Startup manual

SFDEsim-Simulator is made as a part of master's thesis project about usage and benefits of simulation in bachelor's level electrical engineering courses. The SFDEsim-simulator's goal is to provide visual and interactive learning opportunities in simple cases requiring a minimal amount of user knowledge of the subject for use. This manual instructs how to setup the program for use. The program is supported on Windows and Linux operating systems. This manual is for Windows only.

Installing Python

The simulator is built in Python and therefore requires a Python interpreter to work. If you have already installed Python you can skip to next step.

On Windows PCs the simplest way to install Python is from Microsoft Store, which can be found with Windows search startup menu. Other option is to download Python installer form internet (Python.org).

Use search to find Python installations. Any version of Python 3.7 or later should work, versions 3.10 and 3.12 are tested to work and therefore recommended (figure 1).

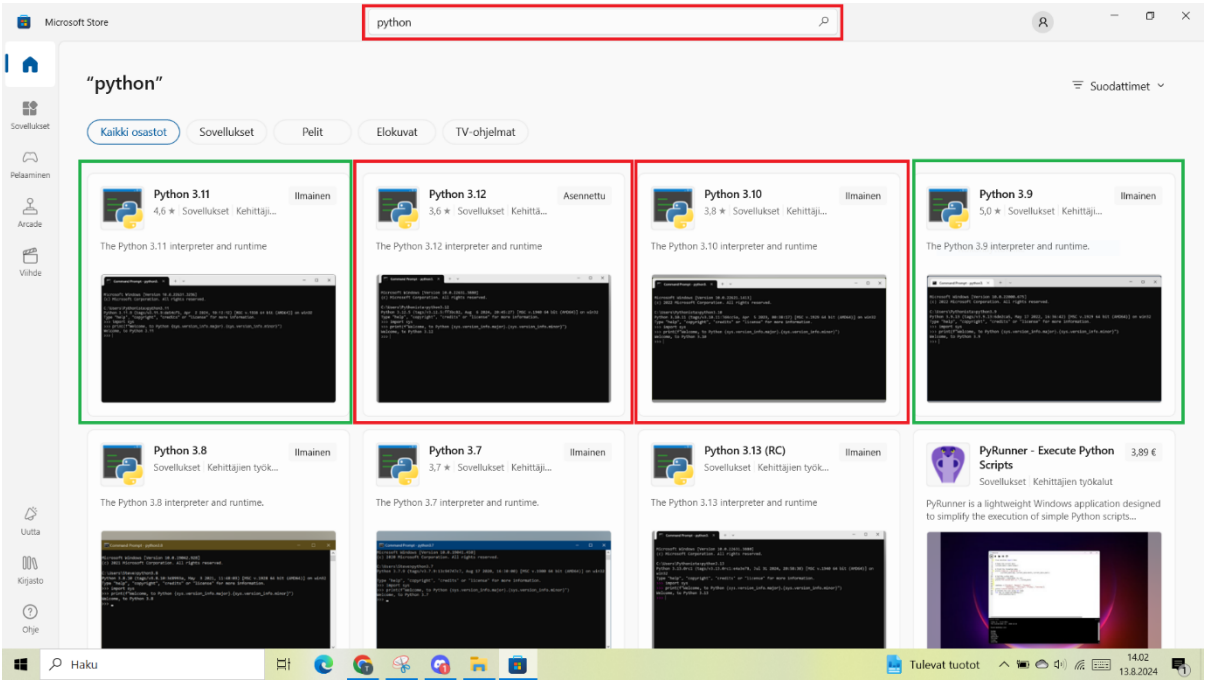

Figure 1. Python search in Microsoft store.

Install by pressing “Get” or “Hanki” (fi) button, depending on your language settings (figure 2).

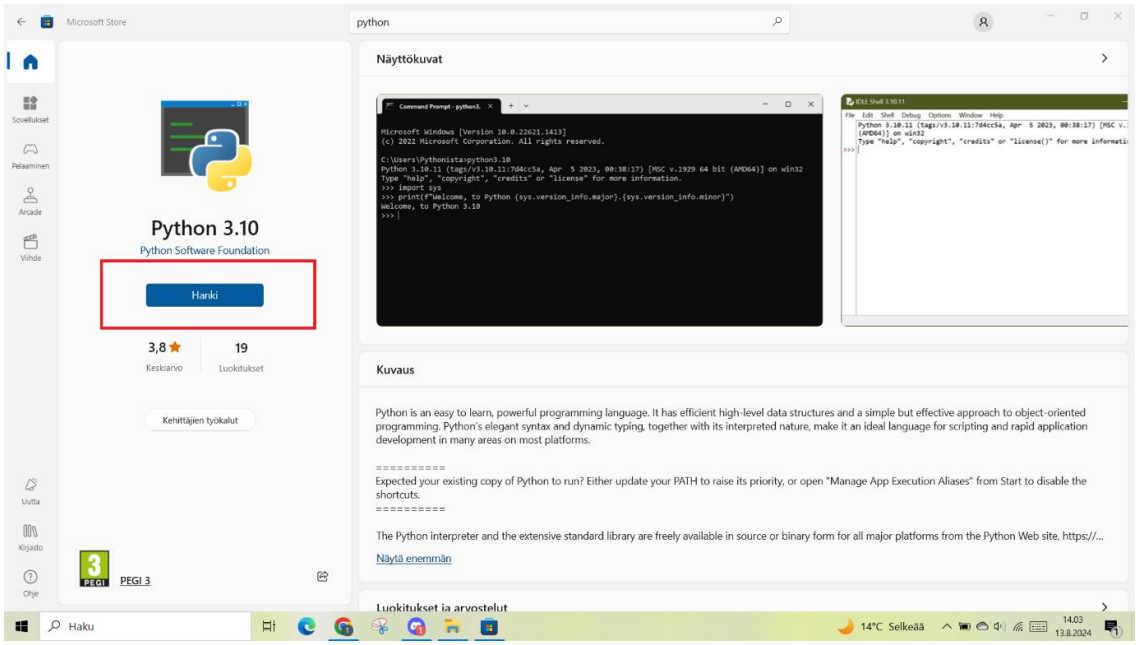

Figure 2. Python 3.10 download page in Microsoft store, installation button circled.

First time launch

The simulator is available for download in Moodle. The folder download is in .zip format and has to be decompressed. This can be done in resource management (extract all). The program can be started by running (double click) **SFDEsim_simulator.py** Python program (figure 3). When running the program for the first time, a number of Python packages has to be installed (PySide6, pyqtgraph, NumPy). Packages are installed from Python Package Index (PyPI), which can be automatically in the simulator.

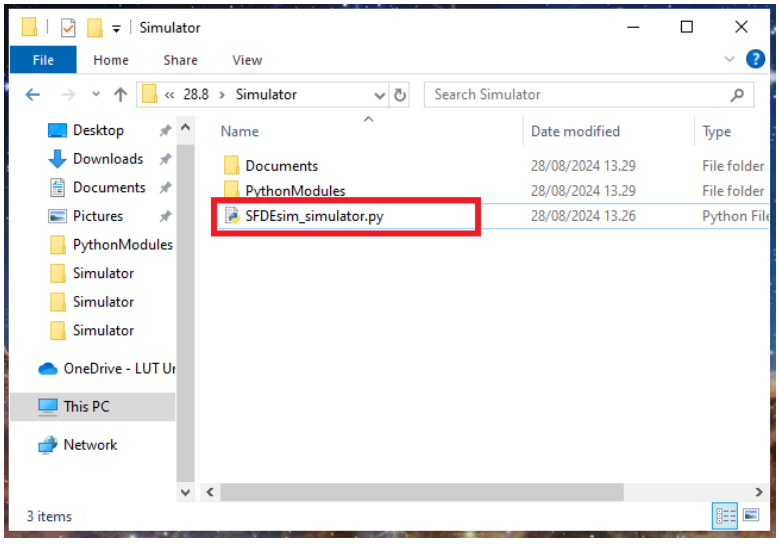

Figure 3. SFDEsim_simulator.py in Simulator folder.

The missing packages are shown in the console window opened by running Simulator.py. Installation can be confirmed typing “y” or “yes” and pressing enter (figure 4).

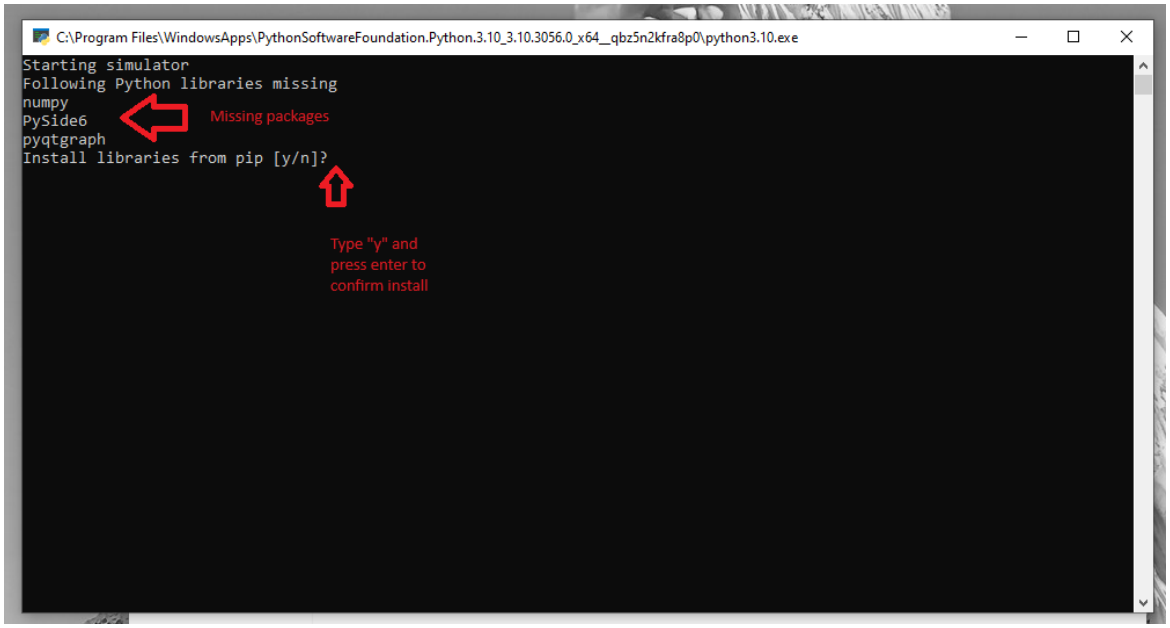

Figure 4. Console window when package installation required.

After the packages are installed, the simulator will launch automatically. The console window can be closed after the simulator has launched (figure 5).

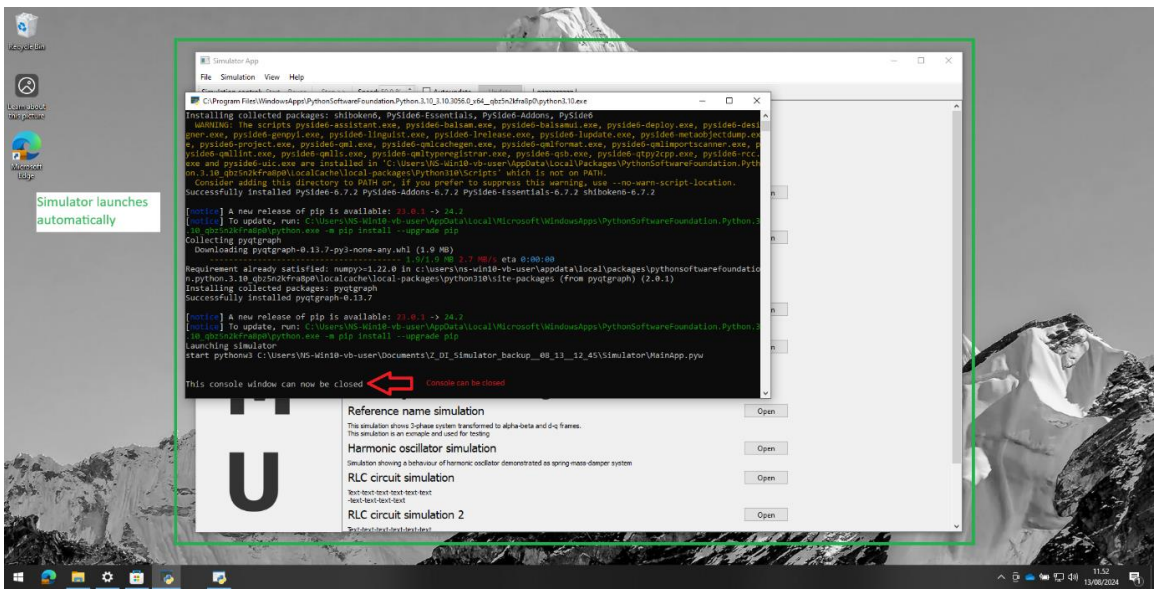

Figure 5. Console window after installation is done and simulator has launched, simulator window at the background.

The simulator is ready to use.

Subsequent launches

When launching the program for the next time, it is done similarly by running the Simulator.py program. Since the packages are installed the program will launch without any user actions, and the console window can be closed.

Problems

If the simulator does not launch from the Simulator.py program, the program can be started from MainApp.pyw program located in the PythonModules folder. This may happen depending on your computer's security settings as launching applications from different directories may not be permitted. If The simulator is not able to launch from MainApp.pyw either, please check the log_file.txt for error messages and contact support through Moodle or email.