

Hands-on exercise 1: Installing MUSE

This hands-on exercise will show you how to install MUSE on your computer.

If at any point you get stuck with any of these exercises, feel free to post a question in the [MUSE Google group](https://groups.google.com/g/muse-model) (<https://groups.google.com/g/muse-model>), or raise an issue on [Github](https://github.com/EnergySystemsModellingLab/MUSE_OS) (https://github.com/EnergySystemsModellingLab/MUSE_OS). You can also consult the full MUSE documentation [here](https://muse-os.readthedocs.io/en/v1.3.3/) (<https://muse-os.readthedocs.io/en/v1.3.3/>).

Creating a virtual environment

MUSE is built as a Python package (called `muse_os`), so installing MUSE requires Python to be installed on your machine. This version of the course is based on `muse_os` version 1.3.3, which requires a Python version between 3.9 and 3.12. In this tutorial, we will show you how to set up an environment with Python 3.12 and install `muse_os` into that environment. If you're already comfortable creating Python environments and installing packages, feel free to do this however you choose. Otherwise, please read on.

The simplest method to install Python is by downloading the [Anaconda distribution](https://www.anaconda.com/) (<https://www.anaconda.com/>). Once installed on your machine, this will allow you to create isolated environments containing all the dependencies needed for a project, with a specific Python version.

If you're using Anaconda, you can create and activate an environment by opening up your terminal and running the following commands:

```
conda create -n muse_env python=3.12
conda activate muse_env
```

Once you've activated the environment, run the following command to install MUSE version 1.3.3:

```
python -m pip install muse_os==1.3.3
```

Later (for example, when tackling future hands-on exercises in this course), you can return to this environment by simply running:

```
conda activate muse_env
```

Confirming the installation

To confirm that the installation has worked, run the following in your command line:

```
python -m muse --model default
```

This command runs the default model that comes with MUSE. If the installation has worked correctly, you should see output text similar to [this](https://muse-os.readthedocs.io/en/v1.3.3/example-output.html) (<https://muse-os.readthedocs.io/en/v1.3.3/example-output.html>).

You should also see that a folder called `Results` has been created in your working directory. Don't worry about the contents of this folder for now, as we will be analysing these results in the next exercise.

Summary

Hopefully you've now managed to install MUSE on your machine, and are ready to tackle the next hands-on exercises. Interested readers can also see [here](https://muse-os.readthedocs.io/en/v1.3.3/installation/index.html) (<https://muse-os.readthedocs.io/en/v1.3.3/installation/index.html>) for more detailed installation instructions, including a number of alternative methods for installing MUSE.