The following steps are needed to set up QML software on Windows 11

These instructions install Python 3.11, PennyLane 0.42.3, PyTorch 2.8.0, and other required Python packages for CPU execution only. Any changes to these versions will need testing.

To run the following commands in a terminal on Windows, we recommend using Git Bash: https://git-scm.com/downloads/win

Step 1. Install Python 3.11 if needed

- To ensure that all course participants have an identical installation, we will use Python 3.11. Check what version of Python you have on your system. If it is version 3.11 you can go to Step 2.
  $ python --version *Your output may look like this:*
  Python 3.12.x
- If not, please install Python 3.11 first (you can download it from this website: https://www.python.org/downloads/release/python-3119)
- $ python3.11 –version *Your output should look similar to this:*
  Python 3.11.x

Step 2. Installing a Python virtual environment (venv)

- If you have Python 3.11 (in fact 3.3 or later) you will be able to create a virtual environment. Create a container directory “~/venv” to hold different Python virtual environments, such as PennyLane with CPU or GPU support, Qiskit, Cirq or Julia with Yao.
  $ mkdir ~/venv
- In the “venv”, we will create an environment for CPU-based PennyLane and call it “pl”.
  $ cd ~/venv
  $ python3.11 -m venv pl
- To use the “pl” environment, we need to activate it - to be done in any PennyLane project.
  $ source ~/venv/pl/Scripts/activate
- Now we can check which Python is active (different from a global default)
  (pl) $ which python *Your output should be similar to this:*
  /home/yourusername/venv/pl/Scripts/python
- The environment will be active until you use the command “deactivate”.
- You can also read about venv here https://docs.python.org/3/library/venv.html

Step 3. Setup your PennyLane+PyTorch virtual Python environment (for QML workshop)

- Download the “requirements.txt” file (https://github.com/ironfrown/qml_workshop_intro_v2/blob/main/requirements.txt) and leave it in “~/Downloads”. This file includes names of packages such as: PennyLane, PyTorch, ML software, Jupyter and data sources.
- While the “pl” environment is active, install all Python packages, with their versions as specified in the “requirements.txt” file:
  (pl) $ pip install -r ~/Downloads/requirements.txt

Step 4. Create a project directory and start Jupyter Lab or Jupyter Notebook

- Create a workshop directory, e.g. “~/workshop”, go there, and start jupyter:
  (pl) $ mkdir ~/workshop
  (pl) $ cd ~/workshop
  (pl) $ jupyter lab
- Follow the instructions on your screen and open http://localhost:8888/tree in your browser