Microcontrollers in Labs **Lab 11 – WiFi** **Fall 2024**

Names ____

Introduction

Finally, we can break the boundaries of the four walls we are sitting in and connect to the wide world out there through the World Wide Web (WWW.) Once you can connect a Pico W to the internet you can read web pages, serve your own web pages, and store data "in the cloud." We are finally getting to the "W" in Pico W.

Start working through Ch. 12 *WiFi Connectivity with Pico W*. When you get **`Robust_connection.py`** working, pause and work through the section below.

Saving WiFi Information in a File

It is a good idea **not** to put sensitive information in your code. Instead save it in a separate file. This way you can safely share your code without sharing your sensitive information. There is a simple format for saving system settings called *TOML* (Tom's Obvious Minimal Language, if you were curious.) The code below reads simple TOML settings you need for WiFi.

The `settings.toml` File

Create a file named **`settings.toml`** like the one below:

```
WIFI_SSID = "GENEVAGAMING"
WIFI_PASSWORD = "GenevaGameConsoles"
```

This should work whereever you can access the Geneva gaming Wifi. If you want to connect to a different WiFi, make the abvious changes in the SSID and PASSWORD lines.

If you are are taking the Pico W home to a completely differents WiFi, I would make two different setting files named something like **`settings_geneva.toml`** and **`setting_home.toml`** and copy the one I want to **`settings.toml`** when I change locations.

Reading the `.toml` File

Since this is a routine task, it is a good idea to put the code that reads this into a module that you import. Create a file **`ReadTOML.py`** with the content below.

```
""" ReadTOML """

settings = {}

with open("settings.toml") as fp:
    line = fp.readline().strip()
    while line:
        if "WIFI_SSID" in line:
            settings["WIFI_SSID"] = line[line.rfind(' "')+2:-1]
        if "WIFI_PASSWORD" in line:
            settings["WIFI_PASSWORD"] = line[line.rfind(' "')+2:-1]

        line = fp.readline().strip()
```

This code creates a *dictionary* of the settings. A dictionary (also called a *hash*) is a collection where the items are found by a *key* instead of an index number. The line

```
settings = {}
```

creates an empty dictionary.

Next the code opens the settings.toml file and reads all of the lines looking for the SSID (network name) and PASSWORD (psk).

Using the `setting.toml` File

Copy the **`ReadTOML.py`** to the Pico W so your code can import it. After the last import line, add the line

```
from ReadTOML import settings
```

Replace the code

```
wlan = network.WLAN(network.STA_IF)
wlan.active(True)
wlan.connect("NetworkName", "Password")
```

with the code

```
ssid = settings["WIFI_SSID"]
psk = settings["WIFI_PASSWORD"]

wlan = network.WLAN(network.STA_IF)
wlan.active(True)
wlan.connect(ssid, psk)
```

Great! Your code is now shareable with revealing your secrets.

Use this in the rest of the code you write for this chapter.

What Does the Web Page Say?

For each of the URL's five URL, including the first one, describe what the web request returns. Check the results aby navigating to the URL using a browser.

- npr.org -
- wttr.in -
- ipecho.net -
- earthquake.usgs.gov -
- artscene.textfiles.com -

Continue

Keep working through the book. When you save the file connect.py, don't forget to use the **`settings.toml`** file and **`connect.py`**.

Posting Data to the Cloud

It is nice to have a web page where you can read a temperature or turn an LED on and off, but what we have done so far requires that you be on the same local WiFi as the Pico W. Another restriction is that the temperature data is lost. When you refresh the web page, it shows the current temperare, but that data is not saved.

What you work on next is not in the book. You will post (send) data to a cloud server where it is save and plotted.

Names ____