
pyportal*_multissidLibraryDocumentation*

Release 1.0

Gregory M Paris

Dec 08, 2020

Contents

1	Dependencies	3
2	Usage Example	5
3	Contributing	7
4	Documentation	9
5	Table of Contents	11
5.1	Simple test	11
5.2	pyportal_multissid	12
5.2.1	Implementation Notes	12
6	Indices and tables	13
	Python Module Index	15
	Index	17

Allows PyPortal to connect to multiple SSIDs

CHAPTER 1

Dependencies

This driver depends on:

- [Adafruit CircuitPython](#)
- [Adafruit CircuitPython PyPortal](#)

Please ensure all dependencies are available on the CircuitPython filesystem. This is easily achieved by downloading the [Adafruit library and driver bundle](#).

CHAPTER 2

Usage Example

This module implements a child class build upon Adafruit's PyPortal class. The class implements just one method, overriding a base method, allowing the PyPortal device to connect to more than one SSID. To use, invoke this class where you would have used the PyPortal class. For example:

```
from pyportal_multissid import PyPortal_MultiSSID
pyportal = PyPortal_MultiSSID(status_neopixel=board.NEOPIXEL)
```

Configure your “home” network as ‘ssid’ and ‘password’ as usual in your *secrets.py* file, then add any additional networks (such as phone hotspot, *MiFi* devices, and networks at places the PyPortal will visit) as described below. The PyPortal then becomes portable.

The alteration of *secrets.py* is to include one extra key/value pair. The key is ‘hotspots’ and the value is a list of pairs of ssid and password strings:

```
secrets = {
    # Whatever is already in your secrets dict remains.
    # Be sure to add a now-needed comma if there isn't one.
    'hotspots': [
        ['myphonessid', 'thepassword'],
        ['mymifissid', 'itspassword'],
        ['myvacationhomessid', 'dontyouwish'],
    ],
}
```

See `examples/pyportal_multissid_simpletest.py`

CHAPTER 3

Contributing

Contributions are welcome! Please read our [Code of Conduct](#) before contributing to help this project stay welcoming.

CHAPTER 4

Documentation

For information on building library documentation, please check out [this guide](#).

5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/pyportal_multissid_simpletest.py

```
1  """Simple test of PyPortal_MultiSSID
2  """
3
4  # This test requires a secrets.py file on your PyPortal.
5  # At a minimum, the secrets dictionary in that file will contain:
6  #   * Your Adafruit IO username and key, required for get_local_time()
7  #   * The ssid and password of your "home" network.
8  #   * A hotspots entry that will be a list of ssid/password tuples
9  #     corresponding to "away" networks, such as your phone operating
10 #     as a hotspot, a "MiFi" device, or any other place you expect
11 #     the device to operate.
12 # Note: Places like hotels and Starbucks are not supported because
13 #       they require visiting a sign-in page and interacting with it.
14 #       This class is for known, passphrase-protected networks with
15 #       no web-based sign-in.
16
17 import time
18 import board
19 from pyportal_multissid import PyPortal_MultiSSID
20
21 # PyPortal_MultiSSID class is invoked exactly the same way as PyPortal.
22 pyportal = PyPortal_MultiSSID(status_neopixel=board.NEOPIXEL)
23 while True:
24     try:
25         pyportal.get_local_time()
26     except RuntimeError:
27         print('get_local_time failed...')
```

(continues on next page)

(continued from previous page)

```
time.sleep(60)
```

5.2 pyportal_multissid

Allows PyPortal to connect to multiple SSIDs

- Author(s): Gregory M Paris

5.2.1 Implementation Notes

Hardware:

- Adafruit PyPortal
- Adafruit PyPortal Titano
- Adafruit PyPortal Pynt

Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>
- Adafruit's PyPortal library: https://github.com/adafruit/Adafruit_CircuitPython_PyPortal

class pyportal_multissid.**PyPortal_MultiSSID** (*args, **kwargs)
Overrides PyPortal class method to allow for multiple SSIDs.

CHAPTER 6

Indices and tables

- `genindex`
- `modindex`
- `search`

p

`pyportal_multissid`, [12](#)

P

`PyPortal_MultiSSID` (*class in `pyportal_multissid`*),
12
`pyportal_multissid` (*module*), 12