About this tutorial, and how to follow it

This tutorial assumes you are already comfortable enough in a Unix command-line environment, but new to HPC. If you have not used the command line before, we recommend working through Tutorial 1 first.

This tutorial also assumes you have an NYU HPC account and have set up your workstation for SSH tunneling to the NYU HPC clusters. If you do not yet have an account you can apply for one here.

In the Prequel section of this tutorial we’ll make sure your workstation is correctly set up to connect to and use the NYU HPC facilities. If you are attending a tutorial class in the library, please work through the Prequel section before the class begins!

The goal of this session is to prepare, submit and monitor the progress of a batch job.

This tutorial will cover:
- **How to find and use software** packages on the HPC clusters
- **How to reserve a part of the cluster** to run a simulation, so that your jobs' access to the CPU and memory resources they need is not impinged on by the jobs of other users, nor does it impinge on other users
  - We'll also touch on how to best use the cluster if you are developing and debugging a model
- Where to put input and output files that makes the best use of the clusters' storage infrastructure (hint: usually $SCRATCH$)
- **How to monitor your job** while it is queued and while it is running.

This tutorial is run as a regular class in the library (see the calendar on the NYU HPC Wiki front page for times) but can also be used for self-directed learning. Much of the tutorial is a walk through relevant pages of the NYU HPC Wiki, interspersed with additional explanation and exercises to try - so have a terminal window open to try things as you go.

The panels on the left are for navigation through the tutorial. There is more material here than we will have time to go over in the class, so the essential topics are indicated with **bold links in the navigation panels**.

**NYU HPC Facilities**

First, a quick look at what systems NYU HPC provides:

Next: Finding and using software with Environment Modules