# Python in the VE

Scripted profiles – daytime & night purge ventilation

What? This scripted profile can be used for controlling Macroflo openings (or an Air exchange); it provides day time summer temperature operation and tests for a room condition of > 23°C at approx. 11pm in order to switch on night purge ventilation

Why? We need to use a scripted profile in this instance because of the need to test at a specific time of day and to persist this state over the time-steps that constitute the night-time period. Scripted profiles offer a means to extend profile functionality when more complex logic or external data is involved
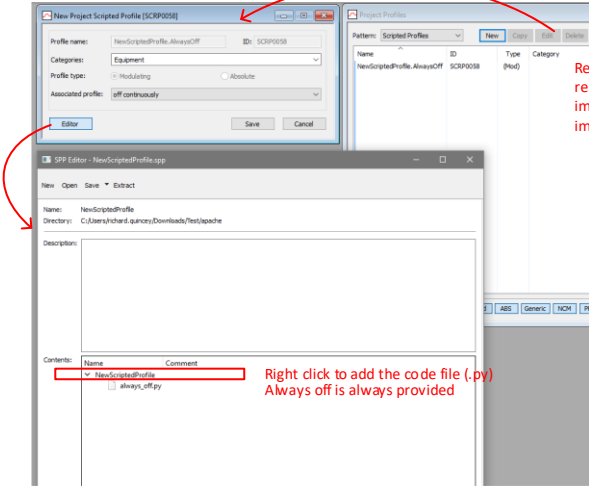
Remember to use reload to display the imported or re-imported profiles.

Right click to add the code file (.py)
Always off is always provided

Scripted profiles can be created in any Python IDE or text editor.

The code is packaged-up when the scripted profile is created in ApPro. It is saved in a .spp file in the Model Apache folder.

Any code edits will require the code file to be replaced and the package re-created.

We recommend testing & debugging any algorithms as a straight script before inserting into a profile class.

We also recommend adding comments to all scripts … use # or "" … "" for blocks of comment to aid maintenance.

Errors for scripted profiles will appear in the Task manager when you simulate.

Also refer to scripted profile help.

Script:

```python
1    # Daytime & secure night cool Window & door opening scripted profile
2
3    # This scripted profile checks:
4    # - Ta on timestep points during the daytime 8-23 hours, and
5    # - Ta at the first simulation timestep point after 11pm
6    # Apache timestep ordinates are always centred about hour so for example for a 30 min
7    # timestep ordinates are XX:45 & XX:15
8    # The context value (Ta) is always the value from the last simulation timestep e.g. for
9    # a 6 min simulation timestep:
10   # - simulation timestep ordinates are at 22:57 & 23:03
11   # - when the script tests the ordinate 23:03 the Ta value used is from the last timestep
12   #   i.e. 22:57
13   # The aps results use a reporting timestep; if this is greater than the simulation
14   # timestep the simulation timestep values are averaged over the reporting timestep for
15   # the asp file, thus if you want to see the logic working against Ta in Vista you need
16   # to chart an aps with a reporting timestep that is the same as the simulation timestep
17
18   import apache
19
20   class Occupied_Day_Secure_Night2 (apache.Profile):
21       # Window opening 8am - 11pm & occupied, Ta 22 to 26 degC, ramp opening 0-1
22       # Window opening 11pm - 8am open if Ta > 23 degC at 11pm
23       category = apache.Modulating
24
25       def setup(self):
26           # Initialise a night purge flag that will persist over timesteps
27           self.night_purge = 0
28           # Initialise opening output
29           self.opening = 0
30           # Get sim interval
31           interval = self.context.simulation.time_step_size_mins
32           # Set mins before or after the hour
33           self.mins = interval/2
34
35       def evaluate(self):
36           # Night time purge flag set each evening
37           # Use first datapoint after 11pm for test; so for 10 min sim timestep 11:05
38           if self.context.simulation.hour == 23 and self.context.simulation.minute == self.mins and self.context.surface.space.ta > 23:
39               self.night_purge = 1
40
41           # Night time purge flag reset each morning
42           if self.context.simulation.hour == 8 and self.context.simulation.minute == self.mins:
43               self.night_purge = 0
44
45           # Daytime opening control
46           if self.context.simulation.hour >= 8 and self.context.simulation.hour < 23 and self.context.surface.space.occ > 0:
47               # Daytime hours occupied ramp control
48               # Set opening value for values outside ramp limits
49               if self.context.surface.space.ta <= 22:
50                   self.opening = 0
51               if self.context.surface.space.ta > 26:
52                   self.opening = 1
53               # Set opening value for values inside ramp  limits
54               if self.context.surface.space.ta > 22 and self.context.surface.space.ta <= 26:
55                   self.opening = (self.context.surface.space.ta - 22) / 4
56           else:
57               # Outside of occupancy and daytime hours then closed
58               self.opening = 0
59
60           # Night purge opening control
61           if (self.context.simulation.hour >= 23 or self.context.simulation.hour < 8) and self.night_purge == 1:
62               # Night time hours night purge control
63               self.opening = 1
64
65           return self.opening
```

Import apache module; this is required to access IESVE scripted profile functionality

A scripted profile is defined by a Python class; one or more classes can be in each script (.py) file

Set the profile type

The setup function initializes data; this is done once. Here we initialise the night purge flag, interval & output variable. Not the use of self; this references the current instance of the class

APS file output is never on the hour; so we check the simulation timestep and set the minutes after the hour for the first sim timestep possible after 23:00; for a typical 6min timestep this is 23:03

The evaluate function is called each simulation time-step and a value returned to the profile

The .context variables are specific to the Apache module; they return the value at the previous timestep e.g. for a 6 mins timestep at 23:03 Ta is given for 22:57 . We check to see if the room temperature is > 23 degC. If it is we set the night purge flag to 1.
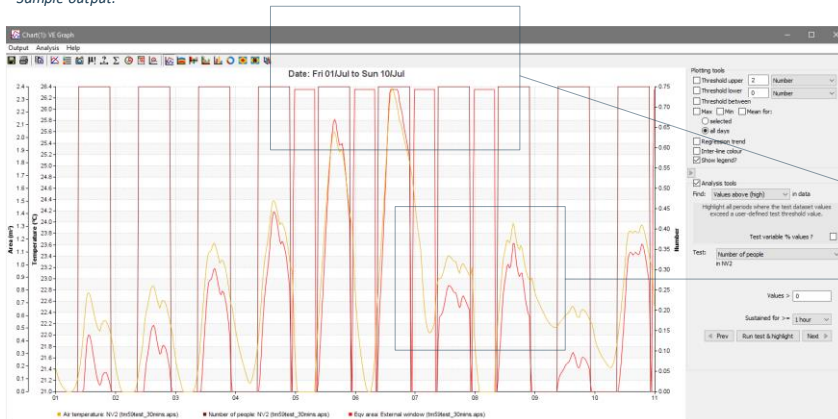
We reset the night purge flag to 0 before morning occupancy

The daytime opening control produces a ramp output of 0-1 during occupied daytime hours

If the night purge flag is set we output to 1 for all night-time hours

The class returns the scripted profile value each time-step

Sample output:

To verify scripted profiles use an aps with a reporting interval = to the simulation timestep

Night cool operation when >23°C test is met

Summer daytime operation during occupancy