

temperature

A CircuitPython helper class for unit conversion of temperature objects. It currently consists of three helpers for converting temperature values from Celsius to Fahrenheit, Fahrenheit to Celsius, Celsius to Kelvin, and Kelvin to Celsius.

- Author(s): JG for Cedar Grove Studios

Implementation Notes

Hardware:

Software and Dependencies:

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

helper_class temperature.celsius_to_fahrenheit(*deg_c*)

Helper representing the degrees Celsius to degrees Fahrenheit converter. Input value is degrees Celsius. Returns a value in degrees Fahrenheit.

Parameters:

- **deg_c** – The input value. Can be any positive or negative integer. No default value.

helper_class temperature.fahrenheit_to_celsius(*deg_f*)

Helper representing the degrees Fahrenheit to degrees Celsius converter. Input value is degrees Fahrenheit. Returns a value in degrees Celsius.

Parameters:

- **deg_f** – The input value. Can be any positive or negative integer. No default value.

helper_class temperature.celsius_to_kelvin(*deg_c*)

Helper representing the degrees Celsius to Kelvin converter. Input value is degrees Celsius. Returns a value in Kelvins.

Parameters:

- **deg_c** – The input value. Can be any positive or negative integer. No default value.

helper_class temperature.kelvin_to_celsius(*kelvins*)

Helper representing the Kelvins to degrees Celsius converter. Input value is Kelvins. Returns a value in degrees Celsius.

Parameters:

- **kelvins** – The input value. Can be any positive or negative integer. No default value.

Example:

```
>>> from cedargrove_unit_converter.temperature import *
>>> celsius_to_fahrenheit(100) # Celsius to Fahrenheit Converter
212.0

>>> celsius_to_fahrenheit(0)
32.0
```

```
>>> fahrenheit_to_celsius(32) # Fahrenheit to Celsius Converter
0.0

>>> fahrenheit_to_celsius(212)
100.0

>>> celsius_to_kelvin(0)      # Celsius to Kelvin Converter
273.15

>>> kelvin_to_celsius(273.15) # Kelvin to Celsius Converter
0.0
```

helper_class temperature.dew_point(*deg_c*, *humidity*, *verbose=False*)

Convert temperature and humidity to a dew point temperature value and observation string. Temperature input value is degrees Celsius in the range of -99 to 99 degrees. Humidity input value is percent humidity in the range of 0 to 100. Returns a dew point value in degrees Celsius and an observation string (if verbose is True) .

- Parameters:**
- **Temperature** – The temperature input value. Can be any positive or negative numeric value in the range of -99 to 99. No default value.
 - **Humidity** – The humidity percent input value. Can be any positive value in the range of 0 to 100. No default value.

Example:

```
>>> from cedargrove_unit_converter.temperature import dew_point
>>> dew_point(30, 50)      # Dew Point Converter
18.46

>>> dew_point(30, 50, verbose=True)
(18.46, 'Caution: Somewhat uncomfortable for most people.')
```

helper_class temperature.heat_index(*deg_c*, *humidity*, *verbose=False*)

Convert temperature and humidity to a heat index (comfort) temperature value and observation string. Temperature input value is degrees Celsius in the range of -99 to 99 degrees. Humidity input value is percent humidity in the range of 0 to 100. Returns a heat index value in degrees Celsius and an observation string (if verbose is True) .

- Parameters:**
- **Temperature** – The temperature input value. Can be any positive or negative numeric value in the range of -99 to 99. No default value.
 - **Humidity** – The humidity percent input value. Can be any positive value in the range of 0 to 100. No default value.

Example:

```
>>> from cedargrove_unit_converter.temperature import heat_index
>>> dew_point(30, 50)      # Dew Point Converter
18.46

>>> dew_point(30, 50, verbose=True)
(18.46, 'Caution: Somewhat uncomfortable for most people.')
```

helper_class temperature.wind_chill(*deg_c*, *wind_vel_kmph*, *verbose=False*)

Convert temperature and wind velocity to a wind chill (“feels like”) temperature value and observation string. Temperature input value is degrees Celsius in the range of -99 to 99 degrees. Wind velocity is kilometers-per-hour in the range of 0 to 100. Returns a wind chill value in degrees Celsius and an observation string (if verbose is True) .

- Parameters:**
- **Temperature** – The temperature input value. Can be any positive or negative numeric value in the range of -99 to 99. No default value.
 - **Wind Velocity** – The wind velocity input value. Can be any positive value in the range of 0 to 100. No default value.

Example:

```
>>> from cedargrove_unit_converter.temperature import wind_chill
>>> wind_chill(tt, ww) # Wind Chill Converter (_under development_)
wcwcwcwcwc
```

helper_class temperature.apparent_temperature(*deg_c*, *humidity*, *wind_vel_kmph*, *verbose=False*)

Convert temperature, humidity, and wind velocity to an apparent temperature (AT) thermal sensation value and observation string. Temperature input value is degrees Celsius in the range of -99 to 99 degrees. Humidity input value is percent humidity in the range of 0 to 100. Wind velocity is kilometers-per-hour in the range of 0 to 100. Returns an apparent temperature value in degrees Celsius and an observation string (if verbose is True) .

- Parameters:**
- **Temperature** – The temperature input value. Can be any positive or negative numeric value in the range of -99 to 99. No default value.
 - **Humidity** – The humidity percent input value. Can be any positive value in the range of 0 to 100. No default value.
 - **Wind Velocity** – The wind velocity input value. Can be any positive value in the range of 0 to 100. No default value.

Example:

```
>>> from cedargrove_unit_converter.temperature import apparent_temperature
>>> apparent_temperature(tt, hh, ww) # Apparent Temperature Converter (_under
development_)
atatatatat
```