlerini yapmak için python programlama dilinin dht11 kütüphanesi kullanıldı.

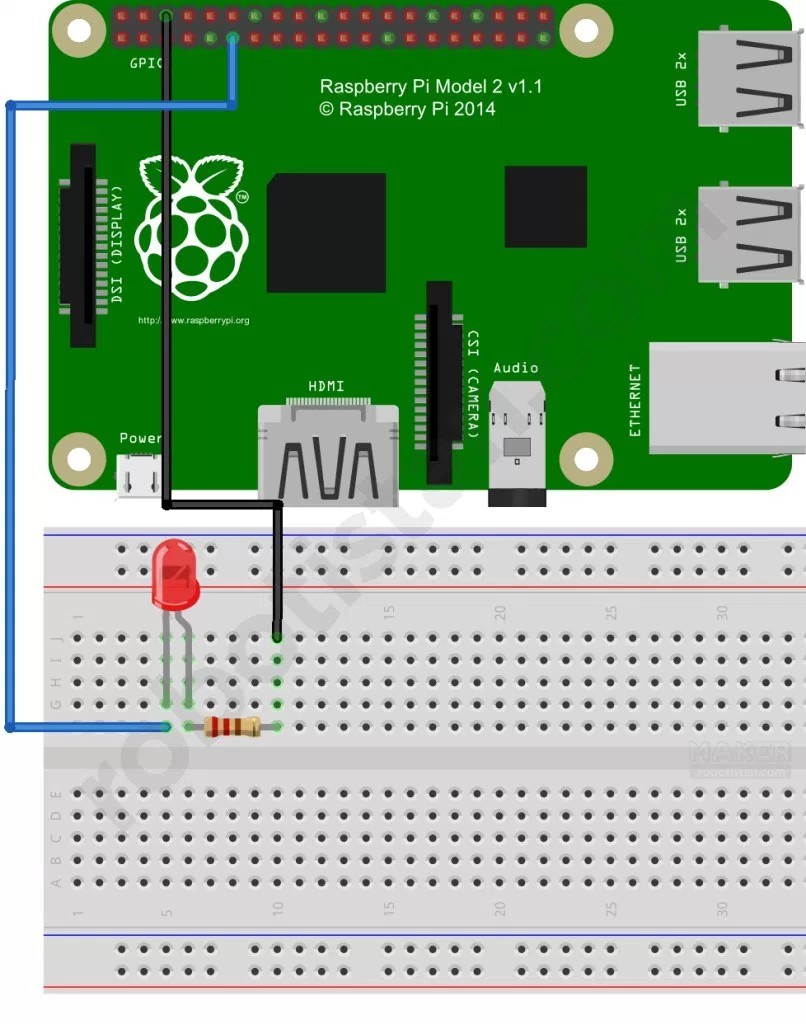


Şekil 5.6:Sıcaklık/Nem Algılama Sensör Testi

Şekil 5.4’deki gibi board üzerinde kurulumu yapılan, Şekil 5.5 gibi kodu yazılarak çalıştırılan Sıcaklık/Nem sensörü¸ Şekil 5.6’daki gibi çalıştırılarak test edildi.

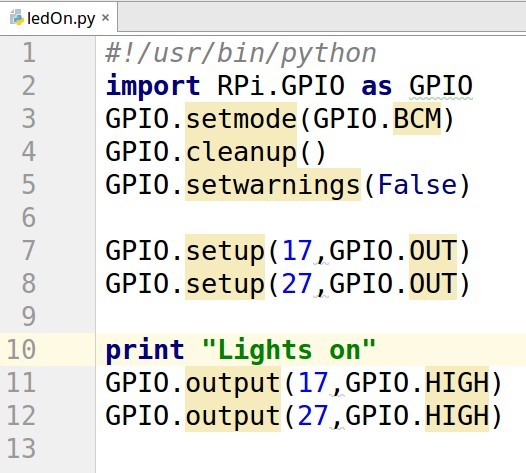
# LED UYGULAMASI

* + 1. **Board Üzerinde Kurulumu**



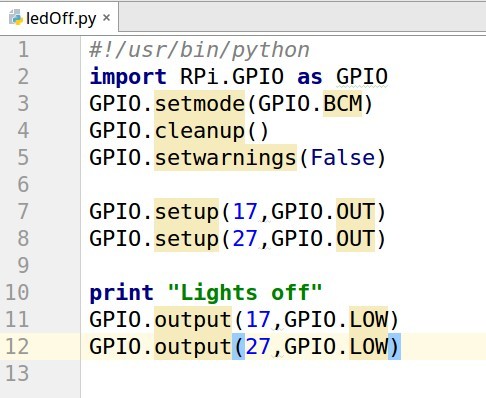
**Şekil 5.7:Led Devresi**

* + 1. **Python Programlama ile Kodlanması**

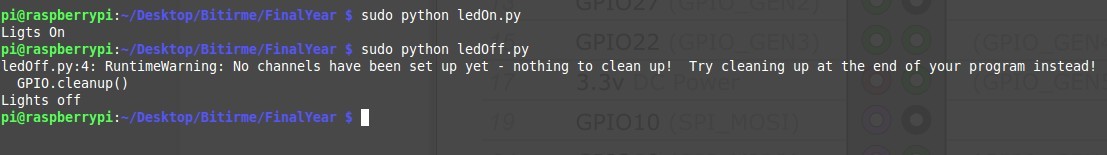


**Şekil 5.8:Led Açma Kodu**

Burada Led açma kapama işlemlerini gerçekleştirmek için **RPi.GPIO as GPIO** kütüphananesi kullanılmıştır. **GPIO.setup(17,GPIO.OUT)**'da GPIO 17 çıkış pini olarak ayarlanmaktadır. **GPIO.output(17,GPIO.HIGH)** kodu ledimizi açmaktadır.



# Şekil 5.9:Led Kapatma Kodu GPIO.output(17,GPIO.LOW) kodu ledimizi kapatmaktadır.



**Şekil 5.10:Led Açma ve Kapatma Testi**

Şekil 5.7’daki gibi board üzerinde kurulumu yapılan, Şekil 5.8 ve Şekil 5.9'daki gibi kodu yazılarak çalıştırılan Ledler¸ Şekil 5.10’daki gibi çalıştırılarak test edildi.

# SERVO MOTOR UYGULAMASI

* + 1. **Servo Motor Hakkında**

Şekil 5.10’da gösterilen Servo motor herhangi bir mekanizmanın işleyişini hatayı algılayarak yan bir geri besleme düzeneğinin yardımıyla denetleyen ve hatayı gideren otomatik aygıttır. Robot teknolojisinde en çok kullanılan motor çeşitidir. Bu sistemler mekanik olabileceği gibi elektronik. hidrolik-pnömatik veya başka alanlarda da kullanılabilmektedir.

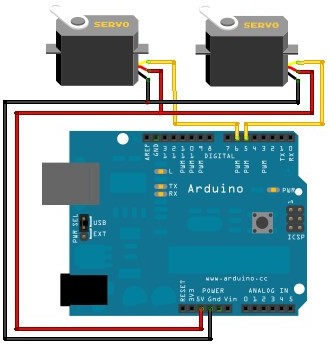
Servo motorlar da çıkış; mekaniksel konum. hız veya ivme gibi parametrelerin kontrol edildiği bir düzenektir. Servo motor içerisinde herhangi bir motor AC, DC veya Step motor bulunmaktadır. Ayrıca sürücü ve kontrol devresini de içerisinde barındırmaktadır. D.C. servo motorları, genel olarak bir D.C. motoru olup, motora gerekli D.C. aşağıdaki metotlardan elde edilir. 1- Bir elektrik yükselteçten. 2- A.C. akımın doyumlu reaktörden geçirilmesiııden. 3- A.C. akımın tristorden geçirilmesinden.4- Amplidin, retotrol, regüleks gibi dönel yükselteçlerden elde edilir.

Servo motor mekanik cihazları kontrol etmekte kullanılır.Balıkları programlanan saat aralığında yem vermek için kullanılabilir.Çiçeklerin belli aralıklarla sulanması için kullanılabilir.

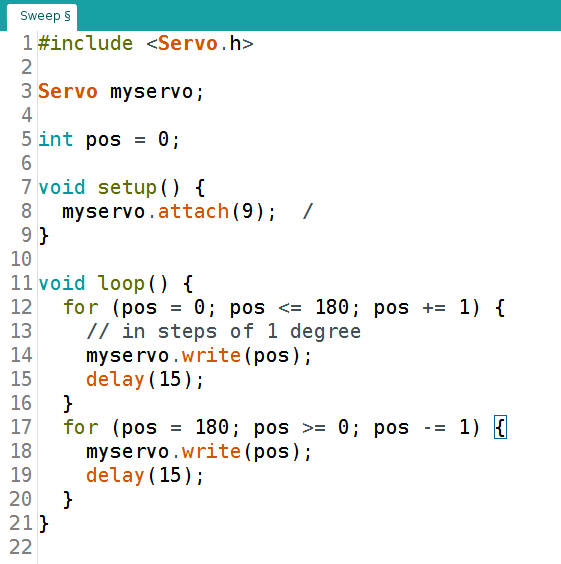


# Şekil 5.10:Servo Motor

* + 1. **Servo Motor Bağlantısı**



**Şekil 5.11:Servo Motor Baglantısı**

* + 1. **Arduino Programlama ile Kodlanması**

**Şekil 5.12:Servo Motor Kodu**

Burada Servo Motoru Arduino'nun 9.pinine bağlandı.Servo Motoru 0 dereceden 180 dereceye kadar derece derece çalışmaktadır.

# RÖLE UYGULAMASI

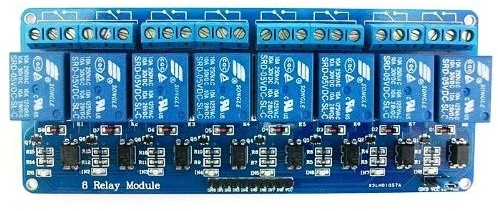
* + 1. **Röle Hakkında**

Röle üzerinden akım geçtiği zaman çalışan elektromanyetik bir devre elemanıdır. Röleler küçük değerli bir akım ile yüksek güçlü bir alıcıyı anahtarlayabilmek için kullanılır. Röleler, tek bir elemanda birden fazla anahtar kontağına sahip olabilir ve böylelikle birden fazla yükü aynı anda açıp kapatabilirler. Bu özellikleri ile röleler, tristör ve triyaklardan daha avantajlıdır.

Rölelerin dezavantajı ise mekanik şekilde çalıştıklarından dolayı sık arıza yapabilmeleridir. Rölenin kontakları defalarca birbirine yapışıp açıldığı için zamanla oluşan elektrik atlamaları ile kontaklar oksitlenebilir ve iletimini kaybedebilir.

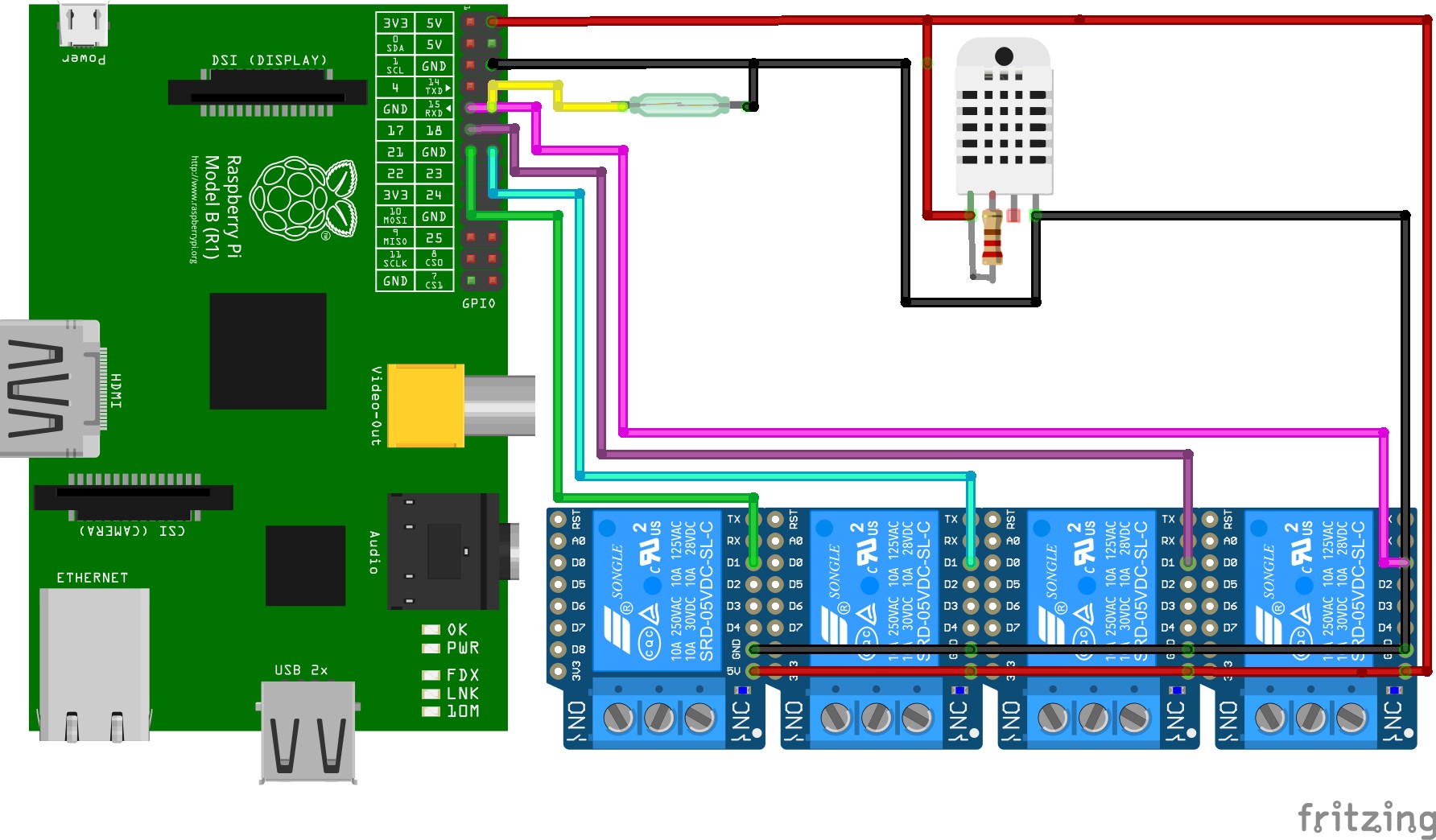
Röleler başka bir elektrik devresinin açılıp kapanmasını sağlayan elektriksel anahtarlardır. Bu özellikleri ile bir nevi transistörler gibidirler. Bobin iki kontağı mıknatısladığı zaman rölenin bir kontağı açılır bir kontağı kapanır.

Röleler, aynı anda farklı frekans ve dalga türlerinde etkilenmeden anahtarlama yapabilirler. Elektromanyetik çalışırlar, yani üzerlerinden akım geçmesiyle aktif hale gelirler. Röleler devrelerin giremediği bölgelerde (yüksek sıcaklık, nem veya sıvısal ortamlar) büyük önem kazanırlar. Tristor ve triyakların kullanımıyla popülerlikleri biraz azalsa da halen aktif olarak kullanılmaktadır.

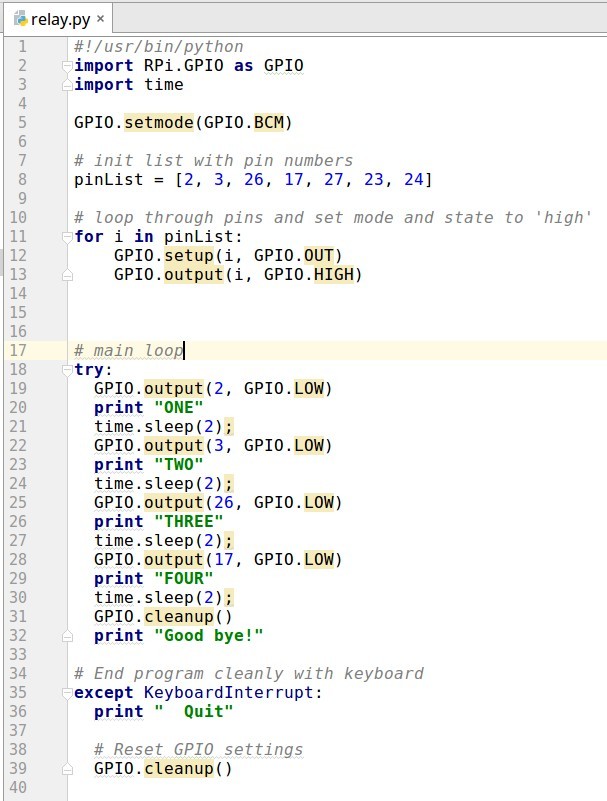


# Şekil 5.13:Röle

* + 1. **Board Üzerinde Kurulumu**

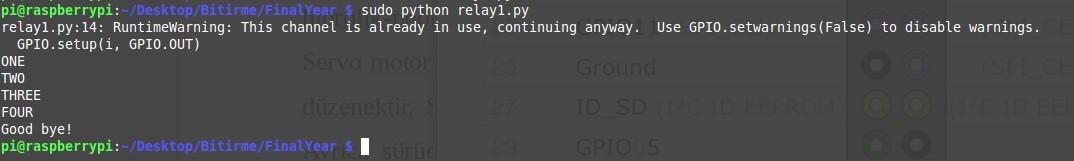


* + 1. **Python Programlama ile Kodlanması**



**Şekil 5.15:Röle Kodu**

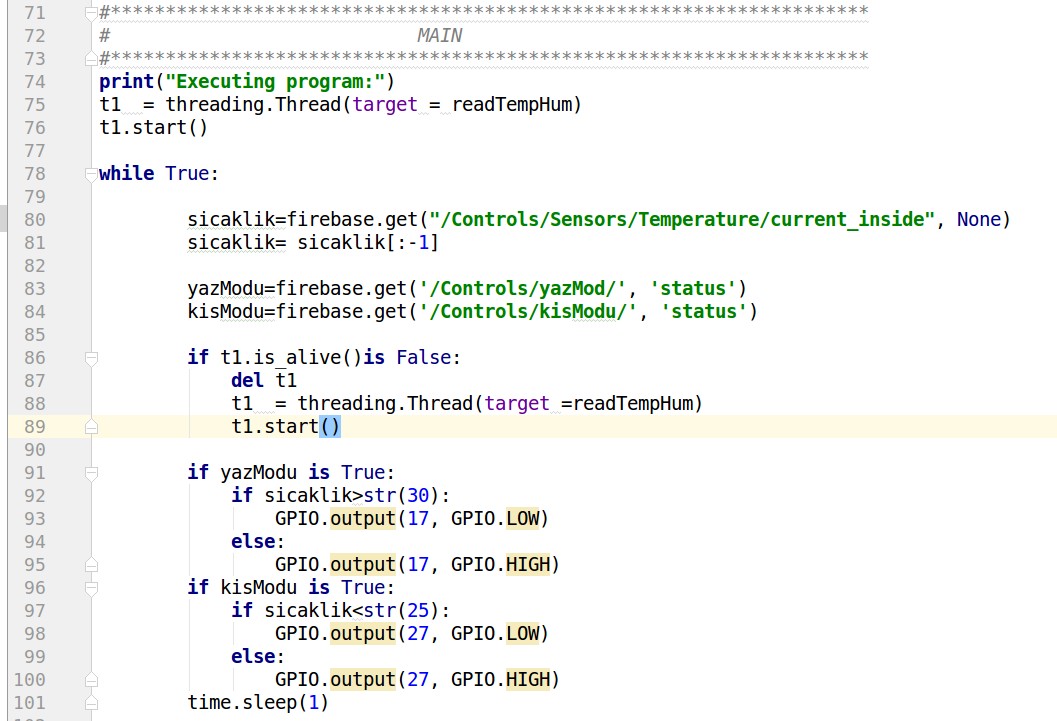
Kullanılacak olan GPIO pinleri bir dizide tutulur. **GPIO.setup(i, GPIO.OUT)** ile pinler çıkış pini olarak ayarlanır.**GPIO.output(i, GPIO.HIGH)** ile pinler pasif duruma getiriliyor.**GPIO.output(i, GPIO.LOW)** ile de pinler aktif duruma getiriliyor.Röle GPIO.LOW da aktif, GPIO.HIGH da pasif çalışır.



Şekil 5.14’deki gibi board üzerinde kurulumu yapılan, Şekil 5.15' deki gibi kodu yazılarak¸ Şekil 5.16’daki gibi çalıştırılarak test edildi.

# 1.5 SICAKLIĞA BAĞLI OLARAK FAN UYGULAMASI

Eğer Android uygulamasında yaz modu aktif edilirse veritabanından gelen sıcaklık değeri 30 C0'dan fazla olursa soğutucu fan otomatik olarak çalışır.Eğer kış modu aktif edilirse veritabanından gelen sıcaklık 25 C0'dan az olursa ısıtıcı fan otomatik olarak çalışır.



# Şekil 5.16:Fan Çalışma Kodu

readTempHum fonksiyonu ile sıcaklık değerleri her 30 sn'de veri tabanına kaydediliyor. sicaklik=firebase.get("/Controls/Sensors/Temperature/current\_inside", None) ile veritabanından güncel sıcaklık değeri alınıyor.

Ayrıca veritabanından yazModu ve kisModu durumları kontrol ediliyor.if bloklarında ise yaz modu ve kış modlarına göre sıcaklık karşılaştırması yapılmıştır.Karşılaştırma sonuçlarına göre ısıtıcı veya soğutucu fan çalışmaktadır.

# RASPBERY PI'da VERİTABANI İŞLEMLERİ

Veritabanı olarak Firebase veritabanı kullanıldı.

# a-)Sensörlerden alınan Sıcaklık ve Nem değerlerinin Veritabanına Kaydedilmesi

firebase=firebase.FirebaseApplication('[https://smarthome-308a5.firebaseio.com](https://smarthome-308a5.firebaseio.com/)') ile veritabanı bağlanılıyor.

firebase.put("/Controls/Sensors","/Humidity/current\_inside","" + str(result.humidity)+"%") ile sensörlerden alınan veriler veritabanındaki hedefe kaydediliyor.Bunun için firebase'ın put methodu kullanılmıştır.

# b-) Veritabanından Gerekli Verilerin Çekilmesi



mutfakLED = firebase.get('/Lights/ODA1/', 'status') ile veritabanından veriler çekildi.Bunun için firebase'in get methodunu kullanıldı.