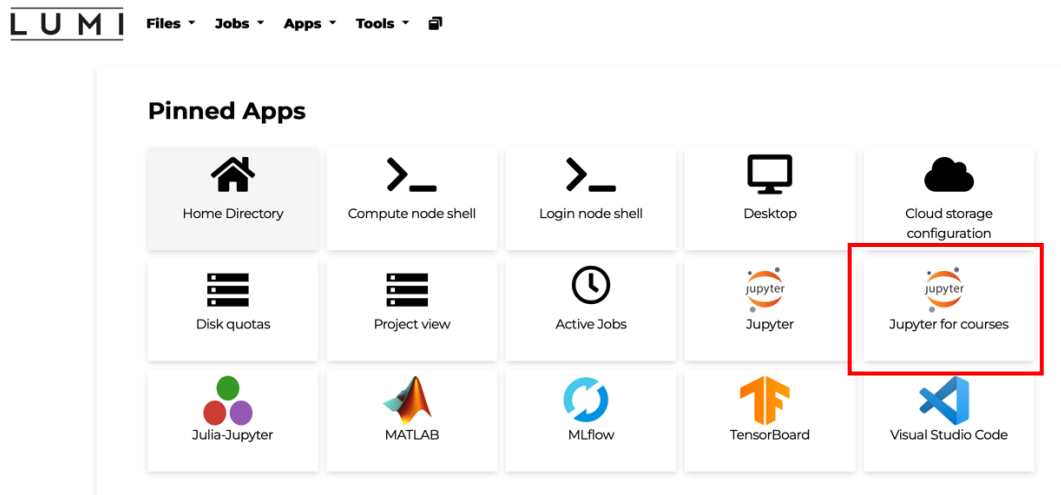


# Using Helmi via the Custom Quantum Course Environment on LUMI Open OnDemand

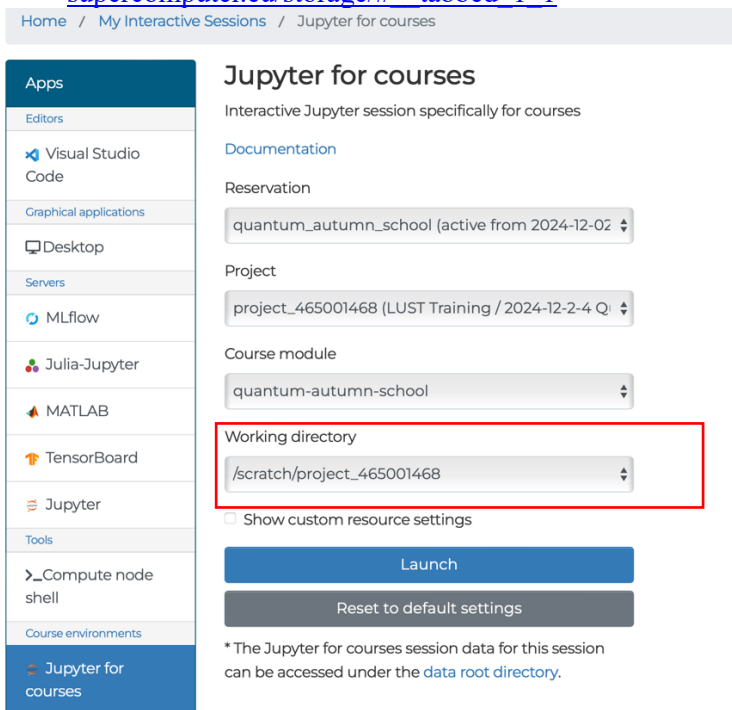
A course environment has been made with preloaded modules([qiskit-iqum](#)) and resource settings for connecting to Helmi. This document offers guidance on accessing this environment.

1. Log in to [LUMI open ondemand/web interface](#) with your MyCSC / Haka account
2. Click on `Jupyter for Courses



3. You should get a prefilled page shown below, you can change a few variables such as your `working directory`.

- Please note what you selected as your working directory when launching your Jupyter Notebook as this will determine where your file is saved.
- You can find more information about storage on LUMI here: [https://docs.lumi-supercomputer.eu/storage/#\\_tabbed\\_1\\_1](https://docs.lumi-supercomputer.eu/storage/#_tabbed_1_1)



4. Once you are ready, click on `Lunch`.

Home / My Interactive Sessions / Jupyter for courses

**Apps**

Editors

- Visual Studio Code

Graphical applications

- Desktop

Servers

- MLflow
- Julia-Jupyter
- MATLAB
- TensorBoard
- Jupyter

Tools

- Compute node shell

Course environments

- Jupyter for courses

### Jupyter for courses

Interactive Jupyter session specifically for courses

[Documentation](#)

Reservation: quantum\_autumn\_school (active from 2024-12-02)

Project: project\_465001468 (LUST Training / 2024-12-2-4 Q)

Course module: quantum-autumn-school

Working directory: /scratch/project\_465001468

Show custom resource settings

**Launch**

Reset to default settings

\* The Jupyter for courses session data for this session can be accessed under the [data root directory](#).

5. Depending on the queue, it might take a few minutes before you can access your session. Once your session is ready, you should see a page like this:

Session was successfully deleted. ✕

Home / My Interactive Sessions

**Apps**

Editors

- Visual Studio Code

Graphical applications

- Desktop

Servers

- MLflow
- Julia-Jupyter
- MATLAB
- TensorBoard
- Jupyter

Tools

### Jupyter for courses (8310225)

**1 node** | **2 cores** | Running

**Host:** >\_ nid002343 ✕ Cancel

**Created at:** 2024-10-31 15:02:55 UTC

**Time Remaining:** 59 minutes

**Session ID:** 23bec8a6-e750-4a35-b131-1e933ca2b72e

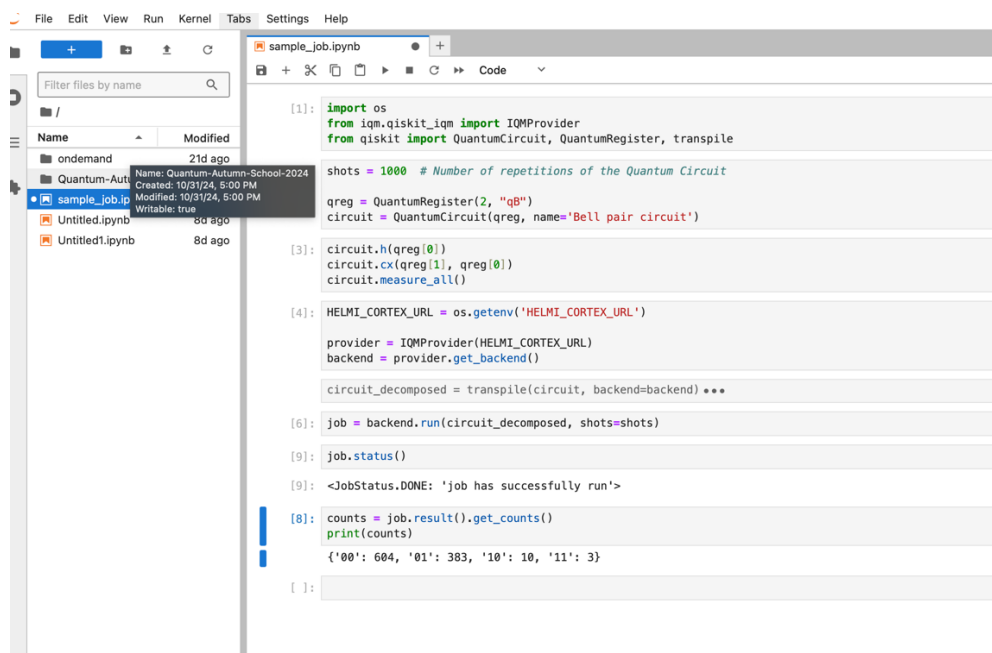
If you run into issues, please include the following log file in the support ticket: [output.log](#)

**Project:** project\_465001468  
**Partition:** q\_fiqci  
**Cores:** 2  
**Memory:** 2048M

**Connect to Jupyter**

6. Click on the big blue **'Connect to Jupyter'** button and it will launch a Jupyter notebook on a separate tab. Navigate to the new tab

7. Now you can create and run codes on a jupyter notebook. A sample is shown in the image below.



```
[1]: import os
      from iqm.qiskit_iqm import IQMProvider
      from qiskit import QuantumCircuit, QuantumRegister, transpile

      shots = 1000 # Number of repetitions of the Quantum Circuit

      qreg = QuantumRegister(2, "qB")
      circuit = QuantumCircuit(qreg, name='Bell pair circuit')

[3]: circuit.h(qreg[0])
      circuit.cx(qreg[1], qreg[0])
      circuit.measure_all()

[4]: HELMI_CORTEX_URL = os.getenv('HELMI_CORTEX_URL')

      provider = IQMProvider(HELMI_CORTEX_URL)
      backend = provider.get_backend()

      circuit_decomposed = transpile(circuit, backend=backend)***

[6]: job = backend.run(circuit_decomposed, shots=shots)

[9]: job.status()

[9]: <JobStatus.DONE: 'job has successfully run'>

[8]: counts = job.result().get_counts()
      print(counts)

      {'00': 604, '01': 383, '10': 10, '11': 3}

[ ]:
```

8. You can save your notebooks like this:

- go to the top left corner, click on file
- click on **'Save notebook'** to save a specific file or **'Save all'** to save all file & changes

9. This environment gives each user 2 hours and 2GB per session. It is configured to load the helmi\_qiskit (qiskit-iqm) module and connect to Helmi.

### Useful Links

- [Introduction to Lumi web](#)
- [Running on Helmi](#)