Introduction to Bash

First Year Bootcamp 2020

Opening bash

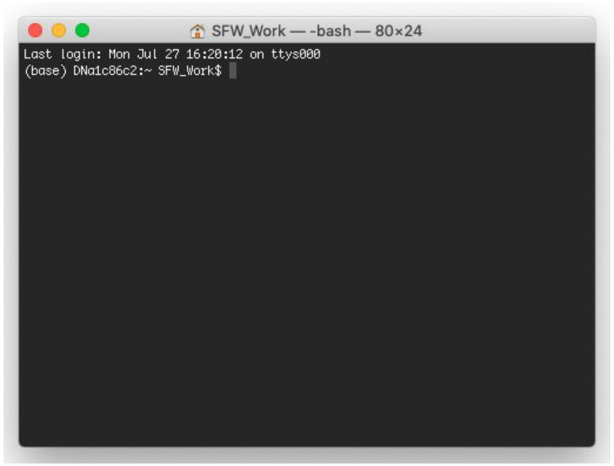
OS X/Linux: Open terminal, bash is default shell

Windows: Go to Start → Git → Git Bash

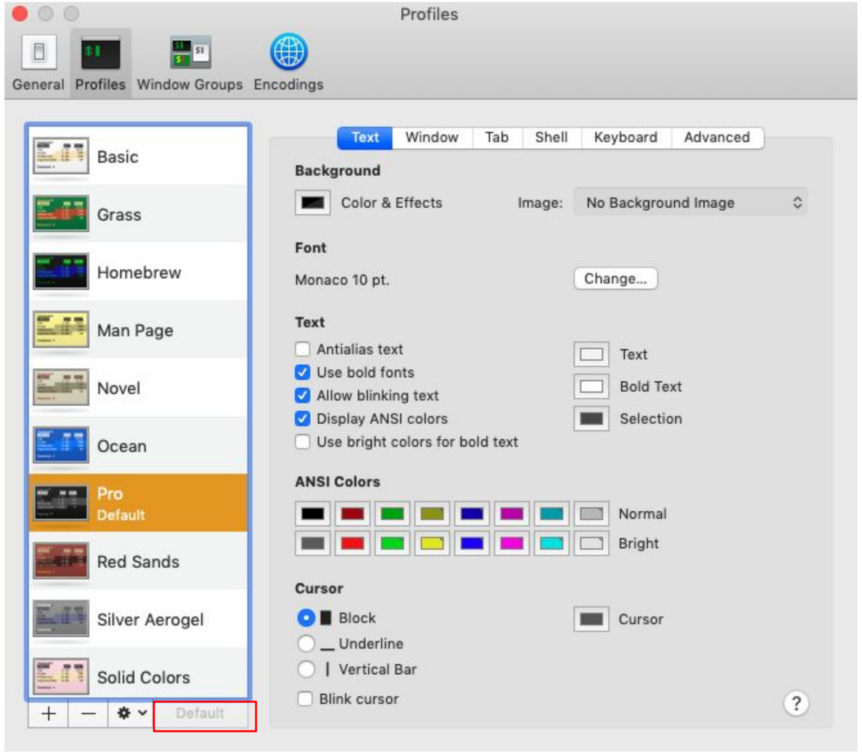
Look like a pro

Terminal > Preferences > Text

Select a Profiles tyle you like and click "Default"


SFW_Work — -bash — 80×24

```
Last login: Mon Jul 27 16:20:12 on ttys000
(base) DNa1c86c2:~ SFW_Work$ 
```


Profiles

General | Profiles | Window Groups | Encodings

Basic
Grass
Homebrew
Man Page
Novel
Ocean
Pro
Default
Red Sands
Silver Aerogel
Solid Colors

+ — Default

Text | Window | Tab | Shell | Keyboard | Advanced

Background
Color & Effects   Image: No Background Image

Font
Monaco 10 pt.   Change...

Text
- [ ] Antialias text
- [x] Use bold fonts
- [x] Allow blinking text
- [x] Display ANSI colors
- [ ] Use bright colors for bold text

Text
Bold Text
Selection

ANSI Colors
Normal
Bright

Cursor
- [✓ Block]
- Underline
- Vertical Bar
- [ ] Blink cursor

Cursor

There are computers other than mine?

Yes, yes there are. Now we’re going to show you how working with files on

other computers from bash isn’t too much harder than working with files

on your own computer, using the tools ssh and scp. For this example, we’ll show you how to use Sherlock.

Sherlock

Sherlock is a High-Performance Computing (HPC) cluster, operated by the Stanford Research Computing Center to provide computing resources to the Stanford community at large.

How much does it cost?

Sherlock is **free** to use for anyone doing sponsored research at Stanford. Any faculty member can request access for research purposes, and get an account with a base storage allocation and unlimited compute time on the global, shared pool of resources. In case those free resources are not sufficient, Stanford Research Computing offers Faculty members the opportunity to invest into the cluster, and get access to additional computing resources for their research teams.

Unlike traditional clusters, Sherlock is a collaborative system where the majority of nodes are purchased and shared by the cluster users. When a user (typically a PI) purchases one or more nodes, they become an *owner*.

The resource scheduler configuration works like this:

- owners and their research teams get immediate and exclusive access to the resources they purchased,
- when those nodes are idle, other owners can use them,
- when the purchasing owners want to use their resources, jobs from other owners that may be running on them are preempted (*ie.* killed and re-queued).

Getting start

To start using Sherlock, you will need:

- an active SUNet ID,
- a Sherlock account,
- a SSH client,
- good understanding of the concepts and terms used throughout that documentation,
- some familiarity with Unix/Linux command-line environments, and notions of shell scripting.

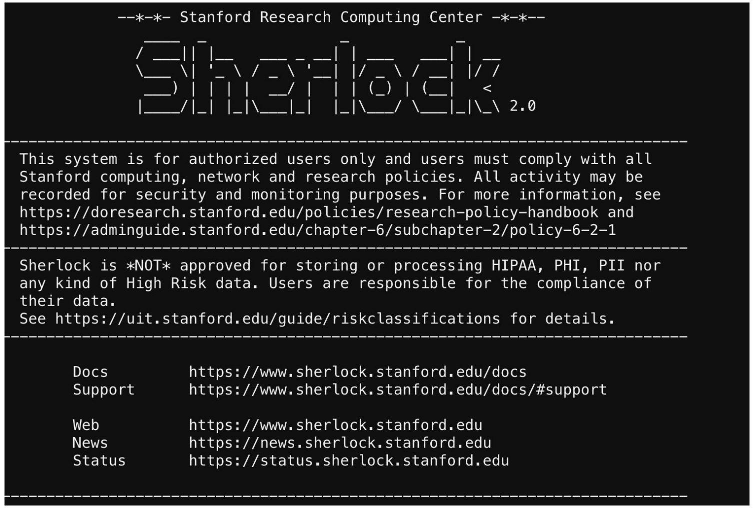
For more information: https://www.sherlock.stanford.edu/docs/getting-started/prerequisites/

Connect to Sherlock

To login to Sherlock, open a terminal and type the following command, where `<sunetid>` should be replaced by your *actual* SUNet ID:

```
$ ssh <sunetid>@login.sherlock.stanford.edu
```

When prompted for you password, type the password that corresponds to your sunet ID. YOu will need to use two-factor authentication.
Then, you should see a welcome screen:



```
            --*-*- Stanford Research Computing Center -*-*--
              ____  _               _            _
             / ___|| |__   ___ _ __| | ___   ___| | __
             \___ \| '_ \ / _ \ '__| |/ _ \ / __| |/ /
              ___) | | | |  __/ |  | | (_) | (__|   <
             |____/|_| |_|\___|_|  |_|\___/ \___|_|\_\ 2.0

-----------------------------------------------------------------------------
 This system is for authorized users only and users must comply with all
 Stanford computing, network and research policies. All activity may be
 recorded for security and monitoring purposes. For more information, see
 https://doresearch.stanford.edu/policies/research-policy-handbook and
 https://adminguide.stanford.edu/chapter-6/subchapter-2/policy-6-2-1
-----------------------------------------------------------------------------
 Sherlock is *NOT* approved for storing or processing HIPAA, PHI, PII nor
 any kind of High Risk data. Users are responsible for the compliance of
 their data.
 See https://uit.stanford.edu/guide/riskclassifications for details.
-----------------------------------------------------------------------------

       Docs         https://www.sherlock.stanford.edu/docs
       Support      https://www.sherlock.stanford.edu/docs/#support

       Web          https://www.sherlock.stanford.edu
       News         https://news.sherlock.stanford.edu
       Status       https://status.sherlock.stanford.edu

-----------------------------------------------------------------------------
```

Requesting resources (i.e., submit a job)

Requesting computing resources on Sherlock is done via a resource scheduler, whose very purpose is to match compute resources in the cluster (CPUs, GPUs, memory, ...) with user resource requests.

The scheduler provides three key functions:

1. it allocates access to resources (compute nodes) to users for some duration of time so they can perform work.
2. it provides a framework for starting, executing, and monitoring work (typically a parallel job such as MPI) on a set of allocated nodes.
3. it arbitrates contention for resources by managing a queue of pending jobs

How to submit a job

A job consists in two parts: resource requests and job steps.

Resource requests describe the amount of computing resource (CPUs, GPUs, memory, expected run time, etc.) that the job will need to successfully run.

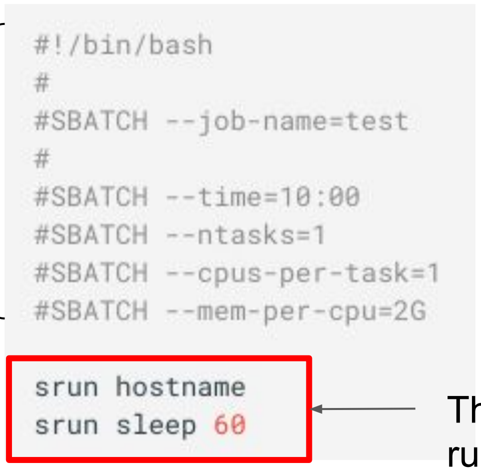
Job steps describe tasks that must be executed.

Batch scripts

The typical way of creating a job is to write a job submission script. A submission script is a shell script (e.g. a Bash script) whose first comments, if they are prefixed with `#SBATCH`, are interpreted by Slurm as parameters describing resource requests and submissions options.

For instance, the following script would request one task with one CPU for 10 minutes, along with 2 GB of memory, in the default partition:

Features of the resources needed to run your task

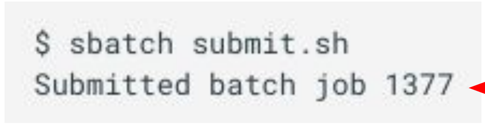


```
#!/bin/bash
#
#SBATCH --job-name=test
#
#SBATCH --time=10:00
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=1
#SBATCH --mem-per-cpu=2G

srun hostname
srun sleep 60
```

This is the task you want to run on Sherlock

Job submission and check job

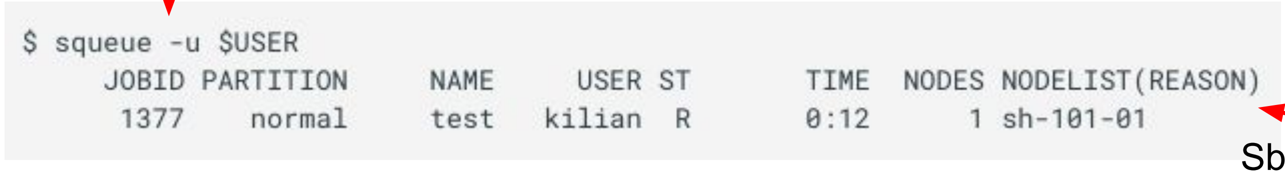
Once the submission script is complete, you can submit the script to the scheduler with the sbatch command:



```
$ sbatch submit.sh
Submitted batch job 1377
```

Sbatch will return the ID it has assigned to the job

Once submitted, the job enters the queue in the Pending state. You can check the job status using the following command:



```
$ squeue -u $USER
     JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
      1377    normal     test   kilian  R       0:12      1 sh-101-01
```

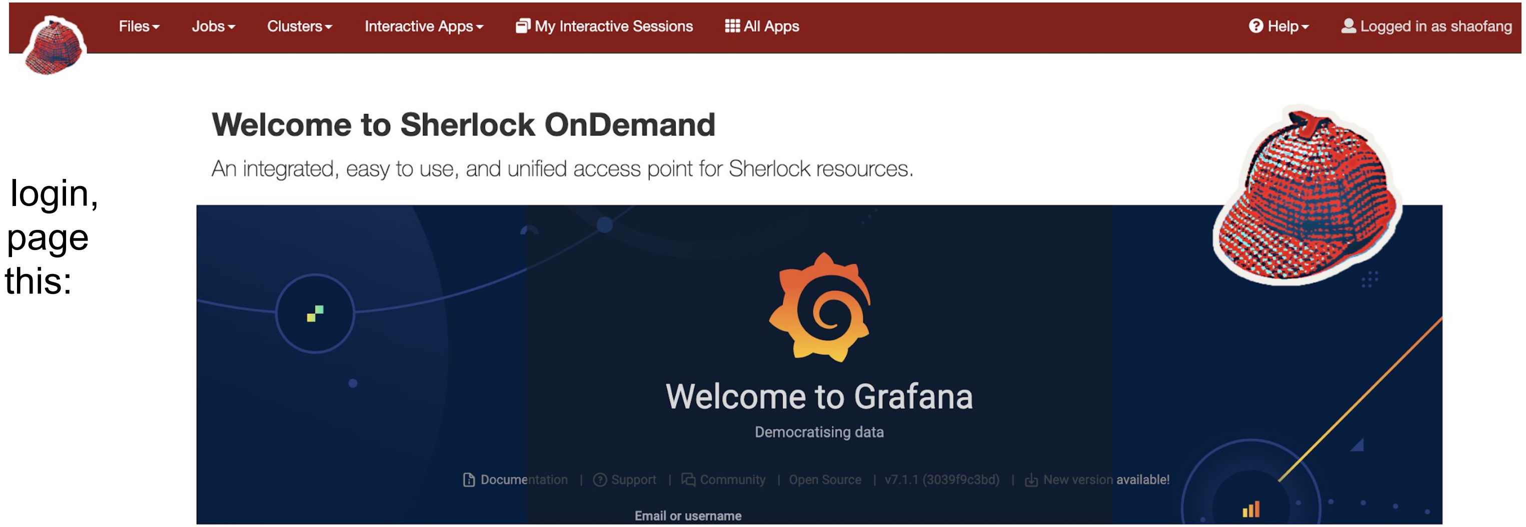
Sbatch will return the job status

Sherlock OnDemand

The Sherlock OnDemand interface allows you to conduct your research on Sherlock through a web browser. You can manage files (create, edit and move them), submit and monitor your jobs, see their output, check the status of the job queue, run a Jupyter notebook and much more, without logging in to Sherlock the traditional way, via a SSH terminal connection.

To connect to Sherlock OnDemand, simply point your browser to **https://login.sherlock.stanford.edu**

Once you login, the home page looks like this:



Welcome to Sherlock OnDemand

An integrated, easy to use, and unified access point for Sherlock resources.

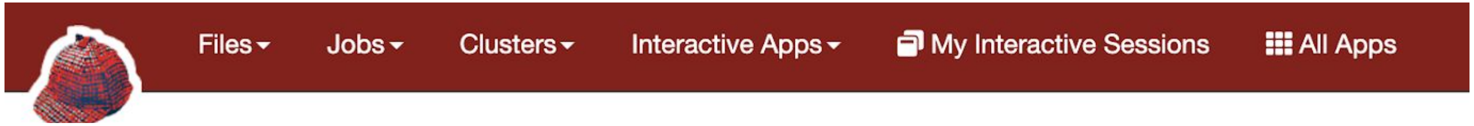
Welcome to Grafana

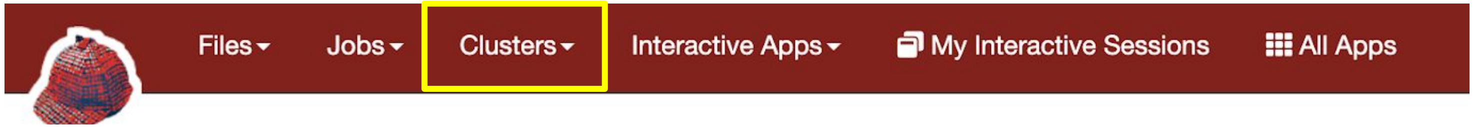
Democratising data

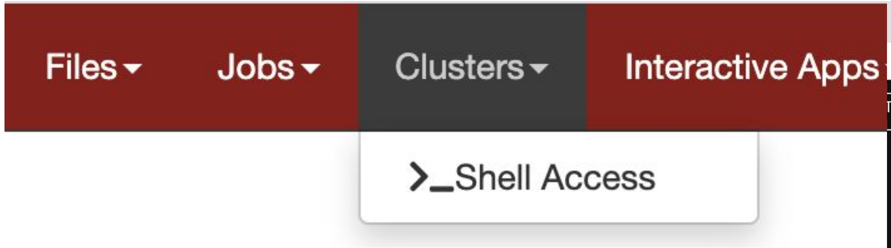
Documentation | Support | Community | Open Source | v7.1.1 (3039f9c3bd) | New version available!

Email or username

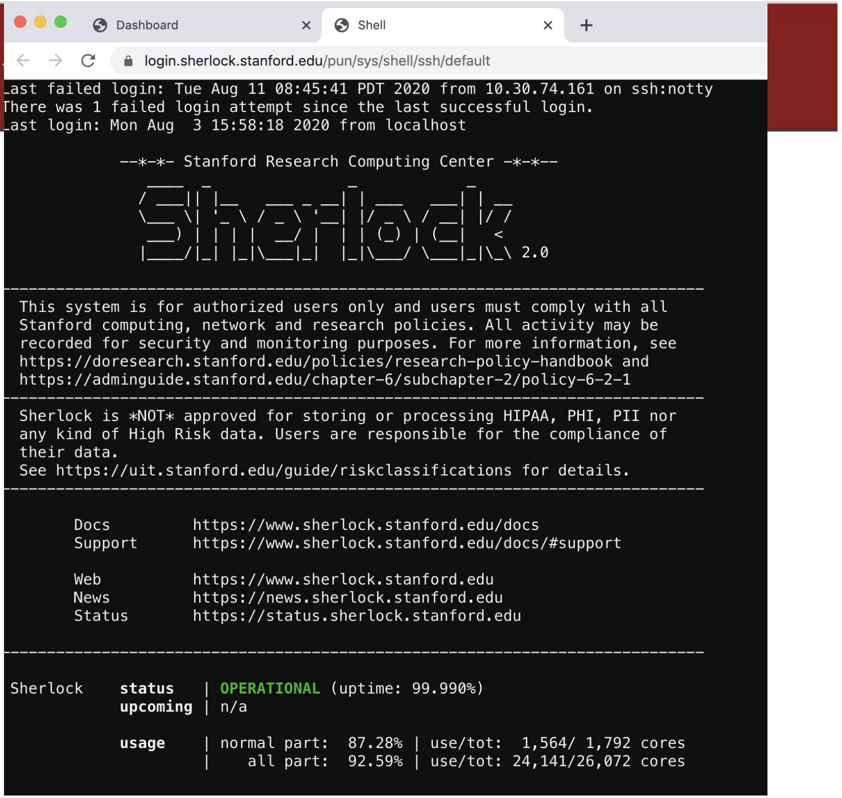
Let's go through a few useful things you can do with Sherlock OnDemand:



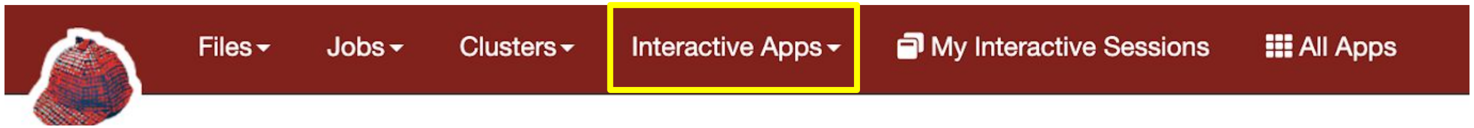


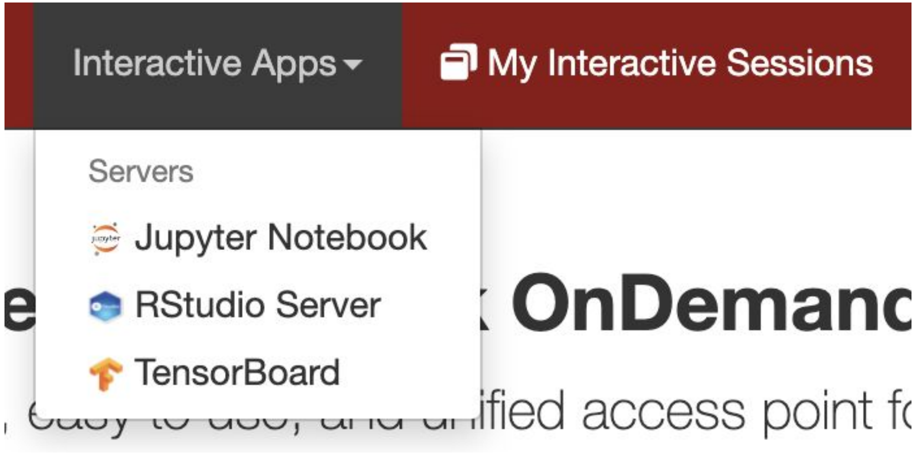


You can get shell access to Sherlock in the browser. This is exactly the same as you ssh to Sherlock using terminal.



```
Last failed login: Tue Aug 11 08:45:41 PDT 2020 from 10.30.74.161 on ssh:notty
There was 1 failed login attempt since the last successful login.
Last login: Mon Aug  3 15:58:18 2020 from localhost

            --*-*- Stanford Research Computing Center -*-*--
               ____  _               _            _
              / ___|| |__   ___ _ __| | ___   ___| | __
              \___ \| '_ \ / _ \ '__| |/ _ \ / __| |/ /
               ___) | | | |  __/ |  | | (_) | (__|   <
              |____/|_| |_|\___|_|  |_|\___/ \___|_|\_\ 2.0

-----------------------------------------------------------------------------
 This system is for authorized users only and users must comply with all
 Stanford computing, network and research policies. All activity may be
 recorded for security and monitoring purposes. For more information, see
 https://doresearch.stanford.edu/policies/research-policy-handbook and
 https://adminguide.stanford.edu/chapter-6/subchapter-2/policy-6-2-1
-----------------------------------------------------------------------------
 Sherlock is *NOT* approved for storing or processing HIPAA, PHI, PII nor
 any kind of High Risk data. Users are responsible for the compliance of
 their data.
 See https://uit.stanford.edu/guide/riskclassifications for details.
-----------------------------------------------------------------------------

       Docs         https://www.sherlock.stanford.edu/docs
       Support      https://www.sherlock.stanford.edu/docs/#support

       Web          https://www.sherlock.stanford.edu
       News         https://news.sherlock.stanford.edu
       Status       https://status.sherlock.stanford.edu

-----------------------------------------------------------------------------

Sherlock    status   | OPERATIONAL (uptime: 99.990%)
            upcoming | n/a

            usage    | normal part:  87.28% | use/tot:  1,564/ 1,792 cores
                     |    all part:  92.59% | use/tot: 24,141/26,072 cores
```

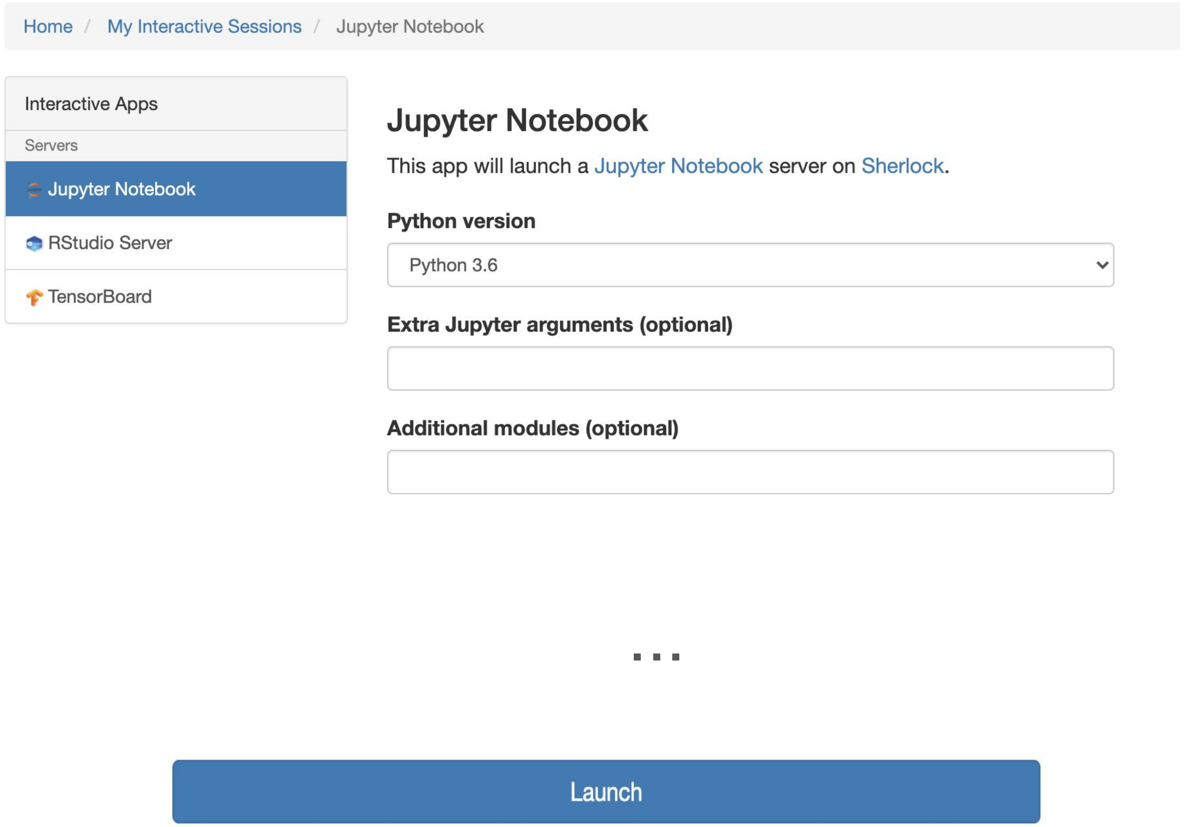




You can run Jupyter Notebook, RStudio, and TensorBoard on Sherlock using interactive Apps.

For example, you can request to run Jupyter Notebook on Sherlock. Once you click “Jupyter Notebook”, you will need to fill out this form to specify the different parameters for your job (time limit, number of nodes, CPUs, partition to use, etc.).

Once you finish the form, you can click the Launch button to start the session.



Jupyter Notebook

This app will launch a Jupyter Notebook server on Sherlock.

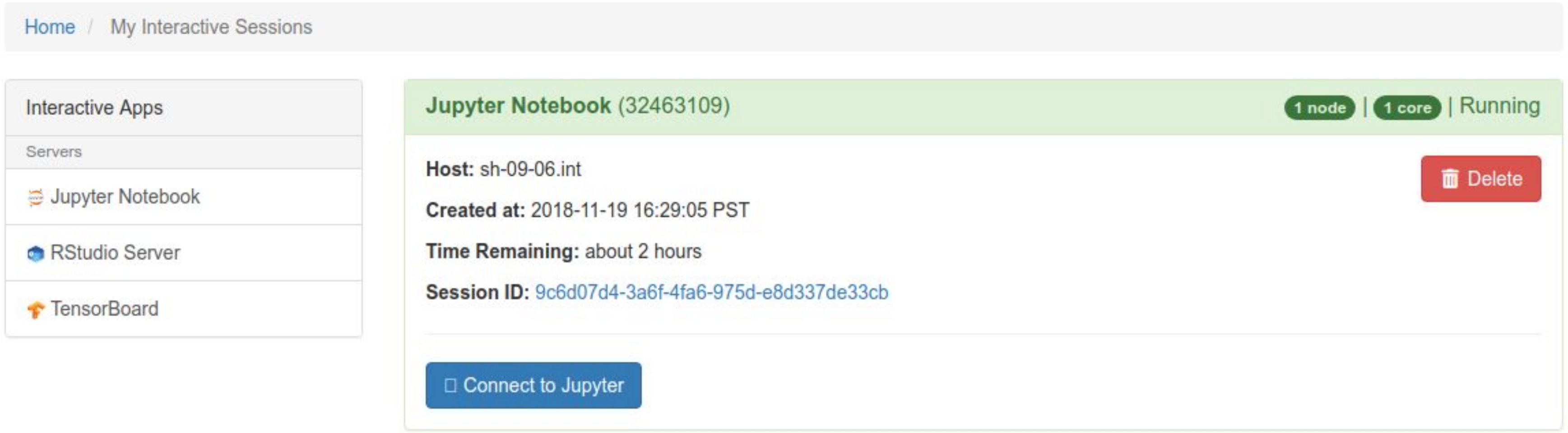
Python version

Python 3.6

Extra Jupyter arguments (optional)

Additional modules (optional)

...


Home / My Interactive Sessions

Interactive Apps

Servers

- Jupyter Notebook
- RStudio Server
- TensorBoard

Jupyter Notebook (32463109) 1 node | 1 core | Running

Delete

Host: sh-09-06.int

Created at: 2018-11-19 16:29:05 PST

Time Remaining: about 2 hours

Session ID: 9c6d07d4-3a6f-4fa6-975d-e8d337de33cb

Connect to Jupyter

Once your session is read, you can see this in “My Interactive Sessions”. Click the blue “Connect to Jupyter” button to open Jupyter Notebook!