# Overview

A photoresistor is a light-controlled variable resistor. The resistance of a photoresistor decreases with increasing incident light intensity. Analog signals collected by the photoresistor are converted to digital signals through PCF8591. Then these digital signals are transmitted to Raspberry Pi and printed on the screen.

# Step 1: Connect It

Use the following wiring diagram to connect the sensor to your Raspberry Pi



**Important Notes:**

* The wire colors do not matter. Just use whatever is available at your station.
* **Make sure that you are plugging the wires into the correct location on both the Pi and the Sensor (especially the power and ground)**
* The wires move around a lot, so make sure they are firmly in place when you are testing

# Step 2: Code It

We have provided you with a Python code template that can run this sensor/device. The code is located at:

**/home/pi/Documents/rpi-iot-demos/photoresistor.py**

Use the Geany editor to open this file (NOTE: this will be slow, so be patient). Once it is open, take some time to read the code to see how it works.

Once you are ready, do the following:

1. Find the function called loop().
   1. This function starts when the script runs, and runs forever
2. Modify this function so that it samples the photoresistor and outputs the resistance value

def loop():

while True:

resistance = ADC.read(0)

print(“value:”, resistance)

time.sleep(0.2)

# Step 3: Run / Test It

1. To run the script, open a Linux console and navigate to the folder with your code
   1. Helpful Linux Commands:  
      **ls** lists the contents of the directory  
      **cd <folder\_name>** opens a folder (don’t type the < > characters)  
      **cd ..** exits the folder you are currently in  
      **cd ~** takes you back to your home folder
2. Run the python program by typing the following:

**python3 photoresistor.py**

1. Try exposing the photosensor to light and/or shade to see what happens!
   1. If it doesn’t work, make sure that your wiring is correct and that you see lights on the sensor / device
   2. How reliable is this sensor? How far away can an object be to sense it?
2. **To exit the program, press Ctrl-C in the terminal**

# Step 4: Make It Control Something

To make your program send MQTT messages, modify your main program to look like the following:

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

# Main Program Starts Here

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

if \_\_name\_\_ == '\_\_main\_\_':

setup()

try:

comm.connect()

print("Program Running. Heading to the Light")

loop()

except KeyboardInterrupt:

destroy()

Now, go to the loop() function you modified earlier. You can use the comm.send function to send a message when the photoresistance is above/below a certain value, etc.

if resistance > 100:

comm.send("CHANNEL NAME GOES HERE", "MESSAGE GOES HERE")