
accim

Daniel Sánchez-García

Jun 21, 2023

TUTORIAL:

1	1. Requirements	1
2	2. Installation	3
3	3. Quick tutorial	5
3.1	3.1 Implementing adaptive setpoint temperatures	5
3.1.1	3.1.1 Short version	5
3.1.2	3.1.2 Long version	5
3.2	3.1 Other uses	7
4	4. Detailed use	9
4.1	4.1 First steps	9
4.2	4.2 Setting up the target IDFs	12
4.3	4.3 Full list of setpoint temperatures	19
4.4	4.4 Putting it into practice: Adaptive setpoint temperatures step by step	19
4.5	4.5 Renaming epw files for later data analysis	26
4.6	4.6 Running simulations	27
4.7	4.7 Functions and methods for data analysis; making figures and tables	29
5	5. Troubleshooting	31
5.1	5.1 Errors you might get	31
5.2	5.2 General recommendations	33
6	6. Citation	35
7	7. Credits	37
8	Using addAccis() to apply adaptive setpoint temperatures	39
8.1	1. General use	39
8.1.1	1.1 Short version	39
8.1.2	1.2 Longer version	56
8.2	2. Other uses	69
8.2.1	2.1 Storing the addAccis() class and inspecting the output IDFs	69
8.2.2	2.2 Specifying the outputs in the IDF	85
8.2.2.1	2.2.1 Using Output_type='custom'	85
8.2.2.2	2.2.2 Using Output_gen_dataframe and Output_take_dataframe	100
9	Full example of adaptive setpoint temperature simulation	127
9.1	1. IDF (using addAccis())	127
9.2	2. EPW (using rename_epw_files())	139
9.3	3. Running the simulation (using runEp())	142

9.4	4. Analysing and visualising the data (using <code>Table()</code>)	145
9.4.1	4.1 Visualizing the data	148
9.4.2	4.2 Analysing the data	153
10	Using <code>rename_epw_files()</code> to rename the EPWs for proper data analysis after simulation	171
11	Using <code>runEp()</code> to directly run simulations with EnergyPlus	175
12	Using the class <code>Table()</code> for data analysis	179
12.1	1. Getting the full dataset of simulation results	180
12.2	2. Filtering results	187
12.3	3. Making graphs	193
12.3.1	3.1 Scatter plot:	193
12.3.2	3.2 Adaptive vs Static data scatter plot:	195
12.4	4. Making tables	196
13	accim	215
13.1	accim package	215
13.1.1	Subpackages	215
13.1.1.1	accim.data package	215
13.1.1.1.1	Submodules	215
13.1.1.1.2	accim.data.data_postprocessing module	215
13.1.1.1.3	accim.data.data_preprocessing module	220
13.1.1.1.4	Module contents	220
13.1.1.2	accim.run package	220
13.1.1.2.1	Submodules	220
13.1.1.2.2	accim.run.run module	220
13.1.1.2.3	Module contents	221
13.1.1.3	accim.sim package	221
13.1.1.3.1	Submodules	221
13.1.1.3.2	accim.sim.accim_Base module	221
13.1.1.3.3	accim.sim.accim_Base_EMS module	221
13.1.1.3.4	accim.sim.accim_ExistingHVAC module	222
13.1.1.3.5	accim.sim.accim_ExistingHVAC_EMS module	222
13.1.1.3.6	accim.sim.accim_IDFgeneration module	223
13.1.1.3.7	accim.sim.accim_Main module	223
13.1.1.3.8	accim.sim.accim_VRFsystem module	225
13.1.1.3.9	accim.sim.accim_VRFsystem_EMS module	226
13.1.1.3.10	accim.sim.accis module	226
13.1.1.3.11	Module contents	227
13.1.2	Module contents	227
14	Indices and tables	229
Python Module Index		231
Index		233

**CHAPTER
ONE**

1. REQUIREMENTS

To use accim, the following must be installed:

- **Python 3.9**
 - When installing it, tick the checkbox “Add Python 3.9 to PATH”
 - Disable path length limit
- EnergyPlus (any version between 9.1 and 23.1 those included) in default path (C:\EnergyPlusV23-1-0)

Also, Jupyter Notebook is recommended, in order to run the notebooks within path accim/sample_files/jupyter_notebook

**CHAPTER
TWO**

2. INSTALLATION

First of all, you need to install the package. To install the latest version, execute:

```
pip install accim
```

To install some specific version, execute the same code followed by “==version number”. For instance, to install version 0.6.11:

```
pip install accim==0.6.11
```


3. QUICK TUTORIAL

3.1 3.1 Implementing adaptive setpoint temperatures

This is a very brief explanation of the usage. Therefore, if you don't get the results you expected or get some error, I would recommend reading the 'Detailed use' section at the documentation in the link below.

accim will take as input IDF files those located at the same path as the script. You only need to run the following code:

3.1.1 3.1.1 Short version

```
from accim.sim import accis
accis.addAccis()
```

Once you run this code, you will be asked to enter some information at the terminal or python console to generate the output IDF files.

3.1.2 3.1.2 Long version

```
from accim.sim import accis
accis.addAccis(
    ScriptType=str, # ScriptType: 'vrf_mm', 'vrf_ac', 'ex_mm', 'ex_ac'. For instance: ↵
    ↵ScriptType='vrf_ac',
    SupplyAirTempInputMethod=str, # SupplyAirTempInputMethod: 'supply air temperature',
    ↵'temperature difference'. For instance: SupplyAirTempInputMethod='supply air temperature',
    Output_keep_existing=bool, # Output_keep_existing: True or False. For instance: ↵
    ↵Output_keep_existing=False,
    Output_type=str, # Output_type: 'simplified', 'standard', 'detailed' or 'custom'. For ↵
    ↵instance: Output_type='standard',
    Output_freqs=list, # Output_freqs: ['timestep', 'hourly', 'daily', 'monthly', 'runperiod']. ↵
    ↵For instance: Output_freqs=['hourly', 'runperiod'],
    Output_gen_dataframe=bool, # Output_gen_dataframe: True or False. For instance: ↵
    ↵Output_gen_dataframe=False,
    Output_take_dataframe=pandas Dataframe,
    EnergyPlus_version=str, # EnergyPlus_version: '9.1', '9.2', '9.3', '9.4', '9.5', '9.6', '22.1
    ↵', '22.2' or '23.1'. For instance: EnergyPlus_version='23.1',
    TempCtrl=str, # TempCtrl: 'temperature' or 'temp', or 'pmv'. For instance: TempCtrl='temp
    ↵',
    ComfStand=list, # it is the Comfort Standard. Can be any integer from 0 to 21. For ↵
```

(continues on next page)

(continued from previous page)

```

→ instance: ComfStand=[0, 1, 2, 3],
    CAT=list, # it is the Category. Can be 1, 2, 3, 80, 85 or 90. For instance: CAT=[3, ↵
→ 80],
    ComfMod=list, # it is Comfort Mode. Can be 0, 1, 2 or 3. For instance: ComfMod=[0, ↵
→ 3],
    SetpointAcc=float, # it is the accuracy of the setpoint temperatures
    CoolSeasonStart=dd/mm date in string format or integer to represent the day of the ↵
→ year, # it is the start date for the cooling season
    CoolSeasonEnd=dd/mm date in string format or integer to represent the day of the ↵
→ year, # it is the end date for the cooling season
    HVACmode=list, # it is the HVAC mode. 0 for Full AC, 1 for NV and 2 for MM. For ↵
→ instance: HVACmode=[0, 2],
    VentCtrl=list, # it is the Ventilation Control. Can be 0 or 1. For instance: ↵
→ VentCtrl=[0, 1],
    MaxTempDiffVOF=float, # When the difference of operative and outdoor temperature ↵
→ exceeds MaxTempDiffVOF, windows will be opened the fraction of MultiplierVOF. For ↵
→ instance: MaxTempDiffVOF=20,
    MinTempDiffVOF=float, # When the difference of operative and outdoor temperature is ↵
→ smaller than MinTempDiffVOF, windows will be fully opened. Between min and max, ↵
→ windows will be linearly opened. For instance: MinTempDiffVOF=1,
    MultiplierVOF=float, # Fraction of window to be opened when temperature difference ↵
→ exceeds MaxTempDiffVOF. For instance: Multiplier=0.2,
    VSToffset=list, # it is the offset for the ventilation setpoint. Can be any number, ↵
→ float or int. For instance: VSToffset=[-1.5, -1, 0, 1, 1.5],
    MinOToffset=list, # it is the offset for the minimum outdoor temperature to ↵
→ ventilate. Can be any positive number, float or int. For instance: MinOToffset=[0.5, 1, ↵
→ 2],
    MaxWindSpeed=list, # it is the maximum wind speed allowed for ventilation. Can be ↵
→ any positive number, float or int. For instance: MinOToffset=[2.5, 5, 10],
    ASTtol_start=float, # it is the start of the tolerance sequence. For instance: ↵
→ ASTtol_start=0,
    ASTtol_end_input=float, # it is the end of the tolerance sequence. For instance: ↵
→ ASTtol_start=2,
    ASTtol_steps=float, # these are the steps of the tolerance sequence. For instance: ↵
→ ASTtol_steps=0.25,
    NameSuffix=str # NameSuffix: some text you might want to add at the end of the ↵
→ output IDF file name. For instance: NameSuffix='whatever',
    verboseMode=bool # verboseMode: True to print all process in screen, False to not to ↵
→ print it. Default is True. For instance: verboseMode=True,
    confirmGen=bool # True to confirm automatically the generation of IDFs; if False, you ↵
→ 'll be asked to confirm in command prompt. Default is False. For instance: ↵
→ confirmGen=False,
)

```

3.2 3.1 Other uses

Although the main use of accim is the implementation of adaptive setpoint temperatures, there are some functions, classes and methods that allow to roughly automate the whole process consisting of preparation of the epw and idf files, the simulation runs and the data analysis. For further information, please refer to the How-to Guides section in this documentation, which contains some Jupyter Notebooks that can also be found in accim's installation folder.

4. DETAILED USE

4.1 4.1 First steps

There has been developed 4 main branches of functions, which are:

- VRF system with full air-conditioning mode: This mode has been developed mainly to support models originated with OpenStudio, which up to date does not support Airflow Network objects and subsequently Calculated Natural Ventilation. It adds standard VRF systems for each occupied zone and applies the adaptive or PMV-based setpoint temperatures, but only works with full air-conditioning mode.
- VRF system with mixed mode: It adds standard VRF systems for each occupied zone and applies the adaptive or PMV-based setpoint temperatures. Works with Calculated Natural Ventilation, although full air-conditioning mode can also be used. If mixed mode is used, the model must be generally developed with DesignBuilder.
- Existing HVAC system only with full air-conditioning mode: Keeps the existing HVAC systems and modify the existing setpoint temperatures to adaptive or PMV-based setpoint temperatures. However, mixed-mode and naturally ventilated modes are not available in this mode.
- Existing HVAC system with mixed mode: UNDER DEVELOPMENT. IT IS NOT ADVISABLE TO USE IT YET. Keeps the existing HVAC systems and modify the existing setpoint temperatures to adaptive or PMV-based setpoint temperatures, considering mixed-mode. In order to properly work, there must be only one object for heating and another for cooling that can be used to monitor if these are turned on at any timestep (such as `Coil:Cooling:Water` and `Coil:Heating:Water`). Also, these objects must be named following the pattern “Zone name” “Object name”. For instance, an `Coil:Heating:Electric` object could be named `Block1:Zone1 PTAC Heating Coil`, given that `Block1:Zone1` is a valid zone name. On the other hand, a `Coil:Cooling:Water` object named `Main Cooling Coil 1` would not be valid, since in this case the room would be `Main`; this is the typical case of some equipment shared by multiple rooms. If this condition is not met, accim will not generate the output IDF files for that input IDF file. For instance, if there are `Coil:Heating:Electric` and `Coil:Heating:DX:SingleSpeed` objects in the same model, simulation will crash. Also, if there is just an `ZoneHVAC:Baseboard:RadiantConvective:Water` used for heating, and cooling is not monitored, simulation will also crash.

Therefore, if you are going to use the VRF system script, you’re supposed to have one or multiple IDFs with fixed setpoint temperature, or even without any HVAC objects at all (it doesn’t matter, since the module is going to add a standard VRF system for each zone, and the simulation is going to be calculated with these VRF systems), and with Calculated Natural Ventilation if you’re going to use the Mixed Mode. On the other hand, if you are going to use any ExistingHVAC script, again you’re supposed to have one or multiple IDFs, however in this case there must be a fully functional HVAC system. Therefore, you must be able to successfully run a simulation with fixed setpoint temperatures in order for the accim package to work. The main difference between ExistingHVAC only with full air-conditioning and with mixed mode is that in the latter, the existing HVAC system needs to be mapped in order to monitor if it needs to be activated or not, and windows need to be actuated in case conditions for natural ventilation are favourable. In both cases, when you export the IDF, please do not request ASHRAE 55 or CEN 15251 results. accim will do so by adding the relevant fields to the People objects.

By using any ExistingHVAC script you might not get the results that you expect, even if there are no errors in the accim and simulation processes. The reason lies on the HVAC system itself, and that is why the VRFsystem script has been developed, because it has been tested that it works.

No matter what type or functions are you going to use, the language of the software used to create the input IDF should be English (for example, if you use Designbuilder in Spanish, accim won't work properly), and it's not recommended to use any non-standard characters in the input IDF, just like written accents or “ñ”.

Said that, accim will transform all the IDF files located in the same path where script is. Therefore, the quickest way to run the script is opening a prompt command dialog in the folder where the IDF files are located (you can do this by holding Ctrl and right-click inside the folder, and click on ‘open PowerShell window here’). Then run Python by typing ‘python’ in the command prompt.

First you need to import the module ‘accis’ (stands for Adaptive-Comfort-Control-Implementation Script):

```
>>> from accim.sim import accis
```

And then, you just need to call the accis function:

```
>>> accis.addAccis()
```

Then you'll be asked in the prompt to enter some information so that python knows how do you want to set up the output IDFs:

```
-----  
Adaptive-Comfort-Control-Implemented Model (ACCIM)  
-----
```

This tool allows to apply adaptive setpoint temperatures.

For further information, please read the documentation:

<https://accim.readthedocs.io/en/master/>

For a visual understanding of the tool, please visit the following jupyter notebooks:

- Using addAccis() to apply adaptive setpoint temperatures

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/addAccis/using_addAccis.ipynb

- Using rename_epw_files() to rename the EPWs for proper data analysis after simulation

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/rename_epw_files/using_rename_epw_files.ipynb

- Using runEp() to directly run simulations with EnergyPlus

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/runEp/using_runEp.ipynb

- Using the class Table() for data analysis

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/Table/using_Table.ipynb

Now, you are going to be asked to enter some information for different arguments to generate the output IDFs with adaptive setpoint temperatures.

If you are not sure about how to use these parameters, please take a look at the documentation in the following link:

<https://accim.readthedocs.io/en/latest/how%20to%20use.html>

Please, enter the following information:

Enter the ScriptType (

(continues on next page)

(continued from previous page)

```

for VRFsystem with full air-conditioning mode: vrf_ac;
for VRFsystem with mixed-mode: vrf_mm;
for ExistingHVAC with mixed mode: ex_mm;
for ExistingHVAC with full air-conditioning mode: ex_ac
): vrf_mm

Enter the SupplyAirTempInputMethod (
for Supply Air Temperature: supply air temperature;
for Temperature Difference: temperature difference;
): temperature difference

Do you want to keep the existing outputs (true or false)?: false

Enter the Output type (standard, simplified, detailed or custom): standard

Enter the Output frequencies separated by space (timestep, hourly, daily, monthly, runperiod): hourly runperiod

Do you want to generate a dataframe to see all outputs? (true or false): false

Enter the EnergyPlus version (9.1 to 23.1): 23.1

Enter the Temperature Control method (temperature or pmv): temperature

```

where:

- ScriptType can be ‘vrf_mm’, ‘vrf_ac’, ‘ex_mm’ or ‘ex_ac’, and it refers to the type of functions as explained above
- SupplyAirTempInputMethod can be ‘supply air temperature’ or ‘temperature difference’, and it is the supply air temperature input method for the VRF systems.
- Existing outputs in the IDF can be kept if entered ‘true’. Otherwise, if entered ‘false’, it will be removed for clarity purposes at results stage.
- Output_type can be ‘standard’, ‘detailed’, ‘simplified’ or ‘custom’ and it refers to the simulation results: ‘standard’ means that results will contain the full selection relevant to accim; ‘detailed’ is mainly used for testing the software tool; ‘simplified’ means that results are just going to be the hourly operative temperature and VRF consumption of each zone, mainly used when you need the results not to be heavy files, because you are going to run a lot of simulations and capacity is limited; and finally, ‘custom’ allows the user to specify the outputs to be kept or removed by entering them in the python console.
- Output_freqs (Output frequencies) can be timestep, hourly, daily, monthly and/or runperiod, and these must be entered separated by space. It will add the specified output type (standard or simplified) in all entered frequencies.
- Also, a pandas DataFrame instance can be created containing all Output:Variable objects. This allows the user to filter the DataFrame as needed, so that it only contains the needed Output:Variable objects, and then it can be entered in the argument Output_take_dataframe
- EnergyPlus_version can be from ‘9.1’ to ‘23.1’. It is the version of EnergyPlus you have installed in your computer. If you enter ‘9.1’, accim will look for the E+9.1.0 IDD file in path “C:\EnergyPlusV9-1-0”.
- Temperature Control method can be ‘temperature’ or ‘temp’, or ‘pmv’. If ‘temp’ is used, the setpoint will be the operative temperature, otherwise if ‘pmv’ is used, the setpoint will be the PMV index.

accis will show on the prompt command dialog all the objects it adds, and those that doesn’t need to be added because were already in the IDF, and finally ask you to enter some values to set up the IDFs as you desire. Please refer to the

section titled ‘Setting up the target IDFs’.

Once you run the simulations, you might get some EnergyPlus warnings and severe errors. This is something I’m currently working on.

4.2 4.2 Setting up the target IDFs

If you have run `accis.addAccis()`, you will be asked in the prompt to enter a few more values separated by space to set up the desired IDFs. However, you can also skip the command prompt process by running `accis` directly including the arguments in the function, whose usage would be:

```
>>> accis.addAccis(str, # ScriptType: 'vrf_mm', 'vrf_ac', 'ex_mm', 'ex_ac'
>>>           str, # SupplyAirTempInputMethod: 'supply air temperature', 'temperature_
->difference'
>>>           bool, # Output_keep_existing: True or False
>>>           str, # Output_type: 'simplified', 'standard', 'detailed' or 'custom'
>>>           list, # Output_freqs: ['timestep', 'hourly', 'daily', 'monthly', 'runperiod']
>>>           bool, # Output_gen_dataframe: True or False
>>>           pandas DataFrame, # Output_take_dataframe
>>>           str, # EnergyPlus_version: '9.1', '9.2', '9.3', '9.4', '9.5', '9.6', '22.1',
->'22.2' or '23.1'
>>>           str, # TempCtrl: 'temperature' or 'temp', or 'pmv'
>>>           list, # ComfStand, which is the Comfort Standard
>>>           list, # CAT, which is the Category
>>>           list, # ComfMod, which is Comfort Mode
>>>           float, # SetpointAcc, which defines the accuracy of the setpoint_
->temperatures
>>>           str containing a date in format dd/mm, or an int # CoolSeasonStart
>>>           str containing a date in format dd/mm, or an int # CoolSeasonEnd
>>>           list, # HVACmode, which is the HVAC mode
>>>           list, # VentCtrl, which is the Ventilation Control
>>>           float, # MaxTempDiffVOF
>>>           float, # MinTempDiffVOF
>>>           float, # MultiplierVOF
>>>           list, # VSToffset
>>>           list, # MinOToffset
>>>           list, # MaxWindSpeed
>>>           float, # ASTtol start
>>>           float, # ASTtol end
>>>           float, # ASTtol steps
>>>           str # NameSuffix: some text you might want to add at the end of the_
->output IDF file name
>>>           bool # verboseMode: True to print all process in screen, False to not_
->to print it. Default is True.
>>>           bool # confirmGen: True to confirm automatically the generation of_
->IDFs; if False, you'll be asked to confirm in command prompt. Default is False.
>>>           )
```

Some example of the usage could be:

```
>>> accis.addAccis(SCRIPTYPE='vrf_mm', # ScriptType: 'vrf_mm', 'vrf_ac', 'ex_mm', 'ex_ac'
>>>           SupplyAirTempInputMethod='supply air temperature', #
```

(continues on next page)

(continued from previous page)

```

→ SupplyAirTempInputMethod: 'supply air temperature', 'temperature difference'
>>> Output_keep_existing=False, # Output_keep_existing: True or False
>>> Output_type='standard', # Output_type: 'simplified' or 'standard'
>>> Output_freqs=['hourly', 'runperiod'], # Output_freqs: ['timestep',
→ 'hourly', 'daily', 'monthly', 'runperiod']
>>> Output_gen_dataframe=False,
>>> # we just omit Output_take_dataframe
>>> EnergyPlus_version='9.5', # EnergyPlus_version: '9.1', '9.2', '9.3', '9.4',
→ '9.5', '9.6', '22.1', '22.2' or '23.1'
>>> TempCtrl='temp', # Temperature Control: 'temperature' or 'temp', or 'pmv'
>>> ComfStand=[0, 1, 2, 3], # ComfStand, which is the Comfort Standard
>>> CAT=[1, 2, 3, 80, 90], # CAT, which is the Category
>>> ComfMod=[0, 1, 2, 3], # ComfMod, which is Comfort Mode
>>> SetpointAcc=10, # Therefore, setpoints will be rounded to the first
→ decimal
>>> # we just omit CoolSeasonStart, since the default date is May 1st
>>> # we just omit CoolSeasonEnd, since the default date is September 1st
>>> HVACmode=[0, 1, 2], # HVACmode, which is the HVAC mode
>>> VentCtrl=[0, 1], # VentCtrl, which is the Ventilation Control
>>> MaxTempDiffVOF=20, # When the difference of operative and outdoor
→ temperature exceeds 20°C, windows will be opened the fraction of MultiplierVOF.
>>> MinTempDiffVOF=1, # When the difference of operative and outdoor
→ temperature is smaller than 1°C, windows will be fully opened. Between min and max, windows will be linearly opened.
>>> MultiplierVOF=20, # Fraction of window to be opened when temperature
→ difference exceeds MaxTempDiffVOF.
>>> VSToffset=[0, 1, 2], # VSToffset, which is the Ventilation Setpoint
→ Temperature offset
>>> MinOToffset=[0, 1, 2], # MinOToffset, which is the Minimum Outdoor
→ Temperature offset
>>> MaxWindSpeed=[10, 20, 30], # MaxWindSpeed, which is the Maximum Wind
→ Speed
>>> ASTtol_start=0, # ASTtol_start, which is the start of the tolerance
→ sequence
>>> ASTtol_end_input=2, # ASTtol_end_input, which is the end of the
→ tolerance sequence
>>> ASTtol_steps=0.25, # ASTtol_steps, which are the steps of the
→ tolerance sequence
>>> NameSuffix='standard' # Name Suffix: for example, just in case you
→ want to clarify the outputs
>>> )

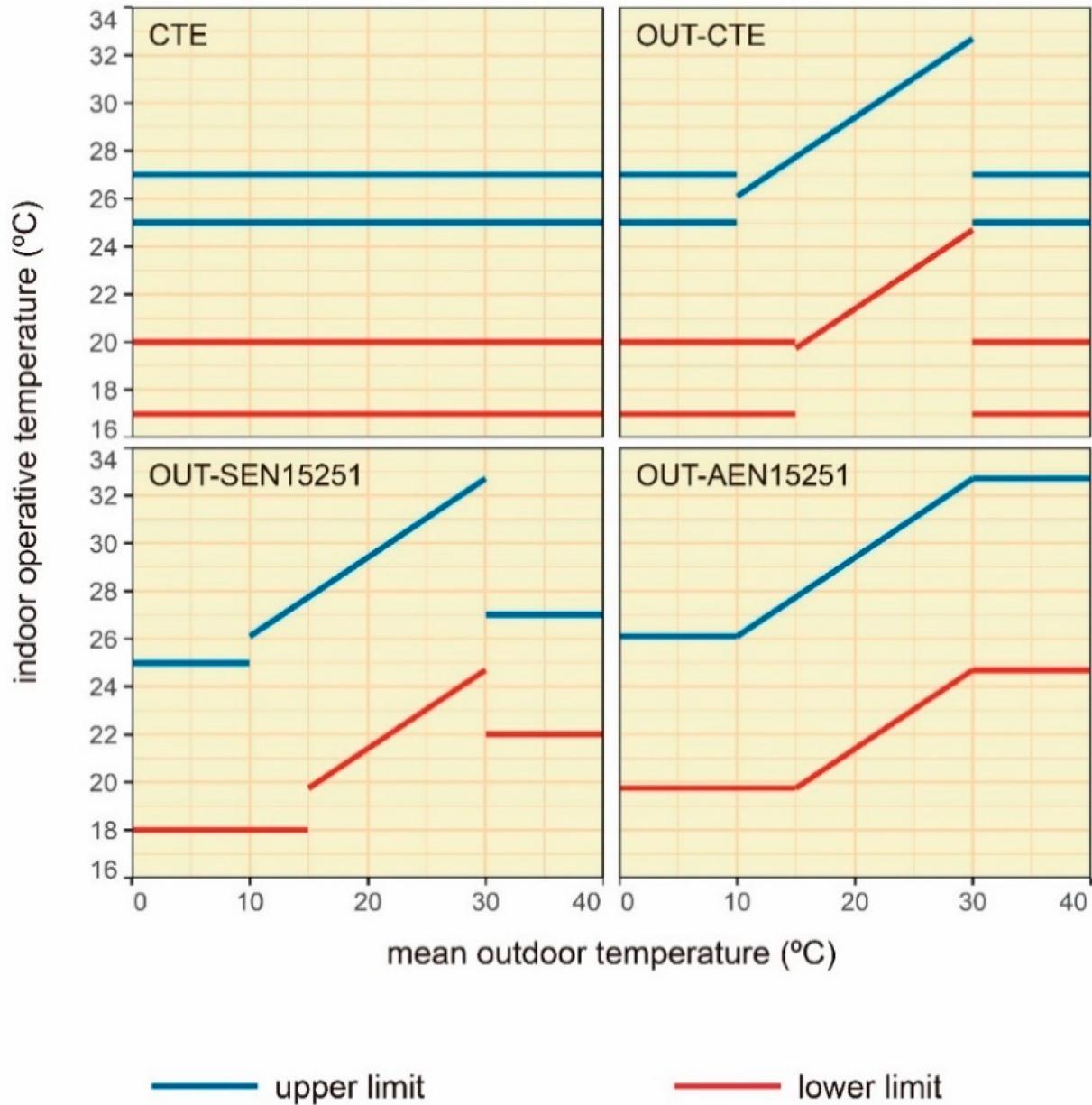
```

For clarity purposes, it's recommended to specify the argument name as well, as shown above. If you don't specify all arguments, you'll be asked to enter them at the prompt command, and these values will be used instead of those specified in the function call. Each argument is explained below:

- ComfStand: refers to the thermal comfort standard or model to be applied. Enter any number from 0 to 21 to select the comfort standard or model to be used; you can see which model is each number in the table below. Readthedocs doesn't render it properly, so please take a look at the [Github repository](#). For example, if you enter '0 1 2 3', you'll get IDFs for CTE, EN16798-1, ASHRAE 55 and the local model developed by Rijal et al for Japanese dwellings. If you don't enter any number, it'll ask you to enter the numbers again.

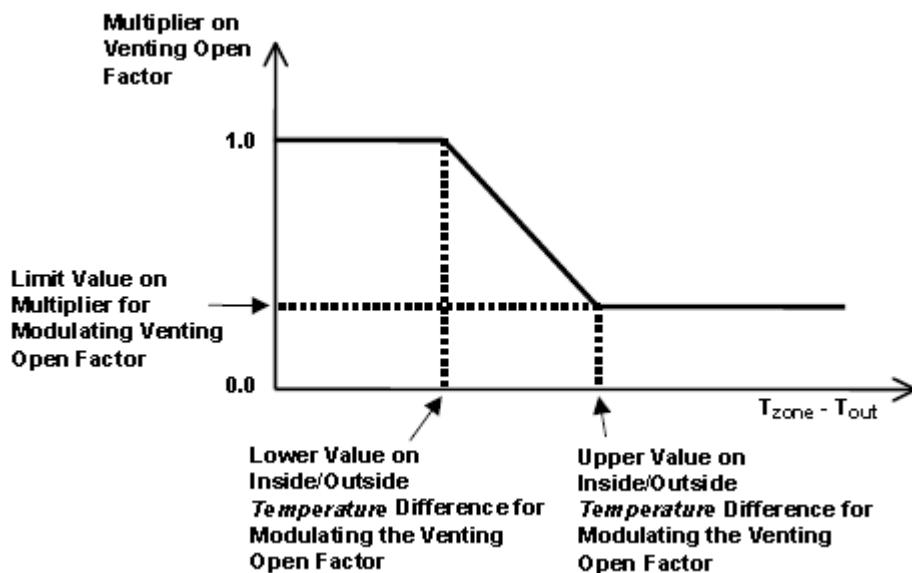
Co m f Sta n d No.	Com f Sta ndNa me	A r e a	Reference	
0	E S P C T E	S p a i n	The Government of Spain. Royal Decree 314/2006. Approving the Spanish Technical Building Code CTE-DB-HE-1 2013:1-43. https://www.boe.es/eli/es/rd/2006/03/17/314 (accessed August 6, 2021).	
1	I N T E N 1 6 7 9 8	E u r o p e	European committee for standardization. EN 16798-1:2019 Energy performance of buildings. Ventilation for buildings. Indoor environmental input parameters for design and assessment of energy performance of buildings addressing indoor air quality, thermal environment, lighting and acoustics. 2019. https://en.tienda.aenor.com/norma-bsi-bs-en-16798-1-2019-000000000030297474 (accessed August 6, 2021).	
2	I N T A S H R A E 5 5	W o r l d w i d e	ASHRAE Standard 55-2020 Thermal Environmental Conditions for Human Occupancy, ASHRAE Standard (2020).	
3	J P N R i j a l	J a p a n	Rijal, H. B., Humphreys, M. A., & Nicol, J. F. (2019). Adaptive model and the adaptive mechanisms for thermal comfort in Japanese dwellings. Energy and Buildings, 202, 109371. https://doi.org/10.1016/j.enbuild.2019.109371	
4	C H N G B T 5 0 7 8 5 C o l d	C h i n a	MOHURD, Evaluation Standard for Indoor Thermal Environment in Civil Buildings (GB/T 50785-2012), Ministry of Housing and Urban-Rural Development (MOHURD), Beijing, China, 2012.	
5	C H N G B T 5 0 7 8 5 H o t M i l d	C h i n a	MOHURD, Evaluation Standard for Indoor Thermal Environment in Civil Buildings (GB/T 50785-2012), Ministry of Housing and Urban-Rural Development (MOHURD), Beijing, China, 2012.	
6	C H N Y a n g	C h i n a	Yang, L., Fu, R., He, W., He, Q., & Liu, Y. (2020). Adaptive thermal comfort and climate responsive building design strategies in dry-hot and dry-cold areas: Case study in Turpan, China. Energy and Buildings, 209, 109678. https://doi.org/10.1016/j.enbuild.2019.109678	
7	I N D I M A C C N V	I n d i a	Manu, S., Shukla, Y., Rawal, R., Thomas, L. E., & de Dear, R. (2016). Field studies of thermal comfort across multiple climate zones for the subcontinent: India Model for Adaptive Comfort (IMAC). Building and Environment, 98, 55–70. https://doi.org/10.1016/j.buildenv.2015.12.019	
8	I N D I M A C C M M	I n d i a	Manu, S., Shukla, Y., Rawal, R., Thomas, L. E., & de Dear, R. (2016). Field studies of thermal comfort across multiple climate zones for the subcontinent: India Model for Adaptive Comfort (IMAC). Building and Environment, 98, 55–70. https://doi.org/10.1016/j.buildenv.2015.12.019	
9	I N D I M A C R 7 D R M	I n d i a	Rawal, R., Shukla, Y., Vardhan, V., Asrani, S., Schweiker, M., de Dear, R., Garg, V., Mathur, J., Prakash, S., Diddi, S., Ranjan, S. V., Siddiqui, A. N., & Soman, G. (2022). Adaptive thermal comfort model based on field studies in five climate zones across India. Building and Environment, 219, 109187. https://doi.org/10.1016/J.BUILENV.2022.109187	
10	I N D I M A C R 3 0 D R M	I n d i a	Rawal, R., Shukla, Y., Vardhan, V., Asrani, S., Schweiker, M., de Dear, R., Garg, V., Mathur, J., Prakash, S., Diddi, S., Ranjan, S. V., Siddiqui, A. N., & Soman, G. (2022). Adaptive thermal comfort model based on field studies in five climate zones across India. Building and Environment, 219, 109187. https://doi.org/10.1016/J.BUILENV.2022.109187	
11	I N D D h a k a	I n d i a	Dhaka, S., Mathur, J., Brager, G., & Honnekeri, A. (2015). Assessment of thermal environmental conditions and quantification of thermal adaptation in naturally ventilated buildings in composite climate of India. Building and Environment, 86, 17–28. https://doi.org/10.1016/J.BUILENV.2014.11.024	
14				Chapter 4. 4. Detailed use
12	R O M U d r e a	R o m a n i	Udrea, I., Croitoru, C., Nastase, I., Crutescu, R., & Badescu, V. (2018). First adaptive thermal comfort equation for naturally ventilated buildings in Bucharest, Romania. International Journal of Ventilation, 17(3), 149–165. https://doi.org/10.1080/14733315 .	

- CAT: refers to the category of the thermal comfort model applied. Most of the Comfort Standards work with 80 and 90% acceptability levels, except the European EN 16798-1 (works with Categories 1, 2 and 3), the Chinese GB/T 50785 (works with categories 1 and 2), and the India Model for Adaptive Comfort - Commercial (which works with 80, 85 and 90% acceptability levels). So, for example, if you are going to use the EN16798-1 (ComfStand = 1), you can enter '1 2 3' to generate setpoint temperatures for Categories 1, 2 and 3. Or, if you are going to use the IMAC Commercial in naturally ventilated mode (ComfStand = 7), you can enter '80 85 90' to generate setpoint temperatures for these acceptability levels. All categories are referenced in the [full list of setpoint temperatures](#) at the end of this section. Please note that the Category values must be consistent with the Comfort Standard values previously entered. If, for instance, you enter '1' in the Comfort Standard value (means you're asking for EN16798 model), but then enter '80' or '90' in the Category value (which are categories used in ASHRAE55), you won't get the results you want.
- ComfMod: is the Comfort Mode, and refers to the comfort modes used in accim. It controls if the setpoints are static (when ComfMod = 0) or adaptive (when ComfMod = 1, 2 or 3). When they are adaptive, it also controls the comfort model applied when the adaptive model is not applicable (that is, when the running mean outdoor temperature limits are exceeded), in which case a PMV-based model is applied. Each ComfMod for each ComfStand and CAT is referenced at the [full list of setpoint temperatures](#). Please refer to the research article <https://www.mdpi.com/1996-1073/12/8/1498> for more information. Figure below shows the variation of setpoint temperatures when ComfMod 0 (upper left), 1 (upper right), 2 (lower left) and 3 (lower right), when ComfStand is 1 (EN 16798-1, although figure shows the superseded standard, but the setpoint behaviour is similar)
- SetpointAcc: refers to the accuracy of the setpoint temperatures. Any number, integer or float, can be entered in this argument. For instance, if 1 was entered, the cooling setpoint would be rounded to the nearest integer below adaptive upper comfort limit minus tolerance (ASTtol), and the heating setpoint would be rounded to the nearest integer above adaptive lower comfort limit plus tolerance. If 27.46 and 20.46 were the upper and lower comfort limits and its tolerances were respectively -0.1 and +0.1, then the nearest integers to 27.36 and 20.56 would be 27 and 21, and therefore, these would be the cooling and heating setpoint temperatures. If 2 was used instead, then the rounding would be done to the nearest half. If 10 were used, the rounding would be done to the first decimal. If 0.5 or 0.1 were used, the rounding would be done respectively every 2 or 10 celsius degrees.
- CoolSeasonStart: it is the start of the cooling season, only used when EN16798-1, ASHRAE 55 or ISO7730 are entered in ComfStand (respectively, ComfStand = 1, 2 and 22) and setpoint behaviour is set to static (i.e. ComfMod = 0). This argument can take the number of the day in the year (i.e. an integer) or a string containing a date in format dd/mm (for instance, "01/05"). Values of CoolSeasonStart greater than CoolSeasonEnd can be used, therefore denoting the location of the EPW file should be in the south hemisphere.
- CoolSeasonEnd: Similar to CoolSeasonStart, but it is the end of the cooling season. Again, only used when EN16798-1, ASHRAE 55 or ISO7730 are entered in ComfStand (respectively, ComfStand = 1, 2 and 22) and setpoint behaviour is set to static (i.e. ComfMod = 0). Again, this argument can take the number of the day in the year (i.e. an integer) or a string containing a date in format dd/mm (for instance, "01/05"). Values of CoolSeasonEnd smaller than CoolSeasonStart can be used, therefore denoting the location of the EPW file should be in the south hemisphere.
- HVACmode: refers to the HVAC mode applied. Enter 0 for Fully Air-conditioned (AC), 1 for Naturally ventilated (NV) and/or 2 for Mixed Mode (MM). Please note that Calculated natural ventilation must be enabled so that Mixed Mode works. So, for example, if you enter '0 1 2' you'll be getting all HVAC modes, or if you just enter '0 1' you'll be getting just Fully Air-conditioned and Naturally ventilated.
- VentCtrl: refers to the ventilation control, only used in for NV and MM. When using NV, If you enter '0', ventilation will be allowed if operative temperature exceeds neutral temperature (also known as comfort temperature); if you enter '1', ventilation will be allowed if operative temperature exceeds the upper comfort limit. In other words, sets the value of the neutral temperature or the upper comfort limit to the Ventilation Setpoint Temperature (VST). When using MM, 0 = Ventilates above neutral temperature and fully opens doors and windows; 1 = Ventilates above lower comfort limit and fully opens doors and windows; 2 = Ventilates above neutral temperature and opens doors and windows based on the customised venting opening factor; and 3 = Ventilates above lower comfort limit and opens doors and windows based on the customised venting opening factor. Either way,



if you enter '0 1' you'll be getting both ventilation control modes.

- MaxTempDiffVOF: Maximum Temperature Difference for Venting Opening Factor. Maximum temperature difference between indoor operative and outdoor temperatures, which when exceeded, windows and doors are opened only the fraction specified in the MultiplierVOF argument. If temperature difference oscillates between maximum and minimum, the windows and doors are opened based on the linear equation. Follows the same operation as explained in [Designbuilder help website](#).
- MinTempDiffVOF: Minimum Temperature Difference for Venting Opening Factor. Minimum temperature difference between indoor operative and outdoor temperatures, which when smaller, windows and doors are fully opened. If temperature difference oscillates between maximum and minimum, the windows and doors are opened based on the linear equation. Follows the same operation as explained in [Designbuilder help website](#).
- MultiplierVOF: Multiplier for modulating the Venting Opening Factor. The fraction of the windows that will be opened when temperature difference exceeds MaxTempDiffVOF. Follows the same operation as explained in [Designbuilder help website](#).



- VSToffset: stands for Ventilation Setpoint Temperature (VST) offset, again only used in Mixed Mode (HVAC Mode '2'). Applies the entered values as an offset to the VST, in Celsius degrees. Values entered can be positive or negative float or integers, and must be space-separated. For example, if you enter '-2 -1 0 1 2' you'll be getting offsets of -2°C, -1°C, 0°C, 1°C and 2°C to the VST. If you don't enter any number, it'll be used '0' as the default value.
- MinOToffset: stands for Minimum Outdoor Temperature offset, again only used in Mixed Mode (HVAC Mode '2'). Sets the minimum outdoor temperature an offset to the heating setpoint temperature. For example, if you enter '1' (please, note that the numbers must be positive), ventilation won't be allowed if outdoor temperature falls below 1°C below the heating setpoint, in order to prevent from entering excessive cold. Therefore, below said limit, windows are closed and, if needed, air conditioning starts to work. Entered values can be float or integers, but always positive numbers, and must be space-separated. For example, if you enter '0 1 2' you'll be getting offsets of 0°C, 1°C and 2°C to the heating setpoint temperature. If you don't enter any number, it'll be used '50' as the default value (that is 50°C below heating setpoint temperature, and therefore no limit is applied).
- MaxWindSpeed: stands for maximum wind speed, again only used in Mixed Mode (HVAC Mode '2'). Sets the maximum wind speed in which ventilation is allowed, in m/s. Therefore, if you enter '20', ventilation won't be allowed if wind speed is greater than 20 m/s. Entered values can be float or integers, but always positive numbers, and must be space-separated. For example, if you enter '5 10 15 20' you'll be getting different IDFs with maximum wind speeds of 5 m/s, 10 m/s, 15 m/s and 20 m/s. If you don't enter any number, it'll be used '50' as the default value (that is 50 m/s, and therefore no limit is applied).

- ASTtol: stands for Adaptive Setpoint Temperature tolerance. It applies the number that you enter as a tolerance for the adaptive heating and cooling setpoint temperatures. The original problem was that, if we assigned the adaptive setpoint straight to the comfort limit (i.e. you enter ‘0’ for ASTtol), there were a few hours that fell outside the comfort zone because of the error in some decimals in the simulation of the operative temperature. Therefore, the original purpose of this feature is to control that all hours are comfortable hours (i.e. operative temperature falls within the comfort zone), and we can make that sure by considering a little tolerance of 0.10 °C. For example, say that adaptive cooling and heating setpoints are originally 29.5 and 21.5°C at some day; if you enter ‘1’ for ASTtol, then the setpoints would be modified to 28.5 and 22.5°C (1°C below original cooling setpoint, and 1°C above original heating setpoint). The function will create a sequence of numbers based on the entered values. So, numbers must be entered in 3 stages: first, the start of the sequence; second, the end of the sequence, and third, the steps. So for example, if you enter ‘0’ for the start, ‘1’ for the end, and ‘0.25’ for the steps, you would be getting ASTtol values of 0°C, 0.25°C, 0.5°C, 0.75°C and 1°C. If you don’t enter any number, it’ll be used ‘0.1’ as the default value (as previously said, to make sure all hours are comfortable hours), and you would be getting only one variation of 0.1°C.
- NameSuffix: the text you would like to add at the end of the file name.
- verboseMode: True to print all process in screen, False to not to print it. Default is True.
- confirmGen: Generally, this argument should be left as default. True to confirm automatically the generation of IDFs; if False, you’ll be asked to confirm in command prompt. Default is False. So, if you are going to set it True, be sure about the number of IDFs you are going to generate, because these might be thousands.

So, below you can see a sample name of an IDF created by using accim’s VRFsystem functions. The package takes the original IDF file as a reference, saves a copy, run all the functions so that setpoint temperatures are transformed from static to adaptive, and changes its name based on the values previously entered:

TestModel_onlyGeometryForVRFsystem[CS_INT EN16798[CA_1[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[sta

where:

- ‘TestModel_onlyGeometryForVRFsystem’ is the name of the original IDF.
- CS refers to the Comfort Standard, and it’s followed by the thermal comfort standard applied (could be ‘ESP CTE’, ‘INT EN16798’, ‘INT ASHRAE55’, ‘JPN Rijal’, etc).
- CA refers to the Category, which could be 1, 2 or 3 if CS is EN16798, 80 or 90 if CS is ASHRAE55 or other models, or 80, 85 or 90 in case of the IMAC C.
- CM refers to the Comfort Mode, which could be 0 (Static), 1, 2, or 3 (Adaptive modes).
- HM refers to the HVAC Mode, which could be 0 (Full air conditioning), 1 (Naturally ventilated), or 2 (Mixed Mode).
- VC refers to the Ventilation Control, which could be 0, 1, 2 or 3.
- VO refers to the Ventilation setpoint temperature offset, which could be any number, float or integer, positive or negative.
- MT refers to the Minimum Outdoor Temperature offset, which could be any number, float or integer, but always positive number.
- MW refers to the Maximum Wind Speed, which could be any number, float or integer, but always positive number.
- AT refers to the Adaptive Setpoint Temperature offset, which could be any number, float or integer, but always positive number. Please remember this number comes from a 3-stage process (refer to the explanation above).
- ‘standard’ is the suffix, which can be whatever you want. For example, this allows you to make a for loop with ‘standard’, ‘simplified’ and ‘timestep’ and run the simulations with all type of outputs.

If some inputs are not used or don’t make sense, you’ll be able to see an ‘X’ in the output IDF file. For example, if you use CTE as Comfort Standard, then the inputs for Category and Comfort Mode (which are only for EN16798-1 and ASHRAE 55) are not used in the process, and the output IDF would contain in its name ‘CS_ESP CTE[CA_X[CM_X’.

Another similar case occurs if you use Full air-conditioning HVAC Mode (i.e. enter ‘0’ for HVAC Mode), or if you use the ‘ex_ac’ ScriptType, where the output IDF would contain in its name ‘[HM_0][VC_X][VO_X][MT_X][MW_X]’.

4.3 4.3 Full list of setpoint temperatures

Depending on the arguments ComfStand, CAT and ComfMod, cooling and heating setpoint temperatures will be the following:

(If it is too small, you can look at it also at the [Github repository](#))

4.4 4.4 Putting it into practice: Adaptive setpoint temperatures step by step

You can see a Jupyter Notebook either in the How-to Guide section of this documentation or in the link below:

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/addAccis/using_addAccis.ipynb

You can also execute it at your computer. You just need to find the folder containing the .ipynb and all other files at the accim package folder within your site_packages path, in

accim/sample_files/jupyter_notebooks/addAccis

The path should be something like this, with your username instead of YOUR_USERNAME:

C:\Users\YOUR_USERNAME\AppData\Local\Programs\Python\Python39\Lib\site-packages\accim\sample_files\jupyter_notebooks\add

Then, you just need to copy the folder to a different path (i.e. Desktop), open a cmd dialog pointing at it, and run “jupyter notebook”. After that, an internet browser will pop up, and you will be able to open the .ipynb file.

You can also see an example below. The input file is included within accim/sample_files/sample_IDFs folder, and it was originally named TestModel1_onlyGeometryForVRFsystem_2zones_CalcVent_V2310.idf, but for clarity purposes in this case has been renamed to “TestModel.idf”.

So, say you have an IDF in some folder, called ‘TestModel.idf’. So, you can either open an IDE or simply a CMD dialog pointing at that path and execute python. Let’s run the functions to get the energy models with adaptive setpoint temperatures.

```
>>> from accim.sim import accis
>>> accis.addAccis()
```

When we hit enter, we’ll be asked to enter some information regarding the ScriptType, the Outputs and the EnergyPlus version:

```
-----  
Adaptive-Comfort-Control-Implemented Model (ACCIM)  
-----
```

This tool allows to apply adaptive setpoint temperatures.

For further information, please read the documentation:

<https://accim.readthedocs.io/en/master/>

For a visual understanding of the tool, please visit the following jupyter notebooks:

- Using addAccis() to apply adaptive setpoint temperatures

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_

(continues on next page)

(continued from previous page)

```

↳ notebooks/addAccis/using_addAccis.ipynb- Using rename_epw_files() to rename the
↳ EPWs for proper data analysis after simulation
https://github.com/dsanchez-garcia/accim/blob/master/accim/sample\_files/jupyter\_notebooks/rename\_epw\_files/using\_rename\_epw\_files.ipynb
- Using runEp() to directly run simulations with EnergyPlus
https://github.com/dsanchez-garcia/accim/blob/master/accim/sample\_files/jupyter\_notebooks/runEp/using\_runEp.ipynb
- Using the class Table() for data analysis
https://github.com/dsanchez-garcia/accim/blob/master/accim/sample\_files/jupyter\_notebooks/Table/using\_Table.ipynb

```

Starting with the process.

Now, you are going to be asked to enter some information for different arguments to generate the output IDFs with adaptive setpoint temperatures.
 If you are not sure about how to use these parameters, please take a look at the documentation in the following link:
<https://accim.readthedocs.io/en/latest/how%20to%20use.html>

Please, enter the following information:

Enter the ScriptType (
 for VRFsystem with full air-conditioning mode: vrf_ac;
 for VRFsystem with mixed-mode: vrf_mm;
 for ExistingHVAC with mixed mode: ex_mm;
 for ExistingHVAC with full air-conditioning mode: ex_ac
): vrf_mm

Enter the SupplyAirTempInputMethod (
 for Supply Air Temperature: supply air temperature;
 for Temperature Difference: temperature difference;
): temperature difference

Do you want to keep the existing outputs (true or false)?: false

Enter the Output type (standard, simplified, detailed or custom): standard

Enter the Output frequencies separated by space (timestep, hourly, daily, monthly,
 ↳ runperiod): hourly runperiod

Do you want to generate a dataframe to see all outputs? (true or false): false

Enter the EnergyPlus version (9.1 to 23.1): 23.1

Enter the Temperature Control method (temperature or pmv): temperature

When we hit enter, it's going to add all the EnergyPlus objects needed:

```

Basic input data:
ScriptType is: vrf_mm
Supply Air Temperature Input Method is: temperature difference
Output type is: standard

```

(continues on next page)

(continued from previous page)

```

Output frequencies are:
['hourly', 'runperiod']
EnergyPlus version is: 23.1
Temperature Control method is: temperature

=====START OF GENERIC IDF FILE GENERATION=====
→PROCESS=====

Starting with file:
TestModel
IDD location is: C:\EnergyPlusV23-1-0\Energy+.idd
The occupied zones in the model TestModel are:
BLOCK1:ZONE2
BLOCK1:ZONE1
The windows and doors in the model TestModel are:
Block1_Zone2_Wall_3_0_0_0_0_0_Win
.

.

.

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting Frequency Runperiod
→Output:Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting Frequency Runperiod
→Output:Variable data
IDF has been saved
Ending with file:
TestModel
=====END OF GENERIC IDF FILE GENERATION PROCESS=====

The following IDFs will not work, and therefore these will be deleted:
None

```

And then ask us to enter the required information to generate the output IDF files (you can omit some by hitting enter without entering any value):

```

=====START OF OUTPUT IDF FILES GENERATION=====
→PROCESS=====

The information you will be required to enter below will be used to generate the
→customised output IDFs:
Enter the Comfort Standard numbers separated by space (
0 = ESP CTE;
1 = INT EN16798-1;
2 = INT ASHRAE55;
3 = JPN Rijal;
4 = CHN GBT50785 Cold;
5 = CHN GBT50785 HotMild;
6 = CHN Yang;
7 = IND IMAC C NV;
8 = IND IMAC C MM;
9 = IND IMAC R 7DRM;
10 = IND IMAC R 30DRM;
11 = IND Dhaka;

```

(continues on next page)

(continued from previous page)

```

12 = ROM Udrea;
13 = AUS Williamson;
14 = AUS DeDear;
15 = BRA Rupp NV;
16 = BRA Rupp AC;
17 = MEX Oropeza Arid;
18 = MEX Oropeza DryTropic;
19 = MEX Oropeza Temperate;
20 = MEX Oropeza HumTropic;
21 = CHL Perez-Fargallo;
22 = INT ISO7730;

Please refer to the full list of setpoint temperatures at https://raw.githack.com/dsanchez-garcia/accim/master/docs/full\_setpoint\_table.html

): 1 2 7
      Are you sure the numbers are correct? [y or [] / n]:

```

For the comfort standard 1 = INT EN16798, the available categories you can choose are:

```

1 = EN16798 Category I
2 = EN16798 Category II
3 = EN16798 Category III

```

For the comfort standard 2 = INT ASHRAE55, the available categories you can choose are:

```

80 = ASHRAE 55 80% acceptability
90 = ASHRAE 55 90% acceptability

```

For the comfort standard 7 = IND IMAC C NV, the available categories you can choose are:

```

80 = 80% acceptability
85 = 85% acceptability
90 = 90% acceptability

```

Enter the Category numbers separated by space (

```

1 = CAT I / CAT A;
2 = CAT II / CAT B;
3 = CAT III / CAT C;
80 = 80% ACCEPT;
85 = 85% ACCEPT;
90 = 90% ACCEPT;

```

```

Please refer to the full list of setpoint temperatures at https://raw.githack.com/dsanchez-garcia/accim/master/docs/full\_setpoint\_table.html
): 2 3 85 90
      Are you sure the numbers are correct? [y or [] / n]:

```

For the comfort standard 1 = INT EN16798, the available ComfMods you can choose are:

```

0 = EN16798 Static setpoints
1 = EN16798 Adaptive setpoints when applicable, otherwise CTE
2 = EN16798 Adaptive setpoints when applicable, otherwise EN16798 Static setpoints
3 = EN16798 Adaptive setpoints when applicable, otherwise EN16798 Adaptive setpoints
      ↳ horizontally extended

```

For the comfort standard 2 = INT ASHRAE55, the available ComfMods you can choose are:

```

0 = ISO 7730 Static setpoints
1 = ASHRAE 55 Adaptive setpoints when applicable, otherwise CTE
2 = ASHRAE 55 Adaptive setpoints when applicable, otherwise ISO 7730 Static setpoints
3 = ASHRAE 55 Adaptive setpoints when applicable, otherwise ASHRAE 55 Adaptive setpoints
      ↳ horizontally extended

```

For the comfort standard 7 = IND IMAC C NV, the available ComfMods you can choose are:

(continues on next page)

(continued from previous page)

```

0 = Indian Building Code Static setpoints
1 = IMAC C NV Model Adaptive setpoints when applicable, otherwise Indian Building Code
    ↵Static setpoints
2 = IMAC C NV Model Adaptive setpoints when applicable, otherwise ISO 7730 Static
    ↵setpoints
3 = IMAC C NV Model Adaptive setpoints when applicable, otherwise Adaptive setpoints
    ↵horizontally extended
Enter the Comfort Mode numbers separated by space (
0 or 0.X = Static;
1, 1.X, 2, 3 = Adaptive;
Please refer to the full list of setpoint temperatures at https://raw.githack.com/
    ↵dsanchez-garcia/accim/master/docs/full_setpoint_table.html
): 0 3
    Are you sure the numbers are correct? [y or [] / n]: 

Enter the setpoint accuracy number (any number greater than 0): 100
    Are you sure the number is correct? [y or [] / n]: 

Enter the start of the cooling season in numeric date format dd/mm or the day of the
    ↵year: 01/05
    Are you sure the number is correct? [y or [] / n]: 

Enter the end of the cooling season in numeric date format dd/mm or the day of the year: 
    ↵01/10
    Are you sure the number is correct? [y or [] / n]: 

Enter the HVAC Mode numbers separated by space (
0 = Fully Air-conditioned;
1 = Naturally ventilated;
2 = Mixed Mode;
): 2
    Are you sure the numbers are correct? [y or [] / n]: 

Enter the Ventilation Control numbers separated by space (
If HVACmode = 1:
    0 = Ventilates above neutral temperature;
    1 = Ventilates above upper comfort limit;
If HVACmode = 2:
    0 = Ventilates above neutral temperature and fully opens doors and windows;
    1 = Ventilates above lower comfort limit and fully opens doors and windows;
    2 = Ventilates above neutral temperature and opens doors and windows based on the
        ↵customised venting opening factor;
    3 = Ventilates above lower comfort limit and opens doors and windows based on the
        ↵customised venting opening factor;
): 2 3
    Are you sure the numbers are correct? [y or [] / n]: 
Enter the maximum temperature difference number for Ventilation Opening Factor (any
    ↵number larger than 0): 15
    Are you sure the number is correct? [y or [] / n]: 
Enter the minimum temperature difference number for Ventilation Opening Factor (any
    ↵number larger than 0 and smaller than the maximum temperature difference number): 1
    Are you sure the number is correct? [y or [] / n]: 
```

(continues on next page)

(continued from previous page)

Enter the multiplier number for Ventilation Opening Factor (any number between 0 and 1): ↵
0.2

Are you sure the number is correct? [y or [] / n]:

Enter the VSToffset numbers separated by space (if omitted, will be 0):

Are you sure the numbers are correct? [y or [] / n]:

Enter the MinOToffset numbers separated by space (if omitted, will be 50):

Are you sure the numbers are correct? [y or [] / n]:

Enter the MaxWindSpeed numbers separated by space (if omitted, will be 50):

Are you sure the numbers are correct? [y or [] / n]:

Enter the ASTtol value from (if omitted, will be 0.1):

Are you sure the numbers are correct? [y or [] / n]:

Enter the ASTtol value to (if omitted, will be 0.1):

Are you sure the numbers are correct? [y or [] / n]:

Enter the ASTtol value steps (if omitted, will be 0.1):

Are you sure the numbers are correct? [y or [] / n]:

Afterwards, ACCIS will let us know which the output IDF's are going to be, the total number of them and will ask for our confirmation to proceed:

The list of output IDF's is going to be:

```
TestModel[CS_INT EN16798[CA_2[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_90[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_90[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_90[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_90[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_85[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_85[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_85[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_85[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_90[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_90[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_90[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_90[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
And the total number of output IDF's is going to be 20
Do you still want to run ACCIS? [y/n]: y
```

If we entered n, the whole process would shut down. Otherwise, if we entered 'y', the generation of output IDF files would start, and ACCIS would print on screen each output IDF name as it generates it. As you can see, we are going to use the EN16798-1, ASHRAE 55 and IMAC Commercial for naturally ventilated buildings, categories 2 and 3 for EN16798, 90% acceptability levels for ASHRAE 55, 85 and 90% acceptability levels for IMAC C NV, all with ComfMod 0 (with static setpoint temperatures) and 3 (with adaptive setpoint temperatures when the model is applicable, otherwise horizontally extending the adaptive setpoint temperatures), Mixed Mode, and we just went ahead

with the remaining default values.

```
Generating the following output IDF files:
TestModel[CS_INT EN16798[CA_2[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_90[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_90[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_90[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_90[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_85[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_85[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_85[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_85[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_90[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_90[CM_0.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_90[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_IND IMAC C NV[CA_90[CM_3.0[HM_2[VC_3[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
=====END OF OUTPUT IDF FILES GENERATION PROCESS=====
```

Afterwards, you just need to run the simulation. Once you have run the simulations you need, you'll get the files you usually get when you run any simulation, including the hourly results on a CSV file.

4.5 Renaming epw files for later data analysis

You can see a Jupyter Notebook either in the How-to Guide section of this documentation or in the link below:

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/rename_epw_files/using_rename_epw_files.ipynb

You can also execute it at your computer. You just need to find the folder containing the .ipynb and all other files at the accim package folder within your site_packages path, in

accim/sample_files/jupyter_notebooks/rename_epw_files

The path should be something like this, with your username instead of YOUR_USERNAME:

C:\Users\YOUR_USERNAME\AppData\Local\Programs\Python\Python39\Lib\site-packages\accim\sample_files\jupyter_notebooks\ren

Then, you just need to copy the folder to a different path (i.e. Desktop), open a cmd dialog pointing at it, and run "jupyter notebook". After that, an internet browser will pop up, and you will be able to open the .ipynb file.

4.6 4.6 Running simulations

You can see a Jupyter Notebook either in the How-to Guide section of this documentation or in the link below:

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/runEp/using_runEp.ipynb

You can also execute it at your computer. You just need to find the folder containing the .ipynb and all other files at the accim package folder within your site_packages path, in

accim/sample_files/jupyter_notebooks/runEp

The path should be something like this, with your username instead of YOUR_USERNAME:

C:\Users\YOUR_USERNAME\AppData\Local\Programs\Python\Python39\Lib\site-packages\accim\sample_files\jupyter_notebooks\runEp

Then, you just need to copy the folder to a different path (i.e. Desktop), open a cmd dialog pointing at it, and run “jupyter notebook”. After that, an internet browser will pop up, and you will be able to open the .ipynb file.

This script has been created by eppy’s development team (<https://eppy.readthedocs.io/en/latest/runningplus.html>, specifically from section ‘Running in parallel processes using Generators’), however I did some changes. Anyway, you probably should check out eppy package, since it’s absolutely awesome.

By using this script, the EnergyPlus version used to simulate the IDFs will be the IDF’s version. Therefore, if your IDF is in version 9.4, but you don’t have EnergyPlus 9.4 installed, you’ll get an error.

The main difference is that this one allows to run simulations with several EPW files. It takes all EPW files and IDF files located in the script folder, and runs them. So for example, say you have 2 no. IDFs (1.idf and 2.idf) and 2 no. EPW files (a.epw and b.epw). Then, this script will run the following simulations: 1[a; 1[b; 2[a; 2[b. The character ‘[’ has been used as separator in order to not to be in conflict with other programs. Besides, there’s a package within accim currently being developed (within folder data) in order to generate tables and graphs automatically.

So, how to use it?

Say you have already run any of the accis functions, and therefore you might have a folder with the following files:

Mode	LastWriteTime	Length	Name
-a---1	20/07/2019 12:42	1407718	Bilbao_2015.epw
-a---1	20/07/2019 12:43	1408160	Bilbao_2016.epw
-a---	27/02/2021 15:01	114617	TestModel_SingleZone.idf
-a---1	27/02/2021 15:01	114617	TestModel_SingleZone_pymod[AS_
EN16798[CA_1[CM_3[AT_0.1.idf			
-a---1	27/02/2021 15:01	114617	TestModel_SingleZone_pymod[AS_
EN16798[CA_2[CM_3[AT_0.1.idf			

So, now we can run the simulations:

```
>>> from accim.run import run
>>> dir(run)
['IDF', '__builtins__', '__cached__', '__doc__', '__file__', '__loader__', '__name__', '__package__', '__spec__', 'make_eplaunch_options', 'os', 'removefiles', 'runEp', 'runIDFs']
>>> run.runEp()
```

runEp() is going to ask you if you want to run the simulations only with IDF files generated by accim. If you enter ‘y’, it’ll run only accim output IDFs, otherwise if you enter ‘n’, it will run all idfs in the folder. As you can see below, we didn’t need to remove the original IDF ‘TestModel_SingleZone.idf’ from the folder. Then, it will let you know the IDFs and EPWs that are going to be used in the simulations. Besides, it’ll let you know the total number of simulations,

and will ask for your confirmation, because you might start thousands of simulations by mistake. Further, `runEp()` can take a total number of 3 arguments, which are 2 boolean arguments, and one integer: `runEp(runOnlyAccim=True, confirmRun=True, num_CPUs=4)`. If you entered these, you would skip the command prompt process and jump straight to the simulation process. Since we entered 4 for the `num_CPUs` argument, the simulations would be run by using 4 CPS at the same time.

```
>>> run.runEp()
Do you want to run only accim output IDFs? [y or n]:y
The IDFs we are going to run are: ['TestModel_SingleZone_pymod[AS_EN16798[CA_1[CM_3[AT_0.
˓→1.idf', 'TestModel_SingleZone_pymod[AS_EN16798[CA_2[CM_3[AT_0.1.idf']]
and the No. of IDFs is going to be 2
The EPWs we are going to run are: ['Bilbao_2015.epw', 'Bilbao_2016.epw']
and the No. of EPWs is going to be 2
Therefore, the simulations are going to be:
TestModel_SingleZone_pymod[AS_EN16798[CA_1[CM_3[AT_0.1.idf[Bilbao_2015.epw
TestModel_SingleZone_pymod[AS_EN16798[CA_1[CM_3[AT_0.1.idf[Bilbao_2016.epw
TestModel_SingleZone_pymod[AS_EN16798[CA_2[CM_3[AT_0.1.idf[Bilbao_2015.epw
TestModel_SingleZone_pymod[AS_EN16798[CA_2[CM_3[AT_0.1.idf[Bilbao_2016.epw
and the No. of simulations is going to be 4
The number of simulations is going to be 4. Do you still want to proceed?[y or n]:y
```

Afterwards, you'll see the calculations progress if you use the windows prompt command, and you'll get an extensive list of simulation files, similar to this:

Mode	LastWriteTime	Length	Name
---	-----	-----	-----
-a---1	20/07/2019 12:42	1407718	Bilbao_2015.epw
-a---1	20/07/2019 12:43	1408160	Bilbao_2016.epw
-a---1	27/02/2021 15:01	114617	TestModel_SingleZone_pymod[AS_
˓→EN16798[CA_1[CM_3[AT_0.1.idf			
-a---1	27/02/2021 16:47	1721	TestModel_SingleZone_pymod[AS_
˓→EN16798[CA_1[CM_3[AT_0[Bilbao_2015.audit			
-a---1	27/02/2021 16:47	9179	TestModel_SingleZone_pymod[AS_
˓→EN16798[CA_1[CM_3[AT_0[Bilbao_2015.bnd		2023160	TestModel_SingleZone_pymod[AS_
-a---1	27/02/2021 16:47	6181	TestModel_SingleZone_pymod[AS_
˓→EN16798[CA_1[CM_3[AT_0[Bilbao_2015.csv		30483	TestModel_SingleZone_pymod[AS_
-a---1	27/02/2021 16:47	99	TestModel_SingleZone_pymod[AS_
˓→EN16798[CA_1[CM_3[AT_0[Bilbao_2015.dxf		5351	TestModel_SingleZone_pymod[AS_
-a---1	27/02/2021 16:47	2968770	TestModel_SingleZone_pymod[AS_
˓→EN16798[CA_1[CM_3[AT_0[Bilbao_2015.eso		0	TestModel_SingleZone_pymod[AS_
-a---1	27/02/2021 16:47	13352	TestModel_SingleZone_pymod[AS_
˓→EN16798[CA_1[CM_3[AT_0[Bilbao_2015.mdd		0	TestModel_SingleZone_pymod[AS_
-a---1	27/02/2021 16:47	1107	TestModel_SingleZone_pymod[AS_
˓→EN16798[CA_1[CM_3[AT_0[Bilbao_2015.mtd			
-a---1	27/02/2021 16:47		
˓→EN16798[CA_1[CM_3[AT_0[Bilbao_2015.rdd			
-a---1	27/02/2021 16:47		
˓→EN16798[CA_1[CM_3[AT_0[Bilbao_2015.rvaudit			

(continues on next page)

(continued from previous page)

```
-a---l 27/02/2021 16:47 2667 TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_1[CM_3[AT_0[Bilbao_2015.shd
-a---l 27/02/2021 16:47 34187 TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_1[CM_3[AT_0[Bilbao_2015Table.csv
-a---l 27/02/2021 16:47 139585 TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_1[CM_3[AT_0[Bilbao_2015Table.htm
-a---l 27/02/2021 16:47 3421 TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_1[CM_3[AT_0[Bilbao_2015Zsz.csv
-
-
-
```

You might need to keep these in order to debug some error, or any other reason, but if you don't need to keep these except csv values with hourly results, you can run the `removefiles()` function:

```
>>> run.removefiles()
```

And now your working directory should look like this:

Mode	LastWriteTime	Length	Name
---	-----	-----	-----
-a---l	20/07/2019 12:42	1407718	Bilbao_2015.epw
-a---l	20/07/2019 12:43	1408160	Bilbao_2016.epw
-a---l	27/02/2021 15:01	114617	TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_1[CM_3[AT_0.1.idf			
-a---l	27/02/2021 16:47	2023160	TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_1[CM_3[AT_0[Bilbao_2015.csv			
-a---l	27/02/2021 16:47	2017212	TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_1[CM_3[AT_0[Bilbao_2016.csv			
-a---l	27/02/2021 15:01	114617	TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_2[CM_3[AT_0.1.idf			
-a---l	27/02/2021 16:47	2023114	TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_2[CM_3[AT_0[Bilbao_2015.csv			
-a---l	27/02/2021 16:47	2017070	TestModel_SingleZone_pymod[AS_
↳ EN16798[CA_2[CM_3[AT_0[Bilbao_2016.csv			

As you can see, `removefiles()` removes everything except EPW files, IDFs, .py scripts and the hourly CSV values which contains the results of the simulations.

4.7 Functions and methods for data analysis; making figures and tables

You can see a Jupyter Notebook either in the How-to Guide section of this documentation or in the link below:

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/Table/using_Table.ipynb

You can also execute it at your computer. You just need to find the folder containing the .ipynb and all other files at the accim package folder within your site_packages path, in

accim/sample_files/jupyter_notebooks/Table

The path should be something like this, with your username instead of YOUR_USERNAME:

C:\Users\YOUR_USERNAME\AppData\Local\Programs\Python\Python39\Lib\site-packages\accim\sample_files\jupyter_notebooks\Table

Then, you just need to copy the folder to a different path (i.e. Desktop), open a cmd dialog pointing at it, and run “jupyter notebook”. After that, an internet browser will pop up, and you will be able to open the .ipynb file.

5. TROUBLESHOOTING

5.1 5.1 Errors you might get

Below you can see a few errors you might get when trying to run addAccis():

```
ModuleNotFoundError: No module named 'numpy'
```

You need to update the numpy package. To do so, you just need to open cmd and enter:

```
pip install numpy --upgrade
```

```
UnicodeEncodeError: 'latin-1' codec can't encode character '\u0144' in position 123174:  
ordinal not in range(256)
```

This error means that there are some unconventional characters within the model. To amend this problem, try the following:

- Open the IDF with a text editor and Check there are no accents or other unconventional characters such as ‘ñ’. For example, you can try replacing all instances of ‘á’ with ‘a’.
 - If it doesn’t work and you are using Designbuilder to model the building, then you can try opening a new file, then importing the building geometry to the new file, exporting only libraries used from the old model (it is recommended to carefully check that all inherent data, from schedules to profiles, constructions and materials within these libraries are in English), and finally importing these to the new model and assigning them to the relevant objects as needed.
-

```
this node -WINDOWPROPERTY:SHADINGCONTROL-is not present in base dictionary  
this node -WINDOWPROPERTY:SHADINGCONTROL-is not present in base dictionary  
1 <class 'int'> None <class 'NoneType'>  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
  File "C:\Users\sanga\AppData\Local\Programs\Python\Python39\lib\site-packages  
\accim\sim\accis.py", line 138, in addAccis  
    z = accim_Main.accimJob(filename_temp=file, ScriptType=ScriptType, EnergyPl  
us_version=EnergyPlus_version, verboseMode=verboseMode)  
  File "C:\Users\sanga\AppData\Local\Programs\Python\Python39\lib\site-packages  
\accim\sim\accim_Main.py", line 66, in __init__  
    self.idf0 = IDF(fname1)  
  File "C:\Users\sanga\AppData\Local\Programs\Python\Python39\lib\site-packages
```

(continues on next page)

(continued from previous page)

```
\eppy\modeleditor.py", line 548, in __init__
    self.read()
  File "C:\Users\sanga\AppData\Local\Programs\Python\Python39\lib\site-packages
\epy\modeleditor.py", line 672, in read
    readout = idfreader1(
  File "C:\Users\sanga\AppData\Local\Programs\Python\Python39\lib\site-packages
\epy\idfreader.py", line 307, in idfreader1
    bunchdt = makebunches_alter(data, commdct, theidf, block)
  File "C:\Users\sanga\AppData\Local\Programs\Python\Python39\lib\site-packages
\epy\idfreader.py", line 109, in makebunches_alter
    bobj = makeabunch(commdct, obj, obj_i, block=block)
  File "C:\Users\sanga\AppData\Local\Programs\Python\Python39\lib\site-packages
\epy\idfreader.py", line 68, in makeabunch
    extlst = extension_of_extensible(commdct[obj_i], block[obj_i], n)
  File "C:\Users\sanga\AppData\Local\Programs\Python\Python39\lib\site-packages
\epy\idfreader.py", line 332, in extension_of_extensible
    n = n // ext
TypeError: unsupported operand type(s) for //: 'int' and 'NoneType'
```

If you get an error similar to the one above, you are probably trying to run addAccis() with an IDF whose EnergyPlus version is 8.9 or older. To solve this problem, you need to update the IDF's EnergyPlus version up to some 9.X.X version. It is recommended to update up to the latest. To do so, you need to run EP-Launch, go to the Utilities tab, look for and select IDFversionUpdater; run IDFVersionUpdater; then choose the file to update, select the new version and finally update file.

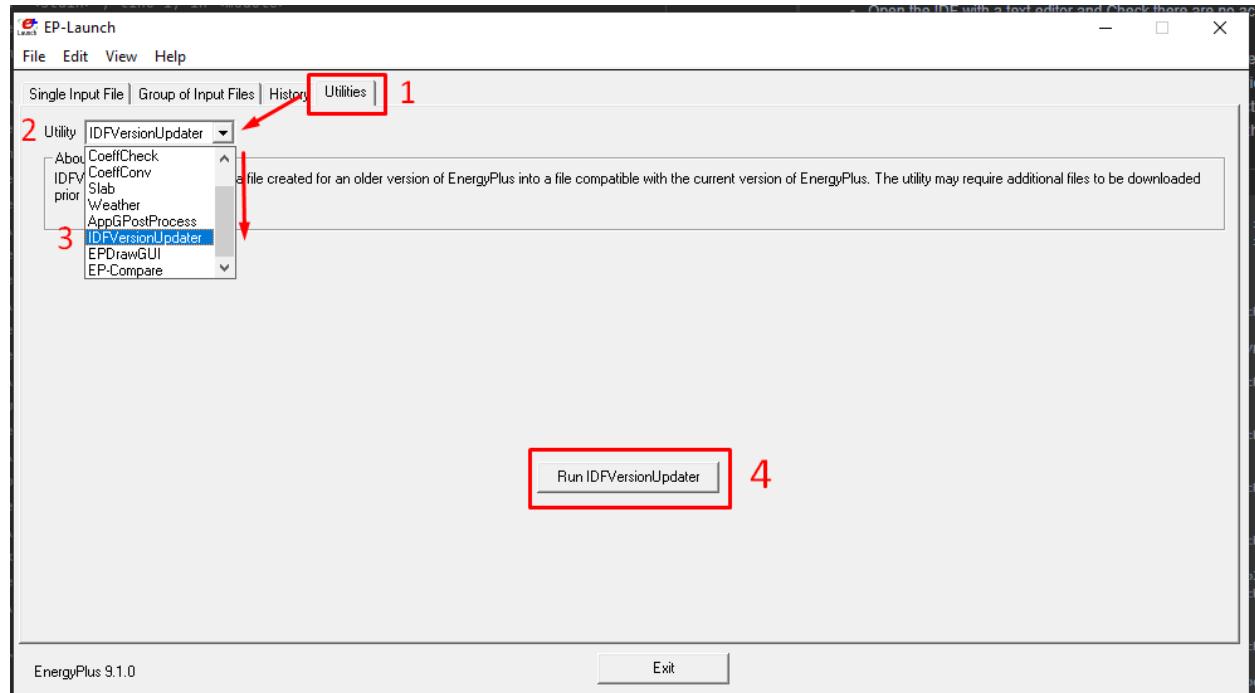


Fig. 1: Troubleshooting_IDFversion_01

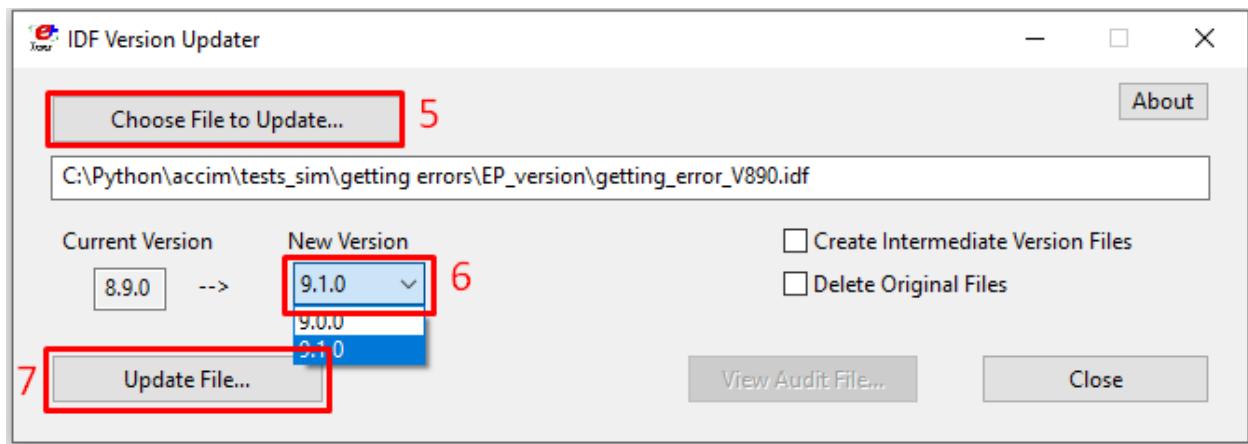


Fig. 2: Troubleshooting_IDFversion_02

5.2 6.2 General recommendations

If you are using only one version of Python in your computer, you should check on the option ‘Add to PATH’. Otherwise, the python interpreter won’t be able to find the accis package.

Also, once you have installed Python 3.9, you should also disable the path length limit:

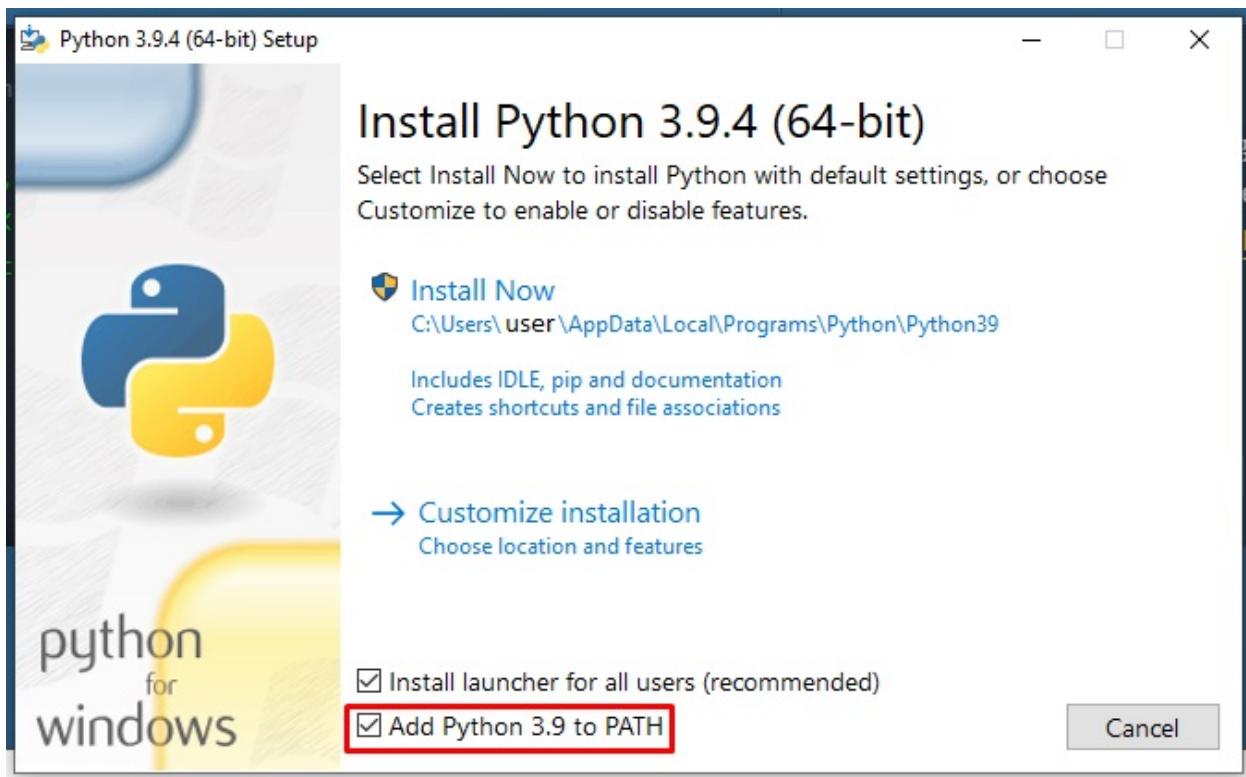


Fig. 3: Troubleshooting_Path

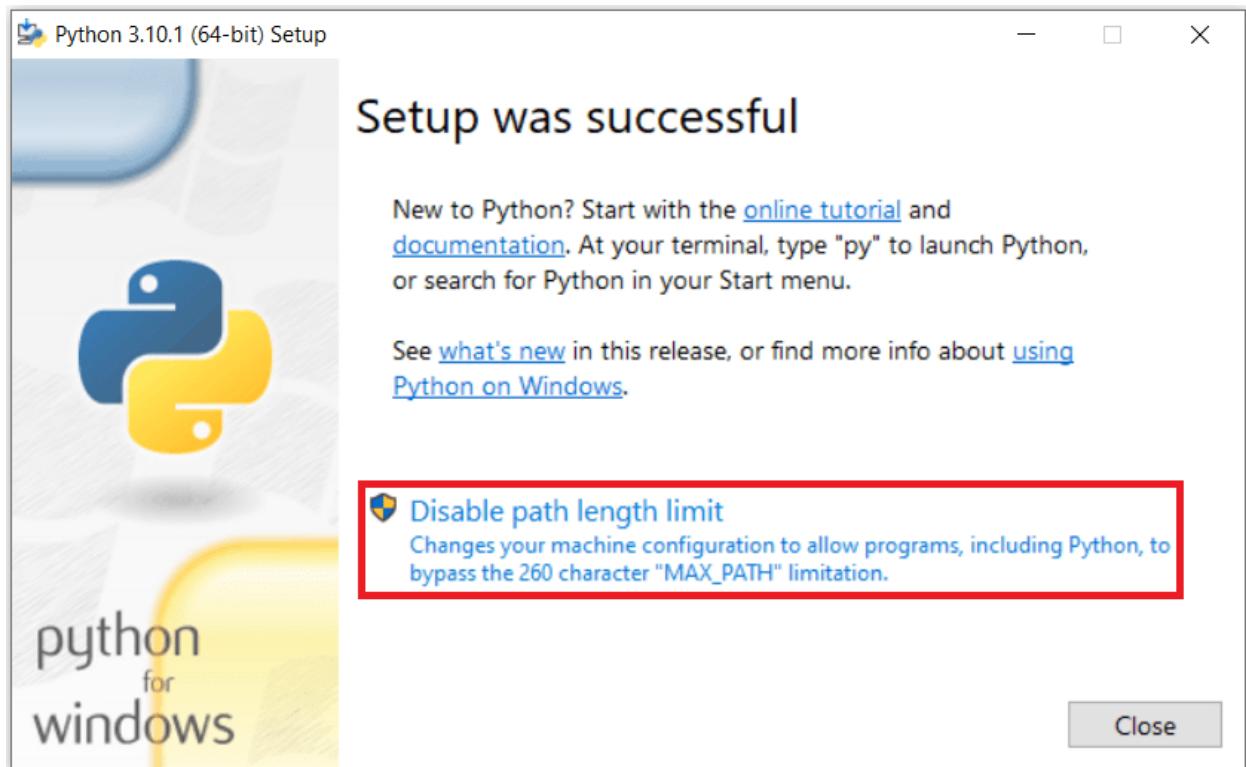


Fig. 4: Troubleshooting_Disable_length_limit

**CHAPTER
SIX**

6. CITATION

If you use this package, please cite us:

Sánchez-García, D., Bienvenido-Huertas, D., Rubio-Bellido, C., 2021. *Computational approach to extend the air-conditioning usage to adaptive comfort: Adaptive-Comfort-Control-Implementation Script*. Automation in Construction. 131, 103900. <https://doi.org/10.1016/j.autcon.2021.103900>

Sánchez-García, D., Martínez-Crespo, J., Hernando, U.R.R., Alonso, C., 2023. *A detailed view of the Adaptive-Comfort-Control-Implementation Script (ACCIS): The capabilities of the automation system for adaptive setpoint temperatures in building energy models*. Energy and Buildings. 288. <https://doi.org/10.1016/j.enbuild.2023.113019>

**CHAPTER
SEVEN**

7. CREDITS

It wouldn't have been possible to develop this python package without [eppy](#), so thank you for such an awesome work.

USING ADDACCIS() TO APPLY ADAPTIVE SETPOINT TEMPERATURES

We are going to apply the adaptive setpoint temperatures to the idf “TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310”, that can be found in the path accim/sample_files/sample_IDFs/input_IDFs. However, for clarity purposes, has been renamed to “TestModel.idf”.

First of all, let's see what files we do have in the folder:

```
[2]: import os  
os.listdir()  
[2]: ['.ipynb_checkpoints', 'TestModel.idf', 'using_addAccis.ipynb', '__init__.py']
```

You can see there is/are a/some IDF file/files there, which is/are:

```
[3]: input_idf = [i for i in os.listdir() if i.endswith('.idf')]  
print(*input_idf, sep='\n')  
TestModel.idf
```

So, let's apply adaptive setpoint temperatures.

8.1 1. General use

8.1.1 1.1 Short version

First, let's do this using the short version. When we run the code below (not entering any argument), accim will request some information to generate the output idfs. We are going to enter the following information in **bold**:

Enter the ScriptType (for VRFsystem with full air-conditioning mode: vrf_ac; for VRFsystem with mixed-mode: vrf_mm; for ExistingHVAC with mixed mode: ex_mm; for ExistingHVAC with full air-conditioning mode: ex_ac):
vrf_mm

Enter the SupplyAirTempInputMethod (for Supply Air Temperature: supply air temperature; for Temperature Difference: temperature difference;): **temperature difference**

Do you want to keep the existing outputs (true or false)?: **false**

Enter the Output type (standard, simplified or detailed): **standard**

Enter the Output frequencies separated by space (timestep, hourly, daily, monthly, runperiod): **hourly runperiod**

Enter the EnergyPlus version (9.1 to 23.1): **23.1**

Enter the Temperature Control method (temperature or pmv): **temperature**

Then, accim will report the information we just entered, and will start generating a generic idf file, which will be modified based on the remaining arguments accim will request in a second stage. Again, we are going to enter the following information in **bold**:

Enter the Comfort Standard numbers separated by space (...): **1 14**

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the Category numbers separated by space (...): **1 2 3 80 90**

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the Comfort Mode numbers separated by space (...): **0 0.1 0.2 0.3 0.4 0.5 1 1.1 1.2 1.3 1.4 1.5 2 3**

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the setpoint accuracy number (any number greater than 0): **100**

Are you sure the number is correct? [y or [] / n]: (**just hit enter**)

Enter the start of the cooling season in numeric date format dd/mm or the day of the year: **01/11**

Are you sure the number is correct? [y or [] / n]: (**just hit enter**)

Enter the end of the cooling season in numeric date format dd/mm or the day of the year: **01/03**

Are you sure the number is correct? [y or [] / n]: (**just hit enter**)

Enter the HVAC Mode numbers separated by space (...): **2**

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the Ventilation Control numbers separated by space (...): **2**

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the maximum temperature difference number for Ventilation Opening Factor (any number larger than 0): **20**

Are you sure the number is correct? [y or [] / n]: (**just hit enter**)

Enter the minimum temperature difference number for Ventilation Opening Factor (any number larger than 0 and smaller than the maximum temperature difference number): **1**

Are you sure the number is correct? [y or [] / n]: (**just hit enter**)

Enter the multiplier number for Ventilation Opening Factor (any number between 0 and 1): **0.2**

Are you sure the number is correct? [y or [] / n]: (**just hit enter**)

Enter the VSToffset numbers separated by space (if omitted, will be 0): (**just hit enter**)

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the MinOToffset numbers separated by space (if omitted, will be 50): (**just hit enter**)

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the MaxWindSpeed numbers separated by space (if omitted, will be 50): (**just hit enter**)

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the ASTtol value from (if omitted, will be 0.1): (**just hit enter**)

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the ASTtol value to (if omitted, will be 0.1): (**just hit enter**)

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

Enter the ASTtol value steps (if omitted, will be 0.1): (**just hit enter**)

Are you sure the numbers are correct? [y or [] / n]: (**just hit enter**)

[3]: from accim.sim import accis
accis.addAccis()

Adaptive-Comfort-Control-Implemented Model (ACCIM)

This tool allows to apply adaptive setpoint temperatures.

For further information, please read the documentation:

<https://accim.readthedocs.io/en/master/>

For a visual understanding of the tool, please visit the following jupyter notebooks:

- Using addAccis() to apply adaptive setpoint temperatures

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/addAccis/using_addAccis.ipynb

- Using rename_epw_files() to rename the EPWs for proper data analysis after simulation

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/rename_epw_files/using_rename_epw_files.ipynb

- Using runEp() to directly run simulations with EnergyPlus

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/runEp/using_runEp.ipynb

- Using the class Table() for data analysis

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/Table/using_Table.ipynb

Starting with the process.

Now, you are going to be asked to enter some information for different arguments to generate the output IDFs with adaptive setpoint temperatures.

If you are not sure about how to use these parameters, please take a look at the documentation in the following link:

<https://accim.readthedocs.io/en/latest/how%20to%20use.html>

Please, enter the following information:

Enter the ScriptType (
for VRFsystem with full air-conditioning mode: vrf_ac;
for VRFsystem with mixed-mode: vrf_mm;
for ExistingHVAC with mixed mode: ex_mm;
for ExistingHVAC with full air-conditioning mode: ex_ac
): vrf_mm

Enter the SupplyAirTempInputMethod (
for Supply Air Temperature: supply air temperature;
for Temperature Difference: temperature difference;
): temperature difference

Do you want to keep the existing outputs (true or false)?: false

Enter the Output type (standard, simplified, detailed or custom): standard

(continues on next page)

(continued from previous page)

Enter the Output frequencies separated by space (timestep, hourly, daily, monthly, ↵runperiod): hourly runperiod

Do you want to generate a dataframe to see all outputs? (true or false): false

Enter the EnergyPlus version (9.1 to 23.1): 23.1

Enter the Temperature Control method (temperature or pmv): temperature

Basic input data:

ScriptType is: vrf_mm

Supply Air Temperature Input Method is: temperature difference

Output type is: standard

Output frequencies are:

['hourly', 'runperiod']

EnergyPlus version is: 23.1

Temperature Control method is: temperature

=====START OF GENERIC IDF FILE GENERATION

PROCESS=====

Starting with file:

TestModel

IDD location is: C:\EnergyPlusV23-1-0\Energy+.idd

The occupied zones in the model TestModel are:

BLOCK1:ZONE2

BLOCK1:ZONE1

The windows and doors in the model TestModel are:

Block1_Zone2_Wall_3_0_0_0_0_0_Win

Block1_Zone2_Wall_4_0_0_0_0_0_Win

Block1_Zone2_Wall_5_0_0_0_0_0_Win

Block1_Zone1_Wall_2_0_0_0_0_0_Win

Block1_Zone1_Wall_3_0_0_0_0_0_Win

Block1_Zone1_Wall_5_0_0_0_0_0_Win

The zones in the model TestModel are:

BLOCK1_ZONE2

BLOCK1_ZONE1

The people objects in the model have been amended.

BLOCK1:ZONE2 Thermostat has been added

BLOCK1:ZONE1 Thermostat has been added

On Schedule already was in the model

TypOperativeTempControlSch Schedule already was in the model

All ZoneHVAC:IdealLoadsAirSystem Heating and Cooling availability schedules has been set
to on

On 24/7 Schedule already was in the model

Control type schedule: Always 4 Schedule has been added

Relative humidity setpoint schedule: Always 50.00 Schedule has been added

Heating Fanger comfort setpoint: Always -0.5 Schedule has been added

Cooling Fanger comfort setpoint: Always 0.1 Schedule has been added

Zone CO2 setpoint: Always 900ppm Schedule has been added

Min CO2 concentration: Always 600ppm Schedule has been added

Generic contaminant setpoint: Always 0.5ppm Schedule has been added

Air distribution effectiveness (always 1) Schedule has been added

(continues on next page)

(continued from previous page)

VRF Heating Cooling (Northern Hemisphere) Schedule has been added
 DefaultFanEffRatioCurve Curve:Cubic Object has been added
 VRFTUCoolCapFT Curve:Cubic Object has been added
 VRFTUHeatCapFT Curve:Cubic Object has been added
 VRCoolCapFTBoundary Curve:Cubic Object has been added
 VRCoolEIRFTBoundary Curve:Cubic Object has been added
 CoolingEIRLowPLR Curve:Cubic Object has been added
 VRFHeatCapFTBoundary Curve:Cubic Object has been added
 VRFHeatEIRFTBoundary Curve:Cubic Object has been added
 HeatingEIRLowPLR Curve:Cubic Object has been added
 DefaultFanPowerRatioCurve Curve:Exponent Object has been added
 DXHtgCoilDefrostEIRFT Curve:Biquadratic Object has been added
 VRCoolCapFT Curve:Biquadratic Object has been added
 VRCoolCapFTHi Curve:Biquadratic Object has been added
 VRCoolEIRFT Curve:Biquadratic Object has been added
 VRCoolEIRFTHi Curve:Biquadratic Object has been added
 VRFHeatCapFT Curve:Biquadratic Object has been added
 VRFHeatCapFTHi Curve:Biquadratic Object has been added
 VRFHeatEIRFT Curve:Biquadratic Object has been added
 VRFHeatEIRFTHi Curve:Biquadratic Object has been added
 CoolingLengthCorrectionFactor Curve:Biquadratic Object has been added
 VRF Piping Correction Factor for Length in Heating Mode Curve:Biquadratic Object has been added
 VRF Heat Recovery Cooling Capacity Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Cooling Energy Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Heating Capacity Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Heating Energy Modifier Curve:Biquadratic Object has been added
 VRFACTCoolCapFFF Curve:Quadratic Object has been added
 CoolingEIRHiPLR Curve:Quadratic Object has been added
 VRFCPLFFPLR Curve:Quadratic Object has been added
 HeatingEIRHiPLR Curve:Quadratic Object has been added
 CoolingCombRatio Curve:Linear Object has been added
 HeatingCombRatio Curve:Linear Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 AirConditioner:VariableRefrigerantFlow Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 AirConditioner:VariableRefrigerantFlow Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 Zone List Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Zone List Object has been added
 BLOCK1:ZONE2 Sizing:Zone Object has been added
 BLOCK1:ZONE1 Sizing:Zone Object has been added
 BLOCK1:ZONE2 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE1 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE2 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE1 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE2 Nodelist Objects has been added
 BLOCK1:ZONE1 Nodelist Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE1 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentList Objects has been added

(continues on next page)

(continued from previous page)

BLOCK1:ZONE1 ZoneHVAC:EquipmentList Object has been added
BLOCK1:ZONE2 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE1 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE2 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE1 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE2 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE1 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE2 Fan:ConstantVolume Object has been added
BLOCK1:ZONE1 Fan:ConstantVolume Object has been added
Vent_SP_temp Schedule has been added
AHST_Sch_BLOCK1_ZONE2 Schedule has been added
ACST_Sch_BLOCK1_ZONE2 Schedule has been added
AHST_Sch_BLOCK1_ZONE1 Schedule has been added
ACST_Sch_BLOCK1_ZONE1 Schedule has been added
Added - SetComfTemp Program
Added - CountHours_BLOCK1_ZONE2 Program
Added - CountHours_BLOCK1_ZONE1 Program
Added - SetAppLimits Program
Added - ApplyCAT Program
Added - SetAST Program
Added - SetASTnoTol Program
Added - CountHoursNoApp_BLOCK1_ZONE2 Program
Added - SetGeoVarBLOCK1_ZONE2 Program
Added - CountHoursNoApp_BLOCK1_ZONE1 Program
Added - SetGeoVarBLOCK1_ZONE1 Program
Added - SetInputData Program
Added - SetVOFinputData Program
Added - SetVST Program
Added - ApplyAST_BLOCK1_ZONE2 Program
Added - ApplyAST_BLOCK1_ZONE1 Program
Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_Win Program
Added - Comfort Temperature Output Variable
Added - Adaptive Cooling Setpoint Temperature Output Variable
Added - Adaptive Heating Setpoint Temperature Output Variable
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Output Variable
Added - Adaptive Heating Setpoint Temperature_No Tolerance Output Variable
Added - Ventilation Setpoint Temperature Output Variable
Added - Minimum Outdoor Temperature for ventilation Output Variable
Added - Minimum Outdoor Temperature Difference for ventilation Output Variable
Added - Maximum Outdoor Temperature Difference for ventilation Output Variable
Added - Multiplier for Ventilation Opening Factor Output Variable

(continues on next page)

(continued from previous page)

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Output Variable
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Output Variable
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Output Variable
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE2 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE1 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE2 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE1 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE2 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE1 Output Variable
 Global variables objects have been added
 Internal variables objects have been added
 Added - RMOT Sensor
 Added - PMOT Sensor
 Added - BLOCK1_ZONE2_OpT Sensor
 Added - BLOCK1_ZONE2_WindSpeed Sensor
 Added - BLOCK1_ZONE2_OutT Sensor
 Added - BLOCK1_ZONE1_OpT Sensor
 Added - BLOCK1_ZONE1_WindSpeed Sensor
 Added - BLOCK1_ZONE1_OutT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - OutT Sensor
 Added - AHST_Act_BLOCK1_ZONE2 Actuator
 Added - ACST_Act_BLOCK1_ZONE2 Actuator
 Added - AHST_Act_BLOCK1_ZONE1 Actuator
 Added - ACST_Act_BLOCK1_ZONE1 Actuator
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator

(continues on next page)

(continued from previous page)

```

Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_VentOpenFact Actuator
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_VentOpenFact Actuator
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
Added - BLOCK1_ZONE2_CoolCoil Sensor
Added - BLOCK1_ZONE2_HeatCoil Sensor
Added - BLOCK1_ZONE1_CoolCoil Sensor
Added - BLOCK1_ZONE1_HeatCoil Sensor
Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
Added - SetComfTemp Program Calling Manager
Added - CountHours_BLOCK1_ZONE2 Program Calling Manager
Added - CountHours_BLOCK1_ZONE1 Program Calling Manager
Added - SetAppLimits Program Calling Manager
Added - ApplyCAT Program Calling Manager
Added - SetAST Program Calling Manager
Added - SetASTnoTol Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE2 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE2 Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE1 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE1 Program Calling Manager
Added - SetInputData Program Calling Manager
Added - SetVOFinputData Program Calling Manager
Added - SetVST Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE2 Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE1 Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - Comfort Temperature Reporting FrequencyHourly Output:Variable data
Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyHourly Output:Variable_
→ data

```

(continues on next page)

(continued from previous page)

Added - Adaptive Heating Setpoint Temperature Reporting FrequencyHourly Output:Variable
 ↳data

Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyHourly
 ↳Output:Variable data

Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyHourly
 ↳Output:Variable data

Added - Ventilation Setpoint Temperature Reporting FrequencyHourly Output:Variable data

Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyHourly Output:
 ↳Variable data

Added - Minimum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly
 ↳Output:Variable data

Added - Maximum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly
 ↳Output:Variable data

Added - Multiplier for Ventilation Opening Factor Reporting FrequencyHourly Output:
 ↳Variable data

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳Variable data

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyHourly Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyHourly Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyHourly Output:Variable data

(continues on next page)

(continued from previous page)

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyHourly Output:
 ↳ Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyHourly
 ↳ Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳ data

Added - AFN Zone Infiltration Volume Reporting FrequencyHourly Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳ data

Added - AFN Zone Ventilation Volume Reporting FrequencyHourly Output:Variable data

Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyHourly Output:Variable
 ↳ data

Added - Site Wind Speed Reporting FrequencyHourly Output:Variable data

Added - Site Outdoor Air Relative Humidity Reporting FrequencyHourly Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyHourly Output:
 ↳ Variable data

Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyHourly Output:
 ↳ Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
 ↳ Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
 ↳ Variable data

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
 ↳ Variable data

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Comfort Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - Adaptive Heating Setpoint Temperature Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod
 ↳ Output:Variable data

Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod
 ↳ Output:Variable data

Added - Ventilation Setpoint Temperature Reporting FrequencyRunperiod Output:Variable
 ↳ data

Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - Minimum Outdoor Temperature Difference for ventilation Reporting
 ↳ FrequencyRunperiod Output:Variable data

Added - Maximum Outdoor Temperature Difference for ventilation Reporting
 ↳ FrequencyRunperiod Output:Variable data

Added - Multiplier for Ventilation Opening Factor Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod

(continues on next page)

(continued from previous page)

```

↳ Output:Variable data
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting
↳ FrequencyRunperiod Output:Variable data
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting
↳ FrequencyRunperiod Output:Variable data
Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting
↳ FrequencyRunperiod Output:Variable data
Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting
↳ FrequencyRunperiod Output:Variable data
Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - Zone Operative Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
↳ Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
↳ Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Cooling Coil Total Cooling Rate Reporting FrequencyRunperiod Output:Variable data
Added - Heating Coil Heating Rate Reporting FrequencyRunperiod Output:Variable data
Added - Facility Total HVAC Electric Demand Power Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyRunperiod Output:
↳ Variable data
Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyRunperiod
↳ Output:Variable data
Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyRunperiod Output:
↳ Variable data
Added - AFN Zone Infiltration Volume Reporting FrequencyRunperiod Output:Variable data
Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyRunperiod Output:
↳ Variable data
Added - AFN Zone Ventilation Volume Reporting FrequencyRunperiod Output:Variable data
Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Site Wind Speed Reporting FrequencyRunperiod Output:Variable data
Added - Site Outdoor Air Relative Humidity Reporting FrequencyRunperiod Output:Variable

```

(continues on next page)

(continued from previous page)

```

→data
Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyRunperiod Output:
→Variable data
Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyRunperiod Output:
→Variable data
Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
→Variable data
Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
→Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
→Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
→Variable data
IDF has been saved
Ending with file:
TestModel

```

=====END OF GENERIC IDF FILE GENERATION PROCESS=====

The following IDFs will not work, and therefore these will be deleted:

None

=====START OF OUTPUT IDF FILES GENERATION
→PROCESS=====

The information you will be required to enter below will be used to generate the
→customised output IDFs:

Enter the Comfort Standard numbers separated by space (

```

0 = ESP CTE;
1 = INT EN16798-1;
2 = INT ASHRAE55;
3 = JPN Rijal;
4 = CHN GBT50785 Cold;
5 = CHN GBT50785 HotMild;
6 = CHN Yang;
7 = IND IMAC C NV;
8 = IND IMAC C MM;
9 = IND IMAC R 7DRM;
10 = IND IMAC R 30DRM;
11 = IND Dhaka;
12 = ROM Udrea;
13 = AUS Williamson;
14 = AUS DeDear;
15 = BRA Rupp NV;
16 = BRA Rupp AC;
17 = MEX Oropeza Arid;
18 = MEX Oropeza DryTropic;
19 = MEX Oropeza Temperate;

```

(continues on next page)

(continued from previous page)

```

20 = MEX Oropeza HumTropic;
21 = CHL Perez-Fargallo;
22 = INT ISO7730;
Please refer to the full list of setpoint temperatures at https://raw.githubusercontent.com/
  ↵dsanchez-garcia/accim/master/docs/full_setpoint_table.html
): 1 14

```

Are you sure the numbers are correct? [y or [] / n]:

For the comfort standard 1 = INT EN16798, the available categories you can choose are:

```

1 = EN16798 Category I
2 = EN16798 Category II
3 = EN16798 Category III

```

For the comfort standard 14 = AUS DeDear, the available categories you can choose are:

```

80 = 80% acceptability
90 = 90% acceptability

```

Enter the Category numbers separated by space (

```

1 = CAT I / CAT A;
2 = CAT II / CAT B;
3 = CAT III / CAT C;
80 = 80% ACCEPT;
85 = 85% ACCEPT;
90 = 90% ACCEPT;

```

```

Please refer to the full list of setpoint temperatures at https://raw.githubusercontent.com/
  ↵dsanchez-garcia/accim/master/docs/full_setpoint_table.html

```

```
): 1 2 3 80 90

```

Are you sure the numbers are correct? [y or [] / n]:

For the comfort standard 1 = INT EN16798, the available ComfMods you can choose are:

```

0 = EN16798 Static setpoints
1 = EN16798 Adaptive setpoints when applicable, otherwise CTE
2 = EN16798 Adaptive setpoints when applicable, otherwise EN16798 Static setpoints
3 = EN16798 Adaptive setpoints when applicable, otherwise EN16798 Adaptive setpoints
  ↵horizontally extended

```

For the comfort standard 14 = AUS DeDear, the available ComfMods you can choose are:

```

0.1 = Australian Building Code Static setpoints for climate zones 1, 2 and 3
0.2 = Australian Building Code Static setpoints for climate zone 4
0.3 = Australian Building Code Static setpoints for climate zone 5
0.4 = Australian Building Code Static setpoints for climate zones 6 and 7
0.5 = Australian Building Code Static setpoints for climate zone 8
1.1 = DeDear Model Adaptive setpoints when applicable, otherwise Australian Building
  ↵Code Static setpoints for climate zones 1, 2 and 3
1.2 = DeDear Model Adaptive setpoints when applicable, otherwise Australian Building
  ↵Code Static setpoints for climate zone 4
1.3 = DeDear Model Adaptive setpoints when applicable, otherwise Australian Building
  ↵Code Static setpoints for climate zone 5
1.4 = DeDear Model Adaptive setpoints when applicable, otherwise Australian Building
  ↵Code Static setpoints for climate zones 6 and 7
1.5 = DeDear Model Adaptive setpoints when applicable, otherwise Australian Building
  ↵Code Static setpoints for climate zone 8
2 = DeDear Model Adaptive setpoints when applicable, otherwise ISO 7730 Static setpoints

```

(continues on next page)

(continued from previous page)

```
3 = DeDear Model Adaptive setpoints when applicable, otherwise Adaptive setpoints
↳ horizontally extended
Enter the Comfort Mode numbers separated by space (
Please refer to the full list of setpoint temperatures at https://raw.githubusercontent.com/
↳ dsanchez-garcia/accim/master/docs/full_setpoint_table.html
): 0 0.1 0.2 0.3 0.4 0.5 1 1.1 1.2 1.3 1.4 1.5 2 3
    Are you sure the numbers are correct? [y or [] / n]:
Enter the setpoint accuracy number (any number greater than 0): 100
    Are you sure the number is correct? [y or [] / n]:
Enter the start of the cooling season in numeric date format dd/mm or the day of the
↳ year: 01/11
    Are you sure the number is correct? [y or [] / n]:
Enter the end of the cooling season in numeric date format dd/mm or the day of the year:
↳ 01/03
    Are you sure the number is correct? [y or [] / n]:

Enter the HVAC Mode numbers separated by space (
0 = Fully Air-conditioned;
1 = Naturally ventilated;
2 = Mixed Mode;
): 2
    Are you sure the numbers are correct? [y or [] / n]:

Enter the Ventilation Control numbers separated by space (
If HVACmode = 1:
    0 = Ventilates above neutral temperature;
    1 = Ventilates above upper comfort limit;
If HVACmode = 2:
    0 = Ventilates above neutral temperature and fully opens doors and windows;
    1 = Ventilates above lower comfort limit and fully opens doors and windows;
    2 = Ventilates above neutral temperature and opens doors and windows based on the
↳ customised venting opening factor;
    3 = Ventilates above lower comfort limit and opens doors and windows based on the
↳ customised venting opening factor;
): 2
    Are you sure the numbers are correct? [y or [] / n]:
Enter the maximum temperature difference number for Ventilation Opening Factor (any
↳ number larger than 0): 20
    Are you sure the number is correct? [y or [] / n]:
Enter the minimum temperature difference number for Ventilation Opening Factor (any
↳ number larger than 0 and smaller than the maximum temperature difference number): 1
    Are you sure the number is correct? [y or [] / n]:
Enter the multiplier number for Ventilation Opening Factor (any number between 0 and 1):
↳ 0.2
    Are you sure the number is correct? [y or [] / n]:

Enter the VSToffset numbers separated by space (if omitted, will be 0):
    Are you sure the numbers are correct? [y or [] / n]:

Enter the MinOToffset numbers separated by space (if omitted, will be 50):
    Are you sure the numbers are correct? [y or [] / n]:
```

(continues on next page)

(continued from previous page)

Enter the MaxWindSpeed numbers separated by space (if omitted, will be 50):

Are you sure the numbers are correct? [y or [] / n]:

Enter the ASTtol value from (if omitted, will be 0.1):

Are you sure the numbers are correct? [y or [] / n]:

Enter the ASTtol value to (if omitted, will be 0.1):

Are you sure the numbers are correct? [y or [] / n]:

Enter the ASTtol value steps (if omitted, will be 0.1):

Are you sure the numbers are correct? [y or [] / n]:

The list of output IDFs is going to be:

```
TestModel[CS_INT EN16798[CA_1[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
And the total number of output IDFs is going to be 36
```

Do you still want to run ACCIS? [y/n]: y

Generating the following output IDF files:

```
TestModel[CS_INT EN16798[CA_1[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

(continues on next page)

(continued from previous page)

```

TestModel[CS_INT EN16798[CA_1[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf

```

=====END OF OUTPUT IDF FILES GENERATION PROCESS=====

[3]: <accim.sim.accis.addAccis at 0x201e2ca3a90>

Let's see what files we do have now in the folder:

```

[4]: os.listdir()
[4]: ['.ipynb_checkpoints',
      'TestModel.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf'],

```

(continues on next page)

(continued from previous page)

```
'TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_80[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_80[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_1[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_1[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_1[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_1[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_2[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_2[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_2[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_2[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_3[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_3[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_3[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_3[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'using_addAccis.ipynb',
'__init__.py']
```

We can see there are new IDFs, as stated previously in the python console:

```
[5]: output_idfs = [i for i in os.listdir() if not(any(i in j for j in input_idf)) and i.
    ↪endswith('.idf')]
print(*output_idfs, sep='\n')
```

```
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

(continues on next page)

(continued from previous page)

```
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3.0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

Now, these are ready for simulation with any EPW file, although it is recommended to have previously renamed them with the class `rename_epw_files()`.

Let's remove the new output IDFs for clarity purposes:

```
[6]: for i in os.listdir():
    if i.endswith('.idf') and input_idf[0] not in i:
        os.remove(i)
```

8.1.2 1.2 Longer version

Let's double check we only have the initial idf we used in the short version:

```
[8]: input_idf = [i for i in os.listdir() if i.endswith('.idf')]
print(*input_idf, sep='\n')

TestModel.idf
```

Now, let's do the same, but using the longer version:

```
[11]: from accim.sim import accis
accis.addAccis(
    ScriptType='vrf_mm',
    SupplyAirTempInputMethod='temperature difference',
    Output_keep_existing=False,
    Output_type='standard',
    Output_freqs=['hourly', 'runperiod'],
    Output_gen_dataframe=False,
    EnergyPlus_version='23.1',
    TempCtrl='temperature',
    ComfStand=[1, 14],
```

(continues on next page)

(continued from previous page)

```

CAT=[1, 2, 3, 80, 90],
ComfMod=[0, 0.1, 0.2, 0.3, 0.4, 0.5, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 2, 3],
SetpointAcc=100,
CoolSeasonStart='01/11',
CoolSeasonEnd='01/03',
HVACmode=[2],
VentCtrl=[2],
MaxTempDiffVOF=20,
MinTempDiffVOF=1,
MultiplierVOF=0.2,
VSToffset=[0],
MinOToffset=[50],
MaxWindSpeed=[50],
ASTtol_steps=0.1,
ASTtol_start=0.1,
ASTtol_end_input=0.1,
confirmGen=True
)

```

Adaptive-Comfort-Control-Implemented Model (ACCIM)

This tool allows to apply adaptive setpoint temperatures.

For further information, please read the documentation:

<https://accim.readthedocs.io/en/master/>

For a visual understanding of the tool, please visit the following jupyter notebooks:

- Using addAccis() to apply adaptive setpoint temperatures

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/addAccis/using_addAccis.ipynb

- Using rename_epw_files() to rename the EPWs for proper data analysis after simulation

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/rename_epw_files/using_rename_epw_files.ipynb

- Using runEp() to directly run simulations with EnergyPlus

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/runEp/using_runEp.ipynb

- Using the class Table() for data analysis

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/Table/using_Table.ipynb

Starting with the process.

Basic input data:

ScriptType is: vrf_mm

Supply Air Temperature Input Method is: temperature difference

Output type is: standard

Output frequencies are:

['hourly', 'runperiod']

EnergyPlus version is: 23.1

Temperature Control method is: temperature

(continues on next page)

(continued from previous page)

```
=====START OF GENERIC IDF FILE GENERATION=====
```

```
→PROCESS=====
```

Starting with file:

TestModel

IDD location is: C:\EnergyPlusV23-1-0\Energy+.idd

The occupied zones in the model TestModel are:

BLOCK1:ZONE2

BLOCK1:ZONE1

The windows and doors in the model TestModel are:

Block1_Zone2_Wall_3_0_0_0_0_0_Win

Block1_Zone2_Wall_4_0_0_0_0_0_Win

Block1_Zone2_Wall_5_0_0_0_0_0_Win

Block1_Zone1_Wall_2_0_0_0_0_0_Win

Block1_Zone1_Wall_3_0_0_0_0_0_Win

Block1_Zone1_Wall_5_0_0_0_0_0_Win

The zones in the model TestModel are:

BLOCK1_ZONE2

BLOCK1_ZONE1

The people objects in the model have been amended.

BLOCK1:ZONE2 Thermostat has been added

BLOCK1:ZONE1 Thermostat has been added

On Schedule already was in the model

TypOperativeTempControlSch Schedule already was in the model

All ZoneHVAC:IdealLoadsAirSystem Heating and Cooling availability schedules has been set

→ to on

On 24/7 Schedule already was in the model

Control type schedule: Always 4 Schedule has been added

Relative humidity setpoint schedule: Always 50.00 Schedule has been added

Heating Fanger comfort setpoint: Always -0.5 Schedule has been added

Cooling Fanger comfort setpoint: Always 0.1 Schedule has been added

Zone CO2 setpoint: Always 900ppm Schedule has been added

Min CO2 concentration: Always 600ppm Schedule has been added

Generic contaminant setpoint: Always 0.5ppm Schedule has been added

Air distribution effectiveness (always 1) Schedule has been added

VRF Heating Cooling (Northern Hemisphere) Schedule has been added

DefaultFanEffRatioCurve Curve:Cubic Object has been added

VRFTUCoolCapFT Curve:Cubic Object has been added

VRFTUHeatCapFT Curve:Cubic Object has been added

VRFCoolCapFTBoundary Curve:Cubic Object has been added

VRFCoolEIRFTBoundary Curve:Cubic Object has been added

CoolingEIRLowPLR Curve:Cubic Object has been added

VRFHeatCapFTBoundary Curve:Cubic Object has been added

VRFHeatEIRFTBoundary Curve:Cubic Object has been added

HeatingEIRLowPLR Curve:Cubic Object has been added

DefaultFanPowerRatioCurve Curve:Exponent Object has been added

DXHtgCoilDefrostEIRFT Curve:Biquadratic Object has been added

VRFCoolCapFT Curve:Biquadratic Object has been added

VRFCoolCapFTHi Curve:Biquadratic Object has been added

VRFCoolEIRFT Curve:Biquadratic Object has been added

VRFCoolEIRFTHi Curve:Biquadratic Object has been added

VRFHeatCapFT Curve:Biquadratic Object has been added

(continues on next page)

(continued from previous page)

VRFHeatCapFTHi Curve:Biquadratic Object has been added
 VRFHeatEIRFT Curve:Biquadratic Object has been added
 VRFHeatEIRFTHi Curve:Biquadratic Object has been added
 CoolingLengthCorrectionFactor Curve:Biquadratic Object has been added
 VRF Piping Correction Factor for Length in Heating Mode Curve:Biquadratic Object has been added
 VRF Heat Recovery Cooling Capacity Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Cooling Energy Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Heating Capacity Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Heating Energy Modifier Curve:Biquadratic Object has been added
 VRFACCoolCapFFF Curve:Quadratic Object has been added
 CoolingEIRHiPLR Curve:Quadratic Object has been added
 VRFCPLFFPLR Curve:Quadratic Object has been added
 HeatingEIRHiPLR Curve:Quadratic Object has been added
 CoolingCombRatio Curve:Linear Object has been added
 HeatingCombRatio Curve:Linear Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 AirConditioner:VariableRefrigerantFlow Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 AirConditioner:VariableRefrigerantFlow Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 Zone List Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Zone List Object has been added
 BLOCK1:ZONE2 Sizing:Zone Object has been added
 BLOCK1:ZONE1 Sizing:Zone Object has been added
 BLOCK1:ZONE2 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE1 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE2 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE1 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE2 Nodelist Objects has been added
 BLOCK1:ZONE1 Nodelist Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE1 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentList Objects has been added
 BLOCK1:ZONE1 ZoneHVAC:EquipmentList Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Fan:ConstantVolume Object has been added
 BLOCK1:ZONE1 Fan:ConstantVolume Object has been added
 Vent_SP_temp Schedule has been added
 AHST_Sch_BLOCK1_ZONE2 Schedule has been added
 ACST_Sch_BLOCK1_ZONE2 Schedule has been added
 AHST_Sch_BLOCK1_ZONE1 Schedule has been added
 ACST_Sch_BLOCK1_ZONE1 Schedule has been added
 Added - SetComfTemp Program
 Added - CountHours_BLOCK1_ZONE2 Program
 Added - CountHours_BLOCK1_ZONE1 Program

(continues on next page)

(continued from previous page)

Added - SetAppLimits Program
Added - ApplyCAT Program
Added - SetAST Program
Added - SetASTnoTol Program
Added - CountHoursNoApp_BLOCK1_ZONE2 Program
Added - SetGeoVarBLOCK1_ZONE2 Program
Added - CountHoursNoApp_BLOCK1_ZONE1 Program
Added - SetGeoVarBLOCK1_ZONE1 Program
Added - SetInputData Program
Added - SetVOFinputData Program
Added - SetVST Program
Added - ApplyAST_BLOCK1_ZONE2 Program
Added - ApplyAST_BLOCK1_ZONE1 Program
Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_Win Program
Added - Comfort Temperature Output Variable
Added - Adaptive Cooling Setpoint Temperature Output Variable
Added - Adaptive Heating Setpoint Temperature Output Variable
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Output Variable
Added - Adaptive Heating Setpoint Temperature_No Tolerance Output Variable
Added - Ventilation Setpoint Temperature Output Variable
Added - Minimum Outdoor Temperature for ventilation Output Variable
Added - Minimum Outdoor Temperature Difference for ventilation Output Variable
Added - Maximum Outdoor Temperature Difference for ventilation Output Variable
Added - Multiplier for Ventilation Opening Factor Output Variable
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Output Variable
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Output Variable
Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Output Variable
Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
Added - Zone Floor Area_BLOCK1_ZONE2 Output Variable
Added - Zone Floor Area_BLOCK1_ZONE1 Output Variable
Added - Zone Air Volume_BLOCK1_ZONE2 Output Variable
Added - Zone Air Volume_BLOCK1_ZONE1 Output Variable
Added - Ventilation Hours_BLOCK1_ZONE2 Output Variable

(continues on next page)

(continued from previous page)

```

Added - Ventilation Hours_BLOCK1_ZONE1 Output Variable
Global variables objects have been added
Internal variables objects have been added
Added - RMOT Sensor
Added - PMOT Sensor
Added - BLOCK1_ZONE2_OpT Sensor
Added - BLOCK1_ZONE2_WindSpeed Sensor
Added - BLOCK1_ZONE2_OutT Sensor
Added - BLOCK1_ZONE1_OpT Sensor
Added - BLOCK1_ZONE1_WindSpeed Sensor
Added - BLOCK1_ZONE1_OutT Sensor
Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OpT Sensor
Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OutT Sensor
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OpT Sensor
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_WindSpeed Sensor
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OutT Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OpT Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OutT Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OpT Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_WindSpeed Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OutT Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OpT Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OutT Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OpT Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OutT Sensor
Added - OutT Sensor
Added - AHST_Act_BLOCK1_ZONE2 Actuator
Added - ACST_Act_BLOCK1_ZONE2 Actuator
Added - AHST_Act_BLOCK1_ZONE1 Actuator
Added - ACST_Act_BLOCK1_ZONE1 Actuator
Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_VentOpenFact Actuator
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_VentOpenFact Actuator
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
Added - BLOCK1_ZONE2_CoolCoil Sensor
Added - BLOCK1_ZONE2_HeatCoil Sensor
Added - BLOCK1_ZONE1_CoolCoil Sensor
Added - BLOCK1_ZONE1_HeatCoil Sensor
Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_HeatCoil Sensor

```

(continues on next page)

(continued from previous page)

```

Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
Added - SetComfTemp Program Calling Manager
Added - CountHours_BLOCK1_ZONE2 Program Calling Manager
Added - CountHours_BLOCK1_ZONE1 Program Calling Manager
Added - SetAppLimits Program Calling Manager
Added - ApplyCAT Program Calling Manager
Added - SetAST Program Calling Manager
Added - SetASTnoTol Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE2 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE2 Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE1 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE1 Program Calling Manager
Added - SetInputData Program Calling Manager
Added - SetVOFinputData Program Calling Manager
Added - SetVST Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE2 Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE1 Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - Comfort Temperature Reporting FrequencyHourly Output:Variable data
Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyHourly Output:Variable↳  
data
Added - Adaptive Heating Setpoint Temperature Reporting FrequencyHourly Output:Variable↳  
data
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyHourly↳  
Output:Variable data
Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyHourly↳  
Output:Variable data
Added - Ventilation Setpoint Temperature Reporting FrequencyHourly Output:Variable data
Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyHourly Output:  
↳ Variable data
Added - Minimum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly↳  
Output:Variable data
Added - Maximum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly↳  
Output:Variable data
Added - Multiplier for Ventilation Opening Factor Reporting FrequencyHourly Output:  
↳ Variable data
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:  
↳ Variable data

```

(continues on next page)

(continued from previous page)

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyHourly Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyHourly Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyHourly Output:Variable data

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyHourly Output:
 ↳ Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyHourly
 ↳ Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳ data

Added - AFN Zone Infiltration Volume Reporting FrequencyHourly Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳ data

Added - AFN Zone Ventilation Volume Reporting FrequencyHourly Output:Variable data

Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyHourly Output:Variable
 ↳ data

Added - Site Wind Speed Reporting FrequencyHourly Output:Variable data

Added - Site Outdoor Air Relative Humidity Reporting FrequencyHourly Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

(continues on next page)

(continued from previous page)

Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyHourly Output:
↳ Variable data
Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyHourly Output:
↳ Variable data
Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
↳ Variable data
Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
↳ Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
↳ Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
↳ Variable data
Added - Comfort Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Adaptive Heating Setpoint Temperature Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Ventilation Setpoint Temperature Reporting FrequencyRunperiod Output:Variable
↳ data
Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Minimum Outdoor Temperature Difference for ventilation Reporting
↳ FrequencyRunperiod Output:Variable data
Added - Maximum Outdoor Temperature Difference for ventilation Reporting
↳ FrequencyRunperiod Output:Variable data
Added - Multiplier for Ventilation Opening Factor Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
↳ Output:Variable data
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting
↳ FrequencyRunperiod Output:Variable data

(continues on next page)

(continued from previous page)

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting
 ↳ FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting
 ↳ FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting
 ↳ FrequencyRunperiod Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyRunperiod Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyRunperiod Output:Variable data

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyRunperiod
 ↳ Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - AFN Zone Infiltration Volume Reporting FrequencyRunperiod Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - AFN Zone Ventilation Volume Reporting FrequencyRunperiod Output:Variable data

Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - Site Wind Speed Reporting FrequencyRunperiod Output:Variable data

Added - Site Outdoor Air Relative Humidity Reporting FrequencyRunperiod Output:Variable
 ↳ data

Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data

(continues on next page)

(continued from previous page)

IDF has been saved
 Ending with file:
 TestModel

=====END OF GENERIC IDF FILE GENERATION PROCESS=====

The following IDFs will not work, and therefore these will be deleted:

None

=====START OF OUTPUT IDF FILES GENERATION
 ↵PROCESS=====

The list of output IDFs is going to be:

```
TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
And the total number of output IDFs is going to be 36
Generating the following output IDF files:
TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

(continues on next page)

(continued from previous page)

```

TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf

```

=====END OF OUTPUT IDF FILES GENERATION PROCESS=====

[11]: <accim.sim.accis.addAccis at 0x201889c10a0>

Let's see what files we do have now in the folder:

```

[12]: os.listdir()
[12]: ['.ipynb_checkpoints',
      'TestModel.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
      'TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',

```

(continues on next page)

(continued from previous page)

```
'TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf',
'using_addAccis.ipynb',
'__init__.py']
```

We can see there are new IDFs, as stated previously in the python console:

```
[13]: output_idfs = [i for i in os.listdir() if not(any(i in j for j in input_idf)) and i.
    ~endswith('.idf')]
print(*output_idfs, sep='\n')

TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf]
```

(continues on next page)

(continued from previous page)

```
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

Now, these are ready for simulation with any EPW file, although it is recommended to have previously renamed them with the class `rename_epw_files()`.

Let's remove the new output IDFs for clarity purposes:

```
[14]: for i in os.listdir():
    if i.endswith('.idf') and input_idf[0] not in i:
        os.remove(i)
```

8.2 2. Other uses

8.2.1 2.1 Storing the `addAccis()` class and inspecting the output IDFs

Additionally to just run `addAccis()` (i.e. instantiate the class), the user can also store the instance and inspect some variables. So let's store the instance in the variable `add_accis_instance` and then we'll take a look what's inside:

```
[13]: from accim.sim import accis
add_accis_instance = accis.addAccis(
    ScriptType='vrf_mm',
    SupplyAirTempInputMethod='temperature difference',
    Output_keep_existing=False,
    Output_type='standard',
    Output_freqs=['hourly', 'runperiod'],
    Output_gen_dataframe=False,
    EnergyPlus_version='23.1',
    TempCtrl='temperature',
    ComfStand=[1, 14],
```

(continues on next page)

(continued from previous page)

```

CAT=[1, 2, 3, 80, 90],
ComfMod=[0, 0.1, 0.2, 0.3, 0.4, 0.5, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 2, 3],
SetpointAcc=100,
CoolSeasonStart='01/11',
CoolSeasonEnd='01/03',
HVACmode=[2],
VentCtrl=[2],
MaxTempDiffVOF=20,
MinTempDiffVOF=1,
MultiplierVOF=0.2,
VSToffset=[0],
MinOToffset=[50],
MaxWindSpeed=[50],
ASTtol_steps=0.1,
ASTtol_start=0.1,
ASTtol_end_input=0.1,
confirmGen=True
)

```

Adaptive-Comfort-Control-Implemented Model (ACCIM)

This tool allows to apply adaptive setpoint temperatures.

For further information, please read the documentation:

<https://accim.readthedocs.io/en/master/>

For a visual understanding of the tool, please visit the following jupyter notebooks:

- Using addAccis() to apply adaptive setpoint temperatures

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/addAccis/using_addAccis.ipynb

- Using rename_epw_files() to rename the EPWs for proper data analysis after simulation

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/rename_epw_files/using_rename_epw_files.ipynb

- Using runEp() to directly run simulations with EnergyPlus

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/runEp/using_runEp.ipynb

- Using the class Table() for data analysis

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/Table/using_Table.ipynb

Starting with the process.

Basic input data:

ScriptType is: vrf_mm

Supply Air Temperature Input Method is: temperature difference

Output type is: standard

Output frequencies are:

['hourly', 'runperiod']

EnergyPlus version is: 23.1

Temperature Control method is: temperature

(continues on next page)

(continued from previous page)

```
=====START OF GENERIC IDF FILE GENERATION=====
```

```
→PROCESS=====
```

Starting with file:

TestModel

IDD location is: C:\EnergyPlusV23-1-0\Energy+.idd

The occupied zones in the model TestModel are:

BLOCK1:ZONE2

BLOCK1:ZONE1

The windows and doors in the model TestModel are:

Block1_Zone2_Wall_3_0_0_0_0_0_Win

Block1_Zone2_Wall_4_0_0_0_0_0_Win

Block1_Zone2_Wall_5_0_0_0_0_0_Win

Block1_Zone1_Wall_2_0_0_0_0_0_Win

Block1_Zone1_Wall_3_0_0_0_0_0_Win

Block1_Zone1_Wall_5_0_0_0_0_0_Win

The zones in the model TestModel are:

BLOCK1:ZONE2

BLOCK1:ZONE1

The people objects in the model have been amended.

BLOCK1:ZONE2 Thermostat has been added

BLOCK1:ZONE1 Thermostat has been added

On Schedule already was in the model

TypOperativeTempControlSch Schedule already was in the model

All ZoneHVAC:IdealLoadsAirSystem Heating and Cooling availability schedules has been set

→ to on

On 24/7 Schedule already was in the model

Control type schedule: Always 4 Schedule has been added

Relative humidity setpoint schedule: Always 50.00 Schedule has been added

Heating Fanger comfort setpoint: Always -0.5 Schedule has been added

Cooling Fanger comfort setpoint: Always 0.1 Schedule has been added

Zone CO2 setpoint: Always 900ppm Schedule has been added

Min CO2 concentration: Always 600ppm Schedule has been added

Generic contaminant setpoint: Always 0.5ppm Schedule has been added

Air distribution effectiveness (always 1) Schedule has been added

VRF Heating Cooling (Northern Hemisphere) Schedule has been added

DefaultFanEffRatioCurve Curve:Cubic Object has been added

VRFTUCoolCapFT Curve:Cubic Object has been added

VRFTUHeatCapFT Curve:Cubic Object has been added

VRFCoolCapFTBoundary Curve:Cubic Object has been added

VRFCoolEIRFTBoundary Curve:Cubic Object has been added

CoolingEIRLowPLR Curve:Cubic Object has been added

VRFHeatCapFTBoundary Curve:Cubic Object has been added

VRFHeatEIRFTBoundary Curve:Cubic Object has been added

HeatingEIRLowPLR Curve:Cubic Object has been added

DefaultFanPowerRatioCurve Curve:Exponent Object has been added

DXHtgCoilDefrostEIRFT Curve:Biquadratic Object has been added

VRFCoolCapFT Curve:Biquadratic Object has been added

VRFCoolCapFTHi Curve:Biquadratic Object has been added

VRFCoolEIRFT Curve:Biquadratic Object has been added

VRFCoolEIRFTHi Curve:Biquadratic Object has been added

VRFHeatCapFT Curve:Biquadratic Object has been added

(continues on next page)

(continued from previous page)

VRFHeatCapFTHi Curve:Biquadratic Object has been added
VRFHeatEIRFT Curve:Biquadratic Object has been added
VRFHeatEIRFTHi Curve:Biquadratic Object has been added
CoolingLengthCorrectionFactor Curve:Biquadratic Object has been added
VRF Piping Correction Factor for Length in Heating Mode Curve:Biquadratic Object has been added
VRF Heat Recovery Cooling Capacity Modifier Curve:Biquadratic Object has been added
VRF Heat Recovery Cooling Energy Modifier Curve:Biquadratic Object has been added
VRF Heat Recovery Heating Capacity Modifier Curve:Biquadratic Object has been added
VRF Heat Recovery Heating Energy Modifier Curve:Biquadratic Object has been added
VRFACCCoolCapFFF Curve:Quadratic Object has been added
CoolingEIRHiPLR Curve:Quadratic Object has been added
VRFCPLFFPLR Curve:Quadratic Object has been added
HeatingEIRHiPLR Curve:Quadratic Object has been added
CoolingCombRatio Curve:Linear Object has been added
HeatingCombRatio Curve:Linear Object has been added
VRF Outdoor Unit_BLOCK1:ZONE2 AirConditioner:VariableRefrigerantFlow Object has been added
VRF Outdoor Unit_BLOCK1:ZONE1 AirConditioner:VariableRefrigerantFlow Object has been added
VRF Outdoor Unit_BLOCK1:ZONE2 Outdoor Air Node Object has been added
VRF Outdoor Unit_BLOCK1:ZONE2 Zone List Object has been added
VRF Outdoor Unit_BLOCK1:ZONE1 Outdoor Air Node Object has been added
VRF Outdoor Unit_BLOCK1:ZONE1 Zone List Object has been added
BLOCK1:ZONE2 Sizing:Zone Object has been added
BLOCK1:ZONE1 Sizing:Zone Object has been added
BLOCK1:ZONE2 Design Specification Outdoor Air Object has been added
BLOCK1:ZONE1 Design Specification Outdoor Air Object has been added
BLOCK1:ZONE2 Design Specification Zone Air Distribution Object has been added
BLOCK1:ZONE1 Design Specification Zone Air Distribution Object has been added
BLOCK1:ZONE2 Nodelist Objects has been added
BLOCK1:ZONE1 Nodelist Objects has been added
BLOCK1:ZONE2 ZoneHVAC:EquipmentConnections Objects has been added
BLOCK1:ZONE1 ZoneHVAC:EquipmentConnections Objects has been added
BLOCK1:ZONE2 ZoneHVAC:EquipmentList Objects has been added
BLOCK1:ZONE1 ZoneHVAC:EquipmentList Objects has been added
BLOCK1:ZONE2 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE1 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE2 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE1 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE2 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE1 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE2 Fan:ConstantVolume Object has been added
BLOCK1:ZONE1 Fan:ConstantVolume Object has been added
Vent_SP_temp Schedule has been added
AHST_Sch_BLOCK1_ZONE2 Schedule has been added
ACST_Sch_BLOCK1_ZONE2 Schedule has been added
AHST_Sch_BLOCK1_ZONE1 Schedule has been added
ACST_Sch_BLOCK1_ZONE1 Schedule has been added
Added - SetComfTemp Program
Added - CountHours_BLOCK1_ZONE2 Program
Added - CountHours_BLOCK1_ZONE1 Program

(continues on next page)

(continued from previous page)

Added - SetAppLimits Program
 Added - ApplyCAT Program
 Added - SetAST Program
 Added - SetASTnoTol Program
 Added - CountHoursNoApp_BLOCK1_ZONE2 Program
 Added - SetGeoVarBLOCK1_ZONE2 Program
 Added - CountHoursNoApp_BLOCK1_ZONE1 Program
 Added - SetGeoVarBLOCK1_ZONE1 Program
 Added - SetInputData Program
 Added - SetVOFinputData Program
 Added - SetVST Program
 Added - ApplyAST_BLOCK1_ZONE2 Program
 Added - ApplyAST_BLOCK1_ZONE1 Program
 Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_Win Program
 Added - Comfort Temperature Output Variable
 Added - Adaptive Cooling Setpoint Temperature Output Variable
 Added - Adaptive Heating Setpoint Temperature Output Variable
 Added - Adaptive Cooling Setpoint Temperature_No Tolerance Output Variable
 Added - Adaptive Heating Setpoint Temperature_No Tolerance Output Variable
 Added - Ventilation Setpoint Temperature Output Variable
 Added - Minimum Outdoor Temperature for ventilation Output Variable
 Added - Minimum Outdoor Temperature Difference for ventilation Output Variable
 Added - Maximum Outdoor Temperature Difference for ventilation Output Variable
 Added - Multiplier for Ventilation Opening Factor Output Variable
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Output Variable
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Output Variable
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Output Variable
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE2 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE1 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE2 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE1 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE2 Output Variable

(continues on next page)

(continued from previous page)

Added - Ventilation Hours_BLOCK1_ZONE1 Output Variable
 Global variables objects have been added
 Internal variables objects have been added
 Added - RMOT Sensor
 Added - PMOT Sensor
 Added - BLOCK1_ZONE2_OpT Sensor
 Added - BLOCK1_ZONE2_WindSpeed Sensor
 Added - BLOCK1_ZONE2_OutT Sensor
 Added - BLOCK1_ZONE1_OpT Sensor
 Added - BLOCK1_ZONE1_WindSpeed Sensor
 Added - BLOCK1_ZONE1_OutT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - OutT Sensor
 Added - AHST_Act_BLOCK1_ZONE2 Actuator
 Added - ACST_Act_BLOCK1_ZONE2 Actuator
 Added - AHST_Act_BLOCK1_ZONE1 Actuator
 Added - ACST_Act_BLOCK1_ZONE1 Actuator
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - BLOCK1_ZONE2_CoolCoil Sensor
 Added - BLOCK1_ZONE2_HeatCoil Sensor
 Added - BLOCK1_ZONE1_CoolCoil Sensor
 Added - BLOCK1_ZONE1_HeatCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_HeatCoil Sensor

(continues on next page)

(continued from previous page)

```

Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
Added - SetComfTemp Program Calling Manager
Added - CountHours_BLOCK1_ZONE2 Program Calling Manager
Added - CountHours_BLOCK1_ZONE1 Program Calling Manager
Added - SetAppLimits Program Calling Manager
Added - ApplyCAT Program Calling Manager
Added - SetAST Program Calling Manager
Added - SetASTnoTol Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE2 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE2 Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE1 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE1 Program Calling Manager
Added - SetInputData Program Calling Manager
Added - SetVOFinputData Program Calling Manager
Added - SetVST Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE2 Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE1 Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - Comfort Temperature Reporting FrequencyHourly Output:Variable data
Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyHourly Output:Variable↳  
data
Added - Adaptive Heating Setpoint Temperature Reporting FrequencyHourly Output:Variable↳  
data
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyHourly↳  
Output:Variable data
Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyHourly↳  
Output:Variable data
Added - Ventilation Setpoint Temperature Reporting FrequencyHourly Output:Variable data
Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyHourly Output:  
↳ Variable data
Added - Minimum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly↳  
Output:Variable data
Added - Maximum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly↳  
Output:Variable data
Added - Multiplier for Ventilation Opening Factor Reporting FrequencyHourly Output:  
↳ Variable data
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:  
↳ Variable data

```

(continues on next page)

(continued from previous page)

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyHourly Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyHourly Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyHourly Output:Variable data

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyHourly Output:
 ↳ Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyHourly
 ↳ Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳ data

Added - AFN Zone Infiltration Volume Reporting FrequencyHourly Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳ data

Added - AFN Zone Ventilation Volume Reporting FrequencyHourly Output:Variable data

Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyHourly Output:Variable
 ↳ data

Added - Site Wind Speed Reporting FrequencyHourly Output:Variable data

Added - Site Outdoor Air Relative Humidity Reporting FrequencyHourly Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

(continues on next page)

(continued from previous page)

Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
 Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
 Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
 Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - Comfort Temperature Reporting FrequencyRunperiod Output:Variable data
 Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Added - Adaptive Heating Setpoint Temperature Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod
 ↳ Output:Variable data
 Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod
 ↳ Output:Variable data
 Added - Ventilation Setpoint Temperature Reporting FrequencyRunperiod Output:Variable
 ↳ data
 Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Added - Minimum Outdoor Temperature Difference for ventilation Reporting
 ↳ FrequencyRunperiod Output:Variable data
 Added - Maximum Outdoor Temperature Difference for ventilation Reporting
 ↳ FrequencyRunperiod Output:Variable data
 Added - Multiplier for Ventilation Opening Factor Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod
 ↳ Output:Variable data
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod
 ↳ Output:Variable data
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod
 ↳ Output:Variable data
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
 ↳ Output:Variable data
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod
 ↳ Output:Variable data
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
 ↳ Output:Variable data
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting
 ↳ FrequencyRunperiod Output:Variable data

(continues on next page)

(continued from previous page)

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting
 ↳ FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting
 ↳ FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting
 ↳ FrequencyRunperiod Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyRunperiod Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyRunperiod Output:Variable data

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyRunperiod
 ↳ Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - AFN Zone Infiltration Volume Reporting FrequencyRunperiod Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - AFN Zone Ventilation Volume Reporting FrequencyRunperiod Output:Variable data

Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - Site Wind Speed Reporting FrequencyRunperiod Output:Variable data

Added - Site Outdoor Air Relative Humidity Reporting FrequencyRunperiod Output:Variable
 ↳ data

Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data

(continues on next page)

(continued from previous page)

IDF has been saved
 Ending with file:
 TestModel

=====END OF GENERIC IDF FILE GENERATION PROCESS=====

The following IDFs will not work, and therefore these will be deleted:

None

=====START OF OUTPUT IDF FILES GENERATION=====
 ↵PROCESS=====

The list of output IDFs is going to be:

```
TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
And the total number of output IDFs is going to be 36
Generating the following output IDF files:
TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

(continues on next page)

(continued from previous page)

```

TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf

```

=====END OF OUTPUT IDF FILES GENERATION PROCESS=====

Now, let's take a look into the variables within the `addAccis()` instance `add_accis_instance`:

```

[14]: [i for i in dir(add_accis_instance) if '__' not in i]
[14]: ['arguments',
      'input_idfs',
      'occupied_zones',
      'occupied_zones_original_name',
      'output_idfs',
      'windows_and_doors',
      'windows_and_doors_original_name']

```

For instance, we can take a look at the arguments we have requested when “`addAccis`” was called:

```
[15]: add_accis_instance.arguments

[15]: {'ScriptType': 'vrf_mm',
       'SupplyAirTempInputMethod': 'temperature difference',
       'Output_type': 'standard',
       'Output_freqs': ['hourly', 'runperiod'],
       'Output_keep_existing': False,
       'Output_gen_dataframe': False,
       'Output_take_dataframe': None,
       'EnergyPlus_version': '23.1',
       'TempCtrl': 'temperature',
       'ComfStand': [1, 14],
       'CAT': [1, 2, 3, 80, 90],
       'ComfMod': [0, 0.1, 0.2, 0.3, 0.4, 0.5, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 2, 3],
       'SetpointAcc': 100,
       'CoolSeasonStart': '01/11',
       'CoolSeasonEnd': '01/03',
       'HVACmode': [2],
       'VentCtrl': [2],
       'MaxTempDiffVOF': 20,
       'MinTempDiffVOF': 1,
       'MultiplierVOF': 0.2,
       'VSToffset': [0],
       'MinOToffset': [50],
       'MaxWindSpeed': [50],
       'ASTtol_start': 0.1,
       'ASTtol_end_input': 0.1,
       'ASTtol_steps': 0.1,
       'NameSuffix': '',
       'verboseMode': True,
       'confirmGen': True}
```

Also, we can see the list of occupied zones and windows and doors in each model, considering the original names and the new names needed by accim:

```
[16]: add_accis_instance.occupied_zones

[16]: {'TestModel': ['BLOCK1_ZONE2', 'BLOCK1_ZONE1']}

[17]: add_accis_instance.occupied_zones_original_name

[17]: {'TestModel': ['BLOCK1:ZONE2', 'BLOCK1:ZONE1']}

[18]: add_accis_instance.windows_and_doors

[18]: {'TestModel': ['Block1_Zone2_Wall_3_0_0_0_0_0_Win',
                  'Block1_Zone2_Wall_4_0_0_0_0_0_Win',
                  'Block1_Zone2_Wall_5_0_0_0_0_0_Win',
                  'Block1_Zone1_Wall_2_0_0_0_0_0_Win',
                  'Block1_Zone1_Wall_3_0_0_0_0_0_Win',
                  'Block1_Zone1_Wall_5_0_0_0_0_0_Win']}
```

```
[19]: add_accis_instance.windows_and_doors_original_name
```

```
[19]: {'TestModel': ['Block1:Zone2_Wall_3_0_0_0_0_0_Win',
  'Block1:Zone2_Wall_4_0_0_0_0_0_Win',
  'Block1:Zone2_Wall_5_0_0_0_0_0_Win',
  'Block1:Zone1_Wall_2_0_0_0_0_0_Win',
  'Block1:Zone1_Wall_3_0_0_0_0_0_Win',
  'Block1:Zone1_Wall_5_0_0_0_0_0_Win']}
```

As you can see, these variables are dictionaries following the pattern: {'input idf filename': [list]}

Also, we can inspect both the input and output idf files using eppy. In both cases, the variables `input_idfs` and `output_idfs` contains a dictionary following the pattern `{'idf filename': eppy.modeleditor.IDF object}`. For instance, let's take a look at the `input_idfs` variable.

```
[20]: add_accis_instance.input_idfs
```

```
[20]: {'TestModel': <eppy.modeleditor.IDF at 0x2103df0d670>}
```

So let's read the `eppy.modeleditor.IDF` object of the idf `TestModel`. For instance, let's see which zones are in the IDF by reading it, and if these are the same as previously seen in `add_accis_instance.occupied_zones_original_name`:

```
[23]: testmodel = add_accis_instance.input_idfs['TestModel']
testmodel.idfobjects['ZONE']
```

```
[23]: [
Zone,
  Block1:Zone2,
    !- Name
    0,
      !- Direction of Relative North
    0,
      !- X Origin
    0,
      !- Y Origin
    0,
      !- Z Origin
    1,
      !- Type
    1,
      !- Multiplier
    ,
      !- Ceiling Height
    126.7064,
      !- Volume
    36.2018,
      !- Floor Area
    TARP,
      !- Zone Inside Convection Algorithm
    ,
      !- Zone Outside Convection Algorithm
    Yes;
      !- Part of Total Floor Area
  ,
Zone,
  Block1:Zone1,
    !- Name
    0,
      !- Direction of Relative North
    0,
      !- X Origin
    0,
      !- Y Origin
    0,
      !- Z Origin
    1,
      !- Type
    1,
      !- Multiplier
    ,
      !- Ceiling Height
    105.3337,
      !- Volume
    30.0953,
      !- Floor Area
    TARP,
      !- Zone Inside Convection Algorithm
    ,
      !- Zone Outside Convection Algorithm
    Yes;
      !- Part of Total Floor Area
]
```

And we can see these are the same. Also, you can take a look at the variable `output_idfs`, which is another dictionary similar to `input_idfs` but with the output files.

[24]: `add_accis_instance.output_idfs`

```
[24]: {'TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':  
    ↳<eppy.modeleditor.IDF at 0x2104265b280>,  
    'TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
```

(continues on next page)

(continued from previous page)

```

↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>,
'TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf':
↳ <eppy.modeleditor.IDF at 0x2104265b280>}
```

Let's read the last one. Let's see, for instance, the Schedule:Compact objects for the thermostats:

```

[25]: ast_testmodel = add_accis_instance.output_idfs[
    'TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf'
]
ast_testmodel.idfobjects['ThermostatSetpoint:DualSetpoint']

[25]: [
THERMOSTATSETPOINT:DUALSETPOINT,
    BLOCK1:ZONE2 Dual SP,      !- Name
    AHST_Sch_BLOCK1_ZONE2,    !- Heating Setpoint Temperature Schedule Name
    ACST_Sch_BLOCK1_ZONE2;   !- Cooling Setpoint Temperature Schedule Name
,
THERMOSTATSETPOINT:DUALSETPOINT,
    BLOCK1:ZONE1 Dual SP,      !- Name
    AHST_Sch_BLOCK1_ZONE1,    !- Heating Setpoint Temperature Schedule Name
    ACST_Sch_BLOCK1_ZONE1;   !- Cooling Setpoint Temperature Schedule Name
]
```

And now let's compare this with the input idf:

```

[26]: testmodel.idfobjects['ThermostatSetpoint:DualSetpoint']

[26]: [
ThermostatSetpoint:DualSetpoint,
    Dual Setpoint - Zone Block1:Zone2,      !- Name
    Block1:Zone2 Heating Setpoint Schedule,  !- Heating Setpoint Temperature Schedule
    ↳Name
    Block1:Zone2 Cooling SP Sch;       !- Cooling Setpoint Temperature Schedule Name

```

(continues on next page)

(continued from previous page)

```

,
ThermostatSetpoint:DualSetpoint,
    Dual Setpoint - Zone Block1:Zone1,      !- Name
    Block1:Zone1 Heating Setpoint Schedule,   !- Heating Setpoint Temperature Schedule
    ↵Name
        Block1:Zone1 Cooling SP Sch;     !- Cooling Setpoint Temperature Schedule Name
]

```

Therefore, we can confirm the Schedule:Compact objects for output idf stored in variable `ast_testmodel` are AHST and ACST, and thus, adaptive setpoint temperatures have been applied. However, this is different in the original idf, stored in variable `testmodel`, in which setpoint temperatures are static.

For more information about the use of eppy, please refer to the documentation at <https://eppy.readthedocs.io/en/latest/> or https://eppy.readthedocs.io/en/latest/Main_Tutorial.html

Let's delete all the output idf files so that we can re-run the notebook.

```
[ ]: for i in os.listdir():
    if i.endswith('.idf') and input_idf[0] not in i:
        os.remove(i)
```

8.2.2 2.2 Specifying the outputs in the IDF

8.2.2.1 2.2.1 Using Output_type='custom'

Imagine you need to perform a large number of simulations, therefore, using `Output_type='standard'` would result in a heavy dataset. Using `Output_type='simplified'` is not an option, since you need more outputs than just the operative temperature, heating coil heating rate and cooling coil total cooling rate. In that case, you can use `Output_type='custom'`. If you do so, accim will iterate through the existing reporting frequencies and will show you all the existing `Output:Variable` objects for that frequency, and then, you'll be asked the following: - firstly, you'll need to say if you want to remove some object or otherwise keep some and remove all others by entering "remove" or "keep" - secondly, you'll have to say all `Output:Variable` objects you want to remove or keep

So let's say we want to keep the following `Output:Variable` objects (you may not be aware of them, but in that case, you can see the existing objects in the list accim prints), which you'll need to enter separated by semicolon (>):

Zone Operative Temperature;Adaptive Cooling Setpoint Temperature_No Tolerance;Adaptive Heating Setpoint Temperature_No Tolerance

Also, in this case, we are going to store the `addAccis()` instance in the variable `add_accis_instance` (similar to previous section 2.1), so that we can inspect it later.

First of all, let's see the idfs we do have in this path:

```
[32]: from os import listdir
input_idfs = [i for i in listdir() if i.endswith('.idf')]
input_idfs
[32]: ['TestModel.idf']
```

Now, we can go for it:

```
[33]: from accim.sim import accis
add_accis_instance = accis.addAccis()
```

(continues on next page)

(continued from previous page)

```

ScriptType='vrf_mm',
SupplyAirTempInputMethod='temperature difference',
Output_keep_existing=False,
Output_type='custom',
Output_freqs=['hourly', 'runperiod'],
Output_gen_dataframe=False,
EnergyPlus_version='23.1',
TempCtrl='temperature',
ComfStand=[1, 14],
CAT=[1, 2, 3, 80, 90],
ComfMod=[0, 0.1, 0.2, 0.3, 0.4, 0.5, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 2, 3],
SetpointAcc=100,
CoolSeasonStart='01/11',
CoolSeasonEnd='01/03',
HVACmode=[2],
VentCtrl=[2],
MaxTempDiffVOF=20,
MinTempDiffVOF=1,
MultiplierVOF=0.2,
VSToffset=[0],
MinOToffset=[50],
MaxWindSpeed=[50],
ASTtol_steps=0.1,
ASTtol_start=0.1,
ASTtol_end_input=0.1,
confirmGen=True
)
-----
```

Adaptive-Comfort-Control-Implemented Model (ACCIM)

This tool allows to apply adaptive setpoint temperatures.

For further information, please read the documentation:

<https://accim.readthedocs.io/en/master/>

For a visual understanding of the tool, please visit the following jupyter notebooks:

- Using addAccis() to apply adaptive setpoint temperatures

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/addAccis/using_addAccis.ipynb

- Using rename_epw_files() to rename the EPWs for proper data analysis after simulation

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/rename_epw_files/using_rename_epw_files.ipynb

- Using runEp() to directly run simulations with EnergyPlus

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/runEp/using_runEp.ipynb

- Using the class Table() for data analysis

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/Table/using_Table.ipynb

Starting with the process.

(continues on next page)

(continued from previous page)

```

Basic input data:
ScriptType is: vrf_mm
Supply Air Temperature Input Method is: temperature difference
Output type is: custom
Output frequencies are:
['hourly', 'runperiod']
EnergyPlus version is: 23.1
Temperature Control method is: temperature

=====START OF GENERIC IDF FILE GENERATION
PROCESS=====

Starting with file:
TestModel
IDD location is: C:\EnergyPlusV23-1-0\Energy+.idd
The occupied zones in the model TestModel are:
BLOCK1:ZONE2
BLOCK1:ZONE1
The windows and doors in the model TestModel are:
Block1_Zone2_Wall_3_0_0_0_0_0_Win
Block1_Zone2_Wall_4_0_0_0_0_0_Win
Block1_Zone2_Wall_5_0_0_0_0_0_Win
Block1_Zone1_Wall_2_0_0_0_0_0_Win
Block1_Zone1_Wall_3_0_0_0_0_0_Win
Block1_Zone1_Wall_5_0_0_0_0_0_Win
The zones in the model TestModel are:
BLOCK1_ZONE2
BLOCK1_ZONE1
The people objects in the model have been amended.
BLOCK1:ZONE2 Thermostat has been added
BLOCK1:ZONE1 Thermostat has been added
On Schedule already was in the model
TypOperativeTempControlSch Schedule already was in the model
All ZoneHVAC:IdealLoadsAirSystem Heating and Cooling availability schedules has been set
to on
On 24/7 Schedule already was in the model
Control type schedule: Always 4 Schedule has been added
Relative humidity setpoint schedule: Always 50.00 Schedule has been added
Heating Fanger comfort setpoint: Always -0.5 Schedule has been added
Cooling Fanger comfort setpoint: Always 0.1 Schedule has been added
Zone CO2 setpoint: Always 900ppm Schedule has been added
Min CO2 concentration: Always 600ppm Schedule has been added
Generic contaminant setpoint: Always 0.5ppm Schedule has been added
Air distribution effectiveness (always 1) Schedule has been added
VRF Heating Cooling (Northern Hemisphere) Schedule has been added
DefaultFanEffRatioCurve Curve:Cubic Object has been added
VRFTUCoolCapFT Curve:Cubic Object has been added
VRFTUHeatCapFT Curve:Cubic Object has been added
VRFCoolCapFTBoundary Curve:Cubic Object has been added
VRFCoolEIRFTBoundary Curve:Cubic Object has been added
CoolingEIRLowPLR Curve:Cubic Object has been added
VRFHeatCapFTBoundary Curve:Cubic Object has been added

```

(continues on next page)

(continued from previous page)

VRFHeatEIRFTBoundary Curve:Cubic Object has been added
HeatingEIRLowPLR Curve:Cubic Object has been added
DefaultFanPowerRatioCurve Curve:Exponent Object has been added
DXHtgCoilDefrostEIRFT Curve:Biquadratic Object has been added
VRCoolCapFT Curve:Biquadratic Object has been added
VRCoolCapFTHi Curve:Biquadratic Object has been added
VRCoolEIRFT Curve:Biquadratic Object has been added
VRCoolEIRFTHi Curve:Biquadratic Object has been added
VRFHeatCapFT Curve:Biquadratic Object has been added
VRFHeatCapFTHi Curve:Biquadratic Object has been added
VRFHeatEIRFT Curve:Biquadratic Object has been added
VRFHeatEIRFTHi Curve:Biquadratic Object has been added
CoolingLengthCorrectionFactor Curve:Biquadratic Object has been added
VRF Piping Correction Factor for Length in Heating Mode Curve:Biquadratic Object has
been added
VRF Heat Recovery Cooling Capacity Modifier Curve:Biquadratic Object has been added
VRF Heat Recovery Cooling Energy Modifier Curve:Biquadratic Object has been added
VRF Heat Recovery Heating Capacity Modifier Curve:Biquadratic Object has been added
VRF Heat Recovery Heating Energy Modifier Curve:Biquadratic Object has been added
VRFACCCoolCapFFF Curve:Quadratic Object has been added
CoolingEIRHiPLR Curve:Quadratic Object has been added
VRFCPLFFPLR Curve:Quadratic Object has been added
HeatingEIRHiPLR Curve:Quadratic Object has been added
CoolingCombRatio Curve:Linear Object has been added
HeatingCombRatio Curve:Linear Object has been added
VRF Outdoor Unit_BLOCK1:ZONE2 AirConditioner:VariableRefrigerantFlow Object has been
added
VRF Outdoor Unit_BLOCK1:ZONE1 AirConditioner:VariableRefrigerantFlow Object has been
added
VRF Outdoor Unit_BLOCK1:ZONE2 Outdoor Air Node Object has been added
VRF Outdoor Unit_BLOCK1:ZONE2 Zone List Object has been added
VRF Outdoor Unit_BLOCK1:ZONE1 Outdoor Air Node Object has been added
VRF Outdoor Unit_BLOCK1:ZONE1 Zone List Object has been added
BLOCK1:ZONE2 Sizing:Zone Object has been added
BLOCK1:ZONE1 Sizing:Zone Object has been added
BLOCK1:ZONE2 Design Specification Outdoor Air Object has been added
BLOCK1:ZONE1 Design Specification Outdoor Air Object has been added
BLOCK1:ZONE2 Design Specification Zone Air Distribution Object has been added
BLOCK1:ZONE1 Design Specification Zone Air Distribution Object has been added
BLOCK1:ZONE2 Nodelist Objects has been added
BLOCK1:ZONE1 Nodelist Objects has been added
BLOCK1:ZONE2 ZoneHVAC:EquipmentConnections Objects has been added
BLOCK1:ZONE1 ZoneHVAC:EquipmentConnections Objects has been added
BLOCK1:ZONE2 ZoneHVAC:EquipmentList Objects has been added
BLOCK1:ZONE1 ZoneHVAC:EquipmentList Objects has been added
BLOCK1:ZONE2 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE1 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE2 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE1 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE2 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE1 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
BLOCK1:ZONE2 Fan:ConstantVolume Object has been added

(continues on next page)

(continued from previous page)

BLOCK1:ZONE1 Fan:ConstantVolume Object has been added
 Vent_SP_temp Schedule has been added
 AHST_Sch_BLOCK1_ZONE2 Schedule has been added
 ACST_Sch_BLOCK1_ZONE2 Schedule has been added
 AHST_Sch_BLOCK1_ZONE1 Schedule has been added
 ACST_Sch_BLOCK1_ZONE1 Schedule has been added
 Added - SetComfTemp Program
 Added - CountHours_BLOCK1_ZONE2 Program
 Added - CountHours_BLOCK1_ZONE1 Program
 Added - SetAppLimits Program
 Added - ApplyCAT Program
 Added - SetAST Program
 Added - SetASTnoTol Program
 Added - CountHoursNoApp_BLOCK1_ZONE2 Program
 Added - SetGeoVarBLOCK1_ZONE2 Program
 Added - CountHoursNoApp_BLOCK1_ZONE1 Program
 Added - SetGeoVarBLOCK1_ZONE1 Program
 Added - SetInputData Program
 Added - SetVOFinputData Program
 Added - SetVST Program
 Added - ApplyAST_BLOCK1_ZONE2 Program
 Added - ApplyAST_BLOCK1_ZONE1 Program
 Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone2_Wall1_3_0_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone2_Wall1_4_0_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone2_Wall1_5_0_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone1_Wall1_2_0_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone1_Wall1_3_0_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone1_Wall1_5_0_0_0_0_0_Win Program
 Added - Comfort Temperature Output Variable
 Added - Adaptive Cooling Setpoint Temperature Output Variable
 Added - Adaptive Heating Setpoint Temperature Output Variable
 Added - Adaptive Cooling Setpoint Temperature_No Tolerance Output Variable
 Added - Adaptive Heating Setpoint Temperature_No Tolerance Output Variable
 Added - Ventilation Setpoint Temperature Output Variable
 Added - Minimum Outdoor Temperature for ventilation Output Variable
 Added - Minimum Outdoor Temperature Difference for ventilation Output Variable
 Added - Maximum Outdoor Temperature Difference for ventilation Output Variable
 Added - Multiplier for Ventilation Opening Factor Output Variable
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Output Variable
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Output Variable
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Output Variable
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Output Variable

(continues on next page)

(continued from previous page)

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE2 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE1 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE2 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE1 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE2 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE1 Output Variable
 Global variables objects have been added
 Internal variables objects have been added
 Added - RMOT Sensor
 Added - PMOT Sensor
 Added - BLOCK1_ZONE2_OpT Sensor
 Added - BLOCK1_ZONE2_WindSpeed Sensor
 Added - BLOCK1_ZONE2_OutT Sensor
 Added - BLOCK1_ZONE1_OpT Sensor
 Added - BLOCK1_ZONE1_WindSpeed Sensor
 Added - BLOCK1_ZONE1_OutT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - OutT Sensor
 Added - AHST_Act_BLOCK1_ZONE2 Actuator
 Added - ACST_Act_BLOCK1_ZONE2 Actuator
 Added - AHST_Act_BLOCK1_ZONE1 Actuator
 Added - ACST_Act_BLOCK1_ZONE1 Actuator
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - BLOCK1_ZONE2_CoolCoil Sensor
 Added - BLOCK1_ZONE2_HeatCoil Sensor
 Added - BLOCK1_ZONE1_CoolCoil Sensor

(continues on next page)

(continued from previous page)

Added - BLOCK1_ZONE1_HeatCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
 Added - SetComfTemp Program Calling Manager
 Added - CountHours_BLOCK1_ZONE2 Program Calling Manager
 Added - CountHours_BLOCK1_ZONE1 Program Calling Manager
 Added - SetAppLimits Program Calling Manager
 Added - ApplyCAT Program Calling Manager
 Added - SetAST Program Calling Manager
 Added - SetASTnoTol Program Calling Manager
 Added - CountHoursNoApp_BLOCK1_ZONE2 Program Calling Manager
 Added - SetGeoVarBLOCK1_ZONE2 Program Calling Manager
 Added - CountHoursNoApp_BLOCK1_ZONE1 Program Calling Manager
 Added - SetGeoVarBLOCK1_ZONE1 Program Calling Manager
 Added - SetInputData Program Calling Manager
 Added - SetVOFinputData Program Calling Manager
 Added - SetVST Program Calling Manager
 Added - ApplyAST_BLOCK1_ZONE2 Program Calling Manager
 Added - ApplyAST_BLOCK1_ZONE1 Program Calling Manager
 Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
 Added - Comfort Temperature Reporting FrequencyHourly Output:Variable data
 Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyHourly Output:Variable[↳ data](#)
 Added - Adaptive Heating Setpoint Temperature Reporting FrequencyHourly Output:Variable[↳ data](#)
 Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyHourly[↳ Output:Variable data](#)
 Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyHourly[↳ Output:Variable data](#)
 Added - Ventilation Setpoint Temperature Reporting FrequencyHourly Output:Variable data
 Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyHourly Output:

(continues on next page)

(continued from previous page)

↳ Variable data
 Added - Minimum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Maximum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Multiplier for Ventilation Opening Factor Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
 Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
 Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
 Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
 Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
 Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
 Added - Zone Operative Temperature Reporting FrequencyHourly Output:Variable data
 Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyHourly Output:Variable data
 Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyHourly Output:Variable data
 Added - Cooling Coil Total Cooling Rate Reporting FrequencyHourly Output:Variable data
 Added - Heating Coil Heating Rate Reporting FrequencyHourly Output:Variable data
 Added - Facility Total HVAC Electric Demand Power Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyHourly Output:
 ↳ Variable data
 Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyHourly
 ↳ Output:Variable data
 Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳ data

(continues on next page)

(continued from previous page)

Added - AFN Zone Infiltration Volume Reporting FrequencyHourly Output:Variable data
 Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳data
 Added - AFN Zone Ventilation Volume Reporting FrequencyHourly Output:Variable data
 Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyHourly Output:Variable
 ↳data
 Added - Site Wind Speed Reporting FrequencyHourly Output:Variable data
 Added - Site Outdoor Air Relative Humidity Reporting FrequencyHourly Output:Variable data
 Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
 Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
 Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
 Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
 Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyHourly Output:
 ↳Variable data
 Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyHourly Output:
 ↳Variable data
 Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
 ↳Variable data
 Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
 ↳Variable data
 Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
 ↳Variable data
 Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
 ↳Variable data
 Added - Comfort Temperature Reporting FrequencyRunperiod Output:Variable data
 Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyRunperiod Output:
 ↳Variable data
 Added - Adaptive Heating Setpoint Temperature Reporting FrequencyRunperiod Output:
 ↳Variable data
 Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod
 ↳Output:Variable data
 Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod
 ↳Output:Variable data
 Added - Ventilation Setpoint Temperature Reporting FrequencyRunperiod Output:Variable
 ↳data
 Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyRunperiod Output:
 ↳Variable data
 Added - Minimum Outdoor Temperature Difference for ventilation Reporting
 ↳FrequencyRunperiod Output:Variable data
 Added - Maximum Outdoor Temperature Difference for ventilation Reporting
 ↳FrequencyRunperiod Output:Variable data
 Added - Multiplier for Ventilation Opening Factor Reporting FrequencyRunperiod Output:
 ↳Variable data
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod
 ↳Output:Variable data
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod
 ↳Output:Variable data
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:
 ↳Variable data
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:
 ↳Variable data
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod

(continues on next page)

(continued from previous page)

```

↳Output:Variable data
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
↳Output:Variable data
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod
↳Output:Variable data
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
↳Output:Variable data
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting
↳FrequencyRunperiod Output:Variable data
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting
↳FrequencyRunperiod Output:Variable data
Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting
↳FrequencyRunperiod Output:Variable data
Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting
↳FrequencyRunperiod Output:Variable data
Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - Zone Operative Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
↳Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
↳Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Cooling Coil Total Cooling Rate Reporting FrequencyRunperiod Output:Variable data
Added - Heating Coil Heating Rate Reporting FrequencyRunperiod Output:Variable data
Added - Facility Total HVAC Electric Demand Power Reporting FrequencyRunperiod Output:
↳Variable data
Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyRunperiod Output:
↳Variable data
Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyRunperiod
↳Output:Variable data
Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyRunperiod Output:
↳Variable data
Added - AFN Zone Infiltration Volume Reporting FrequencyRunperiod Output:Variable data
Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyRunperiod Output:
↳Variable data
Added - AFN Zone Ventilation Volume Reporting FrequencyRunperiod Output:Variable data
Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyRunperiod Output:
↳Variable data
Added - Site Wind Speed Reporting FrequencyRunperiod Output:Variable data
Added - Site Outdoor Air Relative Humidity Reporting FrequencyRunperiod Output:Variable
↳data
Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data
Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data
Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyRunperiod Output:
↳Variable data
Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyRunperiod Output:

```

(continues on next page)

(continued from previous page)

↳ Variable data
 Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
 ↳ Variable data
 Not added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyHourly
 ↳ Output:Variable data
 Not added - AFN Surface Venting Window or Door Opening Factor Reporting
 ↳ FrequencyRunperiod Output:Variable data

The current existing outputs for Hourly Frequency are:

Comfort Temperature
 Adaptive Cooling Setpoint Temperature
 Adaptive Heating Setpoint Temperature
 Adaptive Cooling Setpoint Temperature_No Tolerance
 Adaptive Heating Setpoint Temperature_No Tolerance
 Ventilation Setpoint Temperature
 Minimum Outdoor Temperature for ventilation
 Minimum Outdoor Temperature Difference for ventilation
 Maximum Outdoor Temperature Difference for ventilation
 Multiplier for Ventilation Opening Factor
 Comfortable Hours_No Applicability_BLOCK1_ZONE2
 Comfortable Hours_No Applicability_BLOCK1_ZONE1
 Comfortable Hours_Applicability_BLOCK1_ZONE2
 Comfortable Hours_Applicability_BLOCK1_ZONE1
 Discomfortable Applicable Hot Hours_BLOCK1_ZONE2
 Discomfortable Applicable Hot Hours_BLOCK1_ZONE1
 Discomfortable Applicable Cold Hours_BLOCK1_ZONE2
 Discomfortable Applicable Cold Hours_BLOCK1_ZONE1
 Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2
 Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1
 Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2
 Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1
 Zone Floor Area_BLOCK1_ZONE2
 Zone Floor Area_BLOCK1_ZONE1
 Zone Air Volume_BLOCK1_ZONE2
 Zone Air Volume_BLOCK1_ZONE1
 Ventilation Hours_BLOCK1_ZONE2
 Ventilation Hours_BLOCK1_ZONE1
 Zone Operative Temperature
 Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air Temperature
 Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air Temperature
 Cooling Coil Total Cooling Rate
 Heating Coil Heating Rate
 Facility Total HVAC Electric Demand Power
 Facility Total HVAC Electricity Demand Rate
 AFN Surface Venting Window or Door Opening Factor
 AFN Zone Infiltration Air Change Rate

(continues on next page)

(continued from previous page)

AFN Zone Infiltration Volume
 AFN Zone Ventilation Air Change Rate
 AFN Zone Ventilation Volume
 Site Outdoor Air Drybulb Temperature
 Site Wind Speed
 Site Outdoor Air Relative Humidity
 Schedule Value
 VRF Heat Pump Cooling Electricity Energy
 VRF Heat Pump Heating Electricity Energy
 Do you want to remove some input or keep it and remove all others? Please enter remove
 ↵ or keep:keep
 Please enter these outputs (which must be contained in the list above) separated by
 ↵ semicolon (;):
 @media print {
 .ms-editor-squiggles-container {
 display:none !important;
 }
 }
 .ms-editor-squiggles-container {
 all: initial;
 }Zone Operative Temperature;Adaptive Cooling Setpoint Temperature_No Tolerance;
 ↵Adaptive Heating Setpoint Temperature_No Tolerance

The current existing outputs for Runperiod Frequency are:

Comfort Temperature
 Adaptive Cooling Setpoint Temperature
 Adaptive Heating Setpoint Temperature
 Adaptive Cooling Setpoint Temperature_No Tolerance
 Adaptive Heating Setpoint Temperature_No Tolerance
 Ventilation Setpoint Temperature
 Minimum Outdoor Temperature for ventilation
 Minimum Outdoor Temperature Difference for ventilation
 Maximum Outdoor Temperature Difference for ventilation
 Multiplier for Ventilation Opening Factor
 Comfortable Hours_No Applicability_BLOCK1_ZONE2
 Comfortable Hours_No Applicability_BLOCK1_ZONE1
 Comfortable Hours_Applicability_BLOCK1_ZONE2
 Comfortable Hours_Applicability_BLOCK1_ZONE1
 Discomfortable Applicable Hot Hours_BLOCK1_ZONE2
 Discomfortable Applicable Hot Hours_BLOCK1_ZONE1
 Discomfortable Applicable Cold Hours_BLOCK1_ZONE2
 Discomfortable Applicable Cold Hours_BLOCK1_ZONE1
 Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2
 Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1
 Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2
 Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1
 Zone Floor Area_BLOCK1_ZONE2
 Zone Floor Area_BLOCK1_ZONE1
 Zone Air Volume_BLOCK1_ZONE2
 Zone Air Volume_BLOCK1_ZONE1
 Ventilation Hours_BLOCK1_ZONE2
 Ventilation Hours_BLOCK1_ZONE1

(continues on next page)

(continued from previous page)

Zone Operative Temperature
 Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air Temperature
 Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air Temperature
 Cooling Coil Total Cooling Rate
 Heating Coil Heating Rate
 Facility Total HVAC Electric Demand Power
 Facility Total HVAC Electricity Demand Rate
 AFN Surface Venting Window or Door Opening Factor
 AFN Zone Infiltration Air Change Rate
 AFN Zone Infiltration Volume
 AFN Zone Ventilation Air Change Rate
 AFN Zone Ventilation Volume
 Site Outdoor Air Drybulb Temperature
 Site Wind Speed
 Site Outdoor Air Relative Humidity
 Schedule Value
 VRF Heat Pump Cooling Electricity Energy
 VRF Heat Pump Heating Electricity Energy
 Do you want to remove some input or keep it and remove all others? Please enter remove
 ↵ or keep:keep
 Please enter these outputs (which must be contained in the list above) separated by
 ↵ semicolon (;):
 @media print {
 .ms-editor-squiggles-container {
 display:none !important;
 }
 }
 .ms-editor-squiggles-container {
 all: initial;
 }Zone Operative Temperature;Adaptive Cooling Setpoint Temperature_No Tolerance;
 ↵Adaptive Heating Setpoint Temperature_No Tolerance
 IDF has been saved
 Ending with file:
 TestModel

=====END OF GENERIC IDF FILE GENERATION PROCESS=====

The following IDFs will not work, and therefore these will be deleted:
 None

=====START OF OUTPUT IDF FILES GENERATION
 ↵PROCESS=====

The list of output IDFs is going to be:

```

TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
  
```

(continues on next page)

(continued from previous page)

```

TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf

```

```
=====END OF OUTPUT IDF FILES GENERATION PROCESS=====
```

Now, let's inspect the Output:Variable objects in both the input idf and any of the output idfs, for instance, the first one.

```
[36]: input_testmodel = add_accis_instance.input_idfs['TestModel']
input_testmodel.idfobjects['Output:Variable']
```

```
[36]: []
```

We can see there is no Output:Variable object in the input idf. Now let's see some output idf.

```
[39]: output_testmodel = add_accis_instance.output_idfs[
    'TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf'
]
output_testmodel.idfobjects['Output:Variable']
```

```
[39]: [
    OUTPUT:VARIABLE,
        *,                               !- Key Value
        Adaptive Cooling Setpoint Temperature_No Tolerance,      !- Variable Name
        Hourly,                           !- Reporting Frequency
        ;                                !- Schedule Name
    ,
    OUTPUT:VARIABLE,
        *,                               !- Key Value
        Adaptive Heating Setpoint Temperature_No Tolerance,      !- Variable Name
        Hourly,                           !- Reporting Frequency
        ;                                !- Schedule Name
    ,
    OUTPUT:VARIABLE,
        *,                               !- Key Value
        Zone Operative Temperature,      !- Variable Name
        Hourly,                           !- Reporting Frequency
        ;                                !- Schedule Name
    ,
    OUTPUT:VARIABLE,
        * ,                             !- Key Value
```

(continues on next page)

(continued from previous page)

```

Adaptive Cooling Setpoint Temperature_No Tolerance,      !- Variable Name
Runperiod,                               !- Reporting Frequency
;
;                                     !- Schedule Name

,
OUTPUT:VARIABLE,
*,                                !- Key Value
Adaptive Heating Setpoint Temperature_No Tolerance,    !- Variable Name
Runperiod,                               !- Reporting Frequency
;
;                                     !- Schedule Name

,
OUTPUT:VARIABLE,
*,                                !- Key Value
Zone Operative Temperature,        !- Variable Name
Runperiod,                               !- Reporting Frequency
;
;                                     !- Schedule Name
]

```

We can see in this case 6 Output:Variable objects, which are the ones we previously requested: - Zone Operative Temperature - Adaptive Cooling Setpoint Temperature_No Tolerance - Adaptive Heating Setpoint Temperature_No Tolerance

in both Hourly and Runperiod Reporting Frequencies.

As usual, let's remove all the output idfs, so that we can re-run the notebook.

```
[40]: for i in os.listdir():
    if i.endswith('.idf') and input_idf[0] not in i:
        os.remove(i)
```

8.2.2.2 Using Output_gen_dataframe and Output_take_dataframe

Similar to section 2.2.1, imagine you need to perform a large number of simulations. Therefore, using Output_type='standard' would result in a heavy dataset. Using Output_type='simplified' is not an option, since you need more outputs than just the operative temperature, heating coil heating rate and cooling coil total cooling rate. In that case, you can use Output_gen_dataframe and Output_take_dataframe.

First of all, let's see the idfs we do have in this path:

```
[43]: from os import listdir
input_idfs = [i for i in listdir() if i.endswith('.idf')]
input_idfs
[43]: ['TestModel.idf', 'TestModel_pymod.idf']
```

Now, we can go for it. Let's apply the adaptive setpoint temperatures. We want to see what outputs we would have if we run this script, therefore, we need to specify Output_gen_dataframe=True. Since we don't want to generate the output idfs yet, in this case let's specify confirmGen=False.

```
[45]: from accim.sim import accis
add_accis_instance = accis.addAccis(
    ScriptType='vrf_mm',
    SupplyAirTempInputMethod='temperature difference',
    Output_keep_existing=False,
```

(continues on next page)

(continued from previous page)

```

        Output_type='standard',
        Output_freqs=['hourly', 'runperiod'],
        Output_gen_dataframe=True,
        EnergyPlus_version='23.1',
        TempCtrl='temperature',
        ComfStand=[1, 14],
        CAT=[1, 2, 3, 80, 90],
        ComfMod=[0, 0.1, 0.2, 0.3, 0.4, 0.5, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 2, 3],
        SetpointAcc=100,
        CoolSeasonStart='01/11',
        CoolSeasonEnd='01/03',
        HVACmode=[2],
        VentCtrl=[2],
        MaxTempDiffVOF=20,
        MinTempDiffVOF=1,
        MultiplierVOF=0.2,
        VSToffset=[0],
        MinOToffset=[50],
        MaxWindSpeed=[50],
        ASTtol_steps=0.1,
        ASTtol_start=0.1,
        ASTtol_end_input=0.1,
        confirmGen=False
)

```

----- Adaptive-Comfort-Control-Implemented Model (ACCIM) -----

This tool allows to apply adaptive setpoint temperatures.

For further information, please read the documentation:

<https://accim.readthedocs.io/en/master/>

For a visual understanding of the tool, please visit the following jupyter notebooks:

- Using addAccis() to apply adaptive setpoint temperatures

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/addAccis/using_addAccis.ipynb

- Using rename_epw_files() to rename the EPWs for proper data analysis after simulation

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/rename_epw_files/using_rename_epw_files.ipynb

- Using runEp() to directly run simulations with EnergyPlus

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/runEp/using_runEp.ipynb

- Using the class Table() for data analysis

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/Table/using_Table.ipynb

Starting with the process.

Basic input data:

ScriptType is: vrf_mm

Supply Air Temperature Input Method is: temperature difference

(continues on next page)

(continued from previous page)

```
Output type is: standard
Output frequencies are:
['hourly', 'runperiod']
EnergyPlus version is: 23.1
Temperature Control method is: temperature

=====START OF GENERIC IDF FILE GENERATION
PROCESS=====

Starting with file:
TestModel
IDD location is: C:\EnergyPlusV23-1-0\Energy+.idd
The occupied zones in the model TestModel are:
BLOCK1:ZONE2
BLOCK1:ZONE1
The windows and doors in the model TestModel are:
Block1_Zone2_Wall_3_0_0_0_0_0_Win
Block1_Zone2_Wall_4_0_0_0_0_0_Win
Block1_Zone2_Wall_5_0_0_0_0_0_Win
Block1_Zone1_Wall_2_0_0_0_0_0_Win
Block1_Zone1_Wall_3_0_0_0_0_0_Win
Block1_Zone1_Wall_5_0_0_0_0_0_Win
The zones in the model TestModel are:
BLOCK1_ZONE2
BLOCK1_ZONE1
The people objects in the model have been amended.
BLOCK1:ZONE2 Thermostat has been added
BLOCK1:ZONE1 Thermostat has been added
On Schedule already was in the model
TypOperativeTempControlSch Schedule already was in the model
All ZoneHVAC:IdealLoadsAirSystem Heating and Cooling availability schedules has been set
to on
On 24/7 Schedule already was in the model
Control type schedule: Always 4 Schedule has been added
Relative humidity setpoint schedule: Always 50.00 Schedule has been added
Heating Fanger comfort setpoint: Always -0.5 Schedule has been added
Cooling Fanger comfort setpoint: Always 0.1 Schedule has been added
Zone CO2 setpoint: Always 900ppm Schedule has been added
Min CO2 concentration: Always 600ppm Schedule has been added
Generic contaminant setpoint: Always 0.5ppm Schedule has been added
Air distribution effectiveness (always 1) Schedule has been added
VRF Heating Cooling (Northern Hemisphere) Schedule has been added
DefaultFanEffRatioCurve Curve:Cubic Object has been added
VRFTUCoolCapFT Curve:Cubic Object has been added
VRFTUHeatCapFT Curve:Cubic Object has been added
VRCoolCapFTBoundary Curve:Cubic Object has been added
VRFCoolEIRFTBoundary Curve:Cubic Object has been added
CoolingEIRLowPLR Curve:Cubic Object has been added
VRFHeatCapFTBoundary Curve:Cubic Object has been added
VRFHeatEIRFTBoundary Curve:Cubic Object has been added
HeatingEIRLowPLR Curve:Cubic Object has been added
DefaultFanPowerRatioCurve Curve:Exponent Object has been added
```

(continues on next page)

(continued from previous page)

DXHtgCoilDefrostEIRFT Curve:Biquadratic Object has been added
 VRFCoolCapFT Curve:Biquadratic Object has been added
 VRFCoolCapFTHi Curve:Biquadratic Object has been added
 VRFCoolEIRFT Curve:Biquadratic Object has been added
 VRFCoolEIRFTHi Curve:Biquadratic Object has been added
 VRFHeatCapFT Curve:Biquadratic Object has been added
 VRFHeatCapFTHi Curve:Biquadratic Object has been added
 VRFHeatEIRFT Curve:Biquadratic Object has been added
 VRFHeatEIRFTHi Curve:Biquadratic Object has been added
 CoolingLengthCorrectionFactor Curve:Biquadratic Object has been added
 VRF Piping Correction Factor for Length in Heating Mode Curve:Biquadratic Object has
 been added
 VRF Heat Recovery Cooling Capacity Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Cooling Energy Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Heating Capacity Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Heating Energy Modifier Curve:Biquadratic Object has been added
 VRFACCoolCapFFF Curve:Quadratic Object has been added
 CoolingEIRHiPLR Curve:Quadratic Object has been added
 VRFCPLFFPLR Curve:Quadratic Object has been added
 HeatingEIRHiPLR Curve:Quadratic Object has been added
 CoolingCombRatio Curve:Linear Object has been added
 HeatingCombRatio Curve:Linear Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 AirConditioner:VariableRefrigerantFlow Object has been
 added
 VRF Outdoor Unit_BLOCK1:ZONE1 AirConditioner:VariableRefrigerantFlow Object has been
 added
 VRF Outdoor Unit_BLOCK1:ZONE2 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 Zone List Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Zone List Object has been added
 BLOCK1:ZONE2 Sizing:Zone Object has been added
 BLOCK1:ZONE1 Sizing:Zone Object has been added
 BLOCK1:ZONE2 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE1 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE2 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE1 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE2 Nodelist Objects has been added
 BLOCK1:ZONE1 Nodelist Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE1 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentList Objects has been added
 BLOCK1:ZONE1 ZoneHVAC:EquipmentList Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Fan:ConstantVolume Object has been added
 BLOCK1:ZONE1 Fan:ConstantVolume Object has been added
 Vent_SP_temp Schedule has been added
 AHST_Sch_BLOCK1_ZONE2 Schedule has been added

(continues on next page)

(continued from previous page)

ACST_Sch_BLOCK1_ZONE2 Schedule has been added
AHST_Sch_BLOCK1_ZONE1 Schedule has been added
ACST_Sch_BLOCK1_ZONE1 Schedule has been added
Added - SetComfTemp Program
Added - CountHours_BLOCK1_ZONE2 Program
Added - CountHours_BLOCK1_ZONE1 Program
Added - SetAppLimits Program
Added - ApplyCAT Program
Added - SetAST Program
Added - SetASTnoTol Program
Added - CountHoursNoApp_BLOCK1_ZONE2 Program
Added - SetGeoVarBLOCK1_ZONE2 Program
Added - CountHoursNoApp_BLOCK1_ZONE1 Program
Added - SetGeoVarBLOCK1_ZONE1 Program
Added - SetInputData Program
Added - SetVOFinputData Program
Added - SetVST Program
Added - ApplyAST_BLOCK1_ZONE2 Program
Added - ApplyAST_BLOCK1_ZONE1 Program
Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program
Added - Comfort Temperature Output Variable
Added - Adaptive Cooling Setpoint Temperature Output Variable
Added - Adaptive Heating Setpoint Temperature Output Variable
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Output Variable
Added - Adaptive Heating Setpoint Temperature_No Tolerance Output Variable
Added - Ventilation Setpoint Temperature Output Variable
Added - Minimum Outdoor Temperature for ventilation Output Variable
Added - Minimum Outdoor Temperature Difference for ventilation Output Variable
Added - Maximum Outdoor Temperature Difference for ventilation Output Variable
Added - Multiplier for Ventilation Opening Factor Output Variable
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Output Variable
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Output Variable
Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Output Variable
Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Output Variable

(continues on next page)

(continued from previous page)

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE2 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE1 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE2 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE1 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE2 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE1 Output Variable
 Global variables objects have been added
 Internal variables objects have been added
 Added - RMOT Sensor
 Added - PMOT Sensor
 Added - BLOCK1_ZONE2_OpT Sensor
 Added - BLOCK1_ZONE2_WindSpeed Sensor
 Added - BLOCK1_ZONE2_OutT Sensor
 Added - BLOCK1_ZONE1_OpT Sensor
 Added - BLOCK1_ZONE1_WindSpeed Sensor
 Added - BLOCK1_ZONE1_OutT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - OutT Sensor
 Added - AHST_Act_BLOCK1_ZONE2 Actuator
 Added - ACST_Act_BLOCK1_ZONE2 Actuator
 Added - AHST_Act_BLOCK1_ZONE1 Actuator
 Added - ACST_Act_BLOCK1_ZONE1 Actuator
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - BLOCK1_ZONE2_CoolCoil Sensor
 Added - BLOCK1_ZONE2_HeatCoil Sensor
 Added - BLOCK1_ZONE1_CoolCoil Sensor
 Added - BLOCK1_ZONE1_HeatCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor

(continues on next page)

(continued from previous page)

```
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
Added - SetComfTemp Program Calling Manager
Added - CountHours_BLOCK1_ZONE2 Program Calling Manager
Added - CountHours_BLOCK1_ZONE1 Program Calling Manager
Added - SetAppLimits Program Calling Manager
Added - ApplyCAT Program Calling Manager
Added - SetAST Program Calling Manager
Added - SetASTnoTol Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE2 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE2 Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE1 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE1 Program Calling Manager
Added - SetInputData Program Calling Manager
Added - SetVOFinputData Program Calling Manager
Added - SetVST Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE2 Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE1 Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - Comfort Temperature Reporting FrequencyHourly Output:Variable data
Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyHourly Output:Variable↳data
Added - Adaptive Heating Setpoint Temperature Reporting FrequencyHourly Output:Variable↳data
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyHourly↳Output:Variable data
Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyHourly↳Output:Variable data
Added - Ventilation Setpoint Temperature Reporting FrequencyHourly Output:Variable data
Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyHourly Output:↳Variable data
Added - Minimum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly↳Output:Variable data
```

(continues on next page)

(continued from previous page)

Added - Maximum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly
 ↳Output:Variable data

Added - Multiplier for Ventilation Opening Factor Reporting FrequencyHourly Output:
 ↳Variable data

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳Variable data

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyHourly Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyHourly Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyHourly Output:Variable data

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyHourly Output:
 ↳Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyHourly Output:
 ↳Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyHourly
 ↳Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳data

Added - AFN Zone Infiltration Volume Reporting FrequencyHourly Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳data

(continues on next page)

(continued from previous page)

Added - AFN Zone Ventilation Volume Reporting FrequencyHourly Output:Variable data
Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyHourly Output:Variable data
Added - Site Wind Speed Reporting FrequencyHourly Output:Variable data
Added - Site Outdoor Air Relative Humidity Reporting FrequencyHourly Output:Variable data
Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyHourly Output:
↳ Variable data
Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyHourly Output:
↳ Variable data
Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
↳ Variable data
Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
↳ Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
↳ Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
↳ Variable data
Added - Comfort Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Adaptive Heating Setpoint Temperature Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Ventilation Setpoint Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Minimum Outdoor Temperature Difference for ventilation Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Maximum Outdoor Temperature Difference for ventilation Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Multiplier for Ventilation Opening Factor Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:
↳ Variable data
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:
↳ Variable data

(continues on next page)

(continued from previous page)

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod
 ↳Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
 ↳Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting
 ↳FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting
 ↳FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting
 ↳FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting
 ↳FrequencyRunperiod Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyRunperiod Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyRunperiod Output:Variable data

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyRunperiod
 ↳Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - AFN Zone Infiltration Volume Reporting FrequencyRunperiod Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - AFN Zone Ventilation Volume Reporting FrequencyRunperiod Output:Variable data

Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - Site Wind Speed Reporting FrequencyRunperiod Output:Variable data

Added - Site Outdoor Air Relative Humidity Reporting FrequencyRunperiod Output:Variable
 ↳data

Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
 ↳Variable data

(continues on next page)

(continued from previous page)

```

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
  ↵ Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
  ↵ Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
  ↵ Variable data
IDF has been saved
Ending with file:
TestModel

```

=====END OF GENERIC IDF FILE GENERATION PROCESS=====

The following IDFs will not work, and therefore these will be deleted:

None

=====START OF OUTPUT IDF FILES GENERATION=====
 ↵PROCESS=====

The list of output IDFs is going to be:

```

TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf

```

(continues on next page)

(continued from previous page)

```
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
And the total number of output IDF's is going to be 36
IDF generation has been shut down
```

```
=====END OF OUTPUT IDF FILES GENERATION PROCESS=====
```

We can see no output idf has been generated:

```
[46]: [i for i in listdir() if i.endswith('.idf')]
[46]: ['TestModel.idf']
```

However, in this case, there is a variable within `add_accis_instance` named `df_outputs`, which is a pandas DataFrame that contains all Output:Variable objects. Let's take a look:

```
[48]: add_accis_instance.df_outputs
```

	file	key_value	\
0	TestModel	*	
1	TestModel	*	
2	TestModel	*	
3	TestModel	*	
4	TestModel	*	
..	
101	TestModel	*	
102	TestModel	BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil	
103	TestModel	BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil	
104	TestModel	BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil	
105	TestModel	BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil	
		variable_name	reporting_frequency
0		Comfort Temperature	Hourly
1		Adaptive Cooling Setpoint Temperature	Hourly
2		Adaptive Heating Setpoint Temperature	Hourly
3		Adaptive Cooling Setpoint Temperature_No Toler...	Hourly
4		Adaptive Heating Setpoint Temperature_No Toler...	Hourly
..	
101	VRF Heat Pump Heating Electricity Energy		Runperiod
102	Cooling Coil Total Cooling Rate		Runperiod
103	Heating Coil Heating Rate		Runperiod
104	Cooling Coil Total Cooling Rate		Runperiod
105	Heating Coil Heating Rate		Runperiod
	schedule_name		
0			
1			
2			
3			
4		...	

(continues on next page)

(continued from previous page)

```
101
102
103
104
105

[106 rows x 5 columns]
```

Now, let's save it in a new variable, and let's filter it to keep only the Output:Variable objects we need.

```
[51]: df_outputs_in = add_accis_instance.df_outputs
df_outputs_in = df_outputs_in[
    (
        df_outputs_in['variable_name'].str.contains('Zone Operative Temperature')
        |
        df_outputs_in['variable_name'].str.contains('Adaptive Cooling Setpoint_'
        ↪Temperature_No Tolerance')
        |
        df_outputs_in['variable_name'].str.contains('Adaptive Heating Setpoint_'
        ↪Temperature_No Tolerance')
    )
]
df_outputs_in
```

```
[51]:   file key_value                                variable_name \
3   TestModel      * Adaptive Cooling Setpoint Temperature_No Toler...
4   TestModel      * Adaptive Heating Setpoint Temperature_No Toler...
28  TestModel      *                               Zone Operative Temperature
56  TestModel      * Adaptive Cooling Setpoint Temperature_No Toler...
57  TestModel      * Adaptive Heating Setpoint Temperature_No Toler...
81  TestModel      *                               Zone Operative Temperature

   reporting_frequency schedule_name
3           Hourly
4           Hourly
28          Runperiod
56          Runperiod
57          Runperiod
81          Runperiod
```

Now, we can run again `addAccis`, specifying `Output_gen_dataframe=False`, entering the pandas DataFrame instance `df_outputs_in` we just filtered in the argument `Output_take_dataframe`, and specifying `confirmGen=True`, since now we're ready to apply the adaptive setpoints.

```
[52]: from accim.sim import accis
add_accis_instance = accis.addAccis(
    ScriptType='vrf_mm',
    SupplyAirTempInputMethod='temperature difference',
    Output_keep_existing=False,
    Output_type='standard',
    Output_freqs=['hourly', 'runperiod'],
    Output_gen_dataframe=False,
```

(continues on next page)

(continued from previous page)

```

Output_take_dataframe=df_outputs_in,
EnergyPlus_version='23.1',
TempCtrl='temperature',
ComfStand=[1, 14],
CAT=[1, 2, 3, 80, 90],
ComfMod=[0, 0.1, 0.2, 0.3, 0.4, 0.5, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 2, 3],
SetpointAcc=100,
CoolSeasonStart='01/11',
CoolSeasonEnd='01/03',
HVACmode=[2],
VentCtrl=[2],
MaxTempDiffVOF=20,
MinTempDiffVOF=1,
MultiplierVOF=0.2,
VSToffset=[0],
MinOToffset=[50],
MaxWindSpeed=[50],
ASTtol_steps=0.1,
ASTtol_start=0.1,
ASTtol_end_input=0.1,
confirmGen=True
)

```

Adaptive-Comfort-Control-Implemented Model (ACCIM)

This tool allows to apply adaptive setpoint temperatures.

For further information, please read the documentation:

<https://accim.readthedocs.io/en/master/>

For a visual understanding of the tool, please visit the following jupyter notebooks:

- Using addAccis() to apply adaptive setpoint temperatures

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/addAccis/using_addAccis.ipynb

- Using rename_epw_files() to rename the EPWs for proper data analysis after simulation

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/rename_epw_files/using_rename_epw_files.ipynb

- Using runEp() to directly run simulations with EnergyPlus

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/runEp/using_runEp.ipynb

- Using the class Table() for data analysis

https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/Table/using_Table.ipynb

Starting with the process.

Basic input data:

ScriptType is: vrf_mm

Supply Air Temperature Input Method is: temperature difference

Output type is: standard

Output frequencies are:

(continues on next page)

(continued from previous page)

```
['hourly', 'runperiod']
EnergyPlus version is: 23.1
Temperature Control method is: temperature

=====START OF GENERIC IDF FILE GENERATION
PROCESS=====

Starting with file:
TestModel
IDD location is: C:\EnergyPlusV23-1-0\Energy+.idd
The occupied zones in the model TestModel are:
BLOCK1:ZONE2
BLOCK1:ZONE1
The windows and doors in the model TestModel are:
Block1_Zone2_Wall_3_0_0_0_0_0_Win
Block1_Zone2_Wall_4_0_0_0_0_0_Win
Block1_Zone2_Wall_5_0_0_0_0_0_Win
Block1_Zone1_Wall_2_0_0_0_0_0_Win
Block1_Zone1_Wall_3_0_0_0_0_0_Win
Block1_Zone1_Wall_5_0_0_0_0_0_Win
The zones in the model TestModel are:
BLOCK1_ZONE2
BLOCK1_ZONE1
The people objects in the model have been amended.
BLOCK1:ZONE2 Thermostat has been added
BLOCK1:ZONE1 Thermostat has been added
On Schedule already was in the model
TypOperativeTempControlSch Schedule already was in the model
All ZoneHVAC:IdealLoadsAirSystem Heating and Cooling availability schedules has been set
to on
On 24/7 Schedule already was in the model
Control type schedule: Always 4 Schedule has been added
Relative humidity setpoint schedule: Always 50.00 Schedule has been added
Heating Fanger comfort setpoint: Always -0.5 Schedule has been added
Cooling Fanger comfort setpoint: Always 0.1 Schedule has been added
Zone CO2 setpoint: Always 900ppm Schedule has been added
Min CO2 concentration: Always 600ppm Schedule has been added
Generic contaminant setpoint: Always 0.5ppm Schedule has been added
Air distribution effectiveness (always 1) Schedule has been added
VRF Heating Cooling (Northern Hemisphere) Schedule has been added
DefaultFanEffRatioCurve Curve:Cubic Object has been added
VRFTUCoolCapFT Curve:Cubic Object has been added
VRFTUHeatCapFT Curve:Cubic Object has been added
VRCoolCapFTBoundary Curve:Cubic Object has been added
VRCoolEIRFTBoundary Curve:Cubic Object has been added
CoolingEIRLowPLR Curve:Cubic Object has been added
VRFHeatCapFTBoundary Curve:Cubic Object has been added
VRFHeatEIRFTBoundary Curve:Cubic Object has been added
HeatingEIRLowPLR Curve:Cubic Object has been added
DefaultFanPowerRatioCurve Curve:Exponent Object has been added
DXHtgCoilDefrostEIRFT Curve:Biquadratic Object has been added
VRCoolCapFT Curve:Biquadratic Object has been added
```

(continues on next page)

(continued from previous page)

VRFCoolCapFTHi Curve:Biquadratic Object has been added
 VRFCoolEIRFT Curve:Biquadratic Object has been added
 VRFCoolEIRFTHi Curve:Biquadratic Object has been added
 VRFHeatCapFT Curve:Biquadratic Object has been added
 VRFHeatCapFTHi Curve:Biquadratic Object has been added
 VRFHeatEIRFT Curve:Biquadratic Object has been added
 VRFHeatEIRFTHi Curve:Biquadratic Object has been added
 CoolingLengthCorrectionFactor Curve:Biquadratic Object has been added
 VRF Piping Correction Factor for Length in Heating Mode Curve:Biquadratic Object has been added
 VRF Heat Recovery Cooling Capacity Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Cooling Energy Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Heating Capacity Modifier Curve:Biquadratic Object has been added
 VRF Heat Recovery Heating Energy Modifier Curve:Biquadratic Object has been added
 VRFACCoolCapFFF Curve:Quadratic Object has been added
 CoolingEIRHiPLR Curve:Quadratic Object has been added
 VRFCPLFFPLR Curve:Quadratic Object has been added
 HeatingEIRHiPLR Curve:Quadratic Object has been added
 CoolingCombRatio Curve:Linear Object has been added
 HeatingCombRatio Curve:Linear Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 AirConditioner:VariableRefrigerantFlow Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 AirConditioner:VariableRefrigerantFlow Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 Zone List Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Zone List Object has been added
 BLOCK1:ZONE2 Sizing:Zone Object has been added
 BLOCK1:ZONE1 Sizing:Zone Object has been added
 BLOCK1:ZONE2 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE1 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE2 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE1 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE2 Nodelist Objects has been added
 BLOCK1:ZONE1 Nodelist Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE1 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentList Objects has been added
 BLOCK1:ZONE1 ZoneHVAC:EquipmentList Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Fan:ConstantVolume Object has been added
 BLOCK1:ZONE1 Fan:ConstantVolume Object has been added
 Vent_SP_temp Schedule has been added
 AHST_Sch_BLOCK1_ZONE2 Schedule has been added
 ACST_Sch_BLOCK1_ZONE2 Schedule has been added
 AHST_Sch_BLOCK1_ZONE1 Schedule has been added

(continues on next page)

(continued from previous page)

ACST_Sch_BLOCK1_ZONE1 Schedule has been added
Added - SetComfTemp Program
Added - CountHours_BLOCK1_ZONE2 Program
Added - CountHours_BLOCK1_ZONE1 Program
Added - SetAppLimits Program
Added - ApplyCAT Program
Added - SetAST Program
Added - SetASTnoTol Program
Added - CountHoursNoApp_BLOCK1_ZONE2 Program
Added - SetGeoVarBLOCK1_ZONE2 Program
Added - CountHoursNoApp_BLOCK1_ZONE1 Program
Added - SetGeoVarBLOCK1_ZONE1 Program
Added - SetInputData Program
Added - SetVOFinputData Program
Added - SetVST Program
Added - ApplyAST_BLOCK1_ZONE2 Program
Added - ApplyAST_BLOCK1_ZONE1 Program
Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall1_3_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone2_Wall1_5_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall1_2_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall1_3_0_0_0_0_Win Program
Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_Win Program
Added - SetWindowOperation_Block1_Zone1_Wall1_5_0_0_0_0_Win Program
Added - Comfort Temperature Output Variable
Added - Adaptive Cooling Setpoint Temperature Output Variable
Added - Adaptive Heating Setpoint Temperature Output Variable
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Output Variable
Added - Adaptive Heating Setpoint Temperature_No Tolerance Output Variable
Added - Ventilation Setpoint Temperature Output Variable
Added - Minimum Outdoor Temperature for ventilation Output Variable
Added - Minimum Outdoor Temperature Difference for ventilation Output Variable
Added - Maximum Outdoor Temperature Difference for ventilation Output Variable
Added - Multiplier for Ventilation Opening Factor Output Variable
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Output Variable
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Output Variable
Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Output Variable
Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
Added - Zone Floor Area_BLOCK1_ZONE2 Output Variable

(continues on next page)

(continued from previous page)

Added - Zone Floor Area_BLOCK1_ZONE1 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE2 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE1 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE2 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE1 Output Variable
 Global variables objects have been added
 Internal variables objects have been added
 Added - RMOT Sensor
 Added - PMOT Sensor
 Added - BLOCK1_ZONE2_OpT Sensor
 Added - BLOCK1_ZONE2_WindSpeed Sensor
 Added - BLOCK1_ZONE2_OutT Sensor
 Added - BLOCK1_ZONE1_OpT Sensor
 Added - BLOCK1_ZONE1_WindSpeed Sensor
 Added - BLOCK1_ZONE1_OutT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - OutT Sensor
 Added - AHST_Act_BLOCK1_ZONE2 Actuator
 Added - ACST_Act_BLOCK1_ZONE2 Actuator
 Added - AHST_Act_BLOCK1_ZONE1 Actuator
 Added - ACST_Act_BLOCK1_ZONE1 Actuator
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - BLOCK1_ZONE2_CoolCoil Sensor
 Added - BLOCK1_ZONE2_HeatCoil Sensor
 Added - BLOCK1_ZONE1_CoolCoil Sensor
 Added - BLOCK1_ZONE1_HeatCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_HeatCoil Sensor

(continues on next page)

(continued from previous page)

```

Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
Added - SetComfTemp Program Calling Manager
Added - CountHours_BLOCK1_ZONE2 Program Calling Manager
Added - CountHours_BLOCK1_ZONE1 Program Calling Manager
Added - SetAppLimits Program Calling Manager
Added - ApplyCAT Program Calling Manager
Added - SetAST Program Calling Manager
Added - SetASTnoTol Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE2 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE2 Program Calling Manager
Added - CountHoursNoApp_BLOCK1_ZONE1 Program Calling Manager
Added - SetGeoVarBLOCK1_ZONE1 Program Calling Manager
Added - SetInputData Program Calling Manager
Added - SetVOFinputData Program Calling Manager
Added - SetVST Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE2 Program Calling Manager
Added - ApplyAST_BLOCK1_ZONE1 Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
Added - Comfort Temperature Reporting FrequencyHourly Output:Variable data
Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyHourly Output:Variable_
↪data
Added - Adaptive Heating Setpoint Temperature Reporting FrequencyHourly Output:Variable_
↪data
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyHourly_
↪Output:Variable data
Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyHourly_
↪Output:Variable data
Added - Ventilation Setpoint Temperature Reporting FrequencyHourly Output:Variable data
Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyHourly Output:
↪Variable data
Added - Minimum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly_
↪Output:Variable data
Added - Maximum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly_
↪Output:Variable data

```

(continues on next page)

(continued from previous page)

Added - Multiplier for Ventilation Opening Factor Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳ Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳ Temperature Reporting FrequencyHourly Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyHourly Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyHourly Output:Variable data

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyHourly Output:
 ↳ Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyHourly Output:
 ↳ Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyHourly
 ↳ Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳ data

Added - AFN Zone Infiltration Volume Reporting FrequencyHourly Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳ data

Added - AFN Zone Ventilation Volume Reporting FrequencyHourly Output:Variable data

Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyHourly Output:Variable

(continues on next page)

(continued from previous page)

```

→data
Added - Site Wind Speed Reporting FrequencyHourly Output:Variable data
Added - Site Outdoor Air Relative Humidity Reporting FrequencyHourly Output:Variable data
Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data
Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data
Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyHourly Output:
→Variable data
Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyHourly Output:
→Variable data
Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
→Variable data
Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
→Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
→Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
→Variable data
Added - Comfort Temperature Reporting FrequencyRunperiod Output:Variable data
Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyRunperiod Output:
→Variable data
Added - Adaptive Heating Setpoint Temperature Reporting FrequencyRunperiod Output:
→Variable data
Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod_
→Output:Variable data
Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyRunperiod_
→Output:Variable data
Added - Ventilation Setpoint Temperature Reporting FrequencyRunperiod Output:Variable_
→data
Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyRunperiod Output:
→Variable data
Added - Minimum Outdoor Temperature Difference for ventilation Reporting_
→FrequencyRunperiod Output:Variable data
Added - Maximum Outdoor Temperature Difference for ventilation Reporting_
→FrequencyRunperiod Output:Variable data
Added - Multiplier for Ventilation Opening Factor Reporting FrequencyRunperiod Output:
→Variable data
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod_
→Output:Variable data
Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod_
→Output:Variable data
Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:
→Variable data
Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:
→Variable data
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod_
→Output:Variable data
Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod_
→Output:Variable data
Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod_
→Output:Variable data

```

(continues on next page)

(continued from previous page)

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod
 ↳Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting
 ↳FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting
 ↳FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting
 ↳FrequencyRunperiod Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting
 ↳FrequencyRunperiod Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyRunperiod Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyRunperiod Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyRunperiod Output:Variable data

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyRunperiod
 ↳Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - AFN Zone Infiltration Volume Reporting FrequencyRunperiod Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - AFN Zone Ventilation Volume Reporting FrequencyRunperiod Output:Variable data

Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - Site Wind Speed Reporting FrequencyRunperiod Output:Variable data

Added - Site Outdoor Air Relative Humidity Reporting FrequencyRunperiod Output:Variable
 ↳data

Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyRunperiod Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyRunperiod Output:Variable data

Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
 ↳Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
 ↳Variable data

(continues on next page)

(continued from previous page)

```

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyRunperiod Output:
  ↵ Variable data
Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyRunperiod Output:
  ↵ Variable data
Added - * Adaptive Cooling Setpoint Temperature_No Tolerance Output:Variable data
Added - * Adaptive Heating Setpoint Temperature_No Tolerance Output:Variable data
Added - * Zone Operative Temperature Output:Variable data
Added - * Adaptive Cooling Setpoint Temperature_No Tolerance Output:Variable data
Added - * Adaptive Heating Setpoint Temperature_No Tolerance Output:Variable data
Added - * Zone Operative Temperature Output:Variable data
IDF has been saved
Ending with file:
TestModel

```

=====END OF GENERIC IDF FILE GENERATION PROCESS=====

The following IDFs will not work, and therefore these will be deleted:

None

=====START OF OUTPUT IDF FILES GENERATION
 PROCESS=====

The list of output IDFs is going to be:

```

TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf

```

(continues on next page)

(continued from previous page)

```

TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
And the total number of output IDF's is going to be 36
Generating the following output IDF files:

```

```

TestModel[CS_INT EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_1[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_2[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT EN16798[CA_3[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_80[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_0.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.1[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.4[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_1.5[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_2[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_AUS DeDear[CA_90[CM_3[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf

```

=====END OF OUTPUT IDF FILES GENERATION PROCESS=====

Now, let's inspect the Output:Variable objects in both the input idf and any of the output idfs, for instance, the first one.

[53]: `input_testmodel = add_accis_instance.input_idfs['TestModel']`

(continues on next page)

(continued from previous page)

```
input_testmodel.idfobjects['Output:Variable']
```

[53]: []

We can see there is no Output:Variable object in the input idf. Now let's see some output idf.

```
[54]: output_testmodel = add_accis_instance.output_idfs[
    'TestModel[CS_INT_EN16798[CA_1[CM_0[HM_2[VC_2[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf'
]
output_testmodel.idfobjects['Output:Variable']
```

[54]: [

```
OUTPUT:VARIABLE,
*,                      !- Key Value
Adaptive Cooling Setpoint Temperature_No Tolerance,      !- Variable Name
Hourly,                  !- Reporting Frequency
;                      !- Schedule Name

,
OUTPUT:VARIABLE,
*,                      !- Key Value
Adaptive Heating Setpoint Temperature_No Tolerance,      !- Variable Name
Hourly,                  !- Reporting Frequency
;                      !- Schedule Name

,
OUTPUT:VARIABLE,
*,                      !- Key Value
Zone Operative Temperature,      !- Variable Name
Hourly,                  !- Reporting Frequency
;                      !- Schedule Name

,
OUTPUT:VARIABLE,
*,                      !- Key Value
Adaptive Cooling Setpoint Temperature_No Tolerance,      !- Variable Name
Runperiod,                !- Reporting Frequency
;                      !- Schedule Name

,
OUTPUT:VARIABLE,
*,                      !- Key Value
Adaptive Heating Setpoint Temperature_No Tolerance,      !- Variable Name
Runperiod,                !- Reporting Frequency
;                      !- Schedule Name

,
OUTPUT:VARIABLE,
*,                      !- Key Value
Zone Operative Temperature,      !- Variable Name
Runperiod,                !- Reporting Frequency
;                      !- Schedule Name
]
```

We can see in this case 6 Output:Variable objects, which are the ones we previously requested: - Zone Operative Temperature - Adaptive Cooling Setpoint Temperature_No Tolerance - Adaptive Heating Setpoint Temperature_No Tolerance

in both Hourly and Runperiod Reporting Frequencies.

As usual, let's remove all the output idfs, so that we can re-run the notebook.

```
[40]: for i in os.listdir():
    if i.endswith('.idf') and input_idf[0] not in i:
        os.remove(i)
```


FULL EXAMPLE OF ADAPTIVE SETPOINT TEMPERATURE SIMULATION

In this section, we're going to run a simulation with adaptive setpoint temperatures. Say we want to run some simulations using a **Japanese local comfort model** developed by Hom Bahadur Rijal in some **locations in Japan**, and also we want to **analyse and visualize the data**. First of all, given EnergyPlus (any version between 9.1 and 23.1 included) is installed, and accim has been installed by entering ‘pip install accim’ in the CMD terminal, let's prepare the files we need: the IDF(s) and the EPW(s). Let's see what file we have in the folder and then we'll continue with the IDF(s).

```
[1]: from os import listdir
input_files = [i for i in listdir()]
print(*input_files, sep='\n')

.ipynb_checkpoints
backup
Current_Naha_JA-hour.epw
Current_Sapporo_JA-hour.epw
full_example.ipynb
rcp85_2100_Naha_JA-hour.epw
rcp85_2100_Sapporo_JA-hour.epw
TestModel.idf
__init__.py
```

9.1 1. IDF (using addAccis())

Say we have one or multiple IDF files, with an existing HVAC system (in this case, the use of mixed-mode ScriptType ‘ex_mm’ is not recommended; only full air-conditioning) or with no HVAC system at all (in this case, any of the ‘vrf_ac’ or ‘vrf_mm’ ScriptTypes are recommended). In this example, we are going to use an IDF without HVAC system, and we are going to use ‘vrf_mm’ so that accim adds a generic VRF system.

Let's see what IDFs we do have in our folder:

```
[2]: input_idfs = [i for i in listdir() if i.endswith('.idf')]
print(input_idfs)

['TestModel.idf']
```

So now, we're going to generate building energy models with setpoint temperatures based on the Japanese comfort model (i.e. ComfStand takes the value 3) and ASHRAE 55 (i.e. ComfStand takes the value 2). We're going to select the 80% acceptability levels for both (i.e. CAT takes the value 80), and we're going to select the setpoint behaviour to horizontally extend the setpoint temperatures (or comfort limits) when applicability limits are exceeded (i.e. ComfMod

takes the value 3). There are 2 methods to apply adaptive setpoint temperatures: - Short method, which is running the following 2 lines of code:

```
from accim.sim import accis
accis.addAccis()
```

When we run the 2 lines of code above, accim is going to ask us to enter some information it needs to generate the output IDFs. The data we're going to input, in the same order, is: - Enter the ScriptType: **vrf_mm** - Enter the SupplyAirTempInputMethod: **temperature difference** - Do you want to keep the existing outputs (true or false)?: **false** - Enter the Output type (standard, simplified or detailed): **standard** - Enter the Output frequencies separated by space (timestep, hourly, daily, monthly, runperiod): **hourly monthly** - Enter the EnergyPlus version (9.1 to 23.1): **23.1** - Enter the Temperature Control method (temperature or pmv): **temperature**

After that, accim will let us know the information we have entered, and it will start the generic IDF generation process. Lots of actions are going to be performed, and all of them will be printed on screen. Once this process is done, accim will let us know if any of the IDFs is not going to work for any reason, and then it will start the output IDF files generation process. Then, accim will ask us again to enter some information, this time to generate the output IDF(s). The data we are going to enter now is:

- Enter the Comfort Standard numbers separated by space: **2 3**
- Enter the Category numbers separated by space: **80**
- Enter the Comfort Mode numbers separated by space: **0 3** (where 0 and 3 are respectively static and adaptive setpoints)
- Enter the HVAC Mode numbers separated by space: **1 2** (in this case we have also selected 1 for naturally ventilated, to see the difference with mixed-mode)
- Enter the Ventilation Control numbers separated by space: **0**

For all the remaining arguments, we're going to hit enter to omit it and take the default value. Finally, accim will let us know the list of output IDFs and will ask for confirmation to proceed:

- Do you still want to run ACCIS? [y/n]: **y**

Alternatively, we could specify all the arguments when calling the function, as shown in the cell below:

```
[3]: from accim.sim import accis
accis.addAccis(
    ScriptType='vrf_mm',
    SupplyAirTempInputMethod='temperature difference',
    Output_keep_existing=False,
    Output_type='standard',
    Output_freqs=['hourly', 'monthly'],
    EnergyPlus_version='23.1',
    TempCtrl='temperature',
    ComfStand=[2, 3],
    CAT=[80],
    ComfMod=[0, 3],
    HVACmode=[1, 2],
    VentCtrl=[0],
    VSToffset=[0],
    MinOToffset=[50],
    MaxWindSpeed=[50],
    ASTtol_steps=0.1,
    ASTtol_start=0.1,
    ASTtol_end_input=0.1,
```

(continues on next page)

(continued from previous page)

```
    confirmGen=True
)
```

Adaptive-Comfort-Control-Implemented Model (ACCIM)

This tool allows to apply adaptive setpoint temperatures.
For further information, please read the documentation:
<https://accim.readthedocs.io/en/master/>
For a visual understanding of the tool, please visit the following jupyter notebooks:
- Using addAccis() to apply adaptive setpoint temperatures
https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/addAccis/using_addAccis.ipynb
- Using rename_epw_files() to rename the EPWs for proper data analysis after simulation
https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/rename_epw_files/using_rename_epw_files.ipynb
- Using runEp() to directly run simulations with EnergyPlus
https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/runEp/using_runEp.ipynb
- Using the class Table() for data analysis
https://github.com/dsanchez-garcia/accim/blob/master/accim/sample_files/jupyter_notebooks/Table/using_Table.ipynb

Starting with the process.

Basic input data:

ScriptType is: vrf_mm

Supply Air Temperature Input Method is: temperature difference

Output type is: standard

Output frequencies are:

['hourly', 'monthly']

EnergyPlus version is: 23.1

Temperature Control method is: temperature

=====START OF GENERIC IDF FILE GENERATION
 ↵PROCESS=====

Starting with file:

TestModel

IDD location is: C:\EnergyPlusV23-1-0\Energy+.idd

The occupied zones in the model TestModel are:

BLOCK1:ZONE2

BLOCK1:ZONE1

The windows and doors in the model TestModel are:

Block1_Zone2_Wall_3_0_0_0_0_0_Win

Block1_Zone2_Wall_4_0_0_0_0_0_Win

Block1_Zone2_Wall_5_0_0_0_0_0_Win

Block1_Zone1_Wall_2_0_0_0_0_0_Win

Block1_Zone1_Wall_3_0_0_0_0_0_Win

Block1_Zone1_Wall_5_0_0_0_0_0_Win

(continues on next page)

(continued from previous page)

The zones in the model TestModel are:

BLOCK1_ZONE2

BLOCK1_ZONE1

The people objects in the model have been amended.

BLOCK1:ZONE2 Thermostat has been added

BLOCK1:ZONE1 Thermostat has been added

On Schedule already was in the model

TypOperativeTempControlSch Schedule already was in the model

All ZoneHVAC:IdealLoadsAirSystem Heating and Cooling availability schedules has been setto on

On 24/7 Schedule already was in the model

Control type schedule: Always 4 Schedule has been added

Relative humidity setpoint schedule: Always 50.00 Schedule has been added

Heating Fanger comfort setpoint: Always -0.5 Schedule has been added

Cooling Fanger comfort setpoint: Always 0.1 Schedule has been added

Zone CO2 setpoint: Always 900ppm Schedule has been added

Min CO2 concentration: Always 600ppm Schedule has been added

Generic contaminant setpoint: Always 0.5ppm Schedule has been added

Air distribution effectiveness (always 1) Schedule has been added

VRF Heating Cooling (Northern Hemisphere) Schedule has been added

DefaultFanEffRatioCurve Curve:Cubic Object has been added

VRFTUCoolCapFT Curve:Cubic Object has been added

VRFTUHeatCapFT Curve:Cubic Object has been added

VRFCoolCapFTBoundary Curve:Cubic Object has been added

VRFCoolEIRFTBoundary Curve:Cubic Object has been added

CoolingEIRLowPLR Curve:Cubic Object has been added

VRFHeatCapFTBoundary Curve:Cubic Object has been added

VRFHeatEIRFTBoundary Curve:Cubic Object has been added

HeatingEIRLowPLR Curve:Cubic Object has been added

DefaultFanPowerRatioCurve Curve:Exponent Object has been added

DXHtgCoilDefrostEIRFT Curve:Biquadratic Object has been added

VRFCoolCapFT Curve:Biquadratic Object has been added

VRFCoolCapFTHi Curve:Biquadratic Object has been added

VRFCoolEIRFT Curve:Biquadratic Object has been added

VRFCoolEIRFTHi Curve:Biquadratic Object has been added

VRFHeatCapFT Curve:Biquadratic Object has been added

VRFHeatCapFTHi Curve:Biquadratic Object has been added

VRFHeatEIRFT Curve:Biquadratic Object has been added

VRFHeatEIRFTHi Curve:Biquadratic Object has been added

CoolingLengthCorrectionFactor Curve:Biquadratic Object has been added

VRF Piping Correction Factor for Length in Heating Mode Curve:Biquadratic Object has been added

VRF Heat Recovery Cooling Capacity Modifier Curve:Biquadratic Object has been added

VRF Heat Recovery Cooling Energy Modifier Curve:Biquadratic Object has been added

VRF Heat Recovery Heating Capacity Modifier Curve:Biquadratic Object has been added

VRF Heat Recovery Heating Energy Modifier Curve:Biquadratic Object has been added

VRFAccCoolCapFFF Curve:Quadratic Object has been added

CoolingEIRHiPLR Curve:Quadratic Object has been added

VRFCPLFFPLR Curve:Quadratic Object has been added

HeatingEIRHiPLR Curve:Quadratic Object has been added

CoolingCombRatio Curve:Linear Object has been added

HeatingCombRatio Curve:Linear Object has been added

(continues on next page)

(continued from previous page)

VRF Outdoor Unit_BLOCK1:ZONE2 AirConditioner:VariableRefrigerantFlow Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 AirConditioner:VariableRefrigerantFlow Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE2 Zone List Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Outdoor Air Node Object has been added
 VRF Outdoor Unit_BLOCK1:ZONE1 Zone List Object has been added
 BLOCK1:ZONE2 Sizing:Zone Object has been added
 BLOCK1:ZONE1 Sizing:Zone Object has been added
 BLOCK1:ZONE2 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE1 Design Specification Outdoor Air Object has been added
 BLOCK1:ZONE2 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE1 Design Specification Zone Air Distribution Object has been added
 BLOCK1:ZONE2 Nodelist Objects has been added
 BLOCK1:ZONE1 Nodelist Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE1 ZoneHVAC:EquipmentConnections Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:EquipmentList Objects has been added
 BLOCK1:ZONE1 ZoneHVAC:EquipmentList Objects has been added
 BLOCK1:ZONE2 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 ZoneHVAC:TerminalUnit:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 Coil:Cooling:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE1 Coil:Heating:DX:VariableRefrigerantFlow Object has been added
 BLOCK1:ZONE2 Fan:ConstantVolume Object has been added
 BLOCK1:ZONE1 Fan:ConstantVolume Object has been added
 Vent_SP_temp Schedule has been added
 AHST_Sch_BLOCK1_ZONE2 Schedule has been added
 ACST_Sch_BLOCK1_ZONE2 Schedule has been added
 AHST_Sch_BLOCK1_ZONE1 Schedule has been added
 ACST_Sch_BLOCK1_ZONE1 Schedule has been added
 Added - SetComfTemp Program
 Added - CountHours_BLOCK1_ZONE2 Program
 Added - CountHours_BLOCK1_ZONE1 Program
 Added - SetAppLimits Program
 Added - ApplyCAT Program
 Added - SetAST Program
 Added - SetASTnoTol Program
 Added - CountHoursNoApp_BLOCK1_ZONE2 Program
 Added - SetGeoVarBLOCK1_ZONE2 Program
 Added - CountHoursNoApp_BLOCK1_ZONE1 Program
 Added - SetGeoVarBLOCK1_ZONE1 Program
 Added - SetInputData Program
 Added - SetVOFinputData Program
 Added - SetVST Program
 Added - ApplyAST_BLOCK1_ZONE2 Program
 Added - ApplyAST_BLOCK1_ZONE1 Program
 Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program

(continues on next page)

(continued from previous page)

Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_Win Program
 Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_Win Program
 Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_Win Program
 Added - Comfort Temperature Output Variable
 Added - Adaptive Cooling Setpoint Temperature Output Variable
 Added - Adaptive Heating Setpoint Temperature Output Variable
 Added - Adaptive Cooling Setpoint Temperature_No Tolerance Output Variable
 Added - Adaptive Heating Setpoint Temperature_No Tolerance Output Variable
 Added - Ventilation Setpoint Temperature Output Variable
 Added - Minimum Outdoor Temperature for ventilation Output Variable
 Added - Minimum Outdoor Temperature Difference for ventilation Output Variable
 Added - Maximum Outdoor Temperature Difference for ventilation Output Variable
 Added - Multiplier for Ventilation Opening Factor Output Variable
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Output Variable
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Output Variable
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Output Variable
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Output Variable
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Output Variable
 Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE2 Output Variable
 Added - Zone Floor Area_BLOCK1_ZONE1 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE2 Output Variable
 Added - Zone Air Volume_BLOCK1_ZONE1 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE2 Output Variable
 Added - Ventilation Hours_BLOCK1_ZONE1 Output Variable
 Global variables objects have been added
 Internal variables objects have been added
 Added - RMOT Sensor
 Added - PMOT Sensor
 Added - BLOCK1_ZONE2_OpT Sensor
 Added - BLOCK1_ZONE2_WindSpeed Sensor
 Added - BLOCK1_ZONE2_OutT Sensor
 Added - BLOCK1_ZONE1_OpT Sensor
 Added - BLOCK1_ZONE1_WindSpeed Sensor
 Added - BLOCK1_ZONE1_OutT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_Win_WindSpeed Sensor

(continues on next page)

(continued from previous page)

Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_OutT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OpT Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_WindSpeed Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_OutT Sensor
 Added - OutT Sensor
 Added - AHST_Act_BLOCK1_ZONE2 Actuator
 Added - ACST_Act_BLOCK1_ZONE2 Actuator
 Added - AHST_Act_BLOCK1_ZONE1 Actuator
 Added - ACST_Act_BLOCK1_ZONE1 Actuator
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_VentOpenFact Actuator
 Added - BLOCK1_ZONE2_CoolCoil Sensor
 Added - BLOCK1_ZONE2_HeatCoil Sensor
 Added - BLOCK1_ZONE1_CoolCoil Sensor
 Added - BLOCK1_ZONE1_HeatCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_4_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone2_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone1_Wall_2_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone1_Wall_3_0_0_0_0_0_Win_HeatCoil Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_CoolCoil Sensor
 Added - Block1_Zone1_Wall_5_0_0_0_0_0_Win_HeatCoil Sensor
 Added - SetComfTemp Program Calling Manager
 Added - CountHours_BLOCK1_ZONE2 Program Calling Manager
 Added - CountHours_BLOCK1_ZONE1 Program Calling Manager
 Added - SetAppLimits Program Calling Manager
 Added - ApplyCAT Program Calling Manager
 Added - SetAST Program Calling Manager
 Added - SetASTnoTol Program Calling Manager
 Added - CountHoursNoApp_BLOCK1_ZONE2 Program Calling Manager
 Added - SetGeoVarBLOCK1_ZONE2 Program Calling Manager
 Added - CountHoursNoApp_BLOCK1_ZONE1 Program Calling Manager
 Added - SetGeoVarBLOCK1_ZONE1 Program Calling Manager
 Added - SetInputData Program Calling Manager

(continues on next page)

(continued from previous page)

Added - SetVOFinputData Program Calling Manager
 Added - SetVST Program Calling Manager
 Added - ApplyAST_BLOCK1_ZONE2 Program Calling Manager
 Added - ApplyAST_BLOCK1_ZONE1 Program Calling Manager
 Added - SetMyVOF_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone2_Wall_3_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone2_Wall_4_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone2_Wall_5_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone1_Wall_2_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone1_Wall_3_0_0_0_0_0_Win Program Calling Manager
 Added - SetMyVOF_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
 Added - SetWindowOperation_Block1_Zone1_Wall_5_0_0_0_0_0_Win Program Calling Manager
 Added - Comfort Temperature Reporting FrequencyHourly Output:Variable data
 Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyHourly Output:Variable
 ↵data
 Added - Adaptive Heating Setpoint Temperature Reporting FrequencyHourly Output:Variable
 ↵data
 Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyHourly
 ↵Output:Variable data
 Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyHourly
 ↵Output:Variable data
 Added - Ventilation Setpoint Temperature Reporting FrequencyHourly Output:Variable data
 Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyHourly Output:
 ↵Variable data
 Added - Minimum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly
 ↵Output:Variable data
 Added - Maximum Outdoor Temperature Difference for ventilation Reporting FrequencyHourly
 ↵Output:Variable data
 Added - Multiplier for Ventilation Opening Factor Reporting FrequencyHourly Output:
 ↵Variable data
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↵Variable data
 Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↵Variable data
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyHourly Output:
 ↵Variable data
 Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyHourly Output:
 ↵Variable data
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↵Output:Variable data
 Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↵Output:Variable data
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↵Output:Variable data
 Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↵Output:Variable data
 Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↵Output:Variable data

(continues on next page)

(continued from previous page)

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyHourly
 ↳Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyHourly Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyHourly Output:Variable data

Added - Cooling Coil Total Cooling Rate Reporting FrequencyHourly Output:Variable data

Added - Heating Coil Heating Rate Reporting FrequencyHourly Output:Variable data

Added - Facility Total HVAC Electric Demand Power Reporting FrequencyHourly Output:
 ↳Variable data

Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyHourly Output:
 ↳Variable data

Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyHourly
 ↳Output:Variable data

Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳data

Added - AFN Zone Infiltration Volume Reporting FrequencyHourly Output:Variable data

Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyHourly Output:Variable
 ↳data

Added - AFN Zone Ventilation Volume Reporting FrequencyHourly Output:Variable data

Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyHourly Output:Variable
 ↳data

Added - Site Wind Speed Reporting FrequencyHourly Output:Variable data

Added - Site Outdoor Air Relative Humidity Reporting FrequencyHourly Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyHourly Output:Variable data

Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyHourly Output:Variable data

Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyHourly Output:
 ↳Variable data

Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyHourly Output:
 ↳Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
 ↳Variable data

Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
 ↳Variable data

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyHourly Output:
 ↳Variable data

Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyHourly Output:
 ↳Variable data

Added - Comfort Temperature Reporting FrequencyMonthly Output:Variable data

(continues on next page)

(continued from previous page)

Added - Adaptive Cooling Setpoint Temperature Reporting FrequencyMonthly Output:Variable
 ↳data

Added - Adaptive Heating Setpoint Temperature Reporting FrequencyMonthly Output:Variable
 ↳data

Added - Adaptive Cooling Setpoint Temperature_No Tolerance Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Adaptive Heating Setpoint Temperature_No Tolerance Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Ventilation Setpoint Temperature Reporting FrequencyMonthly Output:Variable data

Added - Minimum Outdoor Temperature for ventilation Reporting FrequencyMonthly Output:
 ↳Variable data

Added - Minimum Outdoor Temperature Difference for ventilation Reporting
 ↳FrequencyMonthly Output:Variable data

Added - Maximum Outdoor Temperature Difference for ventilation Reporting
 ↳FrequencyMonthly Output:Variable data

Added - Multiplier for Ventilation Opening Factor Reporting FrequencyMonthly Output:
 ↳Variable data

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE2 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Comfortable Hours_No Applicability_BLOCK1_ZONE1 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE2 Reporting FrequencyMonthly Output:
 ↳Variable data

Added - Comfortable Hours_Applicability_BLOCK1_ZONE1 Reporting FrequencyMonthly Output:
 ↳Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Discomfortable Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Discomfortable Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE2 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Hot Hours_BLOCK1_ZONE1 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE2 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Discomfortable Non Applicable Cold Hours_BLOCK1_ZONE1 Reporting FrequencyMonthly
 ↳Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE2 Reporting FrequencyMonthly Output:Variable data

Added - Zone Floor Area_BLOCK1_ZONE1 Reporting FrequencyMonthly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE2 Reporting FrequencyMonthly Output:Variable data

Added - Zone Air Volume_BLOCK1_ZONE1 Reporting FrequencyMonthly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE2 Reporting FrequencyMonthly Output:Variable data

Added - Ventilation Hours_BLOCK1_ZONE1 Reporting FrequencyMonthly Output:Variable data

Added - Zone Operative Temperature Reporting FrequencyMonthly Output:Variable data

Added - Zone Thermal Comfort CEN 15251 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyMonthly Output:Variable data

Added - Zone Thermal Comfort ASHRAE 55 Adaptive Model Running Average Outdoor Air
 ↳Temperature Reporting FrequencyMonthly Output:Variable data

(continues on next page)

(continued from previous page)

Added - Cooling Coil Total Cooling Rate Reporting FrequencyMonthly Output:Variable data
 Added - Heating Coil Heating Rate Reporting FrequencyMonthly Output:Variable data
 Added - Facility Total HVAC Electric Demand Power Reporting FrequencyMonthly Output:
 ↳ Variable data
 Added - Facility Total HVAC Electricity Demand Rate Reporting FrequencyMonthly Output:
 ↳ Variable data
 Added - AFN Surface Venting Window or Door Opening Factor Reporting FrequencyMonthly
 ↳ Output:Variable data
 Added - AFN Zone Infiltration Air Change Rate Reporting FrequencyMonthly Output:Variable
 ↳ data
 Added - AFN Zone Infiltration Volume Reporting FrequencyMonthly Output:Variable data
 Added - AFN Zone Ventilation Air Change Rate Reporting FrequencyMonthly Output:Variable
 ↳ data
 Added - AFN Zone Ventilation Volume Reporting FrequencyMonthly Output:Variable data
 Added - Site Outdoor Air Drybulb Temperature Reporting FrequencyMonthly Output:Variable
 ↳ data
 Added - Site Wind Speed Reporting FrequencyMonthly Output:Variable data
 Added - Site Outdoor Air Relative Humidity Reporting FrequencyMonthly Output:Variable
 ↳ data
 Added - AHST_Sch_BLOCK1_ZONE2 Reporting FrequencyMonthly Output:Variable data
 Added - ACST_Sch_BLOCK1_ZONE2 Reporting FrequencyMonthly Output:Variable data
 Added - AHST_Sch_BLOCK1_ZONE1 Reporting FrequencyMonthly Output:Variable data
 Added - ACST_Sch_BLOCK1_ZONE1 Reporting FrequencyMonthly Output:Variable data
 Added - VRF Heat Pump Cooling Electricity Energy Reporting FrequencyMonthly Output:
 ↳ Variable data
 Added - VRF Heat Pump Heating Electricity Energy Reporting FrequencyMonthly Output:
 ↳ Variable data
 Added - BLOCK1_ZONE2 VRF Indoor Unit DX Cooling Coil Reporting FrequencyMonthly Output:
 ↳ Variable data
 Added - BLOCK1_ZONE2 VRF Indoor Unit DX Heating Coil Reporting FrequencyMonthly Output:
 ↳ Variable data
 Added - BLOCK1_ZONE1 VRF Indoor Unit DX Cooling Coil Reporting FrequencyMonthly Output:
 ↳ Variable data
 Added - BLOCK1_ZONE1 VRF Indoor Unit DX Heating Coil Reporting FrequencyMonthly Output:
 ↳ Variable data
 IDF has been saved
 Ending with file:
 TestModel

=====END OF GENERIC IDF FILE GENERATION PROCESS=====

The following IDFs will not work, and therefore these will be deleted:
 None

=====START OF OUTPUT IDF FILES GENERATION
 ↳ PROCESS=====

The list of output IDFs is going to be:

TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf

(continues on next page)

(continued from previous page)

```

TestModel[CS_JPN Rijal[CA_80[CM_0[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
And the total number of output IDFs is going to be 8
Generating the following output IDF files:
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf

```

=====END OF OUTPUT IDF FILES GENERATION PROCESS=====

So, now let's see the list of output IDFs we have generated

```
[4]: output_idfs = [i for i in listdir() if i.endswith('.idf') and i not in input_idfs]
print(*output_idfs, sep='\n')

TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

In this case, we have generated more IDFs than we need, so let's remove the others. We only want a single naturally ventilated IDF, to compare the indoor temperature with the mixed-mode IDF with adaptive setpoints. IDFs are NV when HVACmode takes the value 1.

```
[5]: idfs_to_be_removed = [i for i in listdir() if i.endswith('.idf') and 'HM_1' in i and
                           ~'ASHRAE55[CA_80[CM_3[HM_1' not in i]
print(*idfs_to_be_removed, sep='\n')

TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

```
[6]: from os import remove
for i in idfs_to_be_removed:
    remove(i)
```

Let's see what IDFs we do finally have:

```
[7]: output_idfs = [i for i in listdir() if i.endswith('.idf') and i not in input_idfs]
print(*output_idfs, sep='\n')
```

```
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

So, we're done with the IDFs. You can see these have been named based on the input data, separated by the character '['. Let's move to the EPWs.

9.2 2. EPW (using rename_epw_files())

Let's see the EPWs we are going to use for the simulations:

```
[8]: original_epws = [i for i in listdir() if i.endswith('.epw')]
print(*original_epws, sep='\n')

Current_Naha_JA-hour.epw
Current_Sapporo_JA-hour.epw
rcp85_2100_Naha_JA-hour.epw
rcp85_2100_Sapporo_JA-hour.epw
```

However, we don't want to run the simulations using that name. To ease the later data analysis, we are going to rename the EPW files following the pattern 'Country_City_RCPscenario-Year'. This way, data will be able to be grouped by country, city, RCP scenario and year. So let's rename them running the code in the cell below.

First, accim will try to rename them based on the original name and the geolocation. If no match between those is found, accim will assign the string 'UNKNOWN' to the city. Then, accim will ask you if you want to edit some of the new names. If so, you'll need to enter the IDs: - If any of the city or subcountry names needs some amendment (if you are not happy with any of the available options, you can exclude it from renaming at the next stage), please enter the EPW IDs separated by space:(**hit enter**)

Afterwards, you'll be asked to enter the new city name for each ID you previously entered (in this case, 0 1 2 3). So, - Regarding the file ID: 0 ... Please enter the amended city or subcountry, which must be unique: **Naha** - Regarding the file ID: 1 ... Please enter the amended city or subcountry, which must be unique: **Sapporo** - Regarding the file ID: 2 ... Please enter the amended city or subcountry, which must be unique: **Naha** - Regarding the file ID: 3 ... Please enter the amended city or subcountry, which must be unique: **Sapporo**

Then, accim will let you know the old names, and the new named after amendments. Next, accim will ask you if you want to exclude some EPW from renaming. In this case, we're just going to hit enter to continue because we don't want to exclude any: - If you want to exclude some EPWs from renaming, please enter the new names separated by space, otherwise, hit enter to continue:

Finally, accim will ask for confirmation to proceed with the renaming: Do you want to rename the file or files? [y/n]:y

At this point, accim will make a copy of the EPWs and rename them. Afterwards, we would be asked if we want to delete the older EPWs. In this case, we won't because the deletion has been already set to False in the arguments.

```
[9]: from accim.data.data_preprocessing import rename_epw_files
rename_epw_files(confirm_deletion=False)

C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\data_
    ↪preprocessing.py:136: FutureWarning: The default value of regex will change from True
    ↪to False in a future version.
    epw_df['EPW_names'] = epw_df['EPW_file_names'].str.replace('.epw', '')
C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\data_
    ↪preprocessing.py:136: FutureWarning: The default value of regex will change from True
    ↪to False in a future version.
```

(continues on next page)

(continued from previous page)

```

→ preprocessing.py:138: FutureWarning: The default value of regex will change from True
→ to False in a future version. In addition, single character regular expressions
→ will *not* be treated as literal strings when regex=True.
    epw_df['EPW_mod'] = epw_df['EPW_names'].str.replace('-', '_').str.replace('.', '_').
→ str.split('_')
C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\
→ preprocessing.py:140: FutureWarning: The default value of regex will change from True
→ to False in a future version. In addition, single character regular expressions
→ will *not* be treated as literal strings when regex=True.
    epw_df['EPW_mod_filtered'] = epw_df['EPW_names'].str.replace('-', '_').str.replace('.',
→ '_').str.split('_')

```

Since no match has been found between RCP scenario Year and EPW file name, Present year
→ has been assigned to the following EPW files:

Current_Naha_JA-hour.epw
 Current_Sapporo_JA-hour.epw

The geolocation process has taken: 1.81 seconds (0.45 s/EPW)

The previous and new names of the EPW files and their unique IDs are:

ID: 0 / Current_Naha_JA-hour / Japan_UNKNOWN_Present
 ID: 1 / Current_Sapporo_JA-hour / Japan_UNKNOWN_Present
 ID: 2 / rcp85_2100_Naha_JA-hour / Japan_UNKNOWN_RCP85-2100
 ID: 3 / rcp85_2100_Sapporo_JA-hour / Japan_UNKNOWN_RCP85-2100

Duplicates have been found in the renamed EPW files, therefore these need to be amended
→ in the next stage:
 ['Japan_UNKNOWN_Present', 'Japan_UNKNOWN_RCP85-2100']

"UNKNOWN" city or subcountry have been found in the renamed EPW files, therefore these
→ need to be amended in the next stage.

If any of the city or subcountry names needs some amendment (if you are not happy with
→ any of the available options, you can exclude it from renaming at the next stage),
→ please enter the EPW IDs separated by space; otherwise, hit enter to omit:

Regarding the file ID: 0 / old name: Current_Naha_JA-hour / new name: Japan_UNKNOWN_Present, the address obtained from coordinates is:

Du Jia Fu , Li Jian Cheng Shi , Chong Nawa Xian , JP-47, 901-0292, Ri Ben
 Please enter the amended city or subcountry, which must be unique: Naha

Regarding the file ID: 1 / old name: Current_Sapporo_JA-hour / new name: Japan_UNKNOWN_Present, the address obtained from coordinates is:

Zhong Xue Gao Xiao Guan Li Dong , Jiao Shi Dong , Chuang Cheng Chuan Tong , Bei Er Shi
→ Er Tiao Xi Er Ding Mu , Bei Qu , Zha Huang Shi , Shi Shou Zhen Xing Ju , Bei Hai Dao ,
→ JP-01, 001-8501, Ri Ben

Please enter the amended city or subcountry, which must be unique: Sapporo

Regarding the file ID: 2 / old name: rcp85_2100_Naha_JA-hour / new name: Japan_UNKNOWN_RCP85-2100, the address obtained from coordinates is:

Du Jia Fu , Li Jian Cheng Shi , Chong Nawa Xian , JP-47, 901-0292, Ri Ben
 Please enter the amended city or subcountry, which must be unique: Naha

(continues on next page)

(continued from previous page)

Regarding the file ID: 3 / old name: rcp85_2100_Sapporo_JA-hour / new name: Japan_Unknown_RCP85-2100, the address obtained from coordinates is:
 Zhong Xue Gao Xiao Guan Li Dong , Jiao Shi Dong , Chuang Cheng Chuan Tong , Bei Er Shi
 Er Tiao Xi Er Ding Mu , Bei Qu , Zha Huang Shi , Shi Shou Zhen Xing Ju , Bei Hai Dao ,
 JP-01, 001-8501, Ri Ben

Please enter the amended city or subcountry, which must be unique: Sapporo

The previous and new names of the EPW files after city or subcountry name amendments and their unique IDs are:

```
ID: 0 / Current_Naha_JA-hour / Japan_Naha_Present
ID: 1 / Current_Sapporo_JA-hour / Japan_Sapporo_Present
ID: 2 / rcp85_2100_Naha_JA-hour / Japan_Naha_RCP85-2100
ID: 3 / rcp85_2100_Sapporo_JA-hour / Japan_Sapporo_RCP85-2100
```

The final list of previous and new names of the EPW files and their unique IDs is:

```
ID: 0 / Current_Naha_JA-hour / Japan_Naha_Present
ID: 1 / Current_Sapporo_JA-hour / Japan_Sapporo_Present
ID: 2 / rcp85_2100_Naha_JA-hour / Japan_Naha_RCP85-2100
ID: 3 / rcp85_2100_Sapporo_JA-hour / Japan_Sapporo_RCP85-2100
```

If you want to exclude some EPWs from renaming, please enter the IDs separated by space, otherwise, hit enter to continue:

Do you want to copy and rename the file or files? [y/n]:y
 The file Current_Naha_JA-hour has been renamed to Japan_Naha_Present
 The file Current_Sapporo_JA-hour has been renamed to Japan_Sapporo_Present
 The file rcp85_2100_Naha_JA-hour has been renamed to Japan_Naha_RCP85-2100
 The file rcp85_2100_Sapporo_JA-hour has been renamed to Japan_Sapporo_RCP85-2100

Do you want to delete the original EPW file or files? [y/n]:n

[9]: <accim.data.data_preprocessing.rename_epw_files at 0x1d4cc7c6cd0>

Now, let's see what EPWs we do have:

```
[10]: all_epws = [i for i in listdir() if i.endswith('.epw')]
print(*all_epws, sep='\n')

Current_Naha_JA-hour.epw
Current_Sapporo_JA-hour.epw
Japan_Naha_Present.epw
Japan_Naha_RCP85-2100.epw
Japan_Sapporo_Present.epw
Japan_Sapporo_RCP85-2100.epw
rcp85_2100_Naha_JA-hour.epw
rcp85_2100_Sapporo_JA-hour.epw
```

We can see the new EPWs are:

```
[11]: new_epws = [i for i in listdir() if i.endswith('.epw') if i not in original_epws]
print(*new_epws, sep='\n')

Japan_Naha_Present.epw
Japan_Naha_RCP85-2100.epw
```

(continues on next page)

(continued from previous page)

```
Japan_Sapporo_Present.epw
Japan_Sapporo_RCP85-2100.epw
```

EPWs are correctly renamed, so now let's move the old EPWs to a different folder to save them as a backup.

```
[12]: import shutil
for i in original_epws:
    shutil.move(i, f'backup/{i}')
```

Now, we can move to the next stage.

9.3 3. Running the simulation (using runEp())

At this point, we have prepared the IDF(s) we are going to simulate, which are

```
[13]: print(*output_idfs, sep='\n')

TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
```

as well as the locations where we are going to run those simulations, whose EPWs are:

```
[14]: print(*new_epws, sep='\n')

Japan_Naha_Present.epw
Japan_Naha_RCP85-2100.epw
Japan_Sapporo_Present.epw
Japan_Sapporo_RCP85-2100.epw
```

So, we are going to simulate all IDF(s) with all EPW(s). When we run later the simulations using accim, the output files (i.e. the CSVs) will be named following the pattern '**idf[epw]**', where the character '[' is used as a separator for later data analysis, so that CSV rows can be grouped by EPW location. You may have noticed the same character is used as a separator in the IDF name, in order to group the CSV rows depending on the input data.

To run the simulations, 2 methods can be used: - the shorter, in which the following 2 lines of code needs to be run: `from accim.run import run run.runEp()` After this, you'll be asked to enter the EnergyPlus version (which should coincide with the IDF EnergyPlus version): - Please enter the desired EnergyPlus version: **23.1**

Then, you will need to say if you want to run only output IDFs of accim, or otherwise all existing IDFs in the folder: - Do you want to run only ACCIM output IDFs? [y or n]: **y**

Next, accim will tell you the IDF(s) and EPW(s) it's going to use for the simulations, and finally all the simulations it's going to run based on the name pattern 'idf[epw]'. Finally, it will ask for confirmation to proceed with the simulation: - Do you still want to proceed? [y or n]:**y** - the longer method, in which the parameters are specified when calling the function. We'll use the longer method, so let's run the cell below. Since there are a few simulations, it might take a few minutes.

```
[15]: from accim.run import run
run.runEp(
    runOnlyAccim=True, #only runs output IDFs, that is, IDFs with "[" in its name
```

(continues on next page)

(continued from previous page)

```

confirmRun=True, #to skip confirmation
num CPUs=4, #to specify the number of CPUs to be used
EnergyPlus_version='23.1', #to specify the EnergyPlus version of the IDF, and the
#version of EnergyPlus you are going to run
)

```

The IDFs we are going to run are:

```

TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
and the No. of IDFs is going to be 5

```

The sample_EPWs we are going to run are:

```

Japan_Naha_Present.epw
Japan_Naha_RCP85-2100.epw
Japan_Sapporo_Present.epw
Japan_Sapporo_RCP85-2100.epw
and the No. of sample_EPWs is going to be 4

```

Therefore, the simulations are going to be:

```

TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Naha_Present
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Naha_RCP85-2100
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Sapporo_Present
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Sapporo_RCP85-2100
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Naha_Present
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Naha_RCP85-2100
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Sapporo_Present
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Naha_Present
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Sapporo_Present
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Naha_Present
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Naha_Present
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Sapporo_Present
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
-Sapporo_RCP85-2100

```

(continues on next page)

(continued from previous page)

```
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
˓Naha_Present
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
˓Naha_RCP85-2100
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
˓Sapporo_Present
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
˓Sapporo_RCP85-2100
and the No. of simulations is going to be 20
```

So simulations are done. Let's see the CSV data we have now:

```
[16]: csvs = [i for i in listdir() if i.endswith('.csv') and 'Zsz.csv' not in i and 'Table.csv'  
    ↪ ' not in i]  
print(*csvs, sep='\n')  
  
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_Present.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_RCP85-2100.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_Present.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_RCP85-2100.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_Present.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_RCP85-2100.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_Present.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_RCP85-2100.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_Present.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_RCP85-2100.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_Present.csv  
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_RCP85-2100.csv  
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_Present.csv  
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_RCP85-2100.csv  
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_Present.csv  
TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_RCP85-2100.csv  
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_Present.csv  
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Naha_RCP85-2100.csv  
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_Present.csv  
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_  
    ↪ Sapporo_RCP85-2100.csv
```

(continues on next page)

(continued from previous page)

```

↳ Sapporo_Present.csv
TestModel[CS_JPN_Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Sapporo_RCP85-2100.csv

```

Now, we can move to the last stage, in which data will be analysed and visualized.

9.4 4. Analysing and visualising the data (using Table())

In order to analyse and visualize the data, we need to make a pandas DataFrame out of the CSVs. We will do this by using the `Table()` method. To use this method, a minimum knowledge and experience with Python programming is needed, so if this is not your case, you may struggle to make it work.

Let's create an hourly dataframe, since firstly we are going to compare indoor temperature with and without adaptive setpoint temperatures.

```
[17]: from accim.data.data_postprocessing import Table
dataset_hourly = Table(
    #datasets=list #Since we are not specifying any list, it will use all available CSVs
    ↳ in the folder
    source_frequency='hourly', # This lets accim know which is the frequency of the
    ↳ input CSVs. Input CSVs with multiple frequencies are also allowed. It can be 'hourly',
    ↳ 'daily', 'monthly' and 'runperiod'. It can also be 'timestep' but might generate
    ↳ errors.
    frequency='hourly', # If 'daily', accim will aggregate the rows in days. It can be
    ↳ 'hourly', 'daily', 'monthly' and 'runperiod'. It can also be 'timestep' but might
    ↳ generate errors.
    frequency_agg_func='sum', #this makes the sum or average when aggregating in days,
    ↳ months or runperiod; since the original CSV frequency is in hour, it won't make any
    ↳ aeffect
    standard_outputs=True,
    level=['building'], # A list containing the strings 'block' and/or 'building'. For
    ↳ instance, if ['block', 'building'], accim will generate new columns to sum up or
    ↳ average in blocks and building level.
    level_agg_func=['sum', 'mean'], # A list containing the strings 'sum' and/or 'mean'.
    ↳ For instance, if ['sum', 'mean'], accim will generate the new columns explained in the
    ↳ level argument by summing and averaging.
    level_excluded_zones=[],
    split_epw_names=True, #to split EPW names based on the pattern Country_City_
    ↳ RCPscenario-Year
)
Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.
    ↳ 0[MW_50.0[AT_0.1[NS_X[Japan_Naha_Present.csv is hourly, therefore no aggregation will
    ↳ be performed.
Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.
    ↳ 0[MW_50.0[AT_0.1[NS_X[Japan_Naha_RCP85-2100.csv is hourly, therefore no aggregation
    ↳ will be performed.
Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.
    ↳ 0[MW_50.0[AT_0.1[NS_X[Japan_Sapporo_Present.csv is hourly, therefore no aggregation
    ↳ will be performed.
Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.
```

(continues on next page)

(continued from previous page)

↪ 0[MW_50.0[AT_0.1[NS_X[Japan_Sapporo_RCP85-2100.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Naha_Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Naha_RCP85-2100.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Sapporo_Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Sapporo_RCP85-2100.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Naha_Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Naha_RCP85-2100.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Sapporo_Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Sapporo_RCP85-2100.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Naha_Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Naha_RCP85-2100.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Sapporo_Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Sapporo_RCP85-2100.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Naha_Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_Naha_RCP85-2100.csv is hourly, therefore no aggregation will be performed.

Please bear in mind if you are using CSVs with multiple frequencies, you will get NaNs. ↪ The following rows have NaN values:

(continues on next page)

(continued from previous page)

	Date/Time \
8078	12/03 15:00:00
2308	04/07 05:00:00
	Environment:Site Outdoor Air Drybulb Temperature [C](Hourly) \
8078	4.35
2308	25.80
	Environment:Site Outdoor Air Relative Humidity [%](Hourly) \
8078	85.500000
2308	92.166667
	Environment:Site Wind Speed [m/s](Hourly) \
8078	0.258333
2308	2.883333
	EMS:Comfort Temperature [C](Hourly) \
8078	18.822591
2308	25.408244
	EMS:Adaptive Cooling Setpoint Temperature [C](Hourly) \
8078	24.3
2308	27.9
	EMS:Adaptive Heating Setpoint Temperature [C](Hourly) \
8078	17.5
2308	18.1
	EMS:Adaptive Cooling Setpoint Temperature_No Tolerance [C](Hourly) \
8078	24.4
2308	28.0
	EMS:Adaptive Heating Setpoint Temperature_No Tolerance [C](Hourly) \
8078	17.4
2308	18.0
	EMS:Ventilation Setpoint Temperature [C](Hourly) ... \
8078	20.900000 ...
2308	25.408244 ...
	VRF OUTDOOR UNIT_BLOCK1:ZONE1:VRF Heat Pump Cooling Electricity Energy [J](Hourly) \
8078	0.000000
2308	252442.378302
	VRF OUTDOOR UNIT_BLOCK1:ZONE1:VRF Heat Pump Heating Electricity Energy [J](Hourly) \
8078	0.0
2308	0.0
	Source Month/Day Hour \
8078	TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0... None

(continues on next page)

(continued from previous page)

```
2308 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO...
                                         None

    Month   Day   Minute   Second   count
8078       None     None     None      1
2308       None     None     None      1

[2 rows x 60 columns]
No zones have been excluded from level computations.
```

Let's filter the columns we are going to use:

```
[18]: dataset_hourly.format_table(
    type_of_table='custom', # Used to choose some predefined tables. It can be 'energy',
    #demand', 'comfort hours', 'temperature', 'all' or 'custom'
    custom_cols=[ #if type_of_table is 'custom', custom_cols is used to filter the
    #desired columns to keep
        'Adaptive Cooling Setpoint Temperature_No Tolerance (°C)',
        'Adaptive Heating Setpoint Temperature_No Tolerance (°C)',
        'Building_Total_Zone Operative Temperature (°C) (mean)',
        'BLOCK1:ZONE2ASHRAE 55 Running mean outdoor temperature (°C)',
        'Building_Total_Cooling Energy Demand (kWh/m2) (summed)',
        'Building_Total_Heating Energy Demand (kWh/m2) (summed)',
        'Building_Total_AFN Zone Infiltration Air Change Rate (ach) (summed)'
    ]
)
```

9.4.1 4.1 Visualizing the data

And now, let's generate the figure data (a list of lists and dictionaries with all information to be plotted) with `generate_fig_data()` and afterwards, let's plot the figure with `scatter_plot()`.

```
[19]: dataset_hourly.scatter_plot(
    vars_to_gather_rows=['EPW'], # variables to gather in rows of subplots
    vars_to_gather_cols=['ComfStand', 'ComfMod', 'HVACmode'], # variables to gather in
    #columns of subplots; all categorical columns which have more than 1 different value
    #across the rows, must be specified in this argument, otherwise you'll get an error.
    detailed_cols=['CS_INT ASHRAE55[CM_3[HM_1', 'CS_INT ASHRAE55[CM_3[HM_2'], # a list
    #of the specific combinations of arguments to be plotted joined by [
    data_on_x_axis='BLOCK1:ZONE2ASHRAE 55 Running mean outdoor temperature (°C)', #column name (string) for the data on x axis
    data_on_y_main_axis=[ #list which includes the name of the axis on the first place,
    #and then in the second place, a list which includes the column names you want to plot
    [
        'Indoor Operative Temperature (°C)', [
            'Adaptive Cooling Setpoint Temperature_No Tolerance (°C)', 'Adaptive Heating Setpoint Temperature_No Tolerance (°C)', 'Building_Total_Zone Operative Temperature (°C) (mean)'
        ]
    ],
],
```

(continues on next page)

(continued from previous page)

```

    ],
    colorlist_y_main_axis=[
        [
            'Indoor Operative Temperature (°C)',
            [
                'b',
                'r',
                'g',
            ]
        ],
    ],
    supxlabel='Running Mean Outdoor Temperature (°C)', # data label on x axis
    figname=f'Scatterplot_NV_vs_MM',
    figsize=6,
    ratio_height_to_width=0.33,
    confirm_graph=True
)

```

The number of rows and the list of these is going to be:

No. of rows = 4

List of rows:

Japan_Naha_Present

Japan_Naha_RCP85-2100

Japan_Sapporo_Present

Japan_Sapporo_RCP85-2100

Do you want to rename the rows? [y/n]: y

Please enter the new name for Japan_Naha_Present: Naha Present

Please enter the new name for Japan_Naha_RCP85-2100: Naha RCP8.5 2100

Please enter the new name for Japan_Sapporo_Present: Sapporo Present

Please enter the new name for Japan_Sapporo_RCP85-2100: Sapporo RCP8.5 2100

The renamed rows are going to be:

Naha Present

Naha RCP8.5 2100

Sapporo Present

Sapporo RCP8.5 2100

The number of columns and the list of these is going to be:

No. of columns = 2

List of columns:

CS_INT ASHRAE55[CM_3[HM_1

CS_INT ASHRAE55[CM_3[HM_2

Column names will be the subplot titles. Do you want to rename them? [y/n]: y

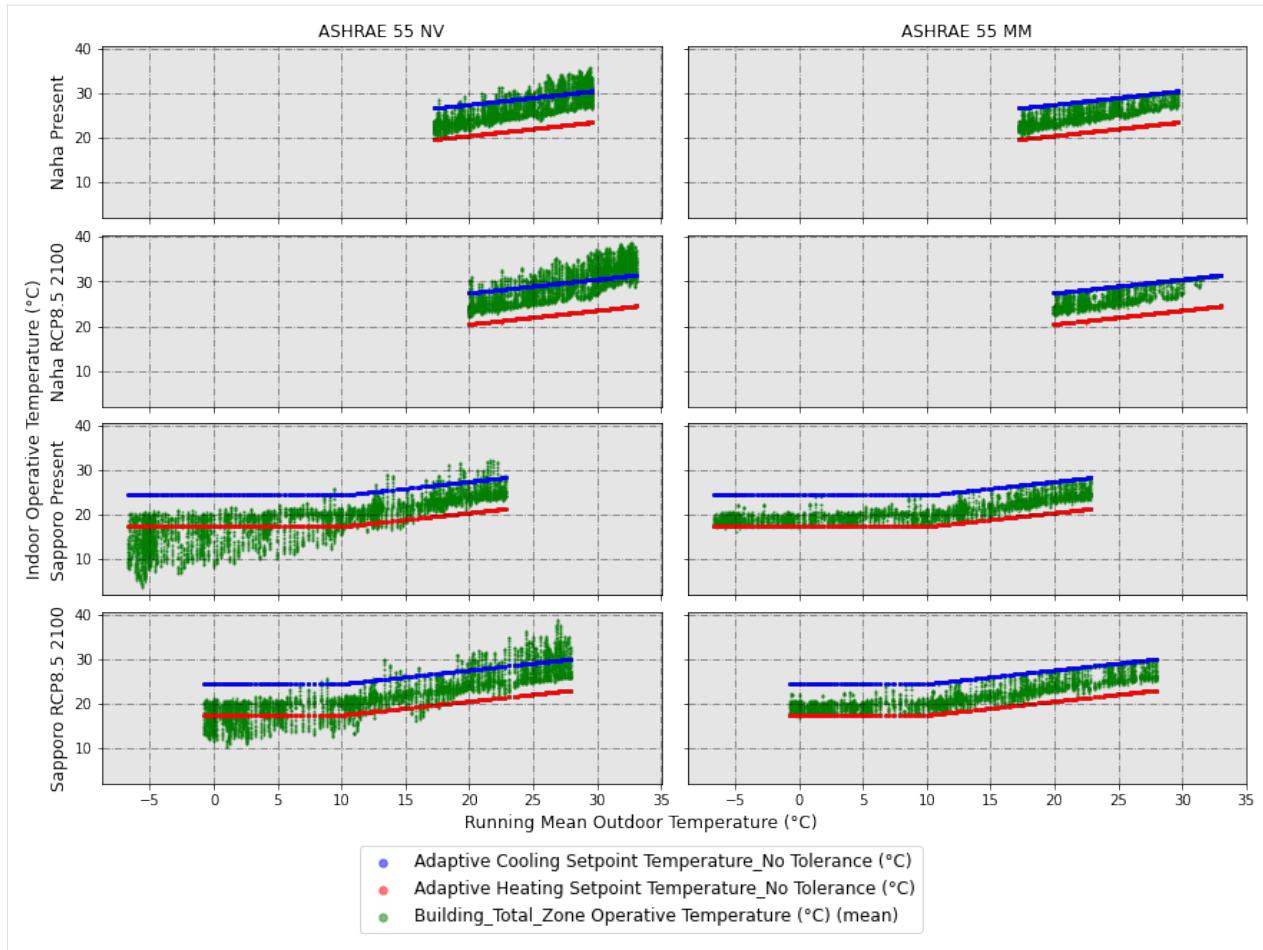
Please enter the new name for CS_INT ASHRAE55[CM_3[HM_1: ASHRAE 55 NV

Please enter the new name for CS_INT ASHRAE55[CM_3[HM_2: ASHRAE 55 MM

The renamed columns are going to be:

ASHRAE 55 NV

ASHRAE 55 MM



In this figure, you can see on the left column the simulations with free-running (or naturally ventilated) mode, while on the right, the same simulations using mixed-mode with adaptive setpoint temperatures, which introduce all hourly indoor temperatures within the adaptive thermal comfort limits.

Next, let's compare the indoor temperatures of ASHRAE 55, the local Japanese model and the static setpoints for Japan, and in this case, we're also going to plot the hourly energy demand on the secondary axis.

```
[20]: dataset_hourly.scatter_plot(
    vars_to_gather_cols=['ComfStand', 'ComfMod', 'HVACmode'], # variables to gather in
    ↵ rows of subplots
    vars_to_gather_rows=['EPW'],# variables to gather in columns of subplots
    detailed_cols=['CS_JPN Rijal[CM_0[HM_2', 'CS_INT ASHRAE55[CM_3[HM_2', 'CS_JPN_
    ↵ Rijal[CM_3[HM_2'], #we only want to see those combinations
    #custom_cols_order=['CS_JPN Rijal[CM_0[HM_2', 'CS_INT ASHRAE55[CM_3[HM_2', 'CS_JPN_
    ↵ Rijal[CM_3[HM_2'],
    data_on_x_axis='BLOCK1:ZONE2_ASHRAE 55 Running mean outdoor temperature (°C)', #column name (string) for the data on x axis
    data_on_y_sec_axis=[ #list which includes the name of the axis on the first place, and then in the second place, a list which includes the column names you want to plot
    [
        'Air renovation (ach)',
        [
            'Building_Total_AFN Zone Infiltration Air Change Rate (ach) (summed)'
    ]]
```

(continues on next page)

(continued from previous page)

```

        ],
        [
            'Operative Temperature (°C)',
            [
                'Adaptive Cooling Setpoint Temperature_No Tolerance (°C)',
                'Adaptive Heating Setpoint Temperature_No Tolerance (°C)',
                'Building_Total_Zone Operative Temperature (°C) (mean)',
            ]
        ],
    ],
    data_on_y_main_axis=[ # similarly to above, a list including the name of the
    ↵secondary y-axis and the column names you want to plot in it
    [
        'Energy (kWh/m2)',
        [
            'Building_Total_Cooling Energy Demand (kWh/m2) (summed)',
            'Building_Total_Heating Energy Demand (kWh/m2) (summed)'
        ]
    ],
    colorlist_y_sec_axis=[
        [
            'Air renovation (ach)',
            [
                'yellow'
            ]
        ],
        [
            'Operative Temperature (°C)',
            [
                'b',
                'r',
                'g',
            ]
        ],
    ],
    colorlist_y_main_axis=[
        [
            'Energy (kWh/m2)',
            [
                'cyan',
                'orange',
            ]
        ],
    ],
    supxlabel='Running Mean Outdoor Temperature (°C)', # data label on x axis
    figname=f'scatterplot_JPN_stat_ASH_adap_JPN_adap',
    figsize=6,
    ratio_height_to_width=0.33,
    confirm_graph=True
]

```

(continues on next page)

(continued from previous page)

)

The number of rows and the list of these is going to be:

No. of rows = 4

List of rows:

Japan_Naha_Present

Japan_Naha_RCP85-2100

Japan_Sapporo_Present

Japan_Sapporo_RCP85-2100

Do you want to rename the rows? [y/n]: y

Please enter the new name for Japan_Naha_Present: Naha Present

Please enter the new name for Japan_Naha_RCP85-2100: Naha RCP8.5 2100

Please enter the new name for Japan_Sapporo_Present: Sapporo Present

Please enter the new name for Japan_Sapporo_RCP85-2100: Sapporo RCP8.5 2100

The renamed rows are going to be:

Naha Present

Naha RCP8.5 2100

Sapporo Present

Sapporo RCP8.5 2100

The number of columns and the list of these is going to be:

No. of columns = 3

List of columns:

CS_JPN Rijal[CM_0[HM_2

CS_INT ASHRAE55[CM_3[HM_2

CS_JPN Rijal[CM_3[HM_2

Column names will be the subplot titles. Do you want to rename them? [y/n]: y

Please enter the new name for CS_JPN Rijal[CM_0[HM_2: JPN Static

Please enter the new name for CS_INT ASHRAE55[CM_3[HM_2: ASHRAE Adaptive

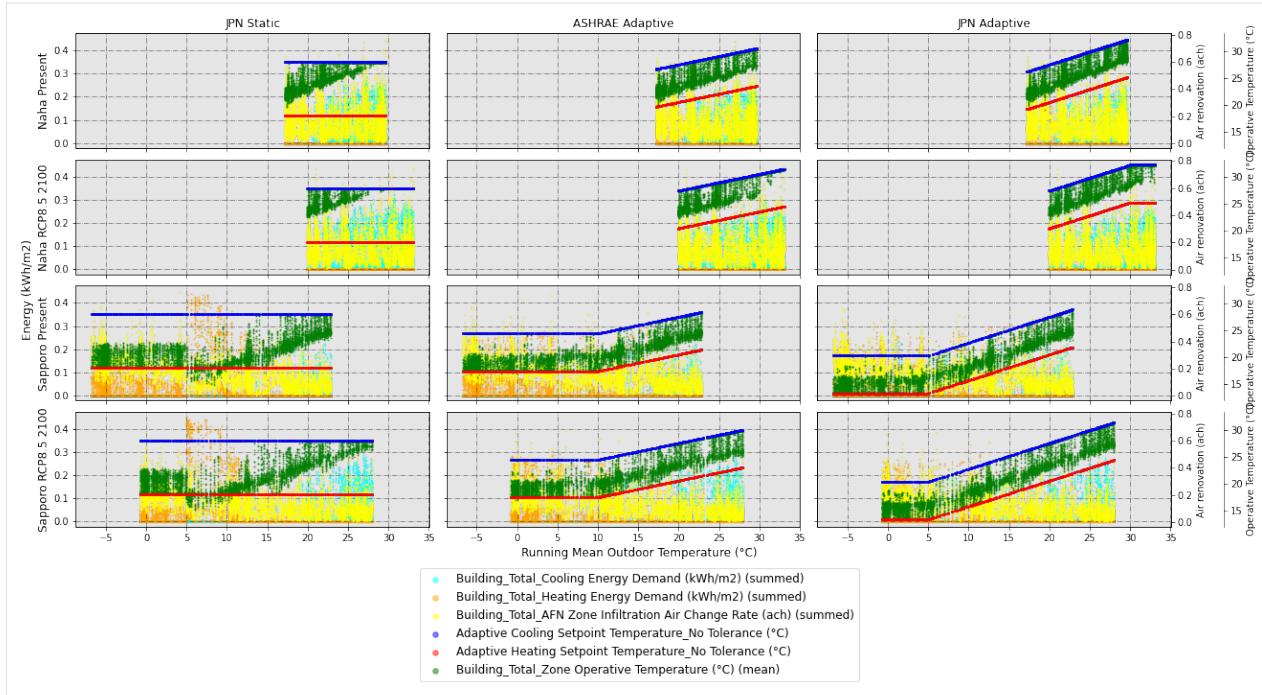
Please enter the new name for CS_JPN Rijal[CM_3[HM_2: JPN Adaptive

The renamed columns are going to be:

JPN Static

ASHRAE Adaptive

JPN Adaptive



9.4.2 4.2 Analysing the data

Now, let's see how many comfort hours were considering the NV mode, and afterwards the MM considering ASHRAE 55, as well as the impact on energy demand. Since we want to see the runperiod totals, we will need to make a new instance of `Table()`, asking for runperiod frequency this time.

```
[21]: from accim.data.data_postprocessing import Table
dataset_runperiod = Table(
    #datasets=list #Since we are not specifying any list, it will use all available CSVs
    #in the folder
    source_frequency='hourly', # This lets accim know which is the frequency of the
    #input CSVs. Input CSVs with multiple frequencies are also allowed. It can be 'hourly',
    #'daily', 'monthly' and 'runperiod'. It can also be 'timestep' but might generate
    #errors.
    frequency='runperiod', # If 'daily', accim will aggregate the rows in days. It can
    #be 'hourly', 'daily', 'monthly' and 'runperiod'. It can also be 'timestep' but might
    #generate errors.
    frequency_agg_func='sum', #this makes the sum or average when aggregating in days,
    #months or runperiod; since the original CSV frequency is in hour, it won't make any
    #affect
    standard_outputs=True,
    level=['building'], # A list containing the strings 'block' and/or 'building'. For
    #instance, if ['block', 'building'], accim will generate new columns to sum up or
    #average in blocks and building level.
    level_agg_func=['sum', 'mean'], # A list containing the strings 'sum' and/or 'mean'.
    #For instance, if ['sum', 'mean'], accim will generate the new columns explained in the
    #level argument by summing and averaging.
    level_excluded_zones=[],
    #match_cities=bool #Only used when EPW file has NOT been previously renamed
```

(continues on next page)

(continued from previous page)

```

#manage_epw_names=bool #Only used when EPW file has NOT been previously renamed
    split_epw_names=True, #to split EPW names based on the pattern Country_City_
    ↵RCPscenario-Year
)

dataset_runperiod.format_table(
    type_of_table='custom',
    custom_cols=[
        'Building_Total_Comfortable Hours_No Applicability (h) (mean)',
        'Building_Total_Total Energy Demand (kWh/m2) (summed)'
    ]
)

dataset_runperiod.wrangled_table(
    reshaping='unstack',
    vars_to_gather=['ComfStand', 'ComfMod', 'HVACmode'],
    baseline='CS_JPN Rijal[CM_3[HM_2',
    comparison_mode=['baseline compared to others'],
    comparison_cols=[]
)

```

dataset_runperiod.wrangled_df_unstacked

No zones have been excluded from level computations.

			Building_Total_Comfortable Hours_No
			Applicability (h) (mean) \ CS_INT
			ASHRAE55[CM_0[HM_2
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present	Present	
↪8706.92	RCP85	2100	
↪8692.08	Present	Present	
Sapporo	Present	Present	
↪8218.67	RCP85	2100	
↪8362.17			
			\
			CS_INT ASHRAE55[CM_3[HM_1
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present	Present	6940.58
	RCP85	2100	5111.58
Sapporo	Present	Present	5632.58
	RCP85	2100	6031.92
			\
			CS_INT ASHRAE55[CM_3[HM_2
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present	Present	8733.33
	RCP85	2100	8714.08
Sapporo	Present	Present	8533.33

(continues on next page)

(continued from previous page)

	RCP85	2100	8563.58
\\			
		CS_JPN Rijal[CM_0[HM_2	
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present	Present	8741.50
	RCP85	2100	8738.67
Sapporo	Present	Present	8471.67
	RCP85	2100	8458.92
\\			
		CS_JPN Rijal[CM_3[HM_2	
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present	Present	8744.50
	RCP85	2100	8704.50
Sapporo	Present	Present	8558.83
	RCP85	2100	8613.75
Building_Total_Total Energy Demand (kWh/m2) ↴			
↪(summed)		CS_INT ↴	
↪ASHRAE55[CM_0[HM_2			
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present	Present	
↪552.11	RCP85	2100	
↪758.73			
Sapporo	Present	Present	
↪594.15	RCP85	2100	
↪585.09			
↪CS_INT ASHRAE55[CM_3[HM_1			
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present	Present	0.0
	RCP85	2100	0.0
Sapporo	Present	Present	0.0
	RCP85	2100	0.0
↪CS_INT ASHRAE55[CM_3[HM_2			
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present	Present	265.82
	RCP85	2100	419.02
Sapporo	Present	Present	292.37
	RCP85	2100	331.92
↪CS_JPN Rijal[CM_0[HM_2			
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present	Present	356.79

(continues on next page)

(continued from previous page)

Sapporo	RCP85 Present RCP85	2100 Present 2100	556.96 334.51 425.29
CS_JPN Rijal[CM_3[HM_2]			
EPW_City_or_subcountry	EPW_Scenario	EPW_Year	
Naha	Present RCP85	Present 2100	183.66 336.46
Sapporo	Present RCP85	Present 2100	254.74 268.65

The table above shows us the comfort hours in NV (CS_INT ASHRAE55[CM_3[HM_1]) mode ranges between 5115.25 and 6945.50 hours, while the same comfort model in mixed-mode with adaptive setpoints (CS_INT ASHRAE55[CM_3[HM_2]) ranges between 8598.15 and 8758.85. Since there is no HVAC system in NV mode, the energy consumption is 0. With adaptive setpoints, the hvac energy consumption ranges between 262.08 and 414.84 (kWh/m2·year).

Now, we could finally export this table to Excel format for later style edition. Since the reshaping argument we used in the `wrangle_table()` method was ‘unstack’, the dataframe we are looking for to be exported is `dataset_runperiod.wrangle_df_unstacked`. If we used the ‘pivot’ argument, the dataframe would have been `dataset_runperiod.wrangle_df_pivoted`. So let’s export it:

```
[22]: dataset_runperiod.wrangle_df_unstacked.to_excel('df_unstacked.xlsx')
```

```
[23]: import pandas as pd
```

```
df = pd.read_excel ('df_unstacked.xlsx')
print (df)
```

```
      Unnamed: 0    Unnamed: 1 Unnamed: 2 \
0          NaN        NaN        NaN
1 EPW_City_or_subcountry  EPW_Scenario  EPW_Year
2          Naha        Present     Present
3          NaN         RCP85      2100
4         Sapporo       Present     Present
5          NaN         RCP85      2100
```

```
Building_Total_Comfortable_Hours_No_Applicability_(h)_mean) \
0           CS_INT ASHRAE55[CM_0[HM_2
1                           NaN
2                         8706.92
3                         8692.08
4                         8218.67
5                         8362.17
```

```
      Unnamed: 4    Unnamed: 5 \
0 CS_INT ASHRAE55[CM_3[HM_1  CS_INT ASHRAE55[CM_3[HM_2
1          NaN        NaN
2         6940.58      8733.33
3         5111.58      8714.08
4         5632.58      8533.33
5         6031.92      8563.58
```

(continues on next page)

(continued from previous page)

	Unnamed: 6	Unnamed: 7 \
0	CS_JPN Rijal[CM_0[HM_2	CS_JPN Rijal[CM_3[HM_2
1	NaN	NaN
2	8741.5	8744.5
3	8738.67	8704.5
4	8471.67	8558.83
5	8458.92	8613.75
 Building_Total_Total Energy Demand (kWh/m2) (summed) \		
0	CS_INT ASHRAE55[CM_0[HM_2	
1		NaN
2		552.11
3		758.73
4		594.15
5		585.09
 Unnamed: 9 Unnamed: 10 \		
0	CS_INT ASHRAE55[CM_3[HM_1	CS_INT ASHRAE55[CM_3[HM_2
1	NaN	NaN
2	0	265.82
3	0	419.02
4	0	292.37
5	0	331.92
 Unnamed: 11 Unnamed: 12		
0	CS_JPN Rijal[CM_0[HM_2	CS_JPN Rijal[CM_3[HM_2
1	NaN	NaN
2	356.79	183.66
3	556.96	336.46
4	334.51	254.74
5	425.29	268.65

Finally, so that we can run this jupyter notebook again, let's leave everything as it was at the beginning.

```
[24]: for i in original_epws:
    shutil.move(f'backup/{i}', i)

[25]: files_to_delete = [i for i in listdir() if i not in input_files]
print(*files_to_delete, sep='\n')

df_unstacked.xlsx
Japan_Naha_Present.epw
Japan_Naha_RCP85-2100.epw
Japan_Sapporo_Present.epw
Japan_Sapporo_RCP85-2100.epw
scatterplot_JPN_stat_ASH_adap_JPN_adap.png
Scatterplot_NV_vs_MM.png
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.audit
TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
```

(continues on next page)

(continued from previous page)

↵ Naha_Present.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_Present.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_PresentTable.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_PresentTable.htm
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_PresentZsz.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.audit
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100,err
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_]

(continues on next page)

(continued from previous page)

↳ Naha_RCP85-2100.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100Table.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100Table.htm
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100Zsz.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.audit
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_PresentTable.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_PresentTable.htm
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_PresentZsz.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.audit
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_]

(continues on next page)

(continued from previous page)

↵Sapporo_RCP85-2100.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100Table.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100Table.htm
 TestModel[CS_INT ASHRAE55[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Sapporo_RCP85-2100Zsz.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.audit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_PresentTable.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_PresentTable.htm

(continues on next page)

(continued from previous page)

TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_PresentZsz.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.audit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100Table.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100Table.htm
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_RCP85-2100Zsz.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.audit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.mdd

(continues on next page)

(continued from previous page)

TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_Present.mtd
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_Present.rdd
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_Present.rvaudit
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_Present.shd
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_PresentTable.csv
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_PresentTable.htm
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_PresentZsz.csv
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.audit
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.bnd
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.csv
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.dxf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.eio
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.end
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.err
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.eso
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.mdd
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.mtd
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.rdd
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.rvaudit
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100.shd
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100Table.csv
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100Table.htm
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_1[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Sapporo_RCP85-2100Zsz.csv
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Naha_Present.audit
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~
↳ Naha_Present.bnd
TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~

(continues on next page)

(continued from previous page)

↵Naha_Present.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_Present.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_PresentTable.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_PresentTable.htm
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_PresentZsz.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.audit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵Naha_RCP85-2100.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_]

(continues on next page)

(continued from previous page)

↳ Naha_RCP85-2100.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100Table.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100Table.htm
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100Zsz.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.audit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_PresentTable.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_PresentTable.htm
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_PresentZsz.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.audit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.bnd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.dxf
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.eio
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.end
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_]

(continues on next page)

(continued from previous page)

↳ Sapporo_RCP85-2100.err
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.eso
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.mdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.mtd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.rdd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.rvaudit
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100.shd
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100Table.csv
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100Table.htm
 TestModel[CS_INT ASHRAE55[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_RCP85-2100Zsz.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.audit
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.bnd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.dxf
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.eio
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.end
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.err
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.eso
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.mdd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.mtd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.rdd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.rvaudit
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_Present.shd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_PresentTable.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_PresentTable.htm
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_PresentZsz.csv

(continues on next page)

(continued from previous page)

TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.audit
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.bnd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.dxf
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.eio
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.end
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.err
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.eso
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.mdd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.mtd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.rdd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.rvaudit
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100.shd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100Table.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100Table.htm
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Naha_RCP85-2100Zsz.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.audit
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.bnd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.dxf
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.eio
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.end
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.err
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.eso
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.mdd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↳ Sapporo_Present.mtd

(continues on next page)

(continued from previous page)

TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.rdd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.rvaudit
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_Present.shd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_PresentTable.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_PresentTable.htm
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_PresentZsz.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.audit
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.bnd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.dxf
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.eio
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.end
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.err
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.eso
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.mdd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.mtd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.rdd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.rvaudit
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100.shd
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100Table.csv
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100Table.htm
 TestModel[CS_JPN Rijal[CA_80[CM_0[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Sapporo_RCP85-2100Zsz.csv
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_Present.audit
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_Present.bnd
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_~Naha_Present.csv
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_

(continues on next page)

(continued from previous page)

```

↳ Naha_Present.dxf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.eio
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.end
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.err
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.eso
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.mdd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.mtd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.rdd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.rvaudit
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_Present.shd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_PresentTable.csv
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_PresentTable.htm
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_PresentZsz.csv
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.audit
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.bnd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.csv
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.dxf
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.eio
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.end
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.err
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.eso
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.mdd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.mtd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.rdd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.rvaudit
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Naha_RCP85-2100.shd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_

```

(continues on next page)

(continued from previous page)

↵ Naha_RCP85-2100Table.csv
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100Table.htm
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Naha_RCP85-2100Zsz.csv
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.audit
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.bnd
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.csv
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.dxf
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.eio
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.end
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.err
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.eso
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.mdd
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.mtd
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.rdd
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.rvaudit
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_Present.shd
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_PresentTable.csv
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_PresentTable.htm
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_PresentZsz.csv
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_RCP85-2100.audit
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_RCP85-2100.bnd
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_RCP85-2100.csv
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_RCP85-2100.dxf
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_RCP85-2100.eio
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_RCP85-2100.end
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
 ↵ Sapporo_RCP85-2100.err
 TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_]

(continues on next page)

(continued from previous page)

```
↳ Sapporo_RCP85-2100.eso
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Sapporo_RCP85-2100.mdd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Sapporo_RCP85-2100.mtd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Sapporo_RCP85-2100.rdd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Sapporo_RCP85-2100.rvaudit
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Sapporo_RCP85-2100.shd
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Sapporo_RCP85-2100Table.csv
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Sapporo_RCP85-2100Table.htm
TestModel[CS_JPN Rijal[CA_80[CM_3[HM_2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[Japan_
↳ Sapporo_RCP85-2100Zsz.csv
```

```
[26]: for i in files_to_delete:
      remove(i)
```

USING RENAME_EPW_FILES() TO RENAME THE EPWS FOR PROPER DATA ANALYSIS AFTER SIMULATION

`rename_epw_files` function will rename your EPW files following the naming convention “Country_City_RCPscenario-Year”. It will get the Country and City fields from EPW coordinates, and the RCPscenario and Year fields from the original name. If there is no reference to this in the original name, it will consider these to be at Present scenario.

usage: `rename_epw_files(filelist=list_of_files_to_rename, # if omitted, it will rename all EPWs in that folder confirm_renaming=True or False, #to skip renaming confirmation on prompt command or console confirm_deletion=True or False #to skip deletion confirmation on prompt command or console match_cities: True or False. Default is False. It's computationally very expensive.)`

First of all, let's see what files we do have in the folder:

```
[1]: import os
os.listdir()

[1]: ['ipynb_checkpoints',
      'GBR_Aberdeen.Dyce.030910_IWEC.epw',
      'GBR_London.Gatwick.037760_IWEC.epw',
      'RCP26_2100_GC03_Ponta_Grossa.epw',
      'using_rename_epw_files.ipynb',
      '__init__.py']
```

You can see there are 3 EPW files, which are:

```
[2]: old_epws = [i for i in os.listdir() if i.endswith('.epw')]
print(old_epws)

['GBR_Aberdeen.Dyce.030910_IWEC.epw', 'GBR_London.Gatwick.037760_IWEC.epw', 'RCP26_2100_
 ↴GC03_Ponta_Grossa.epw']
```

So let's rename them. When we call the function, you will be asked to enter the IDs of the EPW names which are not correct, if there are any. If you enter one or multiple IDs, you will be asked if you want to rename them manually (i.e. typing the correct new name) or not. If you say ‘n’, possible names will be searched by using the geolocation, which can be computationally expensive if there are a moderate number of EPWs (e.g. 30).

```
[3]: from accim.data.data_preprocessing import rename_epw_files
rename_epw_files(
    confirm_deletion=False,
)
```

```
C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\data_
↳ preprocessing.py:136: FutureWarning: The default value of regex will change from True
↳ to False in a future version.
    epw_df['EPW_names'] = epw_df['EPW_file_names'].str.replace('.epw', '')
C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\data_
↳ preprocessing.py:138: FutureWarning: The default value of regex will change from True
↳ to False in a future version. In addition, single character regular expressions
↳ will *not* be treated as literal strings when regex=True.
    epw_df['EPW_mod'] = epw_df['EPW_names'].str.replace('-', '_').str.replace('.', '_')
    .str.split('_')
C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\data_
↳ preprocessing.py:140: FutureWarning: The default value of regex will change from True
↳ to False in a future version. In addition, single character regular expressions
↳ will *not* be treated as literal strings when regex=True.
    epw_df['EPW_mod_filtered'] = epw_df['EPW_names'].str.replace('-', '_').str.replace('.', '_')
    .str.split('_')
```

Since no match has been found between RCP scenarios and EPW file name, Present scenario
↳ has been assigned to the following EPW files:

GBR_Aberdeen.Dyce.030910_IWEC.epw
 GBR_London.Gatwick.037760_IWEC.epw

Since no match has been found between RCP scenario Year and EPW file name, Present year
↳ has been assigned to the following EPW files:

GBR_Aberdeen.Dyce.030910_IWEC.epw
 GBR_London.Gatwick.037760_IWEC.epw

The geolocation process has taken: 1.26 seconds (0.42 s/EPW)

The previous and new names of the EPW files and their unique IDs are:

ID: 0 / GBR_Aberdeen.Dyce.030910_IWEC / United-Kingdom_Aberdeen_Present
 ID: 1 / GBR_London.Gatwick.037760_IWEC / United-Kingdom_UNKNOWN_Present
 ID: 2 / RCP26_2100_GC03_Ponta_Grossa / Brazil_Ponta_RCP26-2100

"UNKNOWN" city or subcountry have been found in the renamed EPW files, therefore these
↳ need to be amended in the next stage.

If any of the city or subcountry names needs some amendment (if you are not happy with
↳ any of the available options, you can exclude it from renaming at the next stage),
↳ please enter the EPW IDs separated by space; otherwise, hit enter to omit:1 2

Regarding the file ID: 1 / old name: GBR_London.Gatwick.037760_IWEC / new name: United-
↳ Kingdom_UNKNOWN_Present, the address obtained from coordinates is:

London Gatwick Airport, Old Brighton Road South, Lowfield Heath, Manor Royal, Crawley,
↳ West Sussex, GB-WSX, England, GB-ENG, RH11 0PT, United Kingdom

Please enter the amended city or subcountry, which must be unique: London

Regarding the file ID: 2 / old name: RCP26_2100_GC03_Ponta_Grossa / new name: Brazil_-
↳ Ponta_RCP26-2100, the address obtained from coordinates is:

Praca Barao de Guarauna, Centro, Centro, Ponta Grossa, Regiao Geografica Imediata de
↳ Ponta Grossa, Regiao Geografica Intermediaria de Ponta Grossa, Parana, BR-PR, Regiao
↳ Sul, 84010-050, Brasil

Please enter the amended city or subcountry, which must be unique: Ponta Grossa

The previous and new names of the EPW files after city or subcountry name amendments and

(continues on next page)

(continued from previous page)

→their unique IDs are:
ID: 1 / GBR_London.Gatwick.037760_IWEC / United-Kingdom_London_Present
ID: 2 / RCP26_2100_GC03_Ponta_Grossa / Brazil_Ponta-Grossa_RCP26-2100

The final list of previous and new names of the EPW files and their unique IDs is:

ID: 0 / GBR_Aberdeen.Dyce.030910_IWEC / United-Kingdom_Aberdeen_Present
ID: 1 / GBR_London.Gatwick.037760_IWEC / United-Kingdom_London_Present
ID: 2 / RCP26_2100_GC03_Ponta_Grossa / Brazil_Ponta-Grossa_RCP26-2100

If you want to exclude some EPWs from renaming, please enter the IDs separated by space, →otherwise, hit enter to continue:

Do you want to copy and rename the file or files? [y/n]:y

The file GBR_Aberdeen.Dyce.030910_IWEC has been renamed to United-Kingdom_Aberdeen_Present

The file GBR_London.Gatwick.037760_IWEC has been renamed to United-Kingdom_London_Present
The file RCP26_2100_GC03_Ponta_Grossa has been renamed to Brazil_Ponta-Grossa_RCP26-2100

[3]: <accim.data.data_preprocessing.rename_epw_files at 0x1a3e60dafa0>

You can see above that there was no reference to RCP scenarios in the original EPW file name in 2 of the instances, therefore these has been considered as Present scenario. The same applies to the Year field. Finally, states the previous and the new names of the EPWs. So, now, let's see what files we do have in the folder.

[4]: os.listdir()

[4]: ['ipynb_checkpoints',
'Brazil_Ponta-Grossa_RCP26-2100.epw',
'GBR_Aberdeen.Dyce.030910_IWEC.epw',
'GBR_London.Gatwick.037760_IWEC.epw',
'RCP26_2100_GC03_Ponta_Grossa.epw',
'United-Kingdom_Aberdeen_Present.epw',
'United-Kingdom_London_Present.epw',
'using_rename_epw_files.ipynb',
'__init__.py']

We can see the new EPWs are:

[5]: new_epws = [i for i in os.listdir() if not(any(i in j for j in old_epws)) and i.endswith('.epw')]
print(new_epws)
['Brazil_Ponta-Grossa_RCP26-2100.epw', 'United-Kingdom_Aberdeen_Present.epw', 'United-Kingdom_London_Present.epw']

Let's delete the new files so that we can run the notebook again.

[6]: for i in new_epws:
os.remove(i)

USING RUNEP() TO DIRECTLY RUN SIMULATIONS WITH ENERGYPLUS

This script has been created by eppy's development team (<https://eppy.readthedocs.io/en/latest/runningeplus.html>, specifically from section 'Running in parallel processes using Generators'), however I did some changes. Anyway, you probably should check out eppy package, since it's absolutely awesome.

This script will take all EPWs and IDFs located in the folder where the script is being executed as input files, and will run simulations based on these, naming the output files with the following naming convention: "IDFname[EPWname]"

First of all, let's see what files we do have in the folder:

```
[1]: import os
original_files = [i for i in os.listdir()]
print(*original_files, sep='\n')

.ipynb_checkpoints
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
United-Kingdom_Aberdeen_Present.epw
using_runEp.ipynb
__init__.py
```

As you can see, the EPW file or files we have is or are:

```
[2]: print([i for i in original_files if i.endswith('.epw')])
['United-Kingdom_Aberdeen_Present.epw']
```

And the IDF file or files we have is or are:

```
[3]: print([i for i in original_files if i.endswith('.idf')])
['TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf']]
```

Based on that, let's use runEp():

```
[4]: from accim.run import run
run.runEp(
    runOnlyAccim=True, #only runs output IDFs, that is, IDFs with "[" in its name
    confirmRun=True, #to skip confirmation
    num_CPUs=2, #to specify the number of CPUs to be used
    EnergyPlus_version='23.1', #to specify the EnergyPlus version of the IDF, and the
    →version of EnergyPlus you are going to run
)
```

The IDFs we are going to run are:
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X.idf
and the No. of IDFs is going to be 1
The sample_EPWs we are going to run are:
United-Kingdom_Aberdeen_Present.epw
and the No. of sample_EPWs is going to be 1
Therefore, the simulations are going to be:
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present
and the No. of simulations is going to be 1

runEp() just informed us about the input files, and the number of simulation runs and name of each one. So now, let's see what are the outputs:

```
[5]: new_files = [i for i in os.listdir() if not(any(i in j for j in original_files))]
print(*new_files, sep='\n')

TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.audit
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.bnd
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.dxf
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.eio
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.end
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.err
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.eso
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.mdd
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.mtd
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.rdd
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.rvaudit
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.shd
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_PresentTable.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_PresentTable.htm
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_
→2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_PresentZsz.csv
```

The CSV files we have generated follow the naming convention ‘IDF|EPW’ and therefore, are suitable to be used with the class Table, within the accim.data.datawrangling module.

```
[6]: print(*[i for i in os.listdir() if i.endswith('.csv') and not(i.endswith('Table.csv'))  
and not (i.endswith('Zsz.csv'))], sep='\n')
```

```
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2310[CS_INT ASHRAE55[CA_80[CM_3[HM_  
2[VC_0[VO_0.0[MT_50.0[MW_50.0[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv
```

Now, let's remove all output files of the simulation run:

```
[7]: for i in new_files:  
    os.remove(i)
```

```
[ ]:
```


USING THE CLASS TABLE() FOR DATA ANALYSIS

Say we have previously renamed the EPW files with `rename_epw_files()`, and the EPWs we currently have are:

- United-Kingdom_Aberdeen_Present
- United-Kingdom_London_Present

We also have transformed the IDF file `TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220.idf`, and the IDFs we currently have are:

- `TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_0[HM_2[VC_0[VO_0[MT_5`
- `TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_3[HM_2[VC_0[VO_0[MT_5`
- `TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_0[HM_2[VC_0[VO_0[MT_5`
- `TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_3[HM_2[VC_0[VO_0[MT_5`
- `TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_0[HM_2[VC_0[VO_0[MT_5`
- `TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_3[HM_2[VC_0[VO_0[MT_5`

Next, we simulate these files with `runEp()`, so the CSVs we get are:

```
[1]: import os
print(*[i for i in os.listdir() if i.endswith('.csv') and 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220' in i], sep='\n')

TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv]
```

(continues on next page)

(continued from previous page)

```

→ 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_3[HM_
→ 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv
TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_3[HM_
→ 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv

```

So now, let's use the class Table() to analyse the data.

12.1 1. Getting the full dataset of simulation results

```
[2]: from accim.data.data_postprocessing import Table
dataset_hourly = Table(
    #datasets=list #Since we are not specifying any list, it will use all available CSVs,
    #in the folder
    source_frequency='hourly', # This lets accim know which is the frequency of the
    #input CSVs. Input CSVs with multiple frequencies are also allowed. It can be 'hourly',
    #'daily', 'monthly' and 'runperiod'. It can also be 'timestep' but might generate
    #errors.
    frequency='hourly', # If 'daily', accim will aggregate the rows in days. It can be
    #'hourly', 'daily', 'monthly' and 'runperiod'. It can also be 'timestep' but might
    #generate errors.
    frequency_agg_func='sum', #this makes the sum or average when aggregating in days,
    #months or runperiod; since the original CSV frequency is in hour, it won't make any
    #affect
    standard_outputs=True,
    level=['building'], # A list containing the strings 'block' and/or 'building'. For
    #instance, if ['block', 'building'], accim will generate new columns to sum up or
    #average in blocks and building level.
    level_agg_func=['sum', 'mean'], # A list containing the strings 'sum' and/or 'mean'.
    #For instance, if ['sum', 'mean'], accim will generate the new columns explained in the
    #level argument by summing and averaging.
    level_excluded_zones=[],
    split_epw_names=True, #to split EPW names based on the pattern Country_City_
    #RCPscenario-Year
)
```

Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_1[CM_0[HM_2[VC_0[VO_0[MT_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
→ Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_1[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
→ Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_1[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
→ Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_1[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
→ Present.csv is hourly, therefore no aggregation will be performed.

Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_2[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
→ Present.csv is hourly, therefore no aggregation will be performed.

(continues on next page)

(continued from previous page)

```

Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_2[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
→ Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_2[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
→ Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_2[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
→ Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_3[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
→ Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_3[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
→ Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_3[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
→ Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
→ INT EN16798[CA_3[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
→ Present.csv is hourly, therefore no aggregation will be performed.
No zones have been excluded from level computations.
All CSVs are for present scenario.

```

A Table object has been created, which includes a number of accessible variables. The most important is the Pandas Dataframe created from the CSVs, which can be accessed with .df (in this case, dataset_hourly.df)

[3]: dataset_hourly.df

	Source \
0	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
1	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
2	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
3	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
4	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
...	...
105115	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
105116	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
105117	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
105118	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
105119	TestModel_onlyGeometryForVRFsystem_2zones_Calc...
	Model \ ComfStand
0	TestModel_onlyGeometryForVRFsystem_2zones_Calc... CS_INT EN16798
1	TestModel_onlyGeometryForVRFsystem_2zones_Calc... CS_INT EN16798
2	TestModel_onlyGeometryForVRFsystem_2zones_Calc... CS_INT EN16798
3	TestModel_onlyGeometryForVRFsystem_2zones_Calc... CS_INT EN16798
4	TestModel_onlyGeometryForVRFsystem_2zones_Calc... CS_INT EN16798
...
105115	TestModel_onlyGeometryForVRFsystem_2zones_Calc... CS_INT EN16798
105116	TestModel_onlyGeometryForVRFsystem_2zones_Calc... CS_INT EN16798
105117	TestModel_onlyGeometryForVRFsystem_2zones_Calc... CS_INT EN16798
105118	TestModel_onlyGeometryForVRFsystem_2zones_Calc... CS_INT EN16798

(continues on next page)

(continued from previous page)

105119		0.012108
Building_Total_Total Energy Consumption (kWh/m2) (mean) \		
0		0.007644
1		0.008654
2		0.009347
3		0.009770
4		0.009982
...		...
105115		0.001985
105116		0.002093
105117		0.002273
105118		0.002643
105119		0.003027
Building_Total_Zone Air Volume (m3) (summed) \		
0		232.0401
1		232.0401
2		232.0401
3		232.0401
4		232.0401
...		...
105115		232.0401
105116		232.0401
105117		232.0401
105118		232.0401
105119		232.0401
Building_Total_Zone Floor Area (m2) (summed) \		
0		66.2971
1		66.2971
2		66.2971
3		66.2971
4		66.2971
...		...
105115		66.2971
105116		66.2971
105117		66.2971
105118		66.2971
105119		66.2971
Building_Total_PMV (summed) Building_Total_PMV (mean) \		
0	0.0	Nan
1	0.0	Nan
2	0.0	Nan
3	0.0	Nan
4	0.0	Nan
...
105115	0.0	Nan
105116	0.0	Nan
105117	0.0	Nan
105118	0.0	Nan

(continues on next page)

(continued from previous page)

105119	0.0	NaN
Building_Total_PPD (%) (summed) Building_Total_PPD (%) (mean)		
0	0.0	NaN
1	0.0	NaN
2	0.0	NaN
3	0.0	NaN
4	0.0	NaN
...
105115	0.0	NaN
105116	0.0	NaN
105117	0.0	NaN
105118	0.0	NaN
105119	0.0	NaN
[105120 rows x 119 columns]		

[4]: dataset_hourly.df.shape

[4]: (105120, 119)

Also, you can see the block list that has been extracted from the zones in the CSV columns, as well as the zones with heating or cooling energy consumption, and the occupied zones:

[5]: dataset_hourly.block_list

[5]: ['BLOCK1']

[6]: dataset_hourly.hvac_zone_list

[6]: ['BLOCK1:ZONE2', 'BLOCK1:ZONE1']

[7]: dataset_hourly.occupied_zone_list

[7]: ['BLOCK1:ZONE2', 'BLOCK1:ZONE1']

In case of lots of CSVs, using the Table class would mean gathering the CSVs to make the Dataframe each time, and it might be very time-consuming. Therefore, the CSVs can be concatenated into a single CSV and store in the same folder with the argument concatenated_csv_name. Afterwards, that concatenated CSV can be read using the argument source_concatenated_csv_filepath each time a Table class object is created.

```
[8]: concatenated_csvname = 'notebook_example'
dataset_hourly_indirect = Table(
    #datasets=list #Since we are not specifying any list, it will use all available CSVs,
    #in the folder
    source_frequency='hourly', # This lets accim know which is the frequency of the
    #input CSVs. Input CSVs with multiple frequencies are also allowed. It can be 'hourly',
    #'daily', 'monthly' and 'runperiod'. It can also be 'timestep' but might generate
    #errors.
    frequency='hourly', # If 'daily', accim will aggregate the rows in days. It can be
    #'hourly', 'daily', 'monthly' and 'runperiod'. It can also be 'timestep' but might
    #generate errors.
    frequency_agg_func='sum', #this makes the sum or average when aggregating in days,
```

(continues on next page)

(continued from previous page)

```

months or runperiod; since the original CSV frequency is in hour, it won't make any
effect
    standard_outputs=True,
    concatenated_csv_name=concatenated_csvname #Useful when working with large datasets.
    It saves the output dataset to a CSV file, so you don't need to re-do some work.
    Afterwards, it can be imported with source_concatenated_csv_filepath argument.
    # All other arguments won't have any effect on the output. This is only to store the
    csv at early stage. Datawrangling will be performed later.

)

Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_1[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_1[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_1[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_1[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_2[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_2[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_2[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_2[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_3[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_3[CM_0[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_3[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_
    Present.csv is hourly, therefore no aggregation will be performed.
Input data frequency in file TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_
    INT EN16798[CA_3[CM_3[HM_2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_
    Present.csv is hourly, therefore no aggregation will be performed.

```

We just generated a CSV, whose name begins with ‘notebook example’, specifies the input data and ends with ‘CSV-concatenated’. If there is some row with NaN, it will be generated another CSV reporting this, but ending with ‘Rows_with_NaNs’. We should be able to see it below:

```
[9]: import os
os.listdir()

[9]: ['.ipynb_checkpoints',
 'notebook_example[srcfreq-hourly[freq-hourly[frequency_agg_func-sum[standard_outputs-
 True[CSVconcatenated.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_0[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_0[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_3[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_1[CM_3[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_0[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_0[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_3[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_2[CM_3[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_0[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_0[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_3[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_Aberdeen_Present.csv',
 'TestModel_onlyGeometryForVRFsystem_2zones_CalcVent_V2220[CS_INT EN16798[CA_3[CM_3[HM_-
 2[VC_0[VO_0[MT_50[MW_50[AT_0.1[NS_X[United-Kingdom_London_Present.csv',
 'using_Table.ipynb',
 '__init__.py']]
```

Now, let's continue working with the CSVconcatenated we have generated:

```
[10]: dataset_hourly_indirect = Table(
    source_concatenated_csv_filepath='notebook_example[srcfreq-hourly[freq-
 hourly[frequency_agg_func-sum[standard_outputs-True[CSVconcatenated.csv'],
    # source_frequency='hourly', # The new source frequency is the previous already_
    # computed frequency, since we have already aggregated rows based on the previous_
    # frequency. Therefore, there is no need to specify this argument.
    # 3 arguments below have been previously specified, and in fact, are stored at the_
    # concatenated CSV filename. So you don't need to specify these again.
    # frequency='hourly',
    # frequency_agg_func='sum',
    # standard_outputs=True,
    level=['building'],
    level_agg_func=['sum', 'mean'],
    level_excluded_zones=[],
    split_epw_names=True, #to split EPW names based on the format Country_City_
    RCPscenario-YEar
)
```

No zones have been excluded from level computations.
All CSVs are for present scenario.

[11]: dataset_hourly_indirect.df.shape

[11]: (105120, 119)

You can see we have obtained the same full dataset. Now, let's delete the concatenated CSV file to save some space.

```
[12]: for i in os.listdir():
    if concatenated_csvname in i:
        os.remove(i)
```

12.2 2. Filtering results

We have used any of the 2 methods to obtain the full dataset of simulation results.

At this point, we should have decided what table or graph we intend to make, and therefore we should know what data we are going to show in it. In this case, we want to make a figure with different subplots, showing the different ComfMod values on the rows and the different EPW_City_or_subcountry values on the columns, and in each subplot, a scatter plot with the indoor operative temperature and adaptive heating and cooling setpoints on the y main axis (left spine), and on the secondary y axis, 2 different spines, one of them to show the heating and cooling energy demand, and the other one the air changes. However, let's say we just want to see the subplots for Category CA_3.

[13]: print(*dataset_hourly.df.columns, sep='\n')

```
Source
Model
ComfStand
CAT
ComfMod
HVACmode
VentCtrl
VSToffset
MinOToffset
MaxWindSpeed
ASTtol
NameSuffix
EPW
Count
EPW_Country_name
EPW_City_or_subcountry
EPW_Scenario-Year
EPW_Scenario
EPW_Year
Date/time
Month/day
Month
Day
Hour
Minute
Second
```

(continues on next page)

(continued from previous page)

Site Outdoor Air Drybulb Temperature (°C)
 Site Outdoor Air Relative Humidity (%)
 Site Wind Speed (m/s)
 Comfort Temperature (°C)
 Adaptive Cooling Setpoint Temperature (°C)
 Adaptive Heating Setpoint Temperature (°C)
 Adaptive Cooling Setpoint Temperature_No Tolerance (°C)
 Adaptive Heating Setpoint Temperature_No Tolerance (°C)
 Ventilation Setpoint Temperature (°C)
 Minimum Outdoor Temperature for ventilation (°C)
 BLOCK1:ZONE2_Comfortable Hours_No Applicability (h)
 BLOCK1:ZONE1_Comfortable Hours_No Applicability (h)
 BLOCK1:ZONE2_Comfortable Hours_Applicability (h)
 BLOCK1:ZONE1_Comfortable Hours_Applicability (h)
 BLOCK1:ZONE2_Discomfortable Applicable Hot Hours (h)
 BLOCK1:ZONE1_Discomfortable Applicable Hot Hours (h)
 BLOCK1:ZONE2_Discomfortable Applicable Cold Hours (h)
 BLOCK1:ZONE1_Discomfortable Applicable Cold Hours (h)
 BLOCK1:ZONE2_Discomfortable Non Applicable Hot Hours (h)
 BLOCK1:ZONE1_Discomfortable Non Applicable Hot Hours (h)
 BLOCK1:ZONE2_Discomfortable Non Applicable Cold Hours (h)
 BLOCK1:ZONE1_Discomfortable Non Applicable Cold Hours (h)
 BLOCK1:ZONE2_Zone Floor Area (m²)
 BLOCK1:ZONE1_Zone Floor Area (m²)
 BLOCK1:ZONE2_Zone Air Volume (m³)
 BLOCK1:ZONE1_Zone Air Volume (m³)
 BLOCK1:ZONE2_Ventilation Hours (h)
 BLOCK1:ZONE1_Ventilation Hours (h)
 BLOCK1:ZONE2_Zone Operative Temperature (°C)
 BLOCK1:ZONE1_Zone Operative Temperature (°C)
 BLOCK1:ZONE2_AFN Zone Infiltration Volume (m³)
 BLOCK1:ZONE2_AFN Zone Infiltration Air Change Rate (ach)
 BLOCK1:ZONE1_AFN Zone Infiltration Volume (m³)
 BLOCK1:ZONE1_AFN Zone Infiltration Air Change Rate (ach)
 Whole Building Facility Total HVAC Electricity Demand Rate (kWh/m²)
 BLOCK1:ZONE2_ASHRAE 55 Running mean outdoor temperature (°C)
 BLOCK1:ZONE2_EN16798-1 Running mean outdoor temperature (°C)
 BLOCK1:ZONE1_ASHRAE 55 Running mean outdoor temperature (°C)
 BLOCK1:ZONE1_EN16798-1 Running mean outdoor temperature (°C)
 BLOCK1:ZONE2_Cooling Energy Demand (kWh/m²)
 BLOCK1:ZONE1_Cooling Energy Demand (kWh/m²)
 BLOCK1:ZONE2_Heating Energy Demand (kWh/m²)
 BLOCK1:ZONE1_Heating Energy Demand (kWh/m²)
 BLOCK1:ZONE2_Cooling Energy Consumption (kWh/m²)
 BLOCK1:ZONE2_Heating Energy Consumption (kWh/m²)
 BLOCK1:ZONE1_Cooling Energy Consumption (kWh/m²)
 BLOCK1:ZONE1_Heating Energy Consumption (kWh/m²)
 BLOCK1:ZONE2_Total Energy Demand (kWh/m²) (summed)
 BLOCK1:ZONE1_Total Energy Demand (kWh/m²) (summed)
 BLOCK1:ZONE2_Total Energy Consumption (kWh/m²) (summed)
 BLOCK1:ZONE1_Total Energy Consumption (kWh/m²) (summed)
 Building_Total_Zone Operative Temperature (°C) (summed)

(continues on next page)

(continued from previous page)

Building_Total_Zone Operative Temperature (°C) (mean)
 Building_Total_Comfortable Hours_No Applicability (h) (summed)
 Building_Total_Comfortable Hours_No Applicability (h) (mean)
 Building_Total_Comfortable Hours_Applicability (h) (summed)
 Building_Total_Comfortable Hours_Applicability (h) (mean)
 Building_Total_Discomfortable Applicable Hot Hours (h) (summed)
 Building_Total_Discomfortable Applicable Hot Hours (h) (mean)
 Building_Total_Discomfortable Applicable Cold Hours (h) (summed)
 Building_Total_Discomfortable Applicable Cold Hours (h) (mean)
 Building_Total_Discomfortable Non Applicable Hot Hours (h) (summed)
 Building_Total_Discomfortable Non Applicable Hot Hours (h) (mean)
 Building_Total_Discomfortable Non Applicable Cold Hours (h) (summed)
 Building_Total_Discomfortable Non Applicable Cold Hours (h) (mean)
 Building_Total_Ventilation Hours (h) (summed)
 Building_Total_Ventilation Hours (h) (mean)
 Building_Total_AFN Zone Infiltration Volume (m³) (summed)
 Building_Total_AFN Zone Infiltration Volume (m³) (mean)
 Building_Total_AFN Zone Infiltration Air Change Rate (ach) (summed)
 Building_Total_AFN Zone Infiltration Air Change Rate (ach) (mean)
 Building_Total_AFN Zone Ventilation Volume (m³) (summed)
 Building_Total_AFN Zone Ventilation Volume (m³) (mean)
 Building_Total_AFN Zone Ventilation Air Change Rate (ach) (summed)
 Building_Total_AFN Zone Ventilation Air Change Rate (ach) (mean)
 Building_Total_Cooling Energy Demand (kWh/m²) (summed)
 Building_Total_Cooling Energy Demand (kWh/m²) (mean)
 Building_Total_Heating Energy Demand (kWh/m²) (summed)
 Building_Total_Heating Energy Demand (kWh/m²) (mean)
 Building_Total_Cooling Energy Consumption (kWh/m²) (summed)
 Building_Total_Cooling Energy Consumption (kWh/m²) (mean)
 Building_Total_Heating Energy Consumption (kWh/m²) (summed)
 Building_Total_Heating Energy Consumption (kWh/m²) (mean)
 Building_Total_Total Energy Demand (kWh/m²) (summed)
 Building_Total_Total Energy Demand (kWh/m²) (mean)
 Building_Total_Total Energy Consumption (kWh/m²) (summed)
 Building_Total_Total Energy Consumption (kWh/m²) (mean)
 Building_Total_Zone Air Volume (m³) (summed)
 Building_Total_Zone Floor Area (m²) (summed)
 Building_Total_PMV (summed)
 Building_Total_PMV (mean)
 Building_Total_PPD (%) (summed)
 Building_Total_PPD (%) (mean)

Now, let's extract the data we are really interested in. For this, we can use the method called `format_table()`:

```
[14]: dataset_hourly.format_table(
    type_of_table='custom', # Used to choose some predefined tables. It can be 'energy_
    ↵ demand', 'comfort hours', 'temperature', 'all' or 'custom'
    custom_cols=[ #if type_of_table is 'custom', custom_cols is used to filter the_
    ↵ desired columns to keep
        'Adaptive Cooling Setpoint Temperature_No Tolerance (°C)',  

        'Adaptive Heating Setpoint Temperature_No Tolerance (°C)',  

        'Building_Total_Zone Operative Temperature (°C) (mean)',
```

(continues on next page)

(continued from previous page)

```
'BLOCK1:ZONE2_EN16798-1 Running mean outdoor temperature (°C)',  
'Building_Total_Cooling Energy Demand (kWh/m2) (summed)',  
'Building_Total_Heating Energy Demand (kWh/m2) (summed)',  
'Building_Total_AFN Zone Infiltration Air Change Rate (ach) (summed)'  
]  
)
```

So, we have already filtered the columns we wanted. Let's look at the result:

[15]: dataset_hourly.df

	Date/time	Model						
0	01/01 01:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
1	01/01 02:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
2	01/01 03:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
3	01/01 04:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
4	01/01 05:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
...						
105115	12/31 20:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
105116	12/31 21:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
105117	12/31 22:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
105118	12/31 23:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
105119	12/31 24:00:00	TestModel_onlyGeometryForVRFsystem_2zones_Calc...						
	ComfStand	CAT	ComfMod	HVACmode	VentCtrl	VSToffset	MinOToffset	\
0	CS_INT	EN16798	CA_1	CM_0	HM_2	VC_0	VO_0	MT_50
1	CS_INT	EN16798	CA_1	CM_0	HM_2	VC_0	VO_0	MT_50
2	CS_INT	EN16798	CA_1	CM_0	HM_2	VC_0	VO_0	MT_50
3	CS_INT	EN16798	CA_1	CM_0	HM_2	VC_0	VO_0	MT_50
4	CS_INT	EN16798	CA_1	CM_0	HM_2	VC_0	VO_0	MT_50
...
105115	CS_INT	EN16798	CA_3	CM_3	HM_2	VC_0	VO_0	MT_50
105116	CS_INT	EN16798	CA_3	CM_3	HM_2	VC_0	VO_0	MT_50
105117	CS_INT	EN16798	CA_3	CM_3	HM_2	VC_0	VO_0	MT_50
105118	CS_INT	EN16798	CA_3	CM_3	HM_2	VC_0	VO_0	MT_50
105119	CS_INT	EN16798	CA_3	CM_3	HM_2	VC_0	VO_0	MT_50
	MaxWindSpeed	...	EPW_Scenario-Year	EPW_Scenario	EPW_Year	\		
0	MW_50	...	Present	Present	Present			
1	MW_50	...	Present	Present	Present			
2	MW_50	...	Present	Present	Present			
3	MW_50	...	Present	Present	Present			
4	MW_50	...	Present	Present	Present			
...		
105115	MW_50	...	Present	Present	Present			
105116	MW_50	...	Present	Present	Present			
105117	MW_50	...	Present	Present	Present			
105118	MW_50	...	Present	Present	Present			
105119	MW_50	...	Present	Present	Present			
	Adaptive Cooling Setpoint Temperature_No Tolerance (°C)	\						
0	25.0							
1	25.0							

(continues on next page)

(continued from previous page)

2	25.0
3	25.0
4	25.0
...	...
105115	26.1
105116	26.1
105117	26.1
105118	26.1
105119	26.1

	Adaptive Heating Setpoint Temperature_No Tolerance (°C) \
0	21.0
1	21.0
2	21.0
3	21.0
4	21.0
...	...
105115	17.1
105116	17.1
105117	17.1
105118	17.1
105119	17.1

	Building_Total_Zone Operative Temperature (°C) (mean) \
0	21.075965
1	21.060730
2	21.061641
3	21.068896
4	21.078253
...	...
105115	17.171267
105116	17.176058
105117	17.178160
105118	17.176237
105119	17.170725

	BLOCK1:ZONE2_EN16798-1 Running mean outdoor temperature (°C) \
0	-1.071071
1	-1.071071
2	-1.071071
3	-1.071071
4	-1.071071
...	...
105115	7.543148
105116	7.543148
105117	7.543148
105118	7.543148
105119	7.543148

	Building_Total_Cooling Energy Demand (kWh/m2) (summed) \
0	0.0
1	0.0

(continues on next page)

(continued from previous page)

2		0.0
3		0.0
4		0.0
...		...
105115		0.0
105116		0.0
105117		0.0
105118		0.0
105119		0.0
Building_Total_Heating Energy Demand (kWh/m2) (summed) \		
0		0.036738
1		0.040240
2		0.042701
3		0.044282
4		0.045109
...		...
105115		0.011349
105116		0.011955
105117		0.012974
105118		0.015072
105119		0.017182
Building_Total_AFN Zone Infiltration Air Change Rate (ach) (summed)		
0		0.217716
1		0.229646
2		0.238609
3		0.242288
4		0.243261
...		...
105115		0.180364
105116		0.175023
105117		0.191868
105118		0.255266
105119		0.281407
[105120 rows x 29 columns]		

As you can see, the number of columns have been reduced to 29, which are 22 columns for the index and categorical data, and the 7 columns we filtered.

12.3 3. Making graphs

We will make the figure with scatter_plot(), however it needs some input data we will previously generate with generate_fig_data().

12.3.1 3.1 Scatter plot:

As said above, we first need to generate the dataset lists with generate_fig_data():

And finally we can plot the figure:

```
[16]: dataset_hourly.scatter_plot(
    vars_to_gather_rows=['ComfMod', 'CAT'], # variables to gather in rows of subplots
    vars_to_gather_cols=['EPW_City_or_subcountry'],# variables to gather in columns of subplots
    detailed_rows=['CM_0[CA_3', 'CM_3[CA_3'], #we only want to see those combinations
    data_on_x_axis='BLOCK1:ZONE2_EN16798-1 Running mean outdoor temperature (°C)', #column name (string) for the data on x axis
    data_on_y_main_axis=[ # similarly to above, a list including the name of the secondary y-axis and the column names you want to plot in it
        [
            'Energy (kWh/m2)',
            [
                'Building_Total_Cooling Energy Demand (kWh/m2) (summed)',
                'Building_Total_Heating Energy Demand (kWh/m2) (summed)'
            ]
        ],
        [
            [
                'Air renovation (ach)',
                [
                    'Building_Total_AFN Zone Infiltration Air Change Rate (ach) (summed)'
                ]
            ],
            [
                'Operative Temperature (°C)',
                [
                    'Adaptive Cooling Setpoint Temperature_No Tolerance (°C)',
                    'Adaptive Heating Setpoint Temperature_No Tolerance (°C)',
                    'Building_Total_Zone Operative Temperature (°C) (mean)'
                ]
            ],
            [
                [
                    'Energy (kWh/m2)',
                    [
                        'cyan',

```

(continues on next page)

(continued from previous page)

```

        'orange',
    ],
],
],
colorlist_y_sec_axis=[
[
    'Air renovation (ach)',
    [
        'yellow'
    ]
],
[
    'Operative Temperature (°C)',
    [
        'b',
        'r',
        'g',
    ]
],
],
),
supxlabel='Running Mean Outdoor Temperature (°C)', # data label on x axis
figname=f'WIP_scatterplot_RMOT',
figsize=6,
ratio_height_to_width=0.33,
confirm_graph=True
)

```

The number of rows and the list of these is going to be:

No. of rows = 2

List of rows:

CM_0[CA_3

CM_3[CA_3

Do you want to rename the rows? [y/n]: y

Please enter the new name for CM_0[CA_3: Static setpoints

Please enter the new name for CM_3[CA_3: Adaptive setpoints

The renamed rows are going to be:

Static setpoints

Adaptive setpoints

The number of columns and the list of these is going to be:

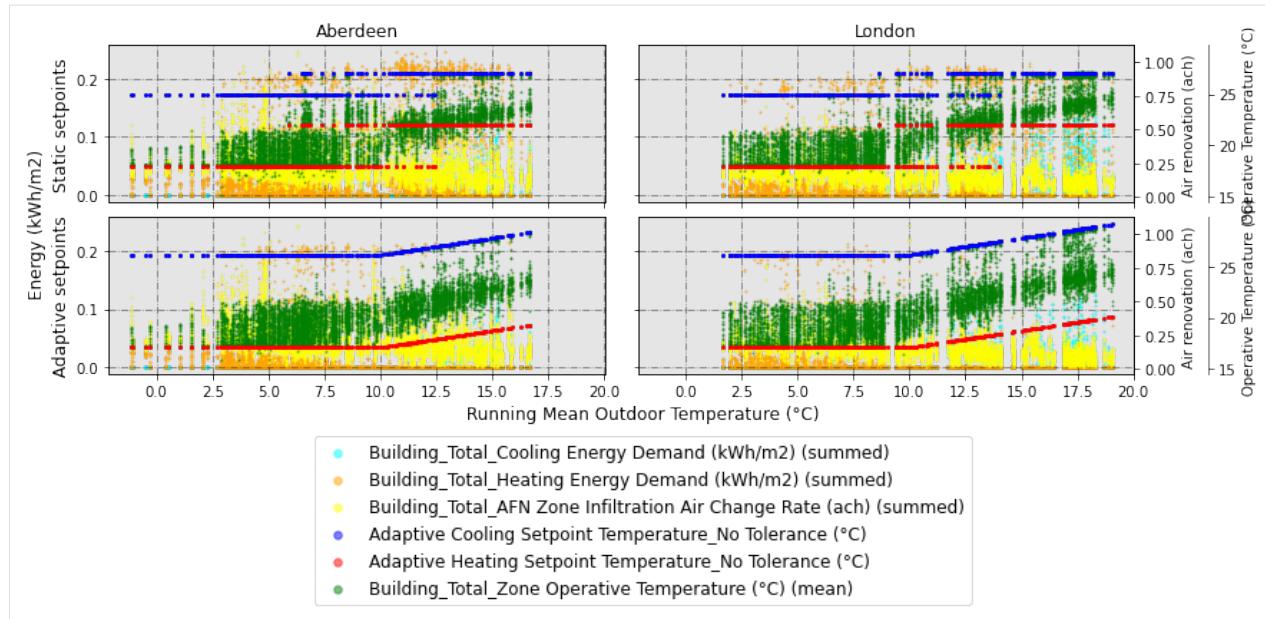
No. of columns = 2

List of columns:

Aberdeen

London

Column names will be the subplot titles. Do you want to rename them? [y/n]: n



12.3.2 3.2 Adaptive vs Static data scatter plot:

A very specific type of scatter plot can be done to show the relationship between data related to adaptive and static setpoint temperatures. In this case, you would need to use the following arguments in generate_fig_data: adap_vs_stat_data_y_main, baseline and colorlist_adap_vs_stat_data; finally, you would need to use the scatter_plot_adap_vs_stat() function.

```
[17]: dataset_hourly.scatter_plot_with_baseline(
    vars_to_gather_rows=['EPW_City_or_subcountry'], #you can enter multiple variables,
    ↪for example: ['EPW_City_or_subcountry', 'EPW_Scenario-Year']
    vars_to_gather_cols=['ComfMod', 'CAT'], #you can enter multiple variables
    data_on_y_axis_baseline_plot=[ # in this case, you only need to specify a list which
    ↪includes the data columns you want to plot
        'Building_Total_Cooling Energy Demand (kWh/m2) (summed)',
        'Building_Total_Heating Energy Demand (kWh/m2) (summed)',
        ],
    baseline='CM_0[CA_1]', # the baseline needs to be in vars_to_gather_cols, and it's
    ↪going to be shown on x axis. Given the variables we have simulated, you can choose
    ↪between 'CM_0[CA_1]', 'CM_0[CA_2]', 'CM_0[CA_3]', 'CM_3[CA_1]', 'CM_3[CA_2' and 'CM_3[CA_3'
    colorlist_baseline_plot_data=[
        'b',
        'r',
        ],
    supxlabel='Static Energy Demand (kWh/m2)',
    supylabel='Adaptive Energy Demand (kWh/m2)',
    figname='WIP_scatterplot_adap_vs_stat',
    figsize=3,
    confirm_graph=True
)
```

The number of rows and the list of these is going to be:

No. of rows = 2

List of rows:

Aberdeen

London

Do you want to rename the rows? [y/n]: n

The number of columns and the list of these is going to be:

No. of columns = 5

List of columns:

CM_0[CA_2

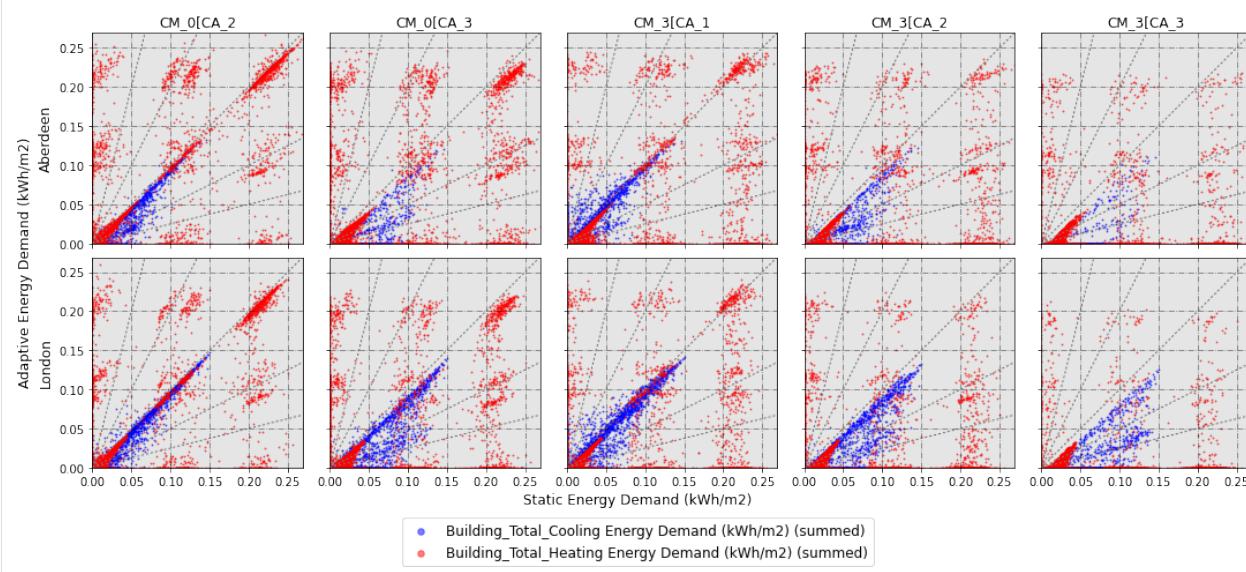
CM_0[CA_3

CM_3[CA_1

CM_3[CA_2

CM_3[CA_3

Column names will be the subplot titles. Do you want to rename them? [y/n]: n



Now, let's delete all the images we have created to save some space:

```
[18]: for i in os.listdir():
    if i.endswith('.png'):
        os.remove(i)
```

12.4 4. Making tables

Now, let's make some tables comparing results. For example, let's make a table to show monthly values of heating and cooling energy demand. First, let's generate the dataset with monthly frequency and let's filter the columns to keep heating and cooling demand at building level.

```
[19]: from accim.data.data_postprocessing import Table
dataset_monthly = Table(
    #datasets=list Since we are not specifying any list, it will use all available CSVs
    #in the folder
    source_frequency='hourly',
```

(continues on next page)

(continued from previous page)

```

frequency='monthly',
frequency_agg_func='sum', #this makes the sum or average when aggregating in days, months or runperiod; since the original CSV frequency is in hour, it won't make any effect
standard_outputs=True,
level=['building'],
level_agg_func=['sum'],
level_excluded_zones=[],
split_epw_names=True, #to split EPW names based on the format Country_City_RCPscenario-YEar
)
dataset_monthly.format_table(
    type_of_table='custom',
    custom_cols=[
        'Building_Total_Cooling Energy Demand (kWh/m2) (summed)',
        'Building_Total_Heating Energy Demand (kWh/m2) (summed)'
    ]
)

```

No zones have been excluded from level computations.
All CSVs are for present scenario.

Now, let's make the table we are looking for. You can choose 'unstack' or 'pivot' as a reshaping method (remember that we are actually working with Pandas Dataframes). Let's go with unstack, and we are going to compare the baseline shown below with all gathered variables

[20]:

```
dataset_monthly.wrangled_table(
    reshaping='unstack', #can be 'unstack' or 'pivot'
    vars_to_gather=['ComfMod', 'CAT'],
    vars_to_keep=['EPW_City_or_subcountry'],
    baseline='CM_0[CA_1',
    comparison_mode=['baseline compared to others'], #can be 'baseline compared to others' or 'others compared to baseline'
    comparison_cols=['relative', 'absolute'] #'relative' to show the difference as a percentage, 'absolute' to show the difference by subtracting
)
```

[21]:

dataset_monthly.wrangled_df_unstacked

Date/time	Month	EPW_City_or_subcountry	Building_Total_Cooling Energy Demand (kWh/m2) (summed)
01/01 01:00:00	01	Aberdeen	0.00
		London	0.12
02/01 01:00:00	02	Aberdeen	0.29
		London	0.43

(continues on next page)

(continued from previous page)

Date/time	Month	EPW_City_or_subcountry	CM_3[CA_1	CM_0[CA_2	CM_3[CA_2
01/01 01:00:00	01	Aberdeen	0.00	0.00	0.00
		London	0.00	0.04	0.00
02/01 01:00:00	02	Aberdeen	0.26	0.14	0.00
		London	0.09	0.07	0.00
03/01 01:00:00	03	Aberdeen	1.06	0.71	0.26
		London	0.95	0.82	0.06
04/01 01:00:00	04	Aberdeen	2.72	1.92	0.79
		London	6.75	5.65	2.76

(continues on next page)

(continued from previous page)

05/01	01:00:00	05	Aberdeen	7.24	5.67	1.67
			London	14.89	11.25	8.51
06/01	01:00:00	06	Aberdeen	11.19	8.94	4.83
			London	16.39	14.03	10.63
07/01	01:00:00	07	Aberdeen	12.88	9.30	5.85
			London	18.93	22.81	10.30
08/01	01:00:00	08	Aberdeen	11.50	8.82	4.58
			London	16.96	19.97	10.88
09/01	01:00:00	09	Aberdeen	4.74	1.39	0.77
			London	10.32	8.71	5.35
10/01	01:00:00	10	Aberdeen	0.38	0.18	0.00
			London	4.53	3.66	1.18
11/01	01:00:00	11	Aberdeen	0.04	0.07	0.00
			London	0.05	0.10	0.00
12/01	01:00:00	12	Aberdeen	0.00	0.00	0.00
			London	0.00	0.00	0.00

CM_0[CA_3 CM_3[CA_3

Date/time	Month	EPW_City_or_subcountry	CM_0[CA_3	CM_3[CA_3
01/01	01:00:00	01	Aberdeen	0.00
			London	0.00
02/01	01:00:00	02	Aberdeen	0.00
			London	0.00
03/01	01:00:00	03	Aberdeen	0.26
			London	0.00
04/01	01:00:00	04	Aberdeen	0.32
			London	2.66
05/01	01:00:00	05	Aberdeen	1.42
			London	4.88
06/01	01:00:00	06	Aberdeen	2.89
			London	9.05
07/01	01:00:00	07	Aberdeen	4.02
			London	15.55
08/01	01:00:00	08	Aberdeen	4.13
			London	13.45
09/01	01:00:00	09	Aberdeen	0.29
			London	4.03
10/01	01:00:00	10	Aberdeen	0.24
			London	3.32
11/01	01:00:00	11	Aberdeen	0.00
			London	0.00
12/01	01:00:00	12	Aberdeen	0.00
			London	0.00

1-(CM_0[CA_1/CM_3[CA_1)

Date/time	Month	EPW_City_or_subcountry	1-(CM_0[CA_1/CM_3[CA_1)	
01/01	01:00:00	01	Aberdeen	NaN
			London	-inf
02/01	01:00:00	02	Aberdeen	-0.14
			London	-3.98

(continues on next page)

(continued from previous page)

03/01	01:00:00	03	Aberdeen	-0.32
			London	-0.41
04/01	01:00:00	04	Aberdeen	-0.24
			London	-0.03
05/01	01:00:00	05	Aberdeen	-0.21
			London	0.03
06/01	01:00:00	06	Aberdeen	-0.09
			London	0.02
07/01	01:00:00	07	Aberdeen	0.06
			London	-0.33
08/01	01:00:00	08	Aberdeen	0.00
			London	-0.33
09/01	01:00:00	09	Aberdeen	0.32
			London	0.01
10/01	01:00:00	10	Aberdeen	0.29
			London	0.15
11/01	01:00:00	11	Aberdeen	-0.11
			London	-7.45
12/01	01:00:00	12	Aberdeen	NaN
			London	NaN

\ CM_3[CA_1 - CM_0[CA_1]

Date/time	Month	EPW_City_or_subcountry	
01/01	01:00:00	01	Aberdeen 0.00
			London -0.12
02/01	01:00:00	02	Aberdeen -0.04
			London -0.34
03/01	01:00:00	03	Aberdeen -0.33
			London -0.39
04/01	01:00:00	04	Aberdeen -0.66
			London -0.20
05/01	01:00:00	05	Aberdeen -1.54
			London 0.52
06/01	01:00:00	06	Aberdeen -0.98
			London 0.29
07/01	01:00:00	07	Aberdeen 0.73
			London -6.28
08/01	01:00:00	08	Aberdeen 0.04
			London -5.53
09/01	01:00:00	09	Aberdeen 1.50
			London 0.09
10/01	01:00:00	10	Aberdeen 0.11
			London 0.68
11/01	01:00:00	11	Aberdeen -0.00
			London -0.35
12/01	01:00:00	12	Aberdeen 0.00
			London 0.00

\ 1-(CM_0[CA_1/CM_0[CA_2])

Date/time	Month	EPW_City_or_subcountry
-----------	-------	------------------------

(continues on next page)

(continued from previous page)

01/01	01:00:00	01	Aberdeen		Nan	
			London		-2.14	
02/01	01:00:00	02	Aberdeen		-1.08	
			London		-5.36	
03/01	01:00:00	03	Aberdeen		-0.97	
			London		-0.64	
04/01	01:00:00	04	Aberdeen		-0.76	
			London		-0.23	
05/01	01:00:00	05	Aberdeen		-0.55	
			London		-0.28	
06/01	01:00:00	06	Aberdeen		-0.36	
			London		-0.15	
07/01	01:00:00	07	Aberdeen		-0.31	
			London		-0.11	
08/01	01:00:00	08	Aberdeen		-0.30	
			London		-0.13	
09/01	01:00:00	09	Aberdeen		-1.34	
			London		-0.17	
10/01	01:00:00	10	Aberdeen		-0.55	
			London		-0.05	
11/01	01:00:00	11	Aberdeen		0.28	
			London		-2.81	
12/01	01:00:00	12	Aberdeen		Nan	
			London		Nan	
						...
						\
						CM_0[CA_2 - CM_0[CA_1]
Date/time	Month	EPW_City_or_subcountry		
01/01	01:00:00	01	Aberdeen	0.00	...	
			London	-0.08	...	
02/01	01:00:00	02	Aberdeen	-0.15	...	
			London	-0.36	...	
03/01	01:00:00	03	Aberdeen	-0.69	...	
			London	-0.52	...	
04/01	01:00:00	04	Aberdeen	-1.45	...	
			London	-1.30	...	
05/01	01:00:00	05	Aberdeen	-3.11	...	
			London	-3.12	...	
06/01	01:00:00	06	Aberdeen	-3.22	...	
			London	-2.07	...	
07/01	01:00:00	07	Aberdeen	-2.85	...	
			London	-2.41	...	
08/01	01:00:00	08	Aberdeen	-2.63	...	
			London	-2.51	...	
09/01	01:00:00	09	Aberdeen	-1.86	...	
			London	-1.52	...	
10/01	01:00:00	10	Aberdeen	-0.10	...	
			London	-0.19	...	
11/01	01:00:00	11	Aberdeen	0.02	...	
			London	-0.29	...	
12/01	01:00:00	12	Aberdeen	0.00	...	
			London	0.00	...	

(continues on next page)

(continued from previous page)

$\text{Building_Total_Heating Energy Demand (kWh/}\text{m}^2\text{) (summed)}$	\backslash	$1 - (\text{CM}_0[\text{CA}_1 / \text{CM}_3[\text{CA}_1])$
Date/time	Month	EPW_City_or_subcountry
01/01 01:00:00	01	Aberdeen
↪ -0.26		London
↪ -0.08		
02/01 01:00:00	02	Aberdeen
↪ -0.12		London
↪ 0.01		
03/01 01:00:00	03	Aberdeen
↪ -0.02		London
↪ 0.05		
04/01 01:00:00	04	Aberdeen
↪ 0.04		London
↪ -0.09		
05/01 01:00:00	05	Aberdeen
↪ -0.36		London
↪ -1.84		
06/01 01:00:00	06	Aberdeen
↪ -0.95		London
↪ -2.43		
07/01 01:00:00	07	Aberdeen
↪ -2.70		London
↪ -0.31		
08/01 01:00:00	08	Aberdeen
↪ -2.32		London
↪ -0.73		
09/01 01:00:00	09	Aberdeen
↪ -2.74		London
↪ -2.71		
10/01 01:00:00	10	Aberdeen
↪ -0.27		London
↪ -0.77		
11/01 01:00:00	11	Aberdeen
↪ -0.05		London
↪ -0.01		
12/01 01:00:00	12	Aberdeen
↪ -0.23		

(continues on next page)

(continued from previous page)

London			
$\hookrightarrow -0.10$			
\backslash			
$CM_3[CA_1] - CM_0[CA_1]$			
Date/time	Month	EPW_City_or_subcountry	
01/01	01:00:00	01 Aberdeen	-3.05
		London	-1.35
02/01	01:00:00	02 Aberdeen	-2.05
		London	0.17
03/01	01:00:00	03 Aberdeen	-0.41
		London	1.29
04/01	01:00:00	04 Aberdeen	1.18
		London	-1.91
05/01	01:00:00	05 Aberdeen	-9.43
		London	-28.72
06/01	01:00:00	06 Aberdeen	-20.62
		London	-30.47
07/01	01:00:00	07 Aberdeen	-36.28
		London	-2.70
08/01	01:00:00	08 Aberdeen	-31.05
		London	-6.28
09/01	01:00:00	09 Aberdeen	-40.09
		London	-31.33
10/01	01:00:00	10 Aberdeen	-6.41
		London	-11.56
11/01	01:00:00	11 Aberdeen	-0.82
		London	-0.15
12/01	01:00:00	12 Aberdeen	-2.81
		London	-1.46
\backslash			
$1 - (CM_0[CA_1]/CM_0[CA_2])$			
Date/time	Month	EPW_City_or_subcountry	
01/01	01:00:00	01 Aberdeen	-0.16
		London	-0.12
02/01	01:00:00	02 Aberdeen	-0.09
		London	-0.04
03/01	01:00:00	03 Aberdeen	-0.03
		London	-0.08
04/01	01:00:00	04 Aberdeen	0.02
		London	-0.10
05/01	01:00:00	05 Aberdeen	0.15
		London	0.15
06/01	01:00:00	06 Aberdeen	0.08
		London	0.04
07/01	01:00:00	07 Aberdeen	0.12
		London	0.22
08/01	01:00:00	08 Aberdeen	0.12
		London	0.17
09/01	01:00:00	09 Aberdeen	0.09
		London	-0.03

(continues on next page)

(continued from previous page)

10/01	01:00:00	10	Aberdeen	-0.25
			London	-0.42
11/01	01:00:00	11	Aberdeen	-0.16
			London	-0.08
12/01	01:00:00	12	Aberdeen	-0.17
			London	-0.20
\				
$CM_0[CA_2 - CM_0[CA_1]$				
Date/time		Month	EPW_City_or_subcountry	
01/01	01:00:00	01	Aberdeen	-2.11
			London	-1.96
02/01	01:00:00	02	Aberdeen	-1.52
			London	-0.78
03/01	01:00:00	03	Aberdeen	-0.72
			London	-1.80
04/01	01:00:00	04	Aberdeen	0.62
			London	-1.95
05/01	01:00:00	05	Aberdeen	6.35
			London	7.73
06/01	01:00:00	06	Aberdeen	3.45
			London	1.86
07/01	01:00:00	07	Aberdeen	6.50
			London	3.16
08/01	01:00:00	08	Aberdeen	5.91
			London	3.10
09/01	01:00:00	09	Aberdeen	5.11
			London	-1.20
10/01	01:00:00	10	Aberdeen	-5.88
			London	-7.85
11/01	01:00:00	11	Aberdeen	-2.45
			London	-1.62
12/01	01:00:00	12	Aberdeen	-2.26
			London	-2.79
\				
$1 - (CM_0[CA_1 / CM_3[CA_2])$				
Date/time		Month	EPW_City_or_subcountry	
01/01	01:00:00	01	Aberdeen	-0.67
			London	-0.54
02/01	01:00:00	02	Aberdeen	-0.37
			London	-0.39
03/01	01:00:00	03	Aberdeen	-0.23
			London	-0.27
04/01	01:00:00	04	Aberdeen	-0.12
			London	-0.47
05/01	01:00:00	05	Aberdeen	-0.69
			London	-4.41
06/01	01:00:00	06	Aberdeen	-1.32
			London	-7.01
07/01	01:00:00	07	Aberdeen	-9.81
			London	-3.47

(continues on next page)

(continued from previous page)

08/01	01:00:00	08	Aberdeen	-7.08
			London	-5.00
09/01	01:00:00	09	Aberdeen	-4.90
			London	-8.28
10/01	01:00:00	10	Aberdeen	-0.96
			London	-2.65
11/01	01:00:00	11	Aberdeen	-0.55
			London	-0.61
12/01	01:00:00	12	Aberdeen	-0.69
			London	-0.75

\ CM_3[CA_2 - CM_0[CA_1]

Date/time	Month	EPW_City_or_subcountry	
01/01	01:00:00	01	Aberdeen
			London
02/01	01:00:00	02	Aberdeen
			London
03/01	01:00:00	03	Aberdeen
			London
04/01	01:00:00	04	Aberdeen
			London
05/01	01:00:00	05	Aberdeen
			London
06/01	01:00:00	06	Aberdeen
			London
07/01	01:00:00	07	Aberdeen
			London
08/01	01:00:00	08	Aberdeen
			London
09/01	01:00:00	09	Aberdeen
			London
10/01	01:00:00	10	Aberdeen
			London
11/01	01:00:00	11	Aberdeen
			London
12/01	01:00:00	12	Aberdeen
			London

\ 1-(CM_0[CA_1/CM_0[CA_3])

Date/time	Month	EPW_City_or_subcountry	
01/01	01:00:00	01	Aberdeen
			London
02/01	01:00:00	02	Aberdeen
			London
03/01	01:00:00	03	Aberdeen
			London
04/01	01:00:00	04	Aberdeen
			London
05/01	01:00:00	05	Aberdeen
			London

(continues on next page)

(continued from previous page)

06/01	01:00:00	06	Aberdeen	-0.00
			London	-0.29
07/01	01:00:00	07	Aberdeen	-0.08
			London	0.23
08/01	01:00:00	08	Aberdeen	-0.10
			London	-0.06
09/01	01:00:00	09	Aberdeen	-0.18
			London	-0.36
10/01	01:00:00	10	Aberdeen	-1.16
			London	-4.72
11/01	01:00:00	11	Aberdeen	-0.57
			London	-0.61
12/01	01:00:00	12	Aberdeen	-0.75
			London	-0.71

\ CM_0[CA_3 - CM_0[CA_1

Date/time	Month	EPW_City_or_subcountry	
01/01	01:00:00	01	Aberdeen
			London
02/01	01:00:00	02	Aberdeen
			London
03/01	01:00:00	03	Aberdeen
			London
04/01	01:00:00	04	Aberdeen
			London
05/01	01:00:00	05	Aberdeen
			London
06/01	01:00:00	06	Aberdeen
			London
07/01	01:00:00	07	Aberdeen
			London
08/01	01:00:00	08	Aberdeen
			London
09/01	01:00:00	09	Aberdeen
			London
10/01	01:00:00	10	Aberdeen
			London
11/01	01:00:00	11	Aberdeen
			London
12/01	01:00:00	12	Aberdeen
			London

\ 1-(CM_0[CA_1/CM_3[CA_3)

Date/time	Month	EPW_City_or_subcountry	
01/01	01:00:00	01	Aberdeen
			London
02/01	01:00:00	02	Aberdeen
			London
03/01	01:00:00	03	Aberdeen
			London

(continues on next page)

(continued from previous page)

04/01	01:00:00	04	Aberdeen	-0.56
			London	-1.79
05/01	01:00:00	05	Aberdeen	-2.12
			London	-16.23
06/01	01:00:00	06	Aberdeen	-4.55
			London	-33.04
07/01	01:00:00	07	Aberdeen	-245.24
			London	-51.57
08/01	01:00:00	08	Aberdeen	-40.97
			London	-46.92
09/01	01:00:00	09	Aberdeen	-12.52
			London	-38.49
10/01	01:00:00	10	Aberdeen	-2.84
			London	-12.53
11/01	01:00:00	11	Aberdeen	-1.48
			London	-2.35
12/01	01:00:00	12	Aberdeen	-1.51
			London	-1.75

CM_3[CA_3 - CM_0[CA_1]

Date/time	Month	EPW_City_or_subcountry	CM_3[CA_3 - CM_0[CA_1]	
01/01	01:00:00	01	Aberdeen	-7.88
			London	-9.53
02/01	01:00:00	02	Aberdeen	-9.65
			London	-10.06
03/01	01:00:00	03	Aberdeen	-11.02
			London	-12.95
04/01	01:00:00	04	Aberdeen	-9.36
			London	-14.23
05/01	01:00:00	05	Aberdeen	-24.07
			London	-41.76
06/01	01:00:00	06	Aberdeen	-34.76
			London	-41.77
07/01	01:00:00	07	Aberdeen	-49.51
			London	-11.18
08/01	01:00:00	08	Aberdeen	-43.37
			London	-14.57
09/01	01:00:00	09	Aberdeen	-50.66
			London	-41.80
10/01	01:00:00	10	Aberdeen	-22.01
			London	-24.53
11/01	01:00:00	11	Aberdeen	-10.69
			London	-15.54
12/01	01:00:00	12	Aberdeen	-9.19
			London	-10.48

[24 rows x 32 columns]

Let's make a different dataset to work with runperiod frequency.

[22]: from accim.data.data_postprocessing import Table

(continues on next page)

(continued from previous page)

```

dataset_runperiod = Table(
    #datasets=list Since we are not specifying any list, it will use all available CSVs
    ↵in the folder
    source_frequency='hourly',
    frequency='runperiod',
    frequency_agg_func='sum', #this makes the sum or average when aggregating in days,
    ↵months or runperiod; since the original CSV frequency is in hour, it won't make any
    ↵affect
    standard_outputs=True,
    level=['building'],
    level_agg_func=['sum'],
    level_excluded_zones=[],
    #match_cities=bool Only used when EPW file has NOT been previously renamed
    #manage_epw_names=bool Only used when EPW file has NOT been previously renamed
    split_epw_names=True, #to split EPW names based on the format Country_City_
    ↵RCPscenario-YEar
)

dataset_runperiod.format_table(
    type_of_table='custom',
    custom_cols=[
        'Building_Total_Cooling Energy Demand (kWh/m2) (summed)',
        'Building_Total_Heating Energy Demand (kWh/m2) (summed)'
    ]
)

```

No zones have been excluded from level computations.
All CSVs are for present scenario.

In this case, we are going to use the ‘pivot’ reshaping option:

```
[23]: dataset_runperiod.wrangled_table(
    reshaping='pivot',
    vars_to_gather=['ComfMod', 'CAT'],
    baseline='CM_0[CA_1',
    comparison_mode=['baseline compared to others'],
    comparison_cols=['relative', 'absolute']
)

C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\data_
↪postprocessing.py:1611: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_
↪guide/indexing.html#returning-a-view-versus-a-copy
    self.df['col_to_pivot'] = 'temp'
C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\data_
↪postprocessing.py:1624: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

(continues on next page)

(continued from previous page)

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
`self.df['col_to_pivot'] = wrangled_df_pivoted['col_to_pivot']`

[24]: dataset_runperiod.wrangled_df_pivoted

```
col_to_pivot           CM_0[CA_1]  CM_0[CA_2] \
Building_Total_Cooling Energy Demand (kWh/m2) (...  154.66   124.24
Building_Total_Heating Energy Demand (kWh/m2) (...  680.07   688.99

col_to_pivot           CM_0[CA_3]  CM_3[CA_1] \
Building_Total_Cooling Energy Demand (kWh/m2) (...  66.51    141.86
Building_Total_Heating Energy Demand (kWh/m2) (...  547.22   413.73

col_to_pivot           CM_3[CA_2]  CM_3[CA_3] \
Building_Total_Cooling Energy Demand (kWh/m2) (...  68.41    26.77
Building_Total_Heating Energy Demand (kWh/m2) (...  274.22   149.52

col_to_pivot           1-(CM_0[CA_1/CM_3[CA_1]) \
Building_Total_Cooling Energy Demand (kWh/m2) (...) -0.09
Building_Total_Heating Energy Demand (kWh/m2) (...) -0.64

col_to_pivot           CM_3[CA_1] - CM_0[CA_1] \
Building_Total_Cooling Energy Demand (kWh/m2) (...) -12.80
Building_Total_Heating Energy Demand (kWh/m2) (...) -266.33

col_to_pivot           1-(CM_0[CA_1/CM_0[CA_2]) \
Building_Total_Cooling Energy Demand (kWh/m2) (...) -0.24
Building_Total_Heating Energy Demand (kWh/m2) (...)  0.01

col_to_pivot           CM_0[CA_2] - CM_0[CA_1] \
Building_Total_Cooling Energy Demand (kWh/m2) (...) -30.42
Building_Total_Heating Energy Demand (kWh/m2) (...)  8.92

col_to_pivot           1-(CM_0[CA_1/CM_3[CA_3]) \
Building_Total_Cooling Energy Demand (kWh/m2) (...) -4.78
Building_Total_Heating Energy Demand (kWh/m2) (...) -3.55

col_to_pivot           CM_3[CA_3] - CM_0[CA_1] \
Building_Total_Cooling Energy Demand (kWh/m2) (...) -127.89
Building_Total_Heating Energy Demand (kWh/m2) (...) -530.55

col_to_pivot           1-(CM_0[CA_1/CM_3[CA_2]) \
Building_Total_Cooling Energy Demand (kWh/m2) (...) -1.26
Building_Total_Heating Energy Demand (kWh/m2) (...) -1.48

col_to_pivot           CM_3[CA_2] - CM_0[CA_1] \
Building_Total_Cooling Energy Demand (kWh/m2) (...) -86.24
Building_Total_Heating Energy Demand (kWh/m2) (...) -405.85

col_to_pivot           1-(CM_0[CA_1/CM_0[CA_3]) \
Building_Total_Cooling Energy Demand (kWh/m2) (...) -1.33
```

(continues on next page)

(continued from previous page)

Building_Total_Heating Energy Demand (kWh/m2) (...	-0.24
col_to_pivot	CM_0[CA_3 - CM_0[CA_1
Building_Total_Cooling Energy Demand (kWh/m2) (...	-88.14
Building_Total_Heating Energy Demand (kWh/m2) (...	-132.84

Again, please remember we are working with Pandas Dataframe objects. That means we can modify the df we have generated. For instance, we can simplify it by removing all rows except those with ‘CA_3’ in the Category column, as shown below.

```
[25]: dataset_runperiod.df = dataset_runperiod.df[
    dataset_runperiod.df['CAT'].isin(['CA_3'])
]
```

```
[26]: dataset_runperiod.df
```

	Model	ComfStand	CAT	\				
8	TestModel_onlyGeometryForVRFsystem_2zones_Calc...	CS_INT	EN16798	CA_3				
9	TestModel_onlyGeometryForVRFsystem_2zones_Calc...	CS_INT	EN16798	CA_3				
10	TestModel_onlyGeometryForVRFsystem_2zones_Calc...	CS_INT	EN16798	CA_3				
11	TestModel_onlyGeometryForVRFsystem_2zones_Calc...	CS_INT	EN16798	CA_3				
	ComfMod	HVACmode	VentCtrl	VSToffset	MinOToffset	MaxWindSpeed	ASTtol	\
8	CM_0	HM_2	VC_0	VO_0	MT_50	MW_50	AT_0.1	
9	CM_0	HM_2	VC_0	VO_0	MT_50	MW_50	AT_0.1	
10	CM_3	HM_2	VC_0	VO_0	MT_50	MW_50	AT_0.1	
11	CM_3	HM_2	VC_0	VO_0	MT_50	MW_50	AT_0.1	
	NameSuffix				EPW	\		
8	NS_X	United-Kingdom_Aberdeen_Present						
9	NS_X	United-Kingdom_London_Present						
10	NS_X	United-Kingdom_Aberdeen_Present						
11	NS_X	United-Kingdom_London_Present						
	Source	EPW_Country_name	\					
8	TestModel_onlyGeometryForVRFsystem_2zones_Calc...	United-Kingdom						
9	TestModel_onlyGeometryForVRFsystem_2zones_Calc...	United-Kingdom						
10	TestModel_onlyGeometryForVRFsystem_2zones_Calc...	United-Kingdom						
11	TestModel_onlyGeometryForVRFsystem_2zones_Calc...	United-Kingdom						
	EPW_City_or_subcountry	EPW_Scenario-Year	EPW_Scenario	EPW_Year	\			
8	Aberdeen	Present	Present	Present				
9	London	Present	Present	Present				
10	Aberdeen	Present	Present	Present				
11	London	Present	Present	Present				
	Building_Total_Cooling Energy Demand (kWh/m2) (summed)	\						
8		13.574438						
9		52.938830						
10		5.687532						
11		21.084511						

(continues on next page)

(continued from previous page)

	Building_Total_Heating Energy Demand (kWh/m2) (summed)
8	318.977783
9	228.246073
10	92.602457
11	56.917599

And now we could make a similar pivoted table as above, but simplified to show only CA_3:

```
[27]: dataset_runperiod.wrangled_table(
    reshaping='pivot',
    vars_to_gather=['ComfMod'],
    baseline='CM_0',
    comparison_mode=['baseline compared to others'],
    comparison_cols=['relative', 'absolute']
)
```

```
[28]: dataset_runperiod.wrangled_df_pivoted
```

col_to_pivot	CM_0	CM_3	\
Building_Total_Cooling Energy Demand (kWh/m2) (...)	66.51	26.77	
Building_Total_Heating Energy Demand (kWh/m2) (...)	547.22	149.52	
col_to_pivot	1-(CM_0/CM_3)	CM_3 - CM_0	
Building_Total_Cooling Energy Demand (kWh/m2) (...)	-1.48	-39.74	
Building_Total_Heating Energy Demand (kWh/m2) (...)	-2.66	-397.70	

Another method to achieve this is generating a different Table object, only reading the CSV files with CA_3 in it name:

```
[29]: import os
from accim.data.data_postprocessing import Table

dataset = [i for i in os.listdir() if i.endswith('.csv') and 'CA_3' in i]

dataset_runperiod_simplified_1 = Table(
    datasets=dataset,
    source_frequency='hourly',
    frequency='runperiod',
    frequency_agg_func='sum',
    # this makes the sum or average when aggregating in days, months or runperiod; since_
    ↵the original CSV frequency is in hour, it won't make any effect
    standard_outputs=True,
    level=['building'],
    level_agg_func=['sum'],
    level_excluded_zones=[],
    split_epw_names=True, # to split EPW names based on the format Country_City_
    ↵RCPscenario-YEar
)

dataset_runperiod_simplified_1.format_table(
    type_of_table='custom',
    custom_cols=[
        'Building_Total_Cooling Energy Demand (kWh/m2) (summed)',
        'Building_Total_Heating Energy Demand (kWh/m2) (summed)'
    ]
)
```

(continues on next page)

(continued from previous page)

```

        ]
)

dataset_runperiod_simplified_1.wrangled_table(
    reshaping='pivot',
    vars_to_gather=['ComfMod'],
    baseline='CM_0',
    comparison_mode=['baseline compared to others'],
    comparison_cols=['relative', 'absolute']
)

```

No zones have been excluded from level computations.
All CSVs are for present scenario.

```
C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\data_
↳postprocessing.py:1611: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_
↳guide/indexing.html#returning-a-view-versus-a-copy
    self.df['col_to_pivot'] = 'temp'
C:\users\sanga\appdata\local\programs\python\python39\lib\site-packages\accim\data\data_
↳postprocessing.py:1624: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_
↳guide/indexing.html#returning-a-view-versus-a-copy
    self.df['col_to_pivot'] = wrangled_df_pivoted['col_to_pivot']
```

You can see below the resulting table is exactly the same:

[30]:	dataset_runperiod_simplified_1.wrangled_df_pivoted
[30]:	col_to_pivot CM_0 CM_3 \
	Building_Total_Cooling Energy Demand (kWh/m2) (... 66.51 26.77
	Building_Total_Heating Energy Demand (kWh/m2) (... 547.22 149.52
	col_to_pivot 1-(CM_0/CM_3) CM_3 - CM_0
	Building_Total_Cooling Energy Demand (kWh/m2) (... -1.48 -39.74
	Building_Total_Heating Energy Demand (kWh/m2) (... -2.66 -397.70

You can finally export the table as xlsx format:

[31]:	dataset_runperiod_simplified_1.wrangled_df_pivoted.to_excel('building_energy_demand.xlsx ↳')
-------	--

Let's delete it now:

```
[32]: for i in os.listdir():
        if i.endswith('.xlsx'):
            os.remove(i)
```

```
[ ]:
```


13.1 accim package

13.1.1 Subpackages

13.1.1.1 accim.data package

13.1.1.1.1 Submodules

13.1.1.1.2 accim.data.data_postprocessing module

```
class accim.data.data_postprocessing.Table(datasets: Optional[list] = None,  
                                         source_concatenated_csv_filepath: Optional[str] = None,  
                                         source_frequency: Optional[str] = None, frequency:  
                                         Optional[str] = None, frequency_agg_func: Optional[str] =  
                                         None, standard_outputs: Optional[bool] = None,  
                                         concatenated_csv_name: Optional[str] = None, level:  
                                         Optional[list] = None, level_agg_func: Optional[list] =  
                                         None, level_excluded_zones: Optional[list] = None,  
                                         block_zone_hierarchy: Optional[dict] = None,  
                                         split_epw_names: bool = False, normalised_energy_units:  
                                         bool = True, rename_cols: bool = True,  
                                         energy_units_in_kwh: bool = True, drop_nan: bool = False,  
                                         name_export_rows_with_NaN: Optional[str] = None,  
                                         name_export_rows_not_corr_agg: Optional[str] = None)
```

Bases: object

custom_order(ordered_list: Optional[list] = None, column_to_order: Optional[str] = None)

Used to order the string values of a column in a custom order. :param ordered_list: A list of strings. Used to order the string values of a column in a custom order. :param column_to_order: A string. It should be the column whose string values should be ordered.

df()

enter_vars_to_gather(vars_to_gather=None)

format_table(type_of_table: str = 'all', custom_cols: Optional[list] = None)

Filter the columns. :param type_of_table: To get previously set out tables. Can be ‘energy demand’ or ‘comfort hours’. :param custom_cols: A list of strings. The strings will be used as a filter, and the columns that match will be selected.

gather_vars_query(*vars_to_gather*: *Optional[list] = None*)

Used to inform the user of the variables suitable to be analysed and the available options from a certain gathered variables :param *vars_to_gather*: A list of variables.

generate_fig_data(*vars_to_gather_cols*: *Optional[list] = None*, *vars_to_gather_rows*: *Optional[list] = None*, *detailed_cols*: *Optional[list] = None*, *detailed_rows*: *Optional[list] = None*, *custom_cols_order*: *Optional[list] = None*, *custom_rows_order*: *Optional[list] = None*, *data_on_y_axis_baseline_plot*: *Optional[list] = None*, *baseline*: *Optional[str] = None*, *colorlist_baseline_plot_data*: *Optional[list] = None*, *data_on_x_axis*: *Optional[str] = None*, *data_on_y_main_axis*: *Optional[list] = None*, *data_on_y_sec_axis*: *Optional[list] = None*, *colorlist_y_main_axis*: *Optional[list] = None*, *colorlist_y_sec_axis*: *Optional[list] = None*, *best_fit_deg_y_main_axis*: *Optional[list] = None*, *best_fit_deg_y_sec_axis*: *Optional[list] = None*, *best_fit_deg*: *Optional[list] = None*, *rows_renaming_dict*: *Optional[dict] = None*, *cols_renaming_dict*: *Optional[dict] = None*)

Generates list of data to be plotted. :param *vars_to_gather_cols*: A list of strings. The list should be the variables you want to show in subplot columns. :param *vars_to_gather_rows*: A list of strings. The list should be the variables you want to show in subplot rows. :param *detailed_cols*: A list of strings. The list should be the specific data you want to show in subplots columns. Used to filter. :param *detailed_rows*: A list of strings. The list should be the specific data you want to show in subplots rows. Used to filter. :param *custom_cols_order*: A list of strings. The list should be the specific order for the items shown in subplot columns. :param *custom_rows_order*: A list of strings. The list should be the specific order for the items shown in subplot rows. :param *data_on_y_axis_baseline_plot*: A list of strings. Used to select the data you want to show in the graph. Should be a list of the column names you want to plot in each subplot. :param *baseline*: A string, used only in *data_on_y_axis_baseline_plot*. The baseline should be one of the combinations in *vars_to_gather_cols*. It will be plotted in x-axis, while the reference combination for comparison in y-axis. :param *colorlist_baseline_plot_data*: A list of strings. Should be the colors using the matplotlib color notation for the columns entered in *data_on_y_axis_baseline_plot* in the same order. :param *data_on_x_axis*: A string. The column name you want to plot in the x-axis. :param *data_on_y_main_axis*: A list with nested lists and strings. Used to select the data you want to show in the scatter plot main y-axis. It needs to follow this structure: [[‘name_on_y_main_axis’, [list of column names you want to plot]]] :param *data_on_y_sec_axis*: A list with nested lists and strings. Used to select the data you want to show in the scatter plot secondary y-axis. It needs to follow this structure: [[‘name_on_1st_y_sec_axis’, [list of column names you want to plot], [‘name_on_2nd_y_sec_axis’, [list of column names you want to plot], etc]]] :param *colorlist_y_main_axis*: A list with nested lists and strings. It should follow the same structure as *data_on_y_main_axis*, but replacing the column names with the colors using the matplotlib notation. :param *colorlist_y_sec_axis*: A list with nested lists and strings. It should follow the same structure as *data_on_y_sec_axis*, but replacing the column names with the colors using the matplotlib notation. :param *rows_renaming_dict*: A dictionary. Should follow the pattern {‘old row name 1’: ‘new row name 1’, ‘old row name 2’: ‘new row name 2’} :param *cols_renaming_dict*: A dictionary. Should follow the pattern {‘old col name 1’: ‘new col name 1’, ‘old col name 2’: ‘new col name 2’} :return:

scatter_plot(*vars_to_gather_cols*: *Optional[list] = None*, *vars_to_gather_rows*: *Optional[list] = None*, *detailed_cols*: *Optional[list] = None*, *detailed_rows*: *Optional[list] = None*, *custom_cols_order*: *Optional[list] = None*, *custom_rows_order*: *Optional[list] = None*, *data_on_x_axis*: *Optional[str] = None*, *data_on_y_main_axis*: *Optional[list] = None*, *data_on_y_sec_axis*: *Optional[list] = None*, *colorlist_y_main_axis*: *Optional[list] = None*, *colorlist_y_sec_axis*: *Optional[list] = None*, *best_fit_deg_y_main_axis*: *Optional[list] = None*, *best_fit_deg_y_sec_axis*: *Optional[list] = None*, *rows_renaming_dict*: *Optional[dict] = None*, *cols_renaming_dict*: *Optional[dict] = None*, *sharex*: *bool = True*, *sharey*: *bool = True*, *sup xlabel*: *Optional[str] = None*, *figname*: *Optional[str] = None*, *figsize*: *float = 1*, *ratio_height_to_width*: *float = 1*, *dpi*: *int = 500*, *confirm_graph*: *bool = False*, *set_facecolor*: *any = (0, 0, 0, 0.1)*, *best_fit_background_linewidth*: *float = 1*, *best_fit_linewidth*: *float = 0.5*, *best_fit_linestyle*: *any = (0, (5, 10))*)

Used to plot a scatter plot. :param vars_to_gather_cols: A list of strings. The list should be the variables you want to show in subplot columns. :param vars_to_gather_rows: A list of strings. The list should be the variables you want to show in subplot rows. :param detailed_cols: A list of strings. The list should be the specific data you want to show in subplots columns. Used to filter. :param detailed_rows: A list of strings. The list should be the specific data you want to show in subplots rows. Used to filter. :param custom_cols_order: A list of strings. The list should be the specific order for the items shown in subplot columns. :param custom_rows_order: A list of strings. The list should be the specific order for the items shown in subplot rows. :param data_on_x_axis: A string. The column name you want to plot in the x-axis. :param data_on_y_main_axis: A list with nested lists and strings. Used to select the data you want to show in the scatter plot main y-axis. It needs to follow this structure: [[‘name_on_y_main_axis’, [list of column names you want to plot]]] :param data_on_y_sec_axis: A list with nested lists and strings. Used to select the data you want to show in the scatter plot secondary y-axis. It needs to follow this structure: [[[‘name_on_1st_y_sec_axis’, [list of column names you want to plot]], [‘name_on_2nd_y_sec_axis’, [list of column names you want to plot]]], etc] :param colorlist_y_main_axis: A list with nested lists and strings. It should follow the same structure as data_on_y_main_axis, but replacing the column names with the colors using the matplotlib notation. :param colorlist_y_sec_axis: A list with nested lists and strings. It should follow the same structure as data_on_y_sec_axis, but replacing the column names with the colors using the matplotlib notation. :param best_fit_deg_y_sec_axis: A list with nested lists and strings. It should follow the same structure as data_on_y_sec_axis, but replacing the column names with the polynomial degree for the best fit lines. :param best_fit_deg_y_main_axis: A list with nested lists and strings. It should follow the same structure as data_on_y_main_axis, but replacing the column names with the polynomial degree for the best fit lines. :param rows_renaming_dict: A dictionary. Should follow the pattern {‘old row name 1’: ‘new row name 1’, ‘old row name 2’: ‘new row name 2’} :param cols_renaming_dict: A dictionary. Should follow the pattern {‘old col name 1’: ‘new col name 1’, ‘old col name 2’: ‘new col name 2’} :param sharex: True to share the x-axis across all subplots :param sharey: True to share the y-axis across all subplots :param supxlabel: A string. The label shown in the x-axis. :param figname: A string. The name of the saved figure without extension. :param figsize: A float. It is the figure size. :param ratio_height_to_width: A float. By default, is 1 (squared). If 0.5 is entered, the figure will be half higher than wide. :param dpi: An integer. The number of dpis for image quality. :param confirm_graph: A bool. True to skip confirmation step. :param set_facecolor: Usage is similar to matplotlib.axes.Axes.set_facecolor :param best_fit_linestyle: Anything in matplotlib linestyle notation. Use to change the style of the best fit lines. :param best_fit_linewidth: A float. Used to change the width of the best fit lines. :param best_fit_background_linewidth: A float. Used to change the width of the background best fit lines. Must be greater than best_fit_linewidth.

```
scatter_plot_with_baseline(vars_to_gather_cols: Optional[list] = None, vars_to_gather_rows: Optional[list] = None, detailed_cols: Optional[list] = None, detailed_rows: Optional[list] = None, custom_cols_order: Optional[list] = None, custom_rows_order: Optional[list] = None, data_on_y_axis_baseline_plot: Optional[list] = None, baseline: Optional[str] = None, colorlist_baseline_plot_data: Optional[list] = None, best_fit_deg: Optional[list] = None, rows_renaming_dict: Optional[dict] = None, cols_renaming_dict: Optional[dict] = None, sup xlabel: Optional[str] = None, supylabel: Optional[str] = None, figname: Optional[str] = None, figsize: int = 1, markersize: int = 1, dpi: int = 500, confirm_graph: bool = False, set_facecolor: any = (0, 0, 0, 0.1), best_fit_background_linewidth: float = 1, best_fit_linewidth: float = 0.5, best_fit_linestyle: any = (0, (5, 10)))
```

Parameters

- **vars_to_gather_cols** – A list of strings. The list should be the variables you want to show in subplot columns.
- **vars_to_gather_rows** – A list of strings. The list should be the variables you want to show in subplot rows.

- **detailed_cols** – A list of strings. The list should be the specific data you want to show in subplots columns. Used to filter.
- **detailed_rows** – A list of strings. The list should be the specific data you want to show in subplots rows. Used to filter.
- **custom_cols_order** – A list of strings. The list should be the specific order for the items shown in subplot columns.
- **custom_rows_order** – A list of strings. The list should be the specific order for the items shown in subplot rows.
- **data_on_y_axis_baseline_plot** – A list of strings. Used to select the data you want to show in the graph. Should be a list of the column names you want to plot in each subplot.
- **baseline** – A string, used only in data_on_y_axis_baseline_plot. The baseline should be one of the combinations in vars_to_gather_cols. It will be plotted in x-axis, while the reference combination for comparison in y-axis.
- **colorlist_baseline_plot_data** – A list of strings. Should be the colors using the matplotlib color notation for the columns entered in data_on_y_axis_baseline_plot in the same order.
- **best_fit_deg** – A list with nested lists and strings. It should follow the same structure as data_on_y_axis_baseline_plot, but replacing the column names with the polynomial degree for the best fit lines.
- **rows_renaming_dict** – A dictionary. Should follow the pattern {‘old row name 1’: ‘new row name 1’, ‘old row name 2’: ‘new row name 2’}
- **cols_renaming_dict** – A dictionary. Should follow the pattern {‘old col name 1’: ‘new col name 1’, ‘old col name 2’: ‘new col name 2’}
- **supxlabel** – A string. The label shown in the x-axis.
- **supylabel** – A string. The label shown in the y-axis.
- **figname** – A string. The name of the saved figure without extension.
- **figsize** – A float. It is the figure size.
- **markersize** – An integer. The size of the markers.
- **dpi** – An integer. The number of dpis for image quality.
- **confirm_graph** – A bool. True to skip confirmation step.
- **best_fit_linestyle** – Anything in matplotlib linestyle notation. Use to change the style of the best fit lines.
- **best_fit_linewidth** – A float. Used to change the width of the best fit lines.
- **best_fit_background_linewidth** – A float. Used to change the width of the background best fit lines. Must be greater than best_fit_linewidth.

```
time_plot(vars_to_gather_cols: Optional[list] = None, vars_to_gather_rows: Optional[list] = None,  

           detailed_cols: Optional[list] = None, detailed_rows: Optional[list] = None, custom_cols_order:  

           Optional[list] = None, custom_rows_order: Optional[list] = None, data_on_y_main_axis:  

           Optional[list] = None, data_on_y_sec_axis: Optional[list] = None, colorlist_y_main_axis:  

           Optional[list] = None, colorlist_y_sec_axis: Optional[list] = None, rows_renaming_dict:  

           Optional[dict] = None, cols_renaming_dict: Optional[dict] = None, sharex: bool = True, sharey:  

           bool = True, figname: Optional[str] = None, figsize: float = 1, ratio_height_to_width: float = 1,  

           dpi: int = 500, confirm_graph: bool = False, set_facecolor: any = (0, 0, 0, 0.1))
```

Used to plot a timeplot. :param vars_to_gather_cols: A list of strings. The list should be the variables you want to show in subplot columns. :param vars_to_gather_rows: A list of strings. The list should be the variables you want to show in subplot rows. :param detailed_cols: A list of strings. The list should be the specific data you want to show in subplots columns. Used to filter. :param detailed_rows: A list of strings. The list should be the specific data you want to show in subplots rows. Used to filter. :param custom_cols_order: A list of strings. The list should be the specific order for the items shown in subplot columns. :param custom_rows_order: A list of strings. The list should be the specific order for the items shown in subplot rows. :param data_on_x_axis: A string. The column name you want to plot in the x-axis. :param data_on_y_main_axis: A list with nested lists and strings. Used to select the data you want to show in the scatter plot main y-axis. It needs to follow this structure: [[‘name_on_y_main_axis’, [list of column names you want to plot]]] :param data_on_y_sec_axis: A list with nested lists and strings. Used to select the data you want to show in the scatter plot secondary y-axis. It needs to follow this structure: [[[‘name_on_1st_y_sec_axis’, [list of column names you want to plot]], [‘name_on_2nd_y_sec_axis’, [list of column names you want to plot]]], etc] :param colorlist_y_main_axis: A list with nested lists and strings. It should follow the same structure as data_on_y_main_axis, but replacing the column names with the colors using the matplotlib notation. :param colorlist_y_sec_axis: A list with nested lists and strings. It should follow the same structure as data_on_y_sec_axis, but replacing the column names with the colors using the matplotlib notation. :param rows_renaming_dict: A dictionary. Should follow the pattern {‘old row name 1’: ‘new row name 1’, ‘old row name 2’: ‘new row name 2’} :param cols_renaming_dict: A dictionary. Should follow the pattern {‘old col name 1’: ‘new col name 1’, ‘old col name 2’: ‘new col name 2’}

Parameters

- **sharey** – True to share the x-axis across all subplots
- **sharex** – True to share the y-axis across all subplots
- **supxlabel** – A string. The label shown in the x-axis.
- **figname** – A string. The name of the saved figure without extension.
- **figsize** – A float. It is the figure size.
- **ratio_height_to_width** – A float. By default, is 1 (squared). If 0.5 is entered, the figure will be half higher than wide.
- **dpi** – An integer. The number of dpis for image quality.
- **confirm_graph** – A bool. True to skip confirmation step.

```
wrangle_table(reshaping: Optional[str] = None, vars_to_gather: Optional[list] = None, baseline: Optional[str] = None, comparison_mode: list = ['others compared to baseline'], comparison_cols: list = ['absolute', 'relative'], check_index_and_cols: bool = False, vars_to_keep: Optional[list] = None, rename_dict: Optional[dict] = None, transpose: bool = False, excel_filename: Optional[str] = None)
```

Creates a table based on the arguments. :param reshaping: A string. Can be ‘pivot’, ‘unstack’ or ‘multiindex’, to perform these actions. :param vars_to_gather: A list of the variables to be transposed from rows to columns. :param baseline: The already transposed column you want to use as a baseline for comparisons. If omitted, you will be asked which one to use. :param comparison_mode: A list of strings. Can be ‘others compared to baseline’ and/or ‘baseline compared to others’. Used to customise the comparison of variables. :param comparison_cols: A list of strings. ‘absolute’ to get the difference or ‘relative’ to get the percentage of reduction. :param check_index_and_cols: A boolean. True to check index and cols, False to skip. :param vars_to_keep: A list of strings. To remove all variables from the multiindex except those to be kept. :param excel_filename: A string. If entered, the wrangled_df will be exported to excel with that string as name. :param transpose: True to transpose the dataframe :param rename_dict: Renames all data in the dataframe based on the format {‘old_string’: ‘new_string’}

```
accim.data.data_postprocessing.genCSVconcatenated(datasets: Optional[list] = None, source_frequency: Optional[str] = None, frequency: Optional[str] = None, datasets_per_chunk: int = 50, concatenated_csv_name: Optional[str] = None, drop_nan: bool = True)
```

13.1.1.1.3 accim.data.data_preprocessing module

```
class accim.data.data_preprocessing.give_address(latitude, longitude)
    Bases: object
class accim.data.data_preprocessing.give_address_openssl(latitude, longitude)
    Bases: object
class accim.data.data_preprocessing.give_address_ssl(latitude, longitude)
    Bases: object
class accim.data.data_preprocessing.rename_epw_files(filelist=None, rename_dict: Optional[dict] = None, confirm_renaming=None, confirm_deletion=None)
    Bases: object
```

13.1.1.1.4 Module contents

13.1.1.2 accim.run package

13.1.1.2.1 Submodules

13.1.1.2.2 accim.run.run module

Multiprocessing runs.

using generators instead of a list when you are running 100 files you have to use generators original script: <https://eppy.readthedocs.io/en/latest/runningsplus.html> slightly modified so that takes multiple sample_EPWs located in the local folder

```
accim.run.run.make_eplaunch_options(idf, epw)
```

Make options for run, so that it runs like EPLaunch on Windows.

```
accim.run.run.removefiles()
```

Delete all files except ‘.py’, ‘.idf’, ‘.epw’, ‘.csv’ and ‘.eso’.

‘Table.csv’, ‘Meter.csv’, ‘Zsz.csv’ files are deleted as well.

```
accim.run.run.runEp(runOnlyAccim=None, confirmRun=None, num_CPUs: int = 2, EnergyPlus_version: Optional[str] = None)
```

Run simulations in the latest EnergyPlus version installed in the computer.

Parameters

- **runOnlyAccim** – Default is None. Enter True to run only ACCIM output IDFs, or False to run all IDFs.
- **confirmRun** – Default is None. Enter True to run all simulations regardless the no. of them,

or False to shut down all runs. :param num CPUs: An integer. The number of CPUs to be used. :param EnergyPlus_version: A string. It should be the EnergyPlus version of the IDFs, from ‘9.0’ to ‘23.1’.

13.1.1.2.3 Module contents

13.1.1.3 accim.sim package

13.1.1.3.1 Submodules

13.1.1.3.2 accim.sim.accim_Base module

Add EnergyPlus objects in common to both ExistingHVAC and VRFSysystem.

`accim.sim.accim_Base.saveaccim(self, verboseMode: bool = True)`

Save IDF.

`accim.sim.accim_Base.setComfFieldsPeople(self, EnergyPlus_version: Optional[str] = None, TempCtrl: Optional[str] = None, verboseMode: bool = True)`

Amend PEOPLE objects so that accim can work.

Copy existing PEOPLE objects and adds AdaptiveASH55 and AdaptiveCEN15251 to Thermal Comfort Model types 1 and 2 fields of the existing People objects.

`accim.sim.accim_Base.setPMVsetpoint(self, verboseMode: bool = True)`

Sets PMV setpoints for temperature control.

13.1.1.3.3 accim.sim.accim_Base_EMS module

Add EMS objects in common to both ExistingHVAC and VRFSysystem.

`accim.sim.accim_Base_EMS.addEMSActuatorsBase(self, ScriptType: Optional[str] = None, verboseMode: bool = True)`

Add EMS actuators for accim.

`accim.sim.accim_Base_EMS.addEMSSoutputVariableBase(self, ScriptType: Optional[str] = None, verboseMode: bool = True)`

Add EMS output variables for Base accim.

Checks if some EMS output variables objects are already in the model, and otherwise adds them.

`accim.sim.accim_Base_EMS.addEMSPCMBase(self, verboseMode: bool = True)`

Add EMS program calling managers for Base accim.

Checks if some EMS program calling manager objects are already in the model, and otherwise adds them.

`accim.sim.accim_Base_EMS.addEMSPProgramsBase(self, ScriptType: Optional[str] = None, verboseMode: bool = True)`

Add EMS programs for Base accim.

Checks if some programs objects are already in the model, and otherwise adds them. :param verboseMode:

`accim.sim.accim_Base_EMS.addEMSSensorsBase(self, ScriptType: Optional[str] = None, verboseMode: bool = True)`

Add EMS sensors for accim.

```
accim.sim.accim_BaseEMS.addGlobVarList(self, ScriptType: Optional[str] = None, verboseMode: bool = True)
```

Remove existing Global Variable objects and add correct Global Variable objects for accim.

```
accim.sim.accim_BaseEMS.addIntVarList(self, verboseMode: bool = True)
```

Add Internal variables objects for accim.

```
accim.sim.accim_BaseEMS.addOutputVariablesDetailed(self, Outputs_freq: Optional[any] = None, verboseMode: bool = True)
```

Add Output:Variable objects for accim.

```
accim.sim.accim_BaseEMS.addOutputVariablesSimplified(self, Outputs_freq: Optional[any] = None, TempCtrl: Optional[str] = None, verboseMode: bool = True)
```

Add simplified Output:Variable objects for accim.

Remove all outputs and add only VFR outdoor unit consumption and operative temperature.

```
accim.sim.accim_BaseEMS.addOutputVariablesStandard(self, Outputs_freq: Optional[any] = None, ScriptType: Optional[str] = None, TempCtrl: Optional[str] = None, verboseMode: bool = True)
```

Add Output:Variable objects for accim.

```
accim.sim.accim_BaseEMS.genOutputDataframe(self, idf_filename)
```

```
accim.sim.accim_BaseEMS.outputsSpecified(self, remove_or_keep: Optional[str] = None)
```

Remove duplicated Output:Variable objects for accim.

```
accim.sim.accim_BaseEMS.removeDuplicatedOutputVariables(self)
```

Remove duplicated Output:Variable objects for accim.

```
accim.sim.accim_BaseEMS.removeExistingOutputVariables(self)
```

Remove existing Output:Variable objects for accim.

```
accim.sim.accim_BaseEMS.takeOutputDataFrame(self, idf_filename, df_outputs_in, verboseMode)
```

13.1.1.3.4 accim.sim.accim_ExistingHVAC module

```
accim.sim.accim_ExistingHVAC.addForScriptSchExistHVAC(self, verboseMode: bool = True)
```

Add FORSCRIPT Schedules for each zone in existing HVAC zones.

13.1.1.3.5 accim.sim.accim_ExistingHVAC_EMSS module

```
accim.sim.accim_ExistingHVAC_EMSS.addEMSSensorsExistHVAC(self, verboseMode: bool = True)
```

13.1.1.3.6 accim.sim.accim_IDFgeneration module

Generate IDFs.

```
accim.sim.accim_IDFgeneration.genIDF(self, ScriptType: Optional[str] = None, TempCtrl: Optional[str] = None, ComfStand=None, CAT=None, ComfMod=None, SetpointAcc=10000, CoolSeasonStart=121, CoolSeasonEnd=274, HVACmode=None, VentCtrl=None, MaxTempDiffVOF=20, MinTempDiffVOF=0.5, MultiplierVOF=0.25, VSToffset=[0], MinOToffset=[50], MaxWindSpeed=[50], ASTtol_start=0.1, ASTtol_end_input=0.1, ASTtol_steps=0.1, NameSuffix='', verboseMode: bool = True, confirmGen: Optional[bool] = None)
```

Generate IDFs.

```
accim.sim.accim_IDFgeneration.inputData(self, ScriptType: Optional[str] = None)
```

Input data for IDF generation.

13.1.1.3.7 accim.sim.accim_Main module

Class for accim.

```
class accim.sim.accim_Main.accimJob(filename_temp, ScriptType: Optional[str] = None, EnergyPlus_version: Optional[str] = None, TempCtrl: Optional[str] = None, verboseMode: bool = True, accimNotWorking: bool = False)
```

Bases: object

Class to start the accim process.

```
addBaseSchedules(verboseMode: bool = True)
```

Amend Schedule:Compact objects.

Checks Schedule:Compact objects needed for accim, and add them in case these are not in the model

```
addCurveObj(verboseMode: bool = True)
```

Add Curve Objects needed for VRFSYSTEM to work.

```
addDetHVACobj(EnergyPlus_version: Optional[str] = None, verboseMode: bool = True, SupplyAirTempInputMethod: Optional[str] = None)
```

Add Detailed HVAC objects for VRFSYSTEM to work.

```
addEMSActuatorsBase(ScriptType: Optional[str] = None, verboseMode: bool = True)
```

Add EMS actuators for accim.

```
addEMSSoutputVariableBase(ScriptType: Optional[str] = None, verboseMode: bool = True)
```

Add EMS output variables for Base accim.

Checks if some EMS output variables objects are already in the model, and otherwise adds them.

```
addEMSPCMBase(verboseMode: bool = True)
```

Add EMS program calling managers for Base accim.

Checks if some EMS program calling manager objects are already in the model, and otherwise adds them.

```
addEMSPProgramsBase(ScriptType: Optional[str] = None, verboseMode: bool = True)
```

Add EMS programs for Base accim.

Checks if some programs objects are already in the model, and otherwise adds them. :param verboseMode:

addEMSSensorsBase(*ScriptType*: *Optional[str]* = *None*, *verboseMode*: *bool* = *True*)
Add EMS sensors for accim.

addEMSSensorsExistHVAC(*verboseMode*: *bool* = *True*)

addEMSSensorsVRFsystem(*ScriptType*: *Optional[str]* = *None*, *verboseMode*: *bool* = *True*)
Add EMS sensors for VRF system accim.

addForscriptSchExistHVAC(*verboseMode*: *bool* = *True*)
Add FORSCRIPT Schedules for each zone in existing HVAC zones.

addForscriptSchVRFsystem(*verboseMode*: *bool* = *True*)
Add FORSCRIPT Schedules for each zone for VRFsystem.

addGlobVarList(*ScriptType*: *Optional[str]* = *None*, *verboseMode*: *bool* = *True*)
Remove existing Global Variable objects and add correct Global Variable objects for accim.

addIntVarList(*verboseMode*: *bool* = *True*)
Add Internal variables objects for accim.

addOpTempTherm(*verboseMode*: *bool* = *True*)
Amend ZoneControl:Thermostat:OperativeTemperature objects.
Add ZoneControl:Thermostat:OperativeTemperature objects for each zone.

addOutputVariablesDetailed(*Outputs_freq*: *Optional[any]* = *None*, *verboseMode*: *bool* = *True*)
Add Output:Variable objects for accim.

addOutputVariablesSimplified(*Outputs_freq*: *Optional[any]* = *None*, *TempCtrl*: *Optional[str]* = *None*, *verboseMode*: *bool* = *True*)
Add simplified Output:Variable objects for accim.
Remove all outputs and add only VFR outdoor unit consumption and operative temperature.

addOutputVariablesStandard(*Outputs_freq*: *Optional[any]* = *None*, *ScriptType*: *Optional[str]* = *None*, *TempCtrl*: *Optional[str]* = *None*, *verboseMode*: *bool* = *True*)
Add Output:Variable objects for accim.

addVRFsystemSch(*verboseMode*: *bool* = *True*)
Amend Schedule:Compact objects for VRFsystem.
Add Schedule:Compact objects needed for VRFsystem to work, other than AHST_Sch and ACST_Sch Schedules.

checkVentIsOn(*verboseMode*: *bool* = *True*)
Check ventilation settings.

genIDF(*ScriptType*: *Optional[str]* = *None*, *TempCtrl*: *Optional[str]* = *None*, *ComfStand*=*None*, *CAT*=*None*, *ComfMod*=*None*, *SetpointAcc*=*10000*, *CoolSeasonStart*=*121*, *CoolSeasonEnd*=*274*, *HVACmode*=*None*, *VentCtrl*=*None*, *MaxTempDiffVOF*=*20*, *MinTempDiffVOF*=*0.5*, *MultiplierVOF*=*0.25*, *VSToffset*=[*0*], *MinOToffset*=[*50*], *MaxWindSpeed*=[*50*], *ASTtol_start*=*0.1*, *ASTtol_end_input*=*0.1*, *ASTtol_steps*=*0.1*, *NameSuffix*=", *verboseMode*: *bool* = *True*, *confirmGen*: *Optional[bool]* = *None*)
Generate IDFs.

genOutputDataframe(*idf_filename*)

inputData(*ScriptType*: *Optional[str]* = *None*)
Input data for IDF generation.

listdir()

Return a list containing the names of the files in the directory.

path can be specified as either str, bytes, or a path-like object. If path is bytes,

the filenames returned will also be bytes; in all other circumstances the filenames returned will be str.

If path is None, uses the path='.'. On some platforms, path may also be specified as an open file descriptor;

the file descriptor must refer to a directory. If this functionality is unavailable, using it raises NotImplementedError.

The list is in arbitrary order. It does not include the special entries '.' and '..' even if they are present in the directory.

```
numpy = <module 'numpy' from '/home/docs/checkouts/readthedocs.org/user_builds/accim/envs/latest/lib/python3.7/site-packages/numpy/__init__.py'>
```

outputsSpecified(remove_or_keep: Optional[str] = None)

Remove duplicated Output:Variable objects for accim.

removeDuplicatedOutputVariables()

Remove duplicated Output:Variable objects for accim.

removeExistingOutputVariables()

Remove existing Output:Variable objects for accim.

saveaccim(verboseMode: bool = True)

Save IDF.

setAvailSchOn(verboseMode: bool = True)

Amend availability schedules.

Assign On Compact:Schedule to heating and cooling availability schedule names.

setComfFieldsPeople(EnergyPlus_version: Optional[str] = None, TempCtrl: Optional[str] = None, verboseMode: bool = True)

Amend PEOPLE objects so that accim can work.

Copy existing PEOPLE objects and adds AdaptiveASH55 and AdaptiveCEN15251 to Thermal Comfort Model types 1 and 2 fields of the existing People objects.

setPMVsetpoint(verboseMode: bool = True)

Sets PMV setpoints for temperature control.

takeOutputDataFrame(idf_filename, df_outputs_in, verboseMode)

13.1.1.3.8 accim.sim.accim_VRFsystem module

Add EnergyPlus objects only for VRFsystem.

accim.sim.accim_VRFsystem.addBaseSchedules(self, verboseMode: bool = True)

Amend Schedule:Compact objects.

Checks Schedule:Compact objects needed for accim, and add them in case these are not in the model

accim.sim.accim_VRFsystem.addCurveObj(self, verboseMode: bool = True)

Add Curve Objects needed for VRFsystem to work.

```
accim.sim.accim_VRFsystem.addDetHVACobj(self, EnergyPlus_version: Optional[str] = None, verboseMode: bool = True, SupplyAirTempInputMethod: Optional[str] = None)
```

Add Detailed HVAC objects for VRFsystem to work.

```
accim.sim.accim_VRFsystem.addForscriptSchVRFsystem(self, verboseMode: bool = True)
```

Add FORSCRIPT Schedules for each zone for VRFsystem.

```
accim.sim.accim_VRFsystem.addOpTempTherm(self, verboseMode: bool = True)
```

Amend ZoneControl:Thermostat:OperativeTemperature objects.

Add ZoneControl:Thermostat:OperativeTemperature objects for each zone.

```
accim.sim.accim_VRFsystem.addVRFsystemSch(self, verboseMode: bool = True)
```

Amend Schedule:Compact objects for VRFsystem.

Add Schedule:Compact objects needed for VRFsystem to work, other than AHST_Sch and ACST_Sch Schedules.

```
accim.sim.accim_VRFsystem.checkVentIsOn(self, verboseMode: bool = True)
```

Check ventilation settings.

```
accim.sim.accim_VRFsystem.setAvailSchOn(self, verboseMode: bool = True)
```

Amend availability schedules.

Assign On Compact:Schedule to heating and cooling availability schedule names.

13.1.1.3.9 accim.sim.accim_VRFsystem_EMS module

Add EMS objects only for VRFsystem.

```
accim.sim.accim_VRFsystem_EMS.addEMSSensorsVRFsystem(self, ScriptType: Optional[str] = None, verboseMode: bool = True)
```

Add EMS sensors for VRF system accim.

13.1.1.3.10 accim.sim.accis module

Run the function below to add the ACCIS.

This function transform fixed setpoint temperature building energy models into adaptive setpoint temperature energy models by adding the Adaptive Comfort Control Implementation Script (ACCIS)

```
class accim.sim.accis.addAccis(ScriptType: Optional[str] = None, SupplyAirTempInputMethod: Optional[str] = None, Output_type: Optional[str] = None, Output_freqs: Optional[any] = None, Output_keep_existing: Optional[bool] = None, Output_gen_dataframe: Optional[bool] = None, Output_take_dataframe: Optional[DataFrame] = None, EnergyPlus_version: Optional[str] = None, TempCtrl: Optional[str] = None, ComfStand: Optional[any] = None, CAT: Optional[any] = None, ComfMod: Optional[any] = None, SetpointAcc: float = 10000, CoolSeasonStart: any = 121, CoolSeasonEnd: any = 274, HVACmode: Optional[any] = None, VentCtrl: Optional[any] = None, MaxTempDiffVOF: float = 20, MinTempDiffVOF: float = 0.5, MultiplierVOF: float = 0.25, VSToffset: any = [0], MinOToffset: any = [50], MaxWindSpeed: any = [50], ASTtol_start: float = 0.1, ASTtol_end_input: float = 0.1, ASTtol_steps: float = 0.1, NameSuffix: str = "", verboseMode: bool = True, confirmGen: Optional[bool] = None)
```

Bases: object

13.1.1.3.11 Module contents

13.1.2 Module contents

CHAPTER
FOURTEEN

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

a

accim, 227
accim.data, 220
accim.data.data_postprocessing, 215
accim.data.data_preprocessing, 220
accim.run, 221
accim.run.run, 220
accim.sim, 227
accim.sim.accim_Base, 221
accim.sim.accim_Base_EMSS, 221
accim.sim.accim_ExistingHVAC, 222
accim.sim.accim_ExistingHVAC_EMSS, 222
accim.sim.accim_IDFgeneration, 223
accim.sim.accim_Main, 223
accim.sim.accim_VRFsystem, 225
accim.sim.accim_VRFsystem_EMSS, 226
accim.sim.accis, 226

INDEX

A

accim
 module, 227
accim.data
 module, 220
accim.data.data_postprocessing
 module, 215
accim.data.data_preprocessing
 module, 220
accim.run
 module, 221
accim.run.run
 module, 220
accim.sim
 module, 227
accim.sim.accim_Base
 module, 221
accim.sim.accim_Base_EMS
 module, 221
accim.sim.accim_ExistingHVAC
 module, 222
accim.sim.accim_ExistingHVAC_EMS
 module, 222
accim.sim.accim_IDFgeneration
 module, 223
accim.sim.accim_Main
 module, 223
accim.sim.accim_VRFsystem
 module, 225
accim.sim.accim_VRFsystem_EMS
 module, 226
accim.sim.accis
 module, 226
accimJob (class in accim.sim.accim_Main), 223
addAccis (class in accim.sim.accis), 226
addBaseSchedules() (accim.sim.accim_Main.accimJob method), 223
addBaseSchedules() (in module accim.sim.accim_VRFsystem), 225
addCurveObj() (accim.sim.accim_Main.accimJob method), 223

addCurveObj() (in module accim.sim.accim_VRFsystem), 225
addDetHVACObj() (accim.sim.accim_Main.accimJob method), 223
addDetHVACObj() (in module accim.sim.accim_VRFsystem), 225
addEMSSActuatorsBase() (accim.sim.accim_Main.accimJob method), 223
addEMSSActuatorsBase() (in module accim.sim.accim_Base_EMS), 221
addEMSSOutputVariableBase() (accim.sim.accim_Main.accimJob method), 223
addEMSSOutputVariableBase() (in module accim.sim.accim_Base_EMS), 221
addEMSPCMBase() (accim.sim.accim_Main.accimJob method), 223
addEMSPCMBase() (in module accim.sim.accim_Base_EMS), 221
addEMSPProgramsBase() (accim.sim.accim_Main.accimJob method), 223
addEMSPProgramsBase() (in module accim.sim.accim_Base_EMS), 221
addEMSSensorsBase() (accim.sim.accim_Main.accimJob method), 223
addEMSSensorsBase() (in module accim.sim.accim_Base_EMS), 221
addEMSSensorsExistHVAC() (accim.sim.accim_Main.accimJob method), 224
addEMSSensorsExistHVAC() (in module accim.sim.accim_ExistingHVAC_EMS), 222
addEMSSensorsVRFsystem() (accim.sim.accim_Main.accimJob method), 224
addEMSSensorsVRFsystem() (in module accim.sim.accim_VRFsystem_EMS), 226
addForscriptSchExistHVAC() (accim.sim.accim_Main.accimJob method),

224
C
 addForscriptSchExistHVAC() (in module *cim.sim.accim_ExistingHVAC*), 222
 addForscriptSchVRFsystem() (in module *cim.sim.accim_Main.accimJob* method), 224
 addForscriptSchVRFsystem() (in module *cim.sim.accim_VRFsystem*), 226
 addGlobVarList() (*accim.sim.accim_Main.accimJob* method), 224
 addGlobVarList() (in module *cim.sim.accim_Base_EMSS*), 221
 addIntVarList() (*accim.sim.accim_Main.accimJob* method), 224
 addIntVarList() (in module *cim.sim.accim_Base_EMSS*), 222
 addOpTempTherm() (*accim.sim.accim_Main.accimJob* method), 224
 addOpTempTherm() (in module *cim.sim.accim_VRFsystem*), 226
 addOutputVariablesDetailed() (in module *cim.sim.accim_Main.accimJob* method), 224
 addOutputVariablesDetailed() (in module *cim.sim.accim_Base_EMSS*), 222
 addOutputVariablesSimplified() (in module *cim.sim.accim_Main.accimJob* method), 224
 addOutputVariablesSimplified() (in module *cim.sim.accim_Base_EMSS*), 222
 addOutputVariablesStandard() (in module *cim.sim.accim_Main.accimJob* method), 224
 addOutputVariablesStandard() (in module *cim.sim.accim_Base_EMSS*), 222
 addVRFsystemSch() (*accim.sim.accim_Main.accimJob* method), 224
 addVRFsystemSch() (in module *cim.sim.accim_VRFsystem*), 226

D
 df() (*accim.data.data_postprocessing.Table* method), 215

E
 enter_vars_to_gather() (ac-

215
F
 format_table() (*accim.data.data_postprocessing.Table* method), 215

G
 gather_vars_query() (in module *cim.data.data_postprocessing.Table* method), 215
 genCSVconcatenated() (in module *cim.data.data_postprocessing*), 219
 generate_fig_data() (in module *cim.data.data_postprocessing.Table* method), 216
 genIDF() (*accim.sim.accim_Main.accimJob* method), 224
 genIDF() (in module *accim.sim.accim_IDFgeneration*), 223
 genOutputDataframe() (in module *cim.sim.accim_Main.accimJob* method), 224
 genOutputDataframe() (in module *cim.sim.accim_Base_EMSS*), 222
 give_address (class *in cim.data.data_preprocessing*), 220
 give_address_openssl (class *in cim.data.data_preprocessing*), 220
 give_address_ssl (class *in cim.data.data_preprocessing*), 220

I
 inputData() (*accim.sim.accim_Main.accimJob* method), 224
 inputData() (in module *cim.sim.accim_IDFgeneration*), 223

L
 listdir() (*accim.sim.accim_Main.accimJob* method), 224

M
 make_eplaunch_options() (in module *accim.run.run*), 220
 module
 accim, 227
 accim.data, 220
 accim.data.data_postprocessing, 215
 accim.data.data_preprocessing, 220
 accim.run, 221
 accim.run.run, 220
 accim.sim, 227

`accim.sim.accim_Base`, 221
`accim.sim.accim_Base_EMs`, 221
`accim.sim.accim_ExistingHVAC`, 222
`accim.sim.accim_ExistingHVAC_EMs`, 222
`accim.sim.accim_IDFgeneration`, 223
`accim.sim.accim_Main`, 223
`accim.sim.accim_VRFsystem`, 225
`accim.sim.accim_VRFsystem_EMs`, 226
`accim.sim.accis`, 226

N

`numpy` (`accim.sim.accim_Main.accimJob` attribute), 225

O

`outputsSpecified()` (`accim.sim.accim_Main.accimJob` method), 225
`outputsSpecified()` (in module `accim.sim.accim_Base_EMs`), 222

R

`removeDuplicatedOutputVariables()` (`accim.sim.accim_Main.accimJob` method), 225
`removeDuplicatedOutputVariables()` (in module `accim.sim.accim_Base_EMs`), 222
`removeExistingOutputVariables()` (`accim.sim.accim_Main.accimJob` method), 225
`removeExistingOutputVariables()` (in module `accim.sim.accim_Base_EMs`), 222
`removefiles()` (in module `accim.run.run`), 220
`rename_epw_files` (class in `accim.data.data_preprocessing`), 220
`runEp()` (in module `accim.run.run`), 220

S

`saveaccim()` (`accim.sim.accim_Main.accimJob` method), 225
`saveaccim()` (in module `accim.sim.accim_Base`), 221
`scatter_plot()` (`accim.data.data_postprocessing.Table` method), 216
`scatter_plot_with_baseline()` (`accim.data.data_postprocessing.Table` method), 217
`setAvailSchOn()` (`accim.sim.accim_Main.accimJob` method), 225
`setAvailSchOn()` (in module `accim.sim.accim_VRFsystem`), 226
`setComfFieldsPeople()` (`accim.sim.accim_Main.accimJob` method), 225
`setComfFieldsPeople()` (in module `accim.sim.accim_Base`), 221

`setPMVsetpoint()` (`accim.sim.accim_Main.accimJob` method), 225
`setPMVsetpoint()` (in module `accim.sim.accim_Base`), 221

T

`Table` (class in `accim.data.data_postprocessing`), 215
`takeOutputDataFrame()` (`accim.sim.accim_Main.accimJob` method), 225
`takeOutputDataFrame()` (in module `accim.sim.accim_Base_EMs`), 222
`time_plot()` (`accim.data.data_postprocessing.Table` method), 218

W

`wrangle_table()` (`accim.data.data_postprocessing.Table` method), 219