**Lesson\_9\_Notes\_and\_Journal**

**2024.05.04**

* Look at Code
* Look at Circuit
* Draw Circuit -- NA
* Build Circuit – Nothing to Build
* Start Lesson

Look for PicoW WiFi Projects on YouTube and/or internet in general

A screenshot of a computer

Description automatically generated

* Consider 37 in 1 sensor kit for a basic weather station
* WiFi Beacon Tracker is a sniffer to locate WiFi Beacons
* WiFi Repeater is a range extender
* IOT Data Logger sends to an internet service like AWS or Google for visualization and analysis

**Code**

* Glance at Code

import network

import socket

from time import sleep

from picozero import pico\_temp\_sensor, pico\_led

import machine

ssid = 'NAME OF YOUR WIFI NETWORK'

password = 'YOUR SECRET PASSWORD'

def connect():

#Connect to WLAN

wlan = network.WLAN(network.STA\_IF)

wlan.active(True)

wlan.connect(ssid, password)

while wlan.isconnected() == False:

print('Waiting for connection...')

sleep(1)

ip = wlan.ifconfig()[0]

print(f'Connected on {ip}')

return ip

def open\_socket(ip):

# Open a socket

address = (ip, 80)

connection = socket.socket()

connection.bind(address)

connection.listen(1)

return connection

def webpage(temperature, state):

#Template HTML

html = f"""

<!DOCTYPE html>

<html>

<form action="./lighton">

<input type="submit" value="Light on" />

</form>

<form action="./lightoff">

<input type="submit" value="Light off" />

</form>

<p>LED is {state}</p>

<p>Temperature is {temperature}</p>

</body>

</html>

"""

return str(html)

def serve(connection):

#Start a web server

state = 'OFF'

pico\_led.off()

temperature = 0

while True:

client = connection.accept()[0]

request = client.recv(1024)

request = str(request)

try:

request = request.split()[1]

except IndexError:

pass

if request == '/lighton?':

pico\_led.on()

state = 'ON'

elif request =='/lightoff?':

pico\_led.off()

state = 'OFF'

temperature = pico\_temp\_sensor.temp

html = webpage(temperature, state)

client.send(html)

client.close()

try:

ip = connect()

connection = open\_socket(ip)

serve(connection)

except KeyboardInterrupt:

machine.reset()

**Circuit**

* Theirs

**A green circuit board with holes

Description automatically generated**

* Mine – None / Nothing to build
* As built -- NA