

Complete specification on Climate and Energy Assessment of Museums (CEAM) software

v. 1.0.1 published on ...

Marcin Zygmunt

KU Leuven, Department of Civil Engineering, Building Physics and Sustainable Design,
Kasteelpark Arenberg 40, BE-3001 Leuven, Belgium

*This document was delivered as an objective of the Climate2Preserv
research project founded by the Belgian Science Policy Office (BELSPO).*



TABLE OF CONTENTS

1. About the CEAM.....	4
2. Installation.....	6
2.1. Stand-alone version.....	6
2.2. Developer version.....	6
3. Input data.....	10
3.1. Input files.....	10
3.2. Input content and how to adjust it.....	11
4. How to use CEAM software.....	14
4.1. GUI overview.....	14
4.2. Computing.....	19
5. Modular structure: overview.....	21
5.1. Module 0:.....	21
5.2. Module 1:.....	23
5.3. Module 2:.....	23
5.4. Module 3:.....	23
5.5. Module 4:.....	23
5.6. Module 5:.....	23
5.7. Additional scripts.....	24
6. Results: overview.....	25
7. Appendixes.....	26
7.1. Quick start user manuals.....	26
7.2. Exemplary application #1: a complete input data.....	26
7.3. Exemplary application #2: an incomplete input data.....	26
7.4. Exemplary application #3: daily input data.....	26
7.5. Exemplary application #4: monthly input data.....	26



1. ABOUT THE CEAM

You can only start to save energy when you see how you are using it. It is essential for buildings housing cultural heritages, where standard approaches used to assess buildings' energy performance cannot be applied. These types of buildings are also uniquely managed, focusing mostly on heritage preservation. Energy-efficiency improvement is possible only when understanding the examined building. The selection of the actions considered for potential improvements is highly dependent on the examined case study and should be executed only after the approval of the delegated building managers. The knowledge of the examined building and its systems is a key factor for the appropriate building assessment no matter the quality and comprehensiveness of the used software.

The **C**limate and **E**nergy **A**ssessment for **M**useums (**CEAM**) was developed to allow its users a valuable and constructive energy- and climate-related overview of the examined heritage buildings. CEAM aims to help Federal Institutions preserve heritage. The developed decision-making support software provides building/collection owners information on how to adapt sustainable solutions to their buildings utilizing indoor climate management. CEAM is an open-source, data-driven tool scripted using Python and consists of a modular structure. This user-friendly tool allows for a comprehensive multicriteria analysis considering heritage preservation, indoor climate management, and energy efficiency improvement. The tool uses raw input data (records of the measured energy consumption and indoor/outdoor climate conditions) to provide useful outputs. The obtained outputs allow for a better understanding of the current building management and provide a helpful guide on how to, and to what extent, improve the energy efficiency of the examined case. Within the generated outputs, the following main aspects can be addressed:

- a comprehensive overview of input data,
- outline of the current building management,
- overview of potential energy savings.

The software is available online, as a stand-alone Windows app¹, as well as raw scripts². In this document, a complex overview of the CEAM software is presented. User guidelines, methodology overview, as well as examples are addressed in the following sections.

CEAM was developed by Marcin Zygmunt³ as a part of the Climate2Preserv (C2P) project⁴, founded by the Belgian Science Policy Office (BELSPO). The C2P project is the combined effort of project partners: KU Leuven⁵, KIK-IRPA⁶, and ULiège⁷.

¹ available here: [...](#)

² available here: <https://github.com/Climate2Preserv>

³ see more: <https://www.linkedin.com/in/marcin-zygmunt-06596115a/>

⁴ see more: <https://www.kikirpa.be/en/projects/climate2preserv>

⁵ see more: <https://www.kuleuven.be/kuleuven/>

⁶ see more: <https://www.kikirpa.be/en/>

⁷ see more: https://www.uliege.be/cms/c_8699436/en/uliege



Please address all the comments related to the CEAM software via the [contact email](mailto:climate2preserv@gmail.com): **climate2preserv@gmail.com**.



2. INSTALLATION

This section goes through the installation process. Based on the needs, CEAM can be used as a standalone program, or as a set of scripts (the so-called developer version) allowing for individual and unique modifications.

2.1. STAND-ALONE VERSION

By default, CEAM is a standalone software, working on a Windows operating system. After downloading¹, the tool is ready to use by the user. There is no need to install any additional software if the updated version of Windows is used. It is recommended to create a new folder and put all the downloaded file(s) there.

The **CEAM.exe** file should be used to run the tool. CEAM operates using the Graphical User Interface (GUI) presented in the next section (see more in section 4). Additionally, input files are required for a successful application of CEAM software (see more in section 3). Depending on the given scope of analysis computing time can take from a few up to dozens of minutes. Users are informed about the computation in progress with the visible pop-up windows; the window is closed automatically when calculations are finished. While computing, the numerous supporting files are generated in the given working directory; all the files should remain intact throughout calculations. Users are informed about the computing time, as well as the execution time when calculations are finished with the status textbox, as well as the status pop-up window, required to accept when analysis is over (followed by the system sound).

2.2. DEVELOPER VERSION

CEAM is an open-source tool, written in Python⁸: a high-level, multi-purpose, and easy-to-read programming language. Python has a straightforward structure and syntax with quite simple to maintain structural code. This programming language provides very high-level dynamic data types and supports dynamic type checking with both object-oriented and procedural programming styles. Finally, it has a sizeable standard library collection, applied for data collection, data exploration, and data visualization among others.

All the CEAM scripts are available² for further, individual use and modification by the users. Keep in mind, that it is the user's responsibility to verify the results obtained from the modified version of the CEAM software or any of its scripts. All the suggestions for further improvements of the tool are highly appreciated and can be provided via the [contact email](#).

In this part, a brief description of how to use CEAM in the so-called developer version is shown. It is assumed that those who decide to use this version of the tool have a basic knowledge of programming, as well as are familiar with Python programming language. For more information visit the GitHub repository².

First of all, you should have installed Python (version no older than 3.12) on your computer. CEAM was developed in the PyCharm⁹ Integrated Development Environment (IDE), but users can use any IDE based on their preferences (with potential adjustments required).

⁸ see more: <https://www.python.org/>

⁹ available here: <https://www.jetbrains.com/pycharm/>



Downloaded scripts should be placed as a new Python project, preferably in the working directory. For a complete application of the developer version, the following Python libraries are necessary to be installed:

- `csv`¹⁰: provides tools for reading and writing csv files,
- `customtkinter`¹¹: an extension of tkinter library,
- `datetime`¹⁰: functionalities for manipulating dates and times,
- `glob`¹⁰: library enables file pattern matching and pathname expansion,
- `holidays`¹²: handles holidays for predefined countries and regions,
- `importlib`¹⁰: allowing to import functionalities from different scripts,
- `keras`¹³: a high-level neural networks Application Programming Interface (API) used for building and training machine learning models,
- `matplotlib`¹⁴: library for visualizations, graphs in particular,
- `numpy`¹⁵: support for large, multi-dimensional arrays and numerical computations,
- `os`¹⁰: library offering functions to interact with the operating system,
- `pandas`¹⁶: library for various data manipulation and analysis,
- `psychrolib`¹⁷: allowing to generate and modify psychrometric charts,
- `re`¹⁰: library facilitates operations involving regular expressions for pattern matching,
- `scipy`¹⁸: provides scientific and technical computing tools,
- `shuttle`¹⁹: to handle file operations (e.g. copying and moving),
- `sklearn`²⁰: a machine learning library for data mining and analysis,
- `subprocess`¹⁰: enables running and managing additional processes,
- `sys`¹⁰: library provides access to system-specific parameters and functions,
- `tensorflow`²¹: a platform for machine learning and deep learning,
- `threading`¹⁰: library to run the subprocess (parallel tasks) in a separate thread,
- `time`¹⁰: functions for working with time-related tasks,
- `tkinter`¹⁰: library for building GUIs,
- `webbrowser`¹⁰: used to open Uniform Resource Locator (URL) in a web browser,
- `winsound`¹⁰: library for playing Windows-specific sound effects.

CEAM consists of 6 modules, contains 22 scripts, which are executed in a specific order regarding the requested scope. 1 additional script is also included as a part of the developer version of CEAM software, even though it is not a part of any module, as well as not used directly for computing. The initial operation of the tool is performed in the so-called mother file named `_00_CEAM_run.py`. Users can run the simulations from this script alone, selecting the preferred scope of analysis (by adjusting the list of executed modules or scripts). When a stand-alone version is used, users will operate on the GUI defined in the `CEAM_GUI.py` file.

¹⁰ see more: <https://docs.python.org/3/>

¹¹ see more: <https://pypi.org/project/customtkinter/0.3/>

¹² see more: <https://pypi.org/project/holidays/>

¹³ see more: <https://pypi.org/project/keras/>

¹⁴ see more: <https://pypi.org/project/matplotlib/>

¹⁵ see more: <https://pypi.org/project/numpy/>

¹⁶ see more: <https://pypi.org/project/pandas/>

¹⁷ see more: <https://github.com/psychrometrics/psychrolib>

¹⁸ see more: <https://pypi.org/project/scipy/>

¹⁹ see more: <https://pypi.org/project/shuttle/>

²⁰ see more: <https://pypi.org/project/scikit-learn/>

²¹ see more: <https://pypi.org/project/tensorflow/>



Multiple predefined links between different modules are present; the correct operation of the CEAM software is possible only if that order is maintained. The necessary adjustments can be made in each of the provided scripts; a compact description of each script is shown in **Table 1**. Each script is presented more in detail in section 5, distinguished by each module. Scripts are presented in the recommended application order, following the modular structure of the tool. Moreover, all the scripts are rich in comments to help users navigate the codes.

Table 1. Developer version of the CEAM software: scripts overview.

#	MODULE	NAME	DESCRIPTION
1	*	<i>CEAM_GUI.py</i>	This script generates the software GUI with all the functionalities.
2	**	<i>_00_CEAM_run.py</i>	The so-called <i>mother script</i> predefined the scope of analysis employing the computed scripts.
3		<i>_11_initial_csv_check.py</i>	This script initially checks the provided input files and performs folder and file management.
4		<i>_01_convert_units.py</i>	Script used to convert IU units to SI units.
5		<i>_02_pv_calc.py</i>	Script used to calculate vapor pressure.
6		<i>_03_degree_days.py</i>	Script used to calculate heating and cooling degree days ²² .
7		<i>_12_data_overview_stats.py</i>	This script verifies the potential incongruities of the provided inputs, as well as generates daily, monthly, and weekly averaged files.
8	0	<i>_04_extreme_weeks.py</i>	Script is used to define the so-called extreme weeks and save them for later assessments.
9		<i>_05_1_combi_files.py</i>	Script combining the input files into the so-called <i>combi_files</i> , with all input variables in one file, based on the frequencies of records.
10		<i>_05_2_adjust_combi_files.py</i>	Script adjusting the pre-generated <i>combi_file</i> with the desired date and time formats.
11		<i>_06_1_Td_Twb_calc.py</i>	Script used to calculate the dew point ²³ and wet bulb temperature ²⁴ .
12		<i>_06_2_factors.py</i>	Script used to calculate other factors for climate and energy assessments.
13		<i>_06_3_abs_humi.py</i>	Script used to provide outputs of absolute humidity ²⁵ .
14	1	<i>_13_data_overview_plots.py</i>	This script provides a graphical overview of the input data.
15	2	<i>_14_correlation.py</i>	This script examines correlations between the provided input variables.
16	3	<i>_15_energy_signature.py</i>	This script performs an energy signature assessment.

²² see more: <https://www.degreedays.net/>

²³ see more: https://en.wikipedia.org/wiki/Dew_point

²⁴ see more: https://en.wikipedia.org/wiki/Wet-bulb_temperature

²⁵ see more: https://en.wikipedia.org/wiki/Humidity#Absolute_humidity



17		<i>_07_CC_file.py</i>	Script used to generate the input file for climate class assessment.
18	4	<i>_08_psychrometric.py</i>	Script used to generate climate class assessment by means of psychrometric charts.
19		<i>_16_CC_overview.py</i>	This script performs climate class assessment.
20	5	<i>_09_prediction_inputs.py</i>	Script used to generate the input files required for the assessment of potential energy savings.
21	5	<i>_17_predictions.py</i>	This script provides the energy predictions by means of black box modeling.
22	5	<i>_18_prediction_outputs.py</i>	This script allows for the overview of the performed energy-saving assessment.
23	***	<i>_00_time_index_generator.py</i>	Script generates a selected data frame using the provided data format for the selected period. The output is provided as a csv file with 'Time' column.

* script providing GUI of the CEAM software

** script defining the scope of analysis performed using the CEAM software

*** supporting script, which is not a part of the CEAM software

The developer version, thus all the Python script, as well as exemplary input files and user manuals are available at the GitHub repository².



3. INPUT DATA

CEAM is a data-driven analytical software. Based on the provided data, a comprehensive overview of indoor climate control, as well as corresponding energy consumption can be performed. Furthermore, the assessment of the indoor climate can be obtained, as well as optimization of energy consumption is possible. In this section, a detailed overview of the necessary structure and content of the input data is discussed.

3.1. INPUT FILES

CEAM software works based on the provided input files; only 3 comma-separated values (csv) input files are required for successful computing. Yet, a certain input data structure is required. A detailed overview of the structure of the required input files, as well as some guidelines on how to prepare correct enters are shown in subsection 3.2. Records in the input files can be in either international²⁶ (SI) or imperial²⁷ (IU) system of units. A precise naming for the used input files is required, as follows:

- **ECD.csv**: Exterior Climate Data; a file containing information about the weather conditions measured preferably on-site or in the closest distance,
- **ICD.csv**: Indoor Climate Data; a file containing information about the inner conditions measured for the selected zone(s),
- **ED.csv**: Energy Data; a file containing historical records of the consumption used for specific needs (e.g. heating) distinguished for the preselected zone(s).

Exterior climate should be measured preferably on-site, or as close to the examined case study as possible. It is possible to use a weather file from a nearby meteorological station if measurements on-site are not possible/available. It is also possible to use (after adjustments) the Typical Meteorological Year²⁸ (TMY) files, yet it is not recommended. Finally, users should keep in mind that their weather data needs to be adjusted to follow the ECD file template. The required variables are exterior temperature (T_e) and exterior relative humidity (RH_e), where total solar radiation (I_s) is recommended, but not obligatory for computing.

Indoor climate should be measured for the selected room/zone. The measurements are done using specialized sensors. It is up to the user to decide how to address the examined case study, in particular, what assumptions to use during the preparation of the input files. The records used in input files should follow the assumptions made. The recommended approach is to select the most representative room/zone of the building and adjust the indoor climate measurement accordingly (i.e. use data from sensors located in the selected area). Data should be averaged if multiple records are available for the selected room/zone, or the records from the most representative sensor (e.g. located in the middle of the area) should be used. Finally, users should keep in mind that their indoor climate data needs to be adjusted to follow the ICD file template: the required variables are interior temperature (T_i) and interior relative humidity (RH_i).

²⁶ see more: https://en.wikipedia.org/wiki/International_System_of_Units

²⁷ see more: https://en.wikipedia.org/wiki/Imperial_units

²⁸ A collation of historical weather data derived from a multi-year time series selected to present the unique weather phenomena with annual averages that are consistent with long-term averages. Some TMY files can be found and downloaded from here: <https://energyplus.net/weather>



Energy consumption can be measured via various approaches and for various levels of detail for the indoor climate management system, typically called HVAC (heating, ventilation, and air conditioning). Moreover, the consumption can be measured at different stages of usage: building-, system-, unit-, zone-, or room-level. The available measured energy consumption should be adjusted in accordance with the examined room/zone. Modern HVAC systems are operated via a building management system (BMS) allowing to access more valuable and distinguished information regarding energy consumption. All the above-mentioned information should be considered while preparing the ED file. It is recommended to work with a system specialist to make a correct input file. Finally, users should keep in mind that their energy consumption data needs to be adjusted to follow the ED file template. The provided ED file can consist of any combination of the following consumptions: heating (*D1*), cooling (*D2*), ventilation (*D3*), humidification (*D4*), and total (*D5*).

The collected and used input data is crucial for a successful CEAM application. The granularity of the data (its frequency) affects the available assessment and its accuracy. Instinctively, the more detailed the data the better and more comprehensive the examination; yet in practice, it is more compound and case-dependent. Detailed interval data (hourly frequencies for CEAM software) allows to reveal of patterns/profiles of energy demand, while daily or monthly inputs are accurate only for simpler methods and data overviews, based on the performed assumptions.

Some exemplary sets of input files are shared with the software^{1,2}. These files can be considered as templates to prepare inputs for future use of CEAM. The description of each set, their application, possible scope of analyses, as well as obtained outputs are reviewed in the section 7.

3.2. INPUT CONTENT AND HOW TO ADJUST IT

For every csv file, we can distinguish headers (present in the first line of the file), separator (symbol separating each record/variable), as well as list(s) of records (e.g. measured data) with corresponding time format. A separator should be the same throughout each file, avoiding confusion with the used decimal point; a typical separator is a comma or semicolon²⁹. The headers used should clearly describe the collected records in the given column. CSV files can be opened by MS Excel, but it is recommended to open them as text files, using e.g. Notepad app³⁰.

CSV input files required by CEAM software have a predefined simple structure that needs to be respected. All the provided input files will be safely copied and saved before any actions are performed by the CEAM software. All the headers should be in the very first row of the file with proper spelling (see more in **Table 2**), yet the tool handles various fonts and placements. Header verifications will be performed; CEAM will detect any incompatibility (font, placement) and reorganize the header to follow the predefined template (sub- and super-scripts are skipped in csv files). Each file must contain the *Time* column, with the same known structure of the formatting³¹ throughout each file. It is essential to know the positioning of the basic

²⁹ The naming of the csv (comma-separated value) file is nonspecific because the used separator can be selected by the user, used software, or operating system.

³⁰ available here: <https://notepad-plus-plus.org/>

³¹ it is impossible to understand correctly the dates without knowing the context of its formatting



components of the date, day, month, and year in particular. The used separator of the components, as well as the style for time formatting, will be automatically evaluated by CEAM. The *Time* format will be adjusted before computing: the default format (*dd.mm.yyyy HH:MM*) will be used with a dot as a separator for all input files. Finally, the data frame frequency (the time between each record) should remain the same throughout the file. The software will detect the most common frequency used, and adjust the input files accordingly. CEAM supports hourly, daily, or monthly frequencies. The used unit system (SI or IU) should be the same for all the input files. The software will automatically check the used separators in the provided csv files, and then modify them if necessary (comma is used as default). The user needs to verify the separator and decimal point not match. It should also be checked if the provided records are aligned with the corresponding headers. The zeroth values should be input as 0, while no record should be left as empty evidence.

Table 2. Information on the headers of the input files for computing using CEAM.

INPUT FILE	HEADER	DESCRIPTION
<i>ECD.csv</i>	<i>Time</i>	Indexing for the given data frame, using the predefined format; the default format is: <i>dd.mm.yyyy HH:MM</i>
	T_e	Exterior air temperature in [C] (SI) or [F] (IU)
	RH_e	Exterior relative humidity in [%]
	I_s	Total solar radiation in [Wh/m ²]
<i>ICD.csv</i>	<i>Time</i>	Indexing for the given data frame, using the predefined format; the default format is: <i>dd.mm.yyyy HH:MM</i>
	T_i	Interior air temperature in [C] (SI) or [F] (IU)
	RH_i	Interior relative humidity in [%]
<i>ED.csv</i>	<i>Time</i>	Indexing for the given data frame, using the predefined format; the default format is: <i>dd.mm.yyyy HH:MM</i>
	$D1$	Demand group no.1: heating, in [Wh] (SI) or [BTU] (IU)
	$D2$	Demand group no.2: cooling, in [Wh] (SI) or [BTU] (IU)
	$D3$	Demand group no.3: ventilation, in [Wh] (SI) or [BTU] (IU)
	$D4$	Demand group no.4: humidification, in [Wh] (SI) or [BTU] (IU)
	$D5$	Demand group no.5: total consumption, in [Wh] (SI) or [BTU] (IU)

Considering all the above mentioned, exemplary file adjustments of the input file that CEAM software will perform before computing are shown in subsection 7.6 (examples #1 and #2).

The available measurement data can be provided to CEAM in various ways. The most universal approach to generate the necessary input files is to create the csv file using MS Excel. User should place the required variables in the obligatory order and put all the records in a single spreadsheet. The proper positioning and length of each column should be verified with the *Time* index. When using MS Excel it should be verified, if the time format was not automatically adjusted by the default setting of the software (the constant and uniform time format should be used in a single csv file). If all the above-mentioned is provided, the file



should be saved as a csv file, using the option 'CSV (Comma delimiter) (*.csv)' with proper naming (e.g. *ECD.csv*).

Additional caution is required when the ED file is generated. The consumption records should be distinguished due to the fuel used. In practice, a separate file should be made for electricity consumption, and another one(s) if other fuel(s) is used (e.g. gas). Separate analyses by means of CEAM are required if the examined building uses more than one fuel, rotating the ED files.

Moreover, the total consumption (*D5*) should be assumed with care: awareness of the collected historical data is required. By default, the *D5* is a sum of all preceding demands, but difficulties in distinguishing individual consumptions are highly possible. For instance, the user might have information on heating demand (*D1*), as well as a total consumption *D5* (from e.g. energy invoice), which combines all the electricity consumption (used for heating, cooling, ventilation, and humidification). Therefore, the user does not know distinguished consumptions for *D2*, *D3*, or *D4*, but still can perform individual assessments for *D1* and *D5* demands. It is recommended to use only *D5* consumption for the evaluation or to combine numerous demands (if available) for an additional variable (*D5*) to be examined. The above-mentioned is presented in examples #3 and #4 in subsection 7.6. The selection of the demands to be included in the analysis is done from the GUI level (see more in section 4).

The analysis by means of CEAM is performed for hourly inputs, yet the provided tool allows the use of daily or monthly energy consumption. If daily or monthly records are provided, the tool will first adjust the given records for hourly archives during preprocessing in Module 0; this process is explained in subsection 5.1.

The user has full flexibility in choosing the span of the examined data frame, nevertheless, it is recommended to adjust all the input files to the same length. Using a full calendar year is also recommended, but not strictly from 1st January till 31st December. The shortest recommended period of analysis is one month.

Within the CEAM software documentation, **four** sets of exemplary input files are provided. Within the provided arrays various combinations of stored records are included and examined in the exemplary application of CEAM software (see more in subsections 7.2-7.5). Users should use them to get familiar with the required template and to help them adjust their input files. The regularity of the inputs is a key to a successful computation with CEAM software.

Finally, the user can use the provided `_00_time_index_generator.py` script² to generate any data frame with the given date and time formats, as well as frequency. The generated file will contain the *Time* column with all the records for the predefined data frame, which can be further used to generate input files for CEAM assessment. This script is not a part of the stand-alone version of the provided software; knowledge of Python programming is required for a successful execution. Moreover, the proper and precise adjustment of the given historical records with the generated data frame is essential, in particular, to respect all the gaps in the examined archives.



4. HOW TO USE CEAM SOFTWARE

CEAM software interface is overviewed in this section. The explanation of the GUI is shown, required actions are discussed, and the computing is introduced.

4.1. GUI OVERVIEW

The GUI of the tool has 3 main parts (see **Figure 1**): the information header (assigned as 1, in yellow), the modules (assigned as 2, in blue), and the additional information (assigned as 3, in green) sections.

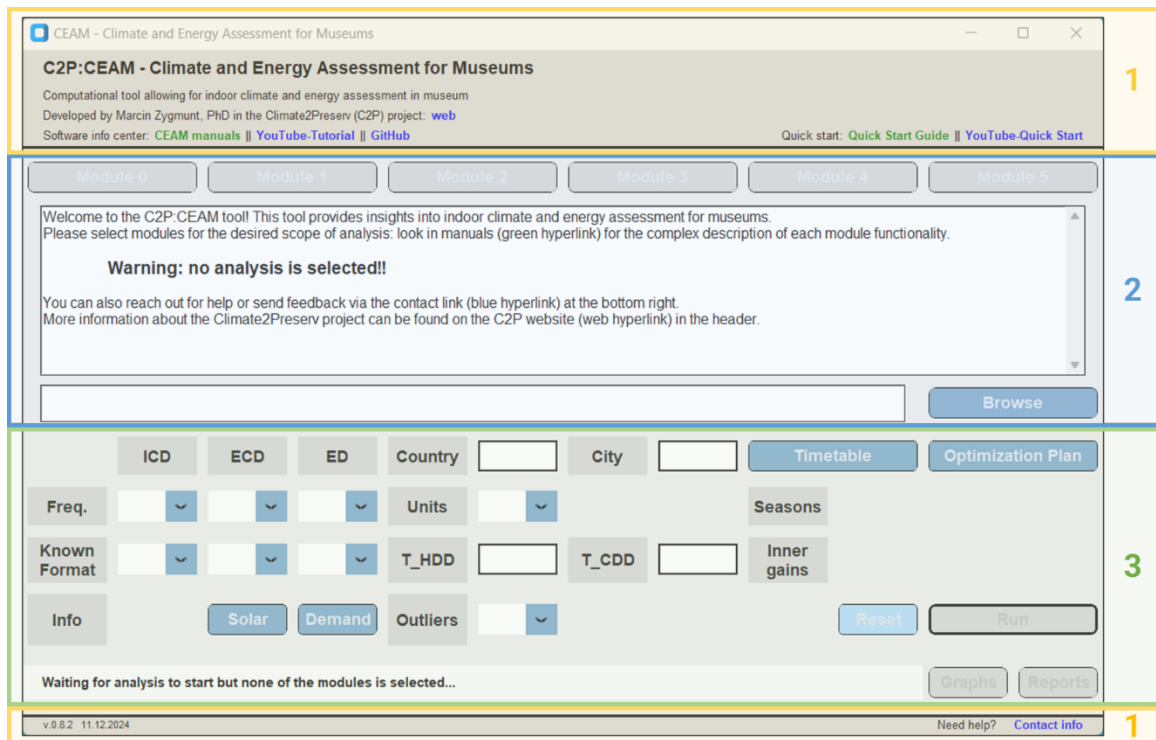


Figure 1. The GUI of the CEAM software with main sections being highlighted.

The information header consists of some contact information (C2P project website⁴), as well as provides active hyperlinks to the software manuals³² (this document in particular), quick start guide³³ (included in subsection 7.1), YouTube tutorials³⁴, as well as GitHub² repository. In the footnote, the information on the current version is shown, with the release date (left alignment), as well as a direct contact link to the developer³ (right alignment).

³² available here: ...

³³ available here: ...

³⁴ available here: ...

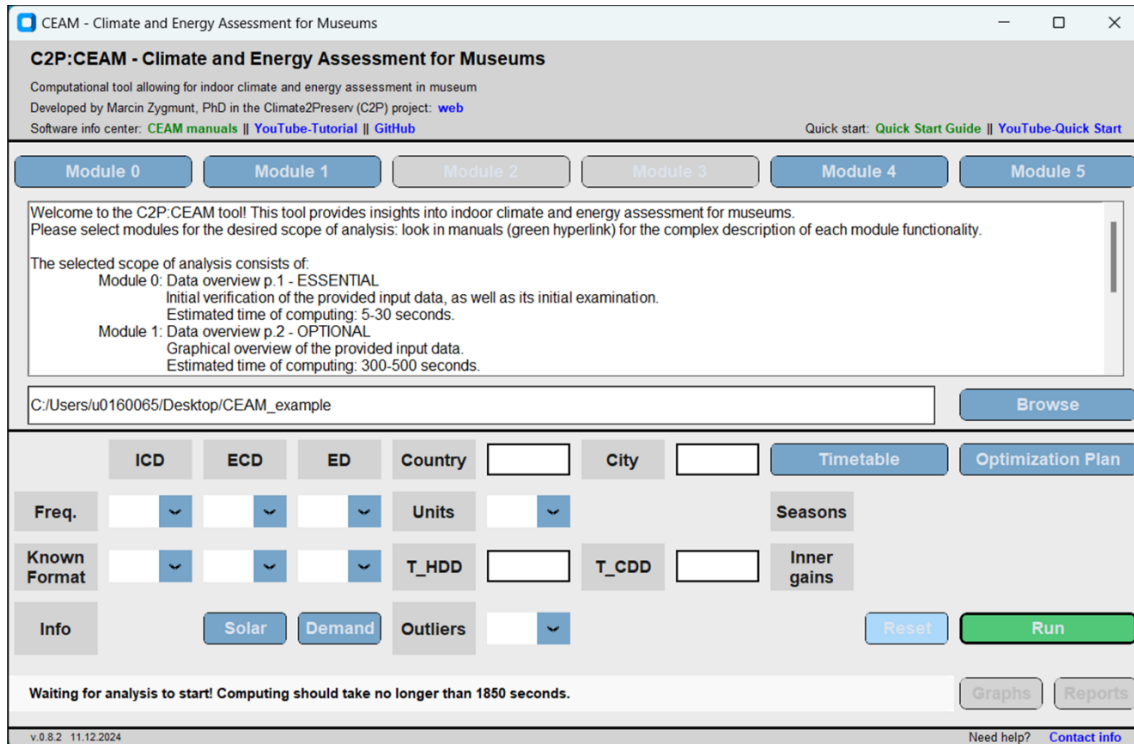
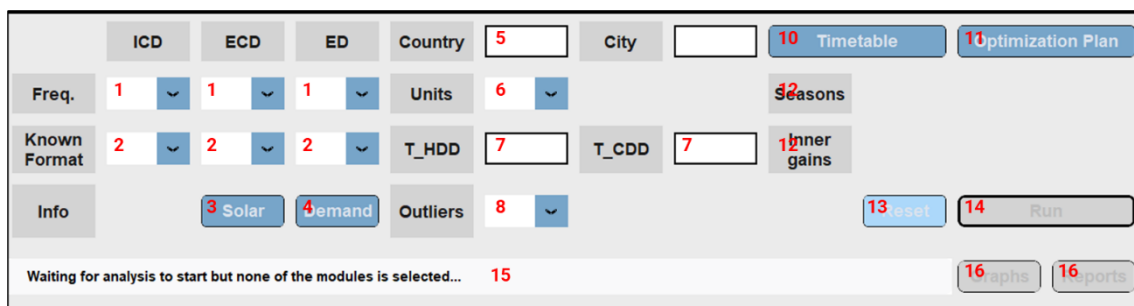


Figure 2. The GUI of the CEAM software with choices in the module section made.

The modules section consists of 6 functional buttons, a scrollable textbox, as well as a browse box. The text displayed in the textbox corresponds to the selected scope of analysis, which is defined by activating the module buttons. By default, all the module buttons are off (in gray), and you can activate them by simply clicking on them (the button will turn blue). Users can freely define the scope of the assessment, yet during the very first run it is obligatory to start with *Module 0* included (see more in subsection 5.1). The displayed text in the textbox will update with the performed selection, showing a brief description of each module (see more in section 5), with the estimated computing time. Finally, the browse button should be used to show the location of the working folder with the input files. By clicking the button, the typical system file explorer pop-up window will open to select the chosen localization (path). The chosen path will appear in the adjacent textbox when selected. When the path is selected, the run button (in the additional information section) will become active (it turns green from gray). The CEAM GUI with exemplary selections made in the module section is shown in **Figure 2**.



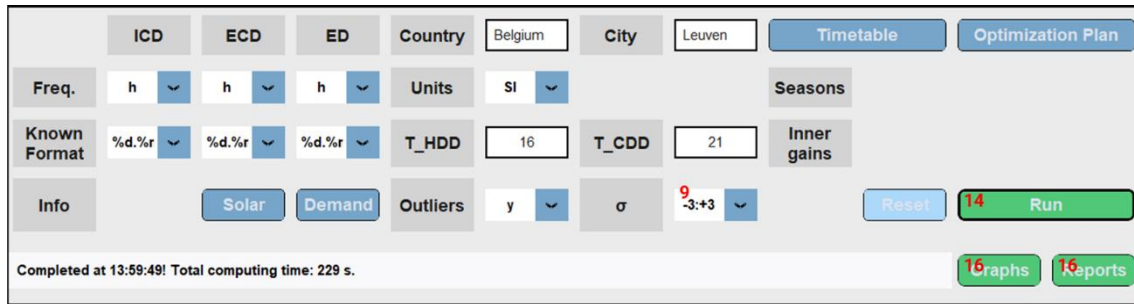


Figure 3. The additional information part of the CEAM GUI: default and empty stage (on top) and the fulfilled stage (on bottom) with assignments (in red).

The information part requires the most attention of the user. In this section, all the necessary additional information regarding the provided inputs and the examined case is required (see **Figure 3**). Firstly, the record frequencies (assigned as 1) in all the input files should be selected from the expandable option menu: the user can choose between monthly (*m*), daily (*d*), and hourly (*h*) granularities. The hourly frequency should also be selected if the provided records have a lower frequency (e.g. 15min). Next, the format of the *Time* records in all the input files should be defined (assigned as 2). The choice in expandable option menus is not active till the selection of the given frequencies is made, as well as the available selection is adjusted by the previous selection. The list of available *Time* formats for given frequencies is shown in **Table 3**.

Table 3. Overview of the *Time* format definition in CEAM.

FREQUENCY	AVAILABLE FORMATS	COMMENTS
(h)ourly	%Y.%m.%d %Y.%d.%m %m.%d.%Y %d.%m.%Y	1. The selection should be made without concerning the separation of the components. CEAM software will verify the used separator and adjust it to the default (dot). 2. For hourly files the time formatting will be handled by the tool. *
(d)aily	%Y.%m.%d %Y.%d.%m %m.%d.%Y %d.%m.%Y	
(m)onthly	%Y.%m %m.%Y	

* as long as the provided format for time records is either only hourly (*HH*), hours and minutes (*HH:MM*), or hours, minutes, and seconds (*HH:MM:SS*).

The ECD file can contain solar radiation data; thus, it must be specified (assigned as 3). After clicking on the *Solar* button, a new pop-up window will open, with a *y/n* (yes/no) expandable option available. The selection should be confirmed with the green *Accept* button (see in **Figure 4**). The ED file can contain various demands available for the assessment; thus it is required to specify (assigned as 4). After clicking on the *Demand* button, a new pop-up window will open, with a checkbox list to be adjusted. By default, none of the demands is selected. After the appropriate selection, the choice should be confirmed with the green *Accept* button (see in **Figure 4**).

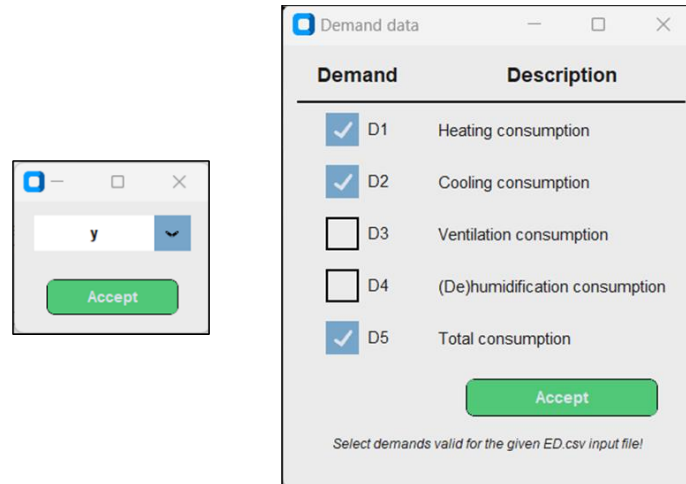


Figure 4. The additional pop-up windows for Solar (on left) and Demand (on right) definitions.

The *Country* textbox (assigned as 5) should be filled in with the examined localization. Users can input the full country name, as well as the corresponding abbreviation. CEAM will handle various fonts, yet the spelling must be correct (see example #5 in subsection 7.6). A full list of the available countries can be found in **Table 5** in subsection 7.8. If no input is provided in the *Country* textbox, Belgium is assumed as default. The input in the *City* textbox has no further functionality in the current version of the CEAM software.

Units used in the prepared input files should be defined. The *Units* expandable option is available (assigned as 6), with SI and IU choices. All the records in input files will be adjusted based on the selection made, in particular, converted into SI if IU is selected. If no selection is made, the SI system is assumed as default.

Two textboxes (assigned as 7) are given to provide information about heating degree day (HDD) and cooling degree day (CDD) temperatures²², named accordingly as T_{HDD} and T_{CDD} . The input values should follow the unit system defined in the *Units* section. If no value is provided, or if the provided input is incorrect (not a number) then the typical values (in °C) for Belgium are assumed³⁵; 16 and 21 respectively for T_{HDD} and T_{CDD} .

CEAM is capable of examining the outliers in the provided records: it should be selected in the *Outliers* expandable option (assigned as 8). If *y* (yes) option is selected, the range of the considered outliers should be defined in σ section (assigned as 9). The outliers examination is performed based on the standard deviation³⁶ series, selected from the given expandable option. The outliers assessment is further explained in the 5.1 subsection.

The operation hours of the examined building should be defined in the *Timetable* section (assigned as 10). After clicking on the *Timetable* button, a new pop-up window will open, with a checkbox list and corresponding textboxes to be adjusted (see **Figure 5**). Users should select the days when the building is open (i.e. the default setpoint temperature and humidity are fixed) and define the corresponding opening hours (in *From* and *Till* textboxes). Only the complete inputs are considered, i.e. day with working hours (a full hour should be input, using 24-hour timing). The definition should be confirmed with the green *Accept* button. An

³⁵ according to: [...](#)

³⁶ see more: https://en.wikipedia.org/wiki/Standard_deviation



exemplary fulfillment of the *Operational Schedule* window is explained in subsection 7.6 (example #6).

The energy optimization is performed based on the short-term modification of the given setpoints for temperature and humidity management (see more in subsection 5.6). In the *Optimization Plan* section (assigned as 11) the predefined scenarios for the optimization should be selected. Users should choose the desired to-be-examined *Schedule*, based on the provided description in the *Details* column (see **Figure 5**). The selection should be confirmed with the green *Accept* button.

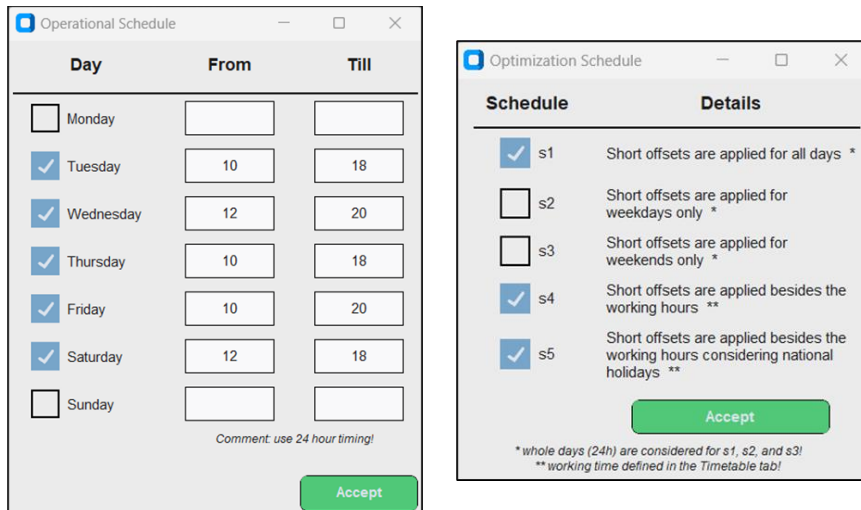


Figure 5. The additional pop-up windows for Operational Schedule (on left) and Optimization Schedule (on right) definitions.

The part of the GUI containing the *Seasons* and *Inner gains* labels (assigned as 12) is not yet developed. It is planned to include these additional features in the future version of CEAM.

The information part of the GUI has a *Reset* button (assigned as 13). All the choices made in the information part can be erased after clicking this button (it does not affect selections made in the module selection).

This part includes the *Run* button (assigned as 14), which initially is inactive (in gray) and turns green after providing the working directory. After starting the calculation, an *Analysis* pop-up window will show (see **Figure 6**) and the *comment textbox* (assigned as 15) will present the status of computing. When computing is finished, the window will update (see **Figure 6**), followed by the system sound. Users are required to confirm the end of the calculation by pressing the *OK* green button.



Figure 6. The computing pop-up Analysis window: calculations in progress (on the left) and finished computing (on the right).

Next to the comment textbox, the last two buttons are located. The *Graphs* and *Reports* buttons (assigned as 16) are initially inactive (in gray), and will become active (in green) if folders with button names are present in the selected working directory (the given path in the



modules section). If the corresponding folder is present, it will open the typical system explorer window after clicking the matching button. Therefore, those two buttons will become active whenever the calculations of *Module 0* are completed (either in the given run of the CEAM or during any of the previous usage).

Some sets of the CEAM GUI inputs are presented in subsections 7.2-7.5 where the exemplary applications are discussed.

4.2. COMPUTING

The computing for the selected scope of the analysis (i.e. modules selection) is performed upon clicking on the *Run* button. The simulation being in progress is indicated with the lively updating *Comment* textbox, as well as the *Analysis* pop-up window (see **Figure 6**). The scope of the analysis is presented in detail in section 5, distinguishing each module. The computing time varies mostly depending on the selected scope of assessment, as well as comparatively with the specs of the used PC. When computing is finished, the user is informed by the sound as well as asked to close the *Analysis* window.

All the generated outputs are saved in the given working directory: the same where the input files are located. The files and outputs management is performed automatically by the software during computing. All the archives are sorted in a way presented in **Table 4**.

Table 4. Files and outputs management by means of CEAM software.

FOLDER/FILE	DESCRIPTION
<i>graphs</i>	The <i>graphs</i> folder contains all the graphical outputs of the performed assessment. The folder includes 5 subfolders, that store the outputs of different modules. Each graph generated is named to help users navigate: a complete list of possible generated graphs is shown in subsection 7.8.
<i>others</i>	The <i>others</i> folder contains all the temporary files generated by the CEAM software for a successful analysis.
<i>raw_data</i>	The <i>raw_data</i> folder stores the original input files, without any changes.
<i>reports</i>	The <i>reports</i> folder consists of the text files with the summaries and comments on the performed assessments. The generated reports will be used in the future version of the CEAM to provide one main report as an extra output of the performed analysis.
<i>d_comb.csv</i>	One of the key data files used during computing, with daily records. The file was generated during <i>Module 0</i> from the input files.
<i>h_comb.csv</i>	One of the key data files used during computing, with hourly records. The file was generated during <i>Module 0</i> from the input files. This file is considered the default input file for the assessments by means of CEAM (<i>Modules 1-5</i>).
<i>h_comb_CC.csv</i>	It is an extended version of the <i>h_comb.csv</i> file required for Climate Class assessment (the file is generated only if <i>Module 4</i> is examined).
<i>m_comb.csv</i>	One of the key data files used during computing, with monthly records. The file was generated during <i>Module 0</i> from the input files.
<i>w_comb.csv</i>	One of the key data files used during computing, with weekly records. The file was generated during <i>Module 0</i> from the input files.
<i>terminal_records.txt</i>	The text file contains all the information which typically shown in the terminal (code-level). In case of facing errors while computing, users



can share this file with the developers for help. The file is constantly updating while the simulation is on. There is no practical application of this file for users.

var.txt

The text file is generated every time the Run button is pressed, collecting all the provided inputs in the CEAM GUI. The *var.txt* file is used by the tool for successful computing. The content of the file is shown in subsection [7.8](#).

It is important to mention that whenever the CEAM is run again (re-run of the analysis) the generated outputs will overwrite the existing ones (even if the input variables are different). Thus, if derivative analysis is required, users should copy the initially obtained results if they are needed. Also, users should not intervene with the files while CEAM is computing.

When the desired assessment is completed, users can freely close the software without any further actions. The input definition process is required to be repeated every time the software is open.



5. MODULAR STRUCTURE: OVERVIEW

CEAM has a modular structure consisting of Python codes. Each of the modules provides a specific functionality, delivered by primary and supplementary scripts. The naming of primary codes is initiated with *1*, while the supplementary scripts start with *0* (see **Table 1**). The numbering of the modules starts with 0 because the first one is essential for suitable computing performed by means of CEAM software. A description of each module is given in the following subsections; each script and its functionality are mentioned, as well as included methods are presented.

5.1. MODULE 0: DATA OVERVIEW PART 1

Module 0 is the first one of the available six. The main purpose of this module is to verify the provided inputs, as well as to perform all the necessary adjustments required for further assessments, performed within the following modules of CEAM software. Thus, *Module 0: Data Overview Part 1* is obligatory to be included in each run of the CEAM software. The result of Module 0 completing is a set of key data files, in particular *d_comb.csv*, *h_comb.csv*, *m_comb.csv*, and *w_comb.csv* (see **Table 4**). This module should be run only once for each assessment, repeated every time the new inputs are provided, using the raw files. It consists of 10 scripts (2 primary and 8 supplementary codes) presented below (in the execution order).

INITIAL CVS CHECK

The initial csv check is performed by the *_11_initial_csv_check.py* script, and the main objective of this part is to verify if the provided input files are correct (following the software requirements). This script generates all the supplementary folders in the working directory to support transparent file management, which includes renaming, moving, coping, merging, as well as deleting the files. There are several main functionalities performed by this script, listed as follows:

- check and (if needed) change the separator,

A function providing a separator (a symbol used to discrete each record) verification, and adjusting it to the default one if needed (see example #2 in subsection 7.6).

- process the input files,

A function is used to adjust the provided structure of the input files to follow the default one. It focuses on the provided headers: their font and placement. If the restructuring is necessary, the whole column is reordered (records follow the header); see example #1 in subsection 7.6.

- check and (if needed) change the date format,

This part of the module is to check and fix the date format in the *Time* column of csv input files. It verifies the given format based on the information provided by users in the GUI and adjusts it to the default one (see example #2 in subsection 7.6).

- adjustment of the ED file with lower (daily or monthly) frequency data,

...

- examine outliers,



A supplementary function that is executed only upon selection in the GUI. If an outliers assessment is selected (see subsection 4.1) the script will detect the records outside of the given range of distribution, and consider them as outliers. All the outliers will not be included in the key data files, keeping empty records of data for the selected periods. The range of the considered data points is defined by the standard deviation, calculated with eq.1 (see subsection 7.7). The visualization of the considered outliers' shares in a normal distribution³⁷ is shown in Figure 7³⁸. The mentioned ranges are based on the mean value (μ) and the scopes defined by subtraction or addition of the σ (i.e. how many standard deviations we are away from the mean). The range of $\mu \pm \sigma$ contains approx. 68.3% of the data, range of $\mu \pm 2\sigma$ approx. 95.4%, while range of $\mu \pm 3\sigma$ approx. 99.7%. Thus, the selection made in the GUI should be thoughtful: if the data is corrupted, with numerous irregularities the narrower range is recommended, while in most cases, where the collected data is accurate, the broader range (or even assessment without outliers examination) should be selected.

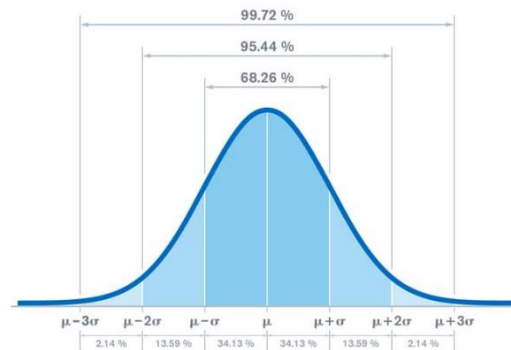


Figure 7. Standard deviations in normal distribution.

- check for the missing values,

A function used to verify if records are missing, thus no data is present for a given *Time* log. The function also reports on the length and appearance of shortages in data.

- most common frequency,

Functionality used to check the granularity of the record (e.g. 15min).

- fill missing timestamps,

A function used to detect the missing gaps in data, as well as to fill it up with empty records (rows); see example # in subsection 7.6. The added records follow the prerecorded frequency, as well as the *Time* format.

- averaging data,

Script performing records averaging to the hourly values (only if frequencies in the provided input files are shorter, e.g. 15min); see example #8 in subsection 7.6.

- convert units.

Functionality is provided in the `_01_convert_units.py` script to substitute the records given in the IU unit system to the default SI system. The considered parameters are temperature (both T_e and T_i in the corresponding ECD and ICD files) as well as all the given energy demands in the ED file.

³⁷ see more: https://en.wikipedia.org/wiki/Normal_distribution

³⁸ source: <https://www.sixsigmadaily.com/standard-deviation-6-steps-to-calculation/>



_02_pv_calc

_03_degree_days

_12_data_overview_stats

_04_extreme_weeks

_05_1_combi_files

_05_2_adjust_combi_files

_06_1_Td_Twb_calc

_06_2_factors

_06_3_abs_humi

The initial check of the provided input files is made with this script: verification of the headers, separators, timestamp formats, as well as missing records in particular.

5.2. MODULE 1: DATA OVERVIEW PART 2

...

5.3. MODULE 2: ...

...

5.4. MODULE 3: ...

...

5.5. MODULE 4: ...

...

5.6. MODULE 5: ...

...



5.7. ADDITIONAL SCRIPTS

...



6. RESULTS: OVERVIEW

...



7. APPENDIXES

In this section, all the appendixes of the provided manuals are presented. In subsection 7.1 the quick start user manuals are included; this document is also available separately. Subsections 7.2-7.5 consist of exemplary applications of CEAM software for energy and climate optimization. Subsection 7.6 includes all the examples with supplementary explanations mentioned throughout this document.

7.1. QUICK START USER MANUALS

...

7.2. EXEMPLARY APPLICATION #1: A COMPLETE INPUT DATA

7.3. EXEMPLARY APPLICATION #2: AN INCOMPLETE INPUT DATA

7.4. EXEMPLARY APPLICATION #3: DAILY INPUT DATA

7.5. EXEMPLARY APPLICATION #4: MONTHLY INPUT DATA

7.6. EXAMPLES

In this subsection, all the examples mentioned throughout the document are presented. The examples are included for a better understanding of the developed CEAM software and to simplify the usage process of the tool.

Example #1

The input of:

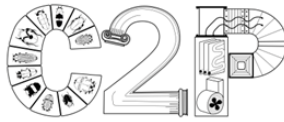
```
Time,d1,D3,d4,d5,D2
01.01.2024 00:00,10,30,40,50,20
01.01.2024 01:00,100,300,400,500,200
...
31.12.2024 23:00,1000,3000,4000,5000,2000
```

will be restructured as:

```
Time,D1,D2,D3,D4,D5
01.01.2024 00:00,10,20,30,40,50
01.01.2024 01:00,100,200,300,400,500
...
31.12.2024 23:00,1000,2000,3000,4000,5000
```

Thus, the tool will restructure the headers (font and positioning), as well as reorganize the given records for each column to correspond with the default template. The following modifications were made:

- verifying the font of the headers and adjusting it (e.g. from *d1* to *D1*),



- reorganization of the columns of records to follow the default structure.

Example #2

The input of:

```
Time;D1;D2;D3;D4;D5
2024/01/01 00;10;0;;;
2024/01/01 01;100;0;;;
...
2024/12/31 23;1000;0;;;
```

will be restructured as:

```
Time,D1,D2,D3,D4,D5
01.01.2024 00:00,10,0,,,
01.01.2024 01:00,100,0,,,
...
31.12.2024 23:00,1000,0,,,
```

Thus, the tool will change the used separator (from semicolon to comma) and adjust the *Time* column to correspond with the default template. The following modifications were made:

- swapping components separator from a forward slash (/) to a dot (.),
- changing the date format (from *yyyy.mm.dd* to *dd.mm.yyyy*),
- changing the time format (from *HH* to *HH:MM*),
- adjust the gap between date and time (from three to one white space).

Example #3

The input for the ED file consists of *D1*, *D2*, and *D4* consumptions. Users can define the *ED.csv* file as:

```
Time,D1,D2,D3,D4,D5
01.01.2024 00:00,1000,0,,200,
01.01.2024 01:00,1000,10,,200,
...
31.12.2024 23:00,1000,0,,200,
```

or with an additional attachment of *D5*, as a sum of the preceding demands, as follows:

```
Time,D1,D2,D3,D4,D5
01.01.2024 00:00,1000,0,,200,1200
01.01.2024 01:00,1000,10,,200,1210
...
31.12.2024 23:00,1000,0,,200,1200
```

Note! The used energy consumption should be out of one fuel (e.g. electricity); otherwise, combining demands is not possible.

Example #4

The user has records of heating (*D1*) and the total (*D5*) consumptions, yet there is no concrete knowledge of allocates from total demands. Users can use both of the records, yet the assessment based on *D5* might be inaccurate (no information on constituents).

```
Time,D1,D2,D3,D4,D5
01.01.2024 00:00,1000,0,,,1200
```



01.01.2024 01:00,1000,,,,1210
 ...
 31.12.2024 23:00,1000,0,,,1200

In that case, the recommended approach is to examine the heating consumption (*D1*) only, with potential *D5* assessment in the future (when more comprehensive knowledge on the energy consumption is gained):

Time,D1,D2,D3,D4,D5
 01.01.2024 00:00,1000,0,,,
 01.01.2024 01:00,1000,,,,
 ...
 31.12.2024 23:00,1000,0,,,

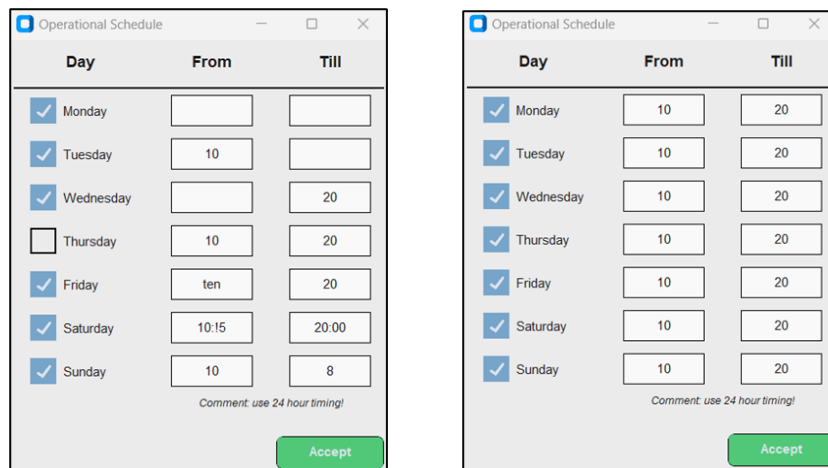
Example #5

This example shows the allowed variability in the *Country* input: the correct spelling is required, following the available list in **Table 5**. If the provided schema is not followed, Belgium (BE) is assumed as the default.

<u>input</u> : Poland	<u>output</u> : Poland
<u>input</u> : poland	<u>output</u> : Poland
<u>input</u> : pOIAND	<u>output</u> : Poland
<u>input</u> : Polaand	<u>output</u> : Belgium (spelling mistake)
<u>input</u> : PL	<u>output</u> : PL
<u>input</u> : pl	<u>output</u> : PL
<u>input</u> : POL	<u>output</u> : BE (incorrect abbreviation)

Example #6

This example shows how to correctly fulfill the Operational Schedule window in CEAM software. The exemplary input is shown below (on the left) with the corresponding correct version (on the right).



The explanation of the performed assessments by the CEAM software is as follows:

- Monday was selected, yet no working hours were provided; this day would not be included in the examined timetable,
- Tuesday was selected, yet no closing hour was provided; this day would not be included in the examined timetable,



- Wednesday was selected, yet no opening hour was provided; this day would not be included in the examined timetable,
- Thursday was not selected despite the input working hours; this day would not be included in the examined timetable,
- Friday was selected and the working hours were provided, but the opening hour is spelled (not as a number); this day would not be included in the examined timetable,
- Saturday was selected and the working hours were provided, but the time is defined with *HH:MM* format; this day would not be included in the examined timetable,
- Sunday was selected and the working hours were provided, but the time is input with the AM/PM format; this day would not be included in the examined timetable.

Example #7

7.7. EQUATIONS

A full collation of the used equations in the CEAM software is presented below.

$\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{n}}$	eq.1
<p><u>where:</u> x_i is the individual value, n is the total number of records, μ is the mean value, and σ is the standard deviation.</p>	
<p>where: ...</p>	...
<p>where: ...</p>	...

7.8. ADDITIONAL INFORMATION

All shorts of the additional information (e.g. lists or specs) mentioned throughout the document are listed.

Table 5. Country headers with the corresponding abbreviations used by the CEAM software.

Albania: AL, Algeria: DZ, Argentina: AR, Armenia: AM, Australia: AU, Austria: AT, Azerbaijan: AZ Bangladesh: BD, Belarus: BY, Belgium: BE, Bolivia: BO, Bosnia and Herzegovina: BA,	Honduras: HN, Hong Kong: HK, Hungary: HU Iceland: IS, India: IN, Indonesia: ID, Ireland: IE, Israel: IL, Italy: IT Jamaica: JM, Japan: JP	Pakistan: PK, Panama: PA, Paraguay: PY, Peru: PE, Philippines: PH, Poland: PL, Portugal: PT Romania: RO, Russia: RU San Marino: SM, Saudi Arabia: SA, Serbia: RS,
--	---	--



Brazil: BR, Bulgaria: BG Canada: CA, Chile: CL, China: CN, Colombia: CO, Costa Rica: CR, Croatia: HR, Czech Republic: CZ, Czechia: CZ Denmark: DK, Dominican Republic: DO Ecuador: EC, Egypt: EG, Estonia: EE Finland: FI, France: FR Georgia: GE, Germany: DE, Greece: GR	Kazakhstan: KZ, Kenya: KE, Kyrgyzstan: KG Latvia: LV, Liechtenstein: LI, Lithuania: LT, Luxembourg: LU Malaysia: MY, Malta: MT, Mexico: MX, Moldova: MD, Mongolia: MN, Morocco: MA Netherlands: NL, New Zealand: NZ, Nicaragua: NI, Nigeria: NG, North Macedonia: MK, Norway: NO	Singapore: SG, Slovakia: SK, Slovenia: SI, South Africa: ZA, South Korea: KR, Spain: ES, Sri Lanka: LK, Sweden: SE, Switzerland: CH Taiwan: TW, Thailand: TH, Tunisia: TN, Turkey: TR Ukraine: UA, United Arab Emirates: AE, United Kingdom: GB, United States: US, Uruguay: UY, Uzbekistan: UZ Venezuela: VE, Vietnam: VN Zambia: ZM, Zimbabwe: ZW
---	--	---

+ list of all figures

The structure of the var.txt file is as shown below. It contains information provided by the user in the CEAM GUI, in particular: path to the working directory, module selection (scope of analysis), records frequencies and time format for input files, information on solar radiation, and the examined demands, country and city info, units selection, heating and cooling degree days temperatures, outliers assessment with normal distribution range, working hours overview, as well as a selection of the optimization schemas.

Working directory: C:/Users/u0160065/Desktop/CEAM_example

Module 0: on
 Module 1: on
 Module 2: on
 Module 3: on
 Module 4: on
 Module 5: on

ICD_freq: h
 ECD_freq: h
 ED_freq: h

ICD_format: %d.%m.%Y
 ECD_format: %d.%m.%Y
 ED_format: %d.%m.%Y

Solar radiation: n

Country: Belgium
 City: Not provided



Units: SI

T_HDD: 16.0

T_CDD: 21.0

Outlier: y

Normal distribution: -3:+3

Operation day: Tuesday, from: 10, till: 18

Operation day: Wednesday, from: 10, till: 18

Operation day: Saturday, from: 10, till: 20

Optimization Schedule: s1, s4, s5

Examined demands: D1
