TC-Python Installation Guide

Thermo-Calc Version 2025b
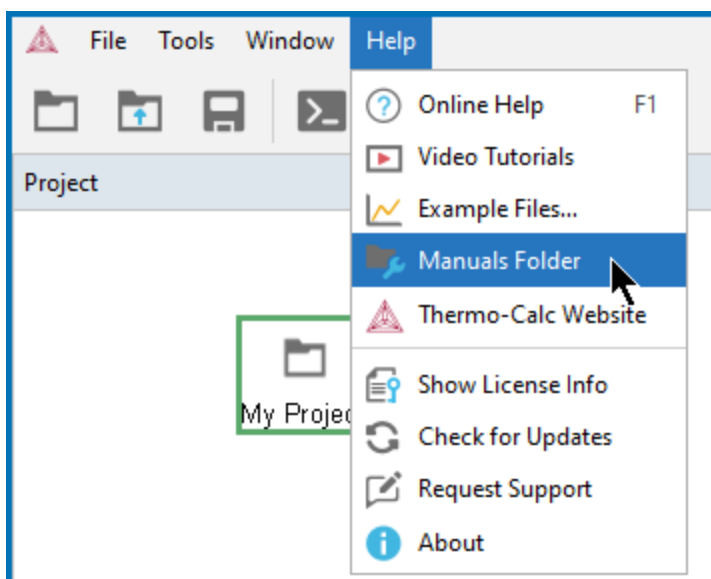
Thermo-Calc
Software

About the TC-Python Documentation

💡 The TC-Python help is available on the Thermo-Calc website where you can also access additional resources. You can alternatively access the locally installed documentation from the **Help** menu in Thermo-Calc.
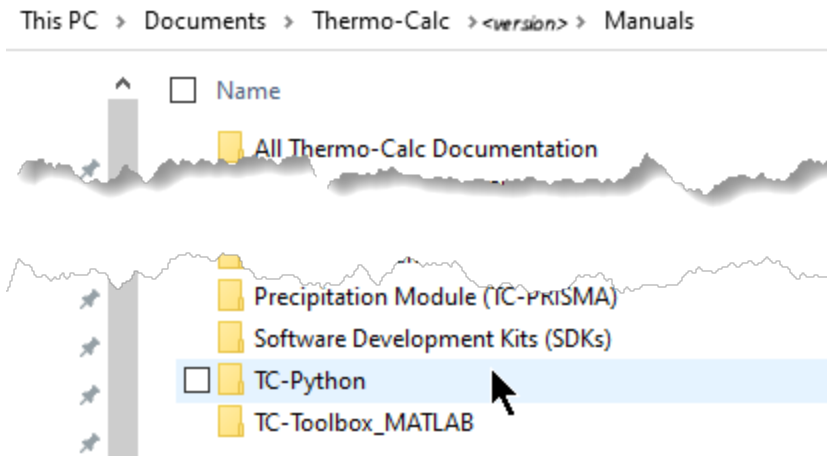
Opening the Locally Installed Documentation

1. In Thermo-Calc from the menu, select **Help → Manuals Folder**.
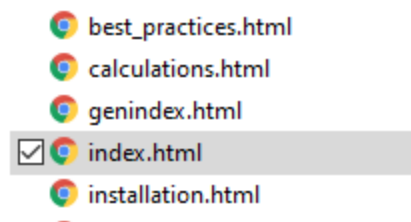

📄 You can also navigate to the **Manuals** folder outside of Thermo-Calc and based on your operating system installation location.

2. In the window that opens, double-click to open the **TC-Python** folder.

This PC > Documents > Thermo-Calc ><version> > Manuals

Name

All Thermo-Calc Documentation

Precipitation Module (TC-PRISMA)

Software Development Kits (SDKs)

TC-Python

TC-Toolbox_MATLAB

3. Double-click the **Index** HTML file (or any of the HTML files).

best_practices.html

calculations.html

genindex.html

index.html

installation.html

4. The API reference documentation opens as a searchable HTML file in your local browser.

Which TC-Python Installation Should I Choose?

> 🔑 A license is required to run TC-Python. Also remember to review the System Requirements page on our website.

There are these options to install TC-Python.

> ▶️ See the Videos about TC-Python page for additional learning resources.

Install with the Bundled Python-interpreter

This is a straightforward installation where TC-Python is preinstalled together with some popular Python packages.

This is the recommended option for new users to TC-Python who may only want to run the examples. The available preinstalled packages (e.g. numpy, scipy, scikit-learn, pyvista, and matplotlib) can also be sufficient for many advanced users.

> ❓ "Installing TC-Python with a Bundled Python-interpreter" on the next page.

> ❓ "Optional Packages" on page 14

Install with a Python-interpreter of Your Choice

This installation requires additional steps and is the recommended option for more advanced use and has full flexibility.

> ❓ "Installing TC-Python Into a Python-interpreter of Your Choice" on page 9.

Installing TC-Python with a Bundled Python-interpreter

TC-Python can be installed using a Python-interpreter that is bundled to Thermo-Calc. This interpreter has TC-Python preinstalled together with some popular Python packages. This is the recommended option for new users to TC-Python who may only want to run the examples.

> The bundled interpreter package is also used to run the Property Models in Thermo-Calc. Do not make changes to interpreter packages or this can stop Thermo-Calc from working correctly. To fix issues caused by changes to an interpreter, completely uninstall Thermo-Calc and start over, either with the bundled interpreter package or using interpreters of your choice.

> If you are planning to do more advanced calculations that require other external packages then you may want to follow these instructions instead: "Installing TC-Python Into a Python-interpreter of Your Choice" on page 9

Limitations

> The following examples cannot be run with the bundled Python-interpreter as additional packages are required: `pyex_M_01_Input_from_file.py` (pandas) and `pyex_M_02_Output_to_file.py` (pandas, lxml, h5py).

The bundled Python 3 interpreter contains the following major packages:

- colour-science
- matplotlib
- numpy
- scikit-learn
- scipy
- pandas

- pyvista
- TC-Python
- tmm

> 🗎 If you have recommendations for additional packages to include (see below), send an email to Thermo-Calc Support.

> 🗎 These TC-Python examples require additional packages that are not available in the bundled Python-interpreter, and thus cannot be run: `pyex_M_01_Input_from_file.py` and `pyex_M_02_Output_to_file.py`.

Step 1. Install an IDE (Integrated Development Environment)

Any editor can be used to write the Python code, but an IDE is recommended, e.g. PyCharm. These instructions are based on the use of PyCharm.

Use of an IDE gives you access to code completion, which helps when you use the API as it gives you the available methods on the objects you are working with.

1. Navigate to the PyCharm website: https://www.jetbrains.com/pycharm/download.
2. Click to choose your OS and then click **Download**. You can use the **Community** version of Pycharm.
3. Follow the instructions. It is recommended you keep all the defaults.

> ❗ For Mac installations, you also need to set some environment variables as described below in "macOS: Setting Environment Variables" on page 24.

Step 2. Configure PyCharm to Use the Bundled Python-interpreter

Open PyCharm and configure the interpreter:

1. Go the menu **File → Settings**.
2. Navigate in the tree to **Project: YourProjectName** and choose **Project Interpreter**.
3. Click on the settings symbol close to the **Project Interpreter** dropdown menu and choose **Add**.

4. Select **System Interpreter** and add the bundled Thermo-Calc Python 3 interpreter, which is located in a folder based on the operating system:

Operating system	Path to the bundled Python-interpreter
Windows	`C:\Program Files\Thermo-Calc\2025b\python\python.exe`
Linux	`/home/UserName/Thermo-Calc/2025b/python/bin/python3`
macOS	`/Applications/Thermo-Calc-2025b.app/Contents/Resources/python/bin/python3`

5. Select the interpreter and confirm.

Step 3. Run a TC-Python Example

It is recommended that you open one or more of the included examples to both check that the installation has worked and to start familiarizing yourself with the code.

> If you have a license activation type based on **user credentials** you first need to activate your license. See "License Activation with User Credentials" on page 17.

Open the TC-Python Project in PyCharm

> When you first open the TC-Python project and examples, it can take a few moments for the Pycharm IDE to index before some of the options are available.

1. Open PyCharm and then choose **File → Open**. The first time you open the project you will need to navigate to the path of the TC-Python installation based on your operating system.

Operating system	Path to the TC-Python folder
Windows	`C:\Users\UserName\Documents\Thermo-Calc\2025b\SDK\TC-Python`
Linux	`/home/UserName/Thermo-Calc/2025b/SDK/TC-Python`

Operating system	Path to the TC-Python folder
macOS	`/Users/Shared/Thermo-Calc/2025b/SDK/TC-Python`

2. Click the **Examples** folder and then click **OK**.

3. From any subfolder:

   - Double-click to open an example file to examine the code.

   - Right-click an example and choose **Run**.

> 📄 If you are not following the recommended approach and create a *new* project (**File → New Project…**), you need to consider that by default the options to choose the interpreter are hidden within the **Create Project** window. For this, click **Project Interpreter: New Virtual Environment** and in most cases choose your *System Interpreter* containing the Python bundled to Thermo-Calc instead of the default *New Virtual Environment*.

Installing TC-Python Into a Python-interpreter of Your Choice

TC-Python can be installed using a Python-interpreter of your choice. This is the recommended option for any more advanced usage and provides full flexibility.

> If you are new to TC-Python and want to mainly run the examples, then it is recommended you install the bundled Python-interpreter. The available preinstalled packages (e.g. numpy, scipy, scikit-learn, pyvista, and matplotlib) can also be sufficient for many advanced users. See "Installing TC-Python with a Bundled Python-interpreter" on page 5.

Step 1. Install a Python Distribution

> If you already have a Python distribution installation, version 3.8 or newer, skip this step.

These instructions are based on using the Anaconda platform for the Python distribution. Install version 3.8 or newer to be able to work with TC-Python, although it is recommended that you use the most recent version.

Install Anaconda

1. Navigate to the Anaconda website: https://www.anaconda.com/download/.
2. Click to choose your OS (operating system) and then click **Download**. Follow the instructions. It is recommended you keep all the defaults.

Step 2. Install Thermo-Calc and the TC-Python SDK

1. Install Thermo-Calc.
2. When the installation is complete, open the TC-Python folder that includes the *.whl file needed for the next step. There is also an **Examples** folder with Python files you can use in the IDE to understand and work with TC-Python.
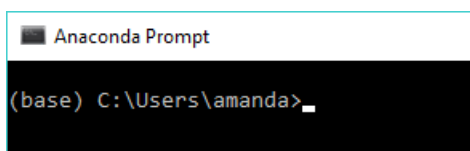
Step 3. Install TC-Python

? "Updating to a Newer Version" on page 23

On Windows, it is recommended you use the Anaconda distribution prompt (i.e. Anaconda), especially if you have other Python installations.

> 🔘 Do not use Virtual Environments unless you have a good reason to do so. Also see "Fixing Potential Issues with the Environment" on page 28.

1. Open the command line. For example, in Anaconda on a Windows OS, go to **Start →** **Anaconda → Anaconda Prompt**.
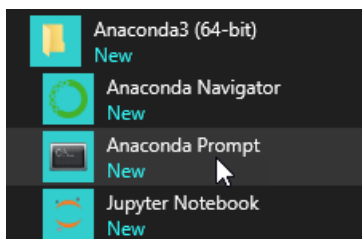
2. At the command line, enter the following. Make sure there are no spaces at the end of the string or in the folder name or it will not run:

```
pip install <path to the TC-Python folder>/TC_Python-<version>-
py3-none-any.whl
```

> 📄 On Linux, and depending on the interpreter, *pip3* is typically used.

Operating system	Path to the TC-Python folder
Windows	C:\Users\UserName\Documents\Thermo-Calc\2025b\SDK\TC-Python
Linux	/home/UserName/Thermo-Calc/2025b/SDK/TC-Python
macOS	/Users/Shared/Thermo-Calc/2025b/SDK/TC-Python

3. Press <Enter>. When the process is completed, there is a confirmation that TC-Python is installed.

> If your computer is located behind a proxy-server, the default pip-command will fail with a network connection error. In that case you need to install the dependency py4j in a special configuration:

```
pip install -proxy user:password@proxy_ip:port py4j jproperties
```

> Go to the TC-Python Help **Troubleshooting** section for detailed information. This can be found on our website or from the installed documentation where you can search for "pip install fails" with "Failed to establish a new network connection" or similar.

Step 4. Install an IDE (Integrated Development Environment)

Any editor can be used to write the Python code, but an IDE is recommended, e.g. PyCharm. These instructions are based on the use of PyCharm.

Use of an IDE gives you access to code completion, which helps when you use the API as it gives you the available methods on the objects you are working with.

1. Navigate to the PyCharm website: https://www.jetbrains.com/pycharm/download.
2. Click to choose your OS and then click **Download**. You can use the **Community** version of Pycharm.
3. Follow the instructions. It is recommended you keep all the defaults.

> For Mac installations, you also need to set some environment variables as described below in "macOS: Setting Environment Variables" on page 24.

Step 5. Open the IDE and Run a TC-Python Example

After you complete all the software installations and set up the pip install path, you are ready to open the IDE to start working with TC-Python.

It is recommended that you open one or more of the included examples to both check that the installation has worked and to start familiarizing yourself with the code.

If you have a license activation type based on **user credentials** you first need to activate your license. See "License Activation with User Credentials" on page 17.

Open the TC-Python Project in PyCharm

When you first open the TC-Python project and examples, it can take a few moments for the Pycharm IDE to index before some of the options are available.

1. Open PyCharm and then choose **File → Open**. The first time you open the project you will need to navigate to the path of the TC-Python installation based on your operating system.

Operating system	Path to the TC-Python folder
Windows	`C:\Users\UserName\Documents\Thermo-Calc\2025b\SDK\TC-Python`
Linux	`/home/UserName/Thermo-Calc/2025b/SDK/TC-Python`
macOS	`/Users/Shared/Thermo-Calc/2025b/SDK/TC-Python`

2. Click the **Examples** folder and then click **OK**.

3. From any subfolder:
   - Double-click to open an example file to examine the code.
   - Right-click an example and choose **Run**.

Also see "Optional Packages" on page 14 if you are using the Additive Manufacturing Module for simulations.

Installation Default Directory: TC-Python

These are the default paths to the TC-Python folder based on your operating system.

Operating system	Path to the TC-Python folder
Windows	`C:\Users\UserName\Documents\Thermo-Calc\2025b\SDK\TC-Python`
Linux	`/home/UserName/Thermo-Calc/2025b/SDK/TC-Python`
macOS	`/Users/Shared/Thermo-Calc/2025b/SDK/TC-Python`

For a bundled Python-interpreter installation, these are the paths to this location based on your operating system.

Operating system	Path to the bundled Python-interpreter
Windows	`C:\Program Files\Thermo-Calc\2025b\python\python.exe`
Linux	`/home/UserName/Thermo-Calc/2025b/python/bin/python3`
macOS	`/Applications/Thermo-Calc-2025b.app/Contents/Resources/python/bin/python3`

Optional Packages

Plotting results of Additive Manufacturing simulations requires the package *pyvista*. It is automatically installed if using the option *[am-plotting]* during the installation:

```
pip install <path to the TC-Python folder>/TC_Python-<version>-py3-none-any.
whl[am-plotting]
```

> The bundled interpreter contains *pyvista* by default.

TC-Python: Configuring a License

TC-Python requires a license to run. If you do not yet have the information about your license, contact Thermo-Calc Support at support@thermocalc.com.

> 🚫 If you have no valid license, an exception is thrown that states: `NO LICENSE: No activated license for TC-Python`.

Depending on the type of license, different steps are required to configure TC-Python to use it.

Environment Variables

Some environment variables need to be set. On Windows this is normally done during the installation of Thermo-Calc, but on Linux and macOS this needs to be performed for each used IDE or terminal.

License Activation with User Credentials

This section is applicable to licenses that are activated after entering user credentials (i.e. user name and password).

> ⚠ The TC-Python license is a separate license from the Thermo-Calc software license. You need to activate the TC-Python license once on your computer after, or in addition to, activating the Thermo-Calc license.

It is recommended to run the included example to perform the activation and deactivation. By default an online activation is performed, but you can choose other methods, which are both included in the example and described in the API documentation.

- Open and run the example `pyex_M_04_license.py`, which is located in the **Miscellaneous** folder.

- The license activation is completed with the methods in the class `tc_python.license.LicenseManager`.

- Depending on the type of license other methods are available, for example for online and offline activation plus the use of proxy settings are described.

> 💡 Once the license is activated, either by using online or offline activation methods, you can view the license status and information in Thermo-Calc. Select **Help → Show License Info** from the menu.

> 🗎 To use the license on a different computer, the license must first be deactivated on the original computer. This is for both online and offline activations.

TC-Python: Setting Environment Variables

macOS and Linux

In order to use TC-Python on macOS and Linux you need to set some environment variables.

```
TC25B_HOME=/Applications/Thermo-Calc-2025b.app/Contents/Resources
```

User Credential License Types

If you have a license based on user credentials:

```
TC_LICENSE_SPRING=y
```

License File Type Activations

If you use a license server:

```
LSHOST=<name-of-the-license-server>
```

If you have a node-locked license:

```
LSHOST=NO-NET
LSERVRC=/Users/Shared/Thermo-Calc/lservrc
```

If running in the terminal these can be set using export `LSHOST=NO-NET`, etc.

PyCharm IDEs

In PyCharm, you can add environment variables in the configurations.

> For other operating systems, the configuration can be done in the same way as described below.

1. Select **Run → Edit Configurations** to open the **Run/Debug Configurations** window.
2. Choose **Templates** and then **Python**.
3. Enter the environment variable(s) by clicking the button to the right of the **Environment Variables** text field. Now the environment variables(s) are set for each new configuration by default.

> Existing configurations need to be removed and recreated to obtain the environment variables in them.

Updating to a Newer Version

> **!** If you are using the bundled Python-interpreter automatically installed with Thermo-Calc, these steps are not required.

> **!** These steps are required if you have previously followed the instructions in "Installing TC-Python Into a Python-interpreter of Your Choice" on page 9.

When updating to a newer version of Thermo-Calc, you always need to also install the latest version of TC-Python. It is not sufficient to run the installer of Thermo-Calc (except for bundled versions of the Python interpreter).

```
pip install <path to the TC-Python folder>/TC_Python-<version>-py3-
none-any.whl
```

In case of problems you may wish to uninstall the previous version of TC-Python in advance:

```
pip uninstall TC-Python
```

```
pip install <path to the TC-Python folder>/TC_Python-<version>-py3-
none-any.whl
```

However, this should not normally be required.

> 📄 On Linux typically *pip3* is used.

You can check the currently installed version of TC-Python by running:

```
pip show TC-Python
```

macOS: Setting Environment Variables

In order to use TC-Python on macOS you need to set some environment variables.

```
TC25B_HOME=/Applications/Thermo-Calc-2025b.app/Contents/Resources
```

User Credentials License Activations

```
TC_LICENSE_SPRING=y
```

License File Type Activations

If you use a license server:

```
LSHOST=<name-of-the-license-server>
```

If you have a node-locked license:

```
LSHOST= NO-NET

LSERVRC=/Users/Shared/Thermo-Calc/lservrc
```

PyCharm IDEs

In PyCharm, you can add environment variables in the configurations.

Select **Run** → **Edit Configurations** to open the **Run/Debug Configurations** window. Enter the environment variable(s) by clicking the button to the right of the **Environment Variables** text field. Now the environment variables(s) will be set for each new configuration by default.

> Existing configurations need to be removed and recreated to obtain the environment variables in them.

The same way for configuring the environment variables can be used on other operating systems as if necessary.
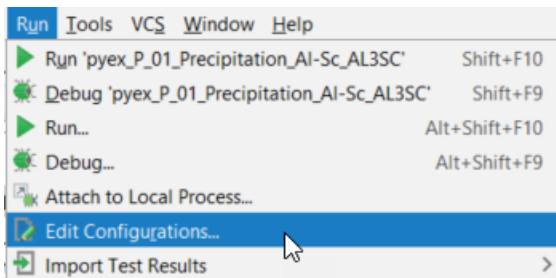
Fixing Potential Issues with the Environment

In most cases you should run TC-Python within your **global** Python 3 interpreter and not use Virtual Environments unless you have a good reason to do so.

If there are problems with the interpreter settings, for example, a common problem when you first use TC-Python is the message *No module named tc_python*, which you can resolve in the Settings window as follows:

1. Go the menu **File → Settings**.

2. Navigate in the tree to *<Project.YourProjectName>* and choose **Project Interpreter**.

3. Click on the settings symbol close to the **Project Interpreter** drop-down menu and choose **Add**.

4. Now choose **System Interpreter** and add your existing Python 3 interpreter.

5. Select your added interpreter and confirm.

> If you are not following the recommended approach and create a *new* project (**File → New Project...**), you need to consider that by default the options to choose the interpreter are hidden within the **Create Project** window. For this, click **Project Interpreter: New Virtual Environment** and in most cases choose your *System Interpreter* containing the Python bundled to Thermo-Calc instead of the default *New Virtual Environment*.

> If you really need to use a Virtual Environment, it is recommended you review the hints provided in the **Best Practices** section in the full TC-Python HTML help or PDF.

TC-Python Diagnostic Script

A diagnostic script is available to help you if you have problems running TC-Python.

> 🛈 It is important to run the diagnostic script EXACTLY the same way as you run your TC-Python script (in the same IDE, same project, same Python environment, same Jupyter notebook, etc).

The script automatically prints useful suggestions that, in many cases, will help you to fix the problem yourself. If you are unable to solve the problem, contact us for support. You can find the script in these locations:

- Miscellaneous example (diagnostics.py)

- In the **Troubleshooting** section of the *TC-Python API Reference Documentation*. Also refer to the **Best Practices** in the same guide to see if there are any other solutions or tips.

- On our website.