Quick Start Guide

Install Python

In order to make use of the PyMedPhys library, you'll need Python installed on your workstation. Below are some recommended instructions for installing Python based upon your OS.

Windows 10 or 11

Open up a command prompt and type:

```
python
```

If python isn't already installed this will open the Windows store. At this point you can click the "Get" button. It will ask you to sign in, but you can skip the sign in step. It will still install.

Once installed, open a new command prompt and test that it has installed by typing:

```
python --version
```

Windows by default also can't utilise paths longer than 260 characters. This will likely be an issue for installing Python packages. As such follow the enable Windows Long Paths guide to enable long paths on your system.

Linux or MacOS

On Linux or MacOS we recommend not using your system Python and instead managing your Python installation using pyenv.

To achieve this first install the python build environment, and then follow the pyenv installation steps. Once pyenv is installed first run the following:

```
pyenv install 3.9.7
```

And then, after that is completed, follow it with:

```
pyenv global 3.9.7
```

You can choose to adjust the version number provided above to be the latest Python version if you wish.

Installing PyMedPhys

Once you have Python you can now install PyMedPhys via pip by typing the following in a terminal or command prompt:

```
pip install pymedphys[user]
```

You can copy this command directly into a Windows command prompt. You can `Right Click` to paste.

You may need to open and close your terminal if you have only just installed Python. The `[user]` option is needed to install pymedphys with its "batteries included" so-to-speak. It will go and install a range of dependencies which you may need during your use of pymedphys.

SSL Issues

Depending on your network set up you may see "SSL" warnings followed by an error message when trying to install PyMedPhys with the above command. This may be due to your network administrator filtering all packets through its own server. Pip by default protects against this as filtering of this sort can intercept the packages to be installed and provide you with something else.

If you trust the network you are on to not be utilising this power maliciously then you can run the following to say that you trust your network's version of pypi.org and files.pythonhosted.org:

```
pip install --trusted-host pypi.org --trusted-host files.pythonhosted.org pymedph
```

Proxy Issues

It's common for private networks, especially those in healthcare, to require outgoing traffic to be sent through a proxy server to reach any servers on the world wide web. If this is the case on your network, you will need to specify a proxy server when using pip.

The following command specifies the proxy server for pip. Ensure you insert your username and password used to authenticate on the proxy server, along with the host and port of the proxy server. If you are unsure of the host and port to use in your environment, reach out to a network administrator to obtain these:

```
pip install --proxy=http://username:password@host:port --trusted-host pypi.org --
```