
AdafruitLogger Library Documentation

Release 1.0

Dave Astels

Sep 24, 2021

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	adafruit_logging	12
5.2.1	Implementation Notes	12
6	Indices and tables	15
	Python Module Index	17
	Index	19

Logging module for CircuitPython

CHAPTER 1

Dependencies

This driver depends on:

- [Adafruit CircuitPython](#)

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

```
import adafruit_logging as logging

logger = logging.getLogger('test')

logger.setLevel(logging.ERROR)
logger.info('Info message')
logger.error('Error message')
```


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/logging_simpletest.py

```
1 # SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
2 # SPDX-License-Identifier: MIT
3
4 # pylint:disable=undefined-variable,wildcard-import,no-name-in-module
5 # pylint:disable=no-member,invalid-name
6
7 """Briefly exercise the logger and null logger."""
8
9 import adafruit_logging as logging
10
11 # This should produce an error output
12
13 logger = logging.getLogger("test")
14
15 logger.setLevel(logging.ERROR)
16 logger.info("Info message")
17 logger.error("Error message")
18
19 # This should produce no output
20
21 null_logger = logging.getLogger(None)
22
23 null_logger.setLevel(logging.ERROR)
24 null_logger.info("Info message")
25 null_logger.error("Error message")
26
27 # This should produce no output
```

(continues on next page)

```
28
29 null_logger = logging.getLogger("")
30
31 null_logger.setLevel(logging.ERROR)
32 null_logger.info("Info message")
33 null_logger.error("Error message")
```

5.2 adafruit_logging

Logging module for CircuitPython

- Author(s): Dave Astels

5.2.1 Implementation Notes

Hardware:

Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>

class `adafruit_logging.Logger`

Provide a logging api.

addHandler (*hdlr*)

Sets the handler of this logger to the specified handler. *NOTE* this is slightly different from the CPython equivalent which adds the handler rather than replacing it.

Parameters *hdlr* – the handler

critical (*format_string*, **args*)

Log a critical message.

Parameters

- **format_string** – the core message string with embedded formatting directives
- **args** – arguments to `format_string.format()`, can be empty

debug (*format_string*, **args*)

Log a debug message.

Parameters

- **format_string** – the core message string with embedded formatting directives
- **args** – arguments to `format_string.format()`, can be empty

error (*format_string*, **args*)

Log a error message.

Parameters

- **format_string** – the core message string with embedded formatting directives
- **args** – arguments to `format_string.format()`, can be empty

getEffectiveLevel ()

Get the effective level for this logger.

Returns the lowest level to output

info (*format_string*, *args)

Log a info message.

Parameters

- **format_string** – the core message string with embedded formatting directives
- **args** – arguments to `format_string.format()`, can be empty

log (*level*, *format_string*, *args)

Log a message.

Parameters

- **level** – the priority level at which to log
- **format_string** – the core message string with embedded formatting directives
- **args** – arguments to `format_string.format()`, can be empty

setLevel (*value*)

Set the logging cutoff level.

Parameters **value** – the lowest level to output

warning (*format_string*, *args)

Log a warning message.

Parameters

- **format_string** – the core message string with embedded formatting directives
- **args** – arguments to `format_string.format()`, can be empty

class `adafruit_logging.LoggingHandler`

Abstract logging message handler.

emit (*level*, *msg*)

Send a message where it should go. Place holder for subclass implementations.

format (*level*, *msg*)

Generate a timestamped message.

Parameters

- **level** – the logging level
- **msg** – the message to log

class `adafruit_logging.NullLogger`

Provide an empty logger. This can be used in place of a real logger to more efficiently disable logging.

addHandler (*hldr*)

Dummy implementation.

critical (*format_string*, *args)

Dummy implementation.

debug (*format_string*, *args)

Dummy implementation.

error (*format_string*, *args)

Dummy implementation.

getEffectiveLevel ()
Dummy implementation.

info (*format_string*, *args)
Dummy implementation.

log (*level*, *format_string*, *args)
Dummy implementation.

setLevel (*value*)
Dummy implementation.

warning (*format_string*, *args)
Dummy implementation.

class adafruit_logging.**PrintHandler**
Send logging messages to the console by using print.

emit (*level*, *msg*)
Send a message to teh console.

Parameters

- **level** – the logging level
- **msg** – the message to log

adafruit_logging.**getLogger** (*name*)
Create or retrieve a logger by name.

Parameters **name** – the name of the logger to create/retrieve None will cause the NullLogger instance to be returned.

adafruit_logging.**level_for** (*value*)
Convert a numeric level to the most appropriate name.

Parameters **value** – a numeric level

CHAPTER 6

Indices and tables

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

a

`adafruit_logging`, [12](#)

A

adafruit_logging (*module*), 12
addHandler() (*adafruit_logging.Logger method*), 12
addHandler() (*adafruit_logging.NullLogger method*), 13

C

critical() (*adafruit_logging.Logger method*), 12
critical() (*adafruit_logging.NullLogger method*), 13

D

debug() (*adafruit_logging.Logger method*), 12
debug() (*adafruit_logging.NullLogger method*), 13

E

emit() (*adafruit_logging.LoggingHandler method*), 13
emit() (*adafruit_logging.PrintHandler method*), 14
error() (*adafruit_logging.Logger method*), 12
error() (*adafruit_logging.NullLogger method*), 13

F

format() (*adafruit_logging.LoggingHandler method*), 13

G

getEffectiveLevel() (*adafruit_logging.Logger method*), 12
getEffectiveLevel() (*adafruit_logging.NullLogger method*), 13
getLogger() (*in module adafruit_logging*), 14

I

info() (*adafruit_logging.Logger method*), 13
info() (*adafruit_logging.NullLogger method*), 14

L

level_for() (*in module adafruit_logging*), 14
log() (*adafruit_logging.Logger method*), 13

log() (*adafruit_logging.NullLogger method*), 14
Logger (*class in adafruit_logging*), 12
LoggingHandler (*class in adafruit_logging*), 13

N

NullLogger (*class in adafruit_logging*), 13

P

PrintHandler (*class in adafruit_logging*), 14

S

setLevel() (*adafruit_logging.Logger method*), 13
setLevel() (*adafruit_logging.NullLogger method*), 14

W

warning() (*adafruit_logging.Logger method*), 13
warning() (*adafruit_logging.NullLogger method*), 14