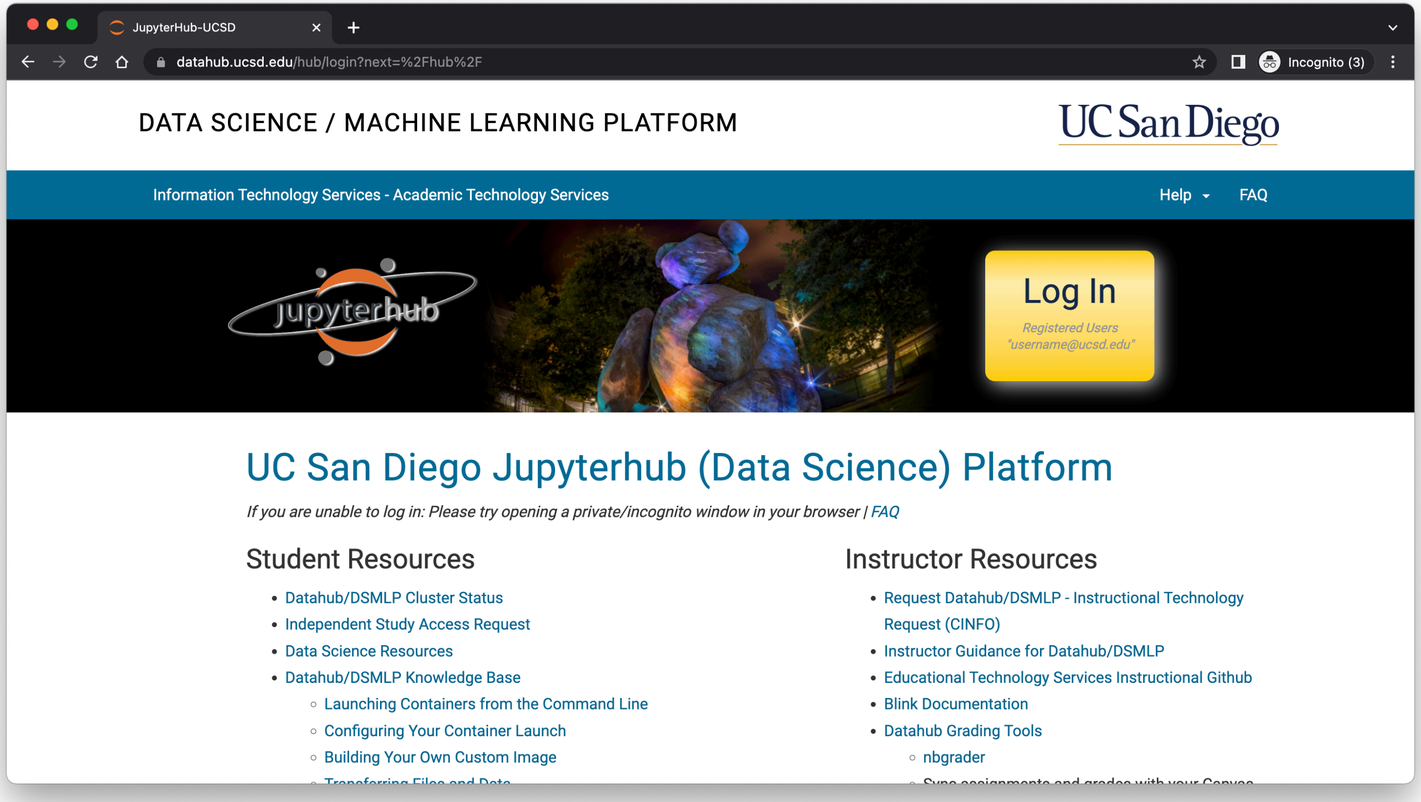
PHYS 142/242

Lab 01: DataHub, Jupyter, Python

Javier Duarte — January 9, 2023

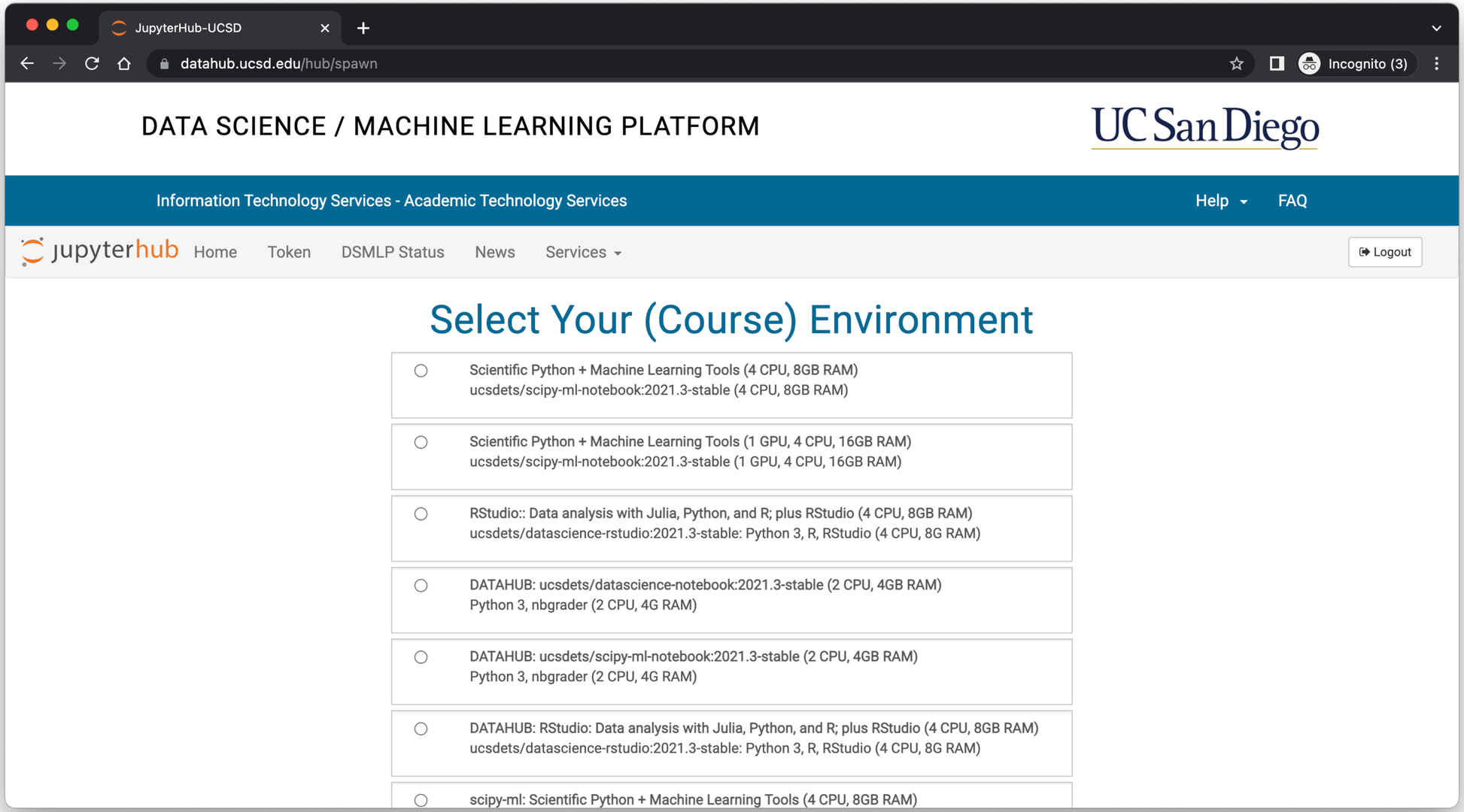
DataHub

- We will use DataHub for in-class hands-on portions
  - Recommend to use it for homework, final project, etc.
- Address: datahub.ucsd.edu
- Similar to public, free services Google Colab, but with access to better CPUs and GPUs and run by UCSD
- Provides a “Jupyter notebook” interface (Python-based but interactive coding like MATLAB/Mathematica)



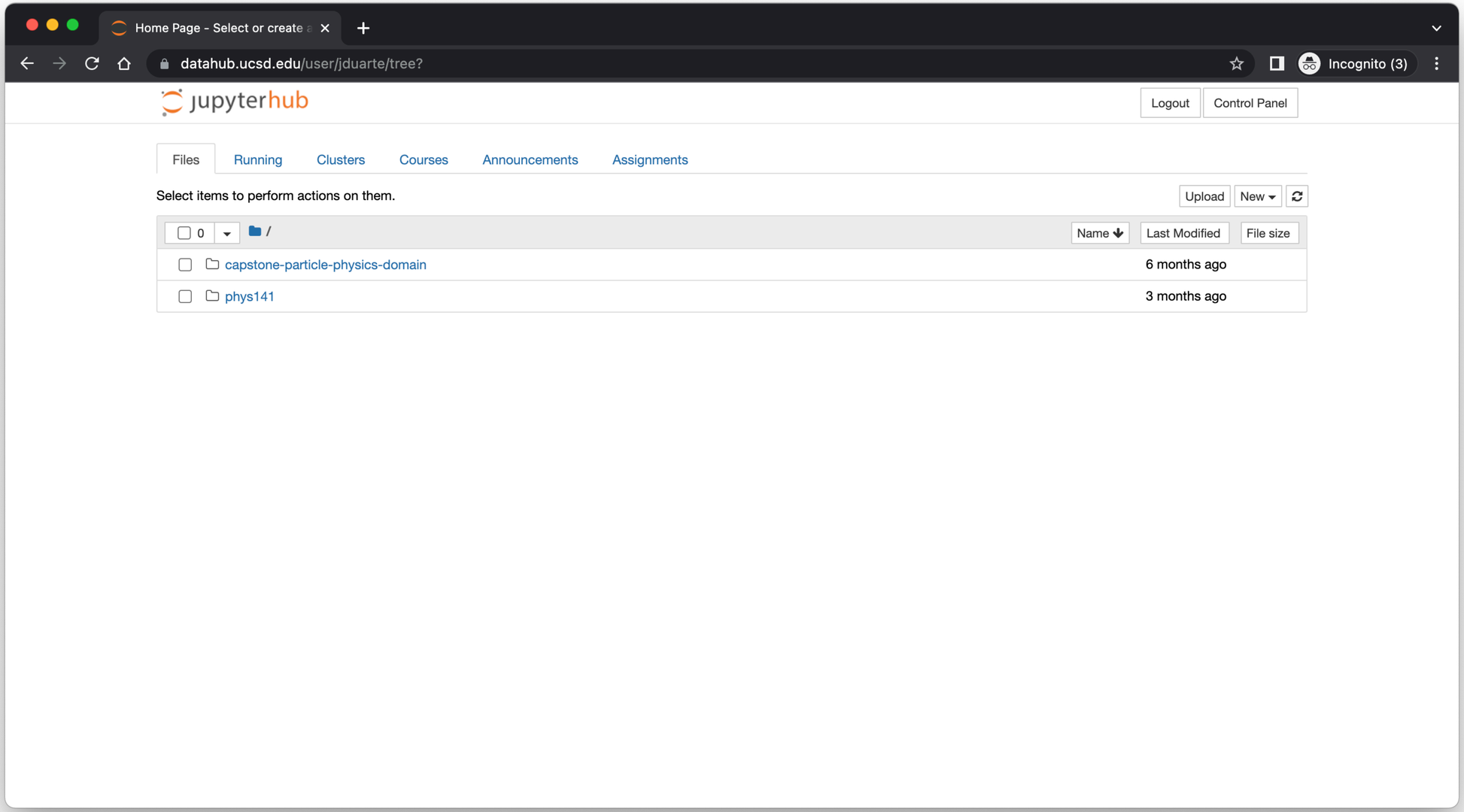
Logging in

- After logging in, choose a course environment…
  PHYS 141 PHYS 241 - Comp Phys I:Prob Models & Simu - Duarte [SP23]
  jmduarte/phys141:latest (2 CPU, 4G RAM, 1 GPU)



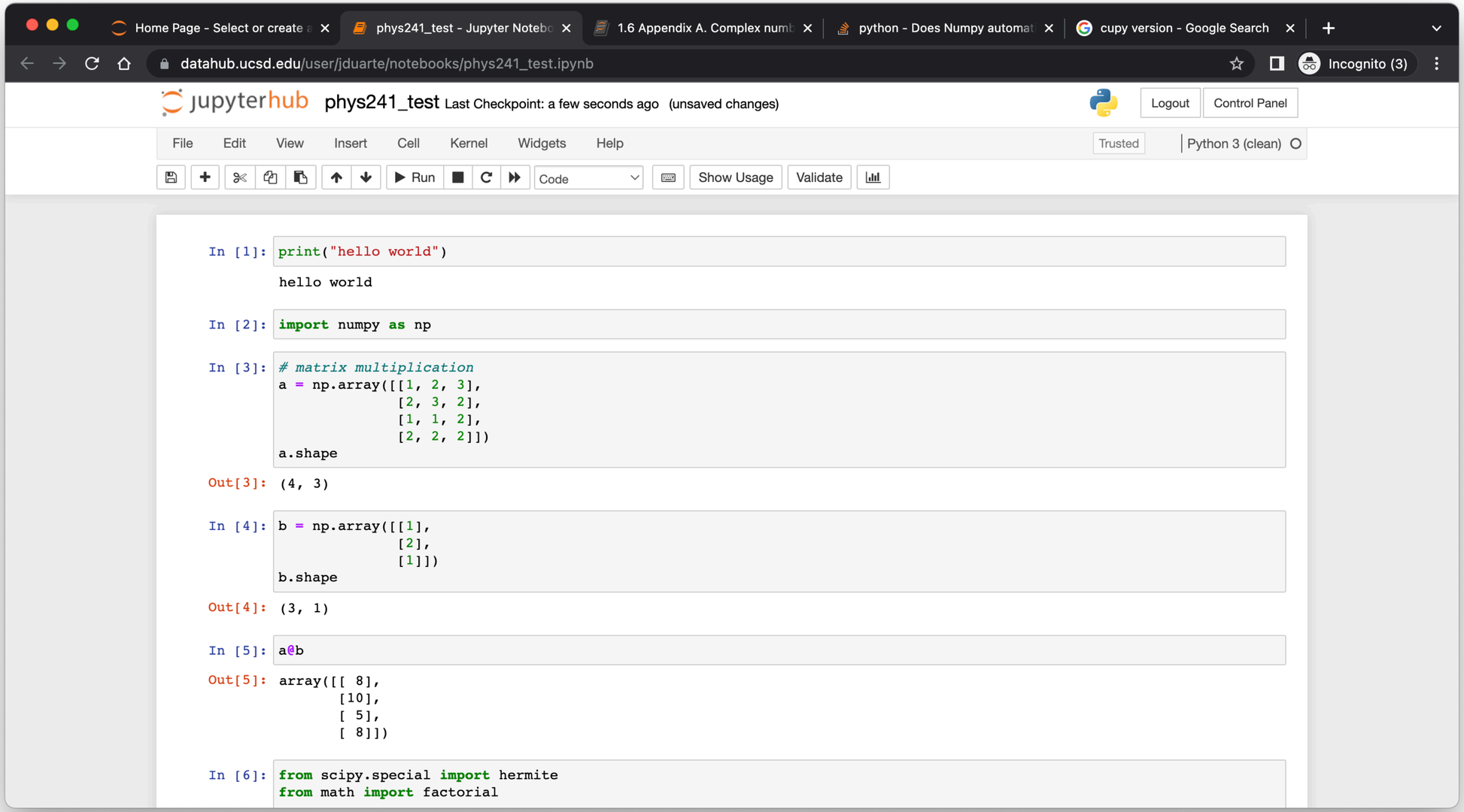
Jupyter interface

- Spawns a “JupyterHub” (let’s go step by step through all the buttons)



Start coding!

- Coding in Jupyter notebooks



Plotting

- Plotting can be done easily with Matplotlib

Home Page - Select or create a | phys241_test - Jupyter Notebo | 1.6 Appendix A. Complex numb | python - Does Numpy automat | cupy version - Google Search

datahub.ucsd.edu/user/jduarte/notebooks/phys241_test.ipynb

jupyterhub phys241_test Last Checkpoint: 2 minutes ago (autosaved)

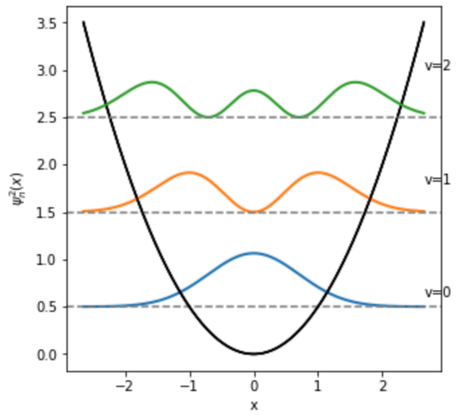
Logout | Control Panel

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (clean)

Run | Code | Show Usage | Validate

```
# Range of x determine by classical tunring points:
xmin, xmax = -np.sqrt(2 * E(VMAX)), np.sqrt(2 * E(VMAX))
x = np.linspace(xmin, xmax, 10000)
fig, ax = plt.subplots(figsize=(5,5))
for v in range(VMAX):
    # plot potential V(x)
    ax.plot(x, V(x), color='black')
    # plot psi squared which we shift up by values of energy
    ax.plot(x, psi(v,x) ** 2 + E(v), lw=2)
    # add lines and labels
    ax.axhline(E(v), color='gray', linestyle='--')
    ax.text(xmax, 1.2 * E(v), f"v={v}")
ax.set_xlabel('x')
ax.set_ylabel('$\psi^2_n(x)$')
plt.show()
```



Installing a missing library

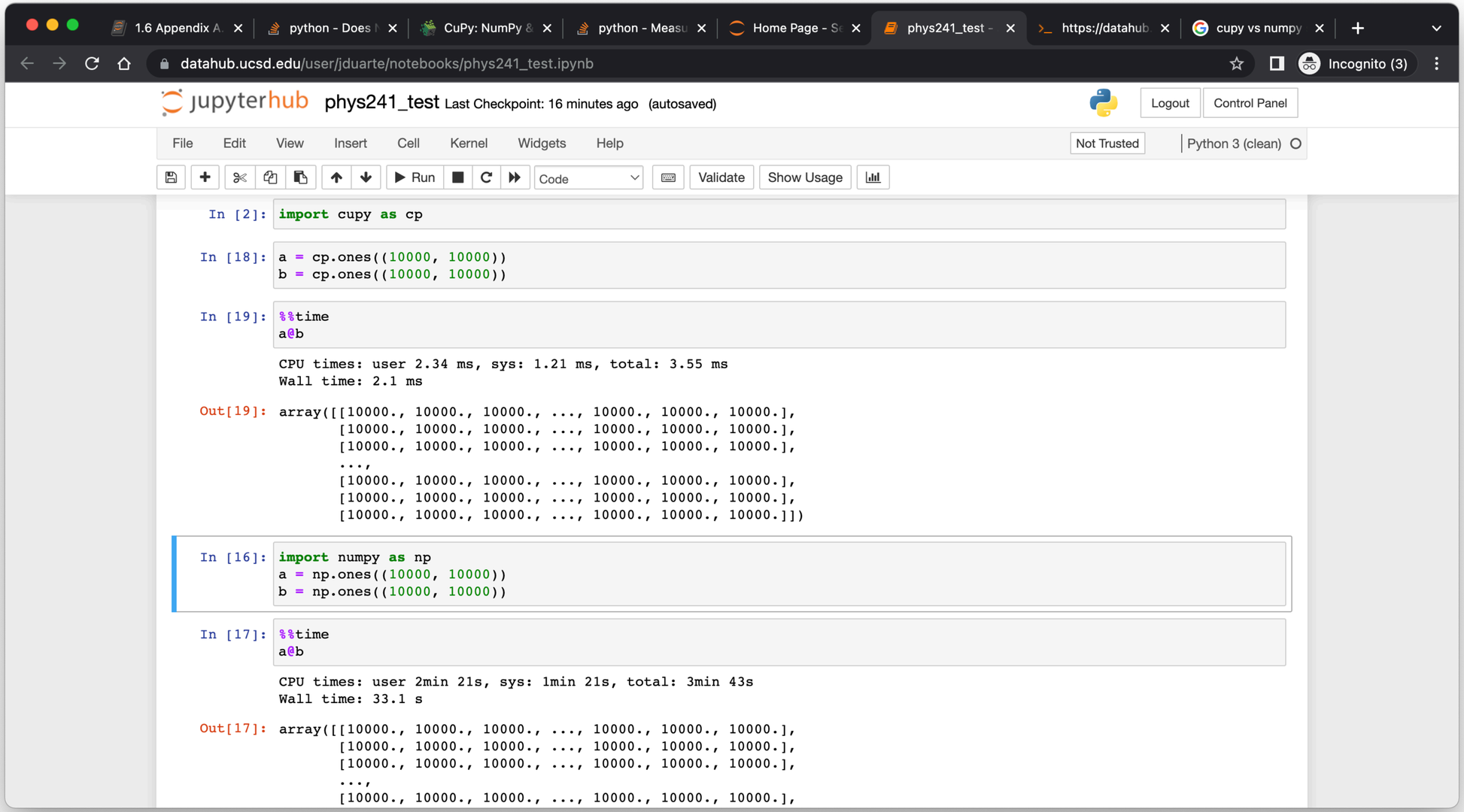
- Not all libraries are preloaded, but it’s easy to install a new one
- Note: restart your “kernel” after doing this

```
In [1]: !pip install cupy-cuda112 --user

        Collecting cupy-cuda112
          Using cached cupy_cuda112-10.3.1-cp39-cp39-manylinux1_x86_64.whl (78.9 MB)
        Collecting fastrlock>=0.5
          Using cached fastrlock-0.8-cp39-cp39-manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_12_x86_64.manylinux2010_x86
        _64.whl (49 kB)
        Requirement already satisfied: numpy<1.25,>=1.18 in /opt/conda/lib/python3.9/site-packages (from cupy-cuda112) (1.19.
        5)
        Installing collected packages: fastrlock, cupy-cuda112
        Successfully installed cupy-cuda112-10.3.1 fastrlock-0.8
```

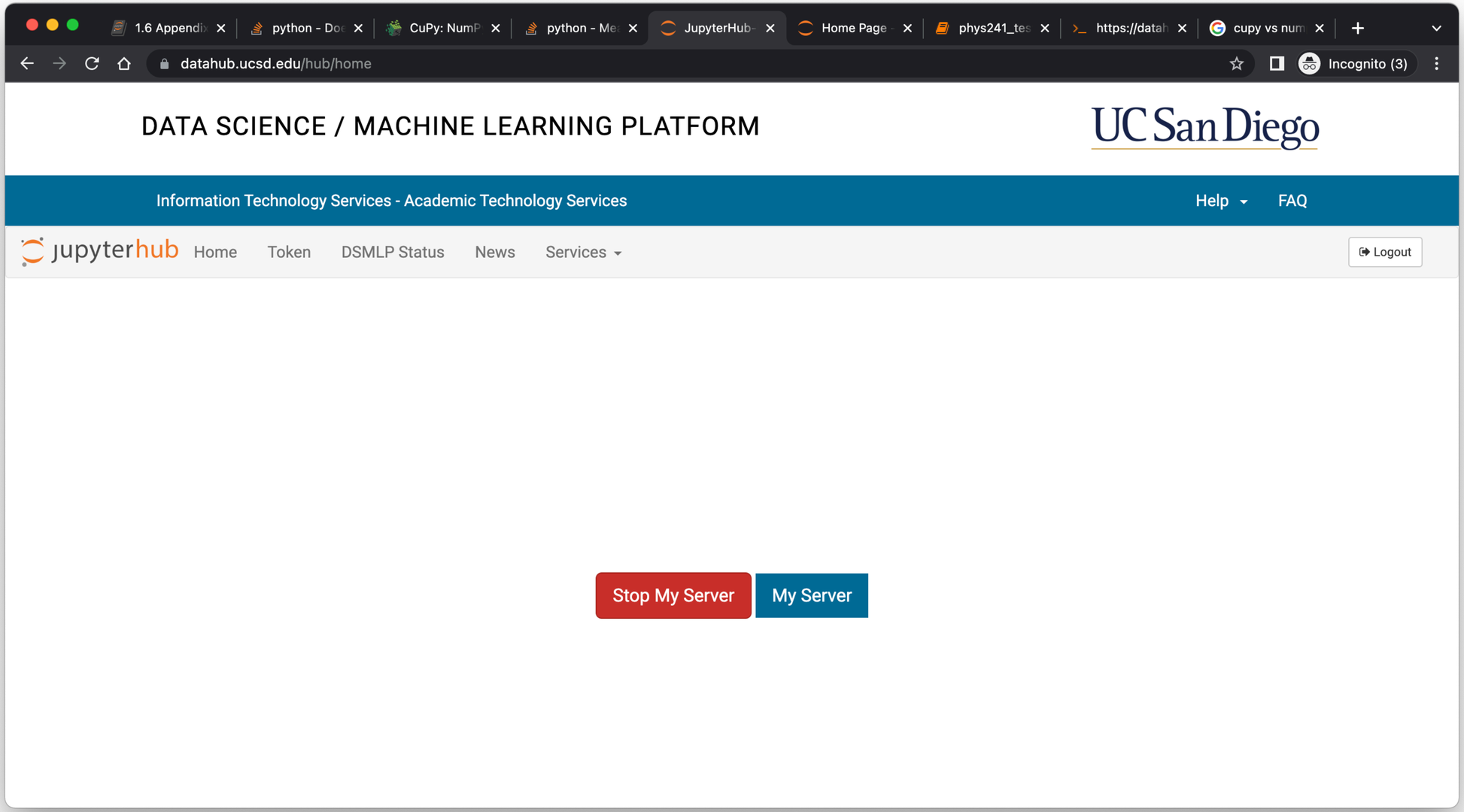
Speed up numerical calculations with the GPU

- Many libraries to do this: TensorFlow, CuPy, PyTorch, …



Exiting / shutting down server

- When you're done; hit control panel and "stop your server"
  When you start it again, all your data will still be there



Data Science & Machine Learning Platform (DSMLP)

- UCSD Data Science & Machine Learning Platform (DSMLP)
  - Based on docker and **Kubernetes** (K8s): open-source system for automating deployment, scaling, and management of containerized applications
- Overview:

  - You log in to a remote "login" node
  - From that login node, you can launch a “pod” running a container
    - The pod automatically starts a “JupyterHub” web server (only accessible from campus so you must VPN if off campus)
    - You are also automatically logged into that pod so you can run interactive terminal commands, etc.

More Resources

- UCSD DSMLP Cluster
  - Using the DSMLP server
    - Customizing the DSMLP Containers
- Terminal and Command-Line Interface
  - Bash Scripting Reference
  - Git(Hub) Resources