TACC
TEXAS ADVANCED COMPUTING CENTER
WWW.TACC.UTEXAS.EDU

TEXAS
The University of Texas at Austin

Machine Learning on HPC: Introduction to ML Support at TACC

Amit Gupta
Scalable Computational Intelligence Group
Texas Advanced Computing Center
University of Texas at Austin

Overview

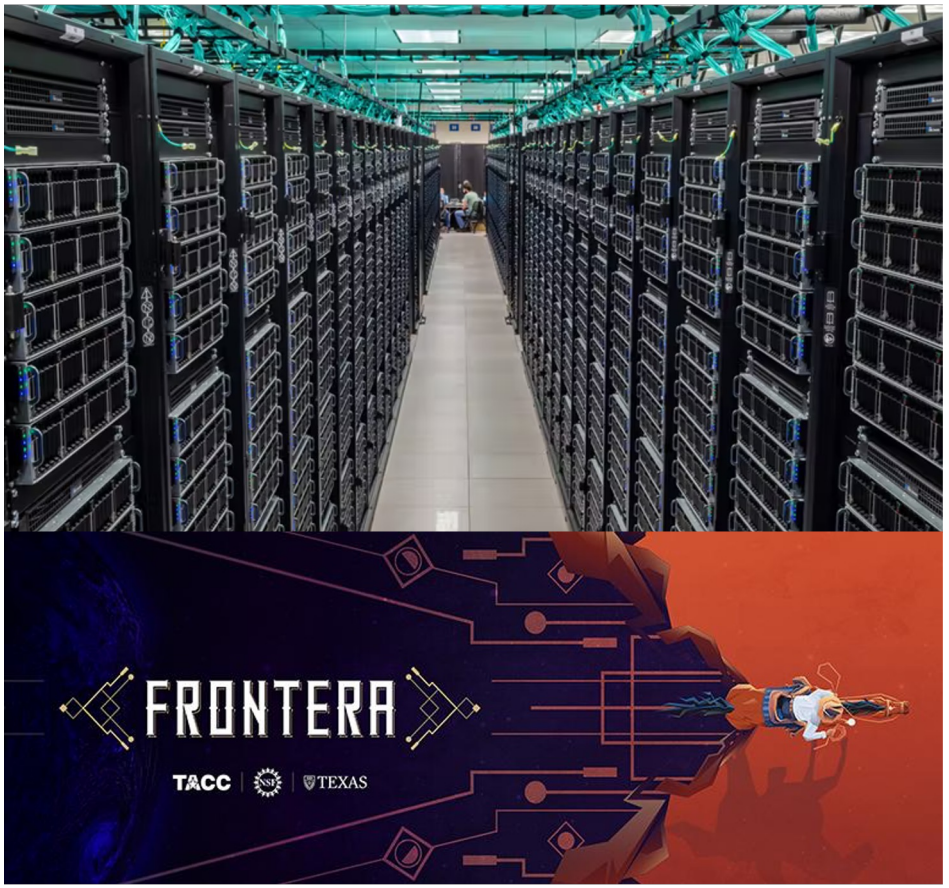
Practical introduction to using TACC resources

- How to Remotely Access TACC
- File system and nodes on TACC
- Navigating the File System on TACC machines
- TACC specific tools

Accessing TACC Machines Remotely

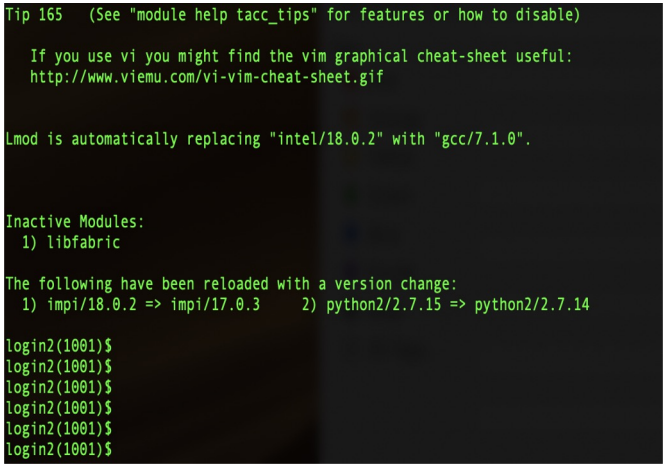
Personal Laptop

Remote Access:
ssh
Analysis Portal



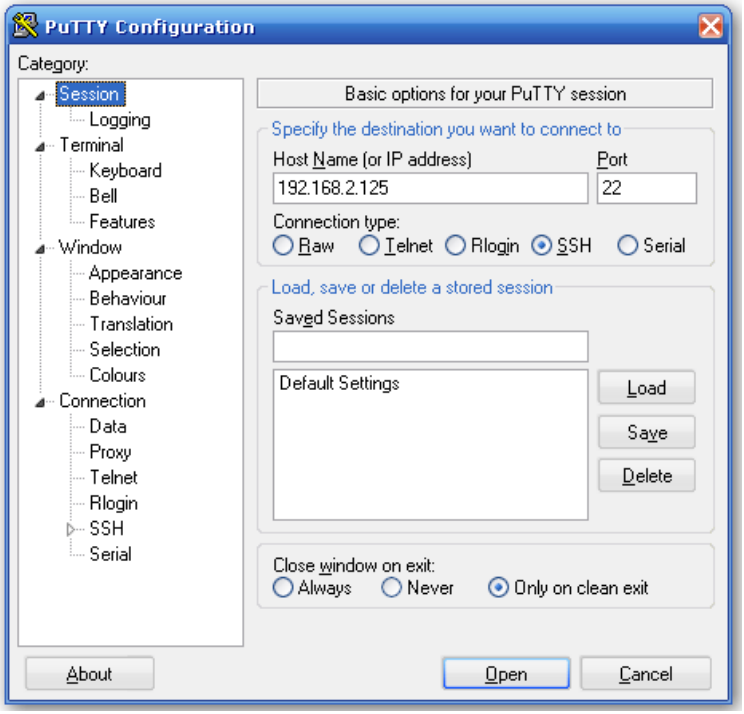
Secure Shell (ssh): Accessing Shell

- Secure way to connect to a remote machine
  - Uses encryption so as to make communication secure
- Upon successful login
  - User given a shell prompt
- Run commands
  - As per permissions assigned to your user
- Disconnect when done
  - Closes the secure connection
- Read User Guide for more information on how to access frontera with ssh



Secure Shell (ssh): Accessing Shell

- On Linux/Mac/Unix based machines
  - Open the Terminal program for your OS
  - Run ssh, using your TACC username
  - **ssh username@frontera.tacc.utexas.edu**
- On Windows
  - Download Putty
  (https://putty.org)
  - Hostname
    - **frontera.tacc.utexas.edu**
    - **Choose option SSH**
    - **Authenticate with TACC credentials**



TACC Analysis Portal

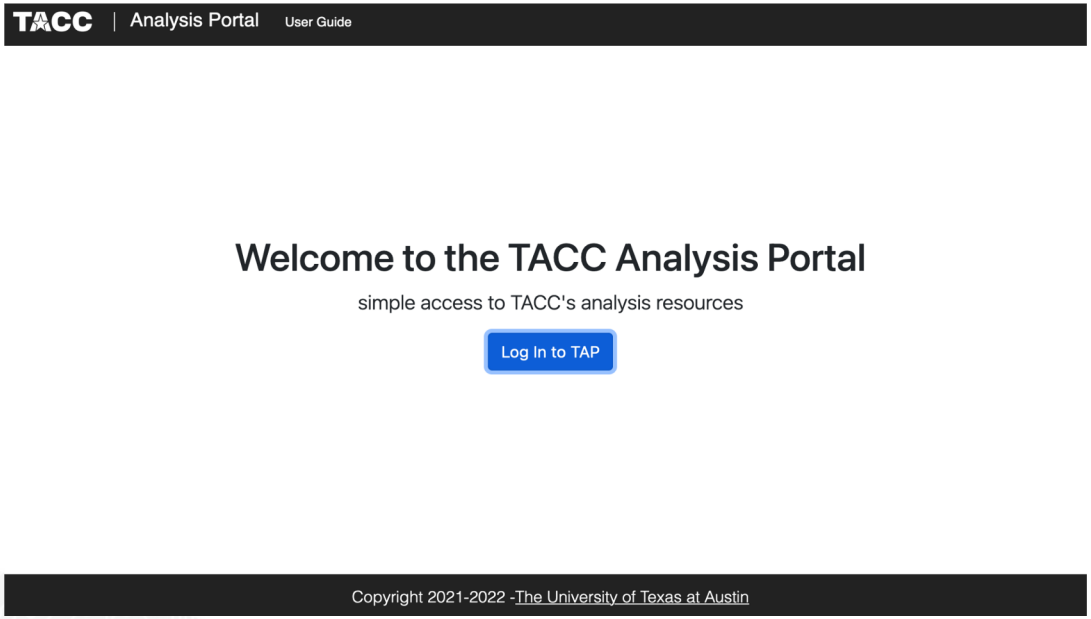
- TACC Analysis Portal (TAP) provides access to interactive sessions on TACC machines
- Allows user to utilize web-based interactions on TACC computers
  - e.g. Jupyter notebook, Rstudio

Development Environment

- Jupyter Notebook
  - http://jupyter.org/
  - web application for creating and sharing computational documents.
  - It offers a simple, streamlined, document-centric experience.
  - Requires a Python installation
  - Provides an interactive interface in the browser
    - Can view output and plots inline
    - Makes Iterative/Interactive Development easy
- Available on TACC machines
  - Frontera
  - Stampede3
  - Lonestar6

TACC Analysis Portal

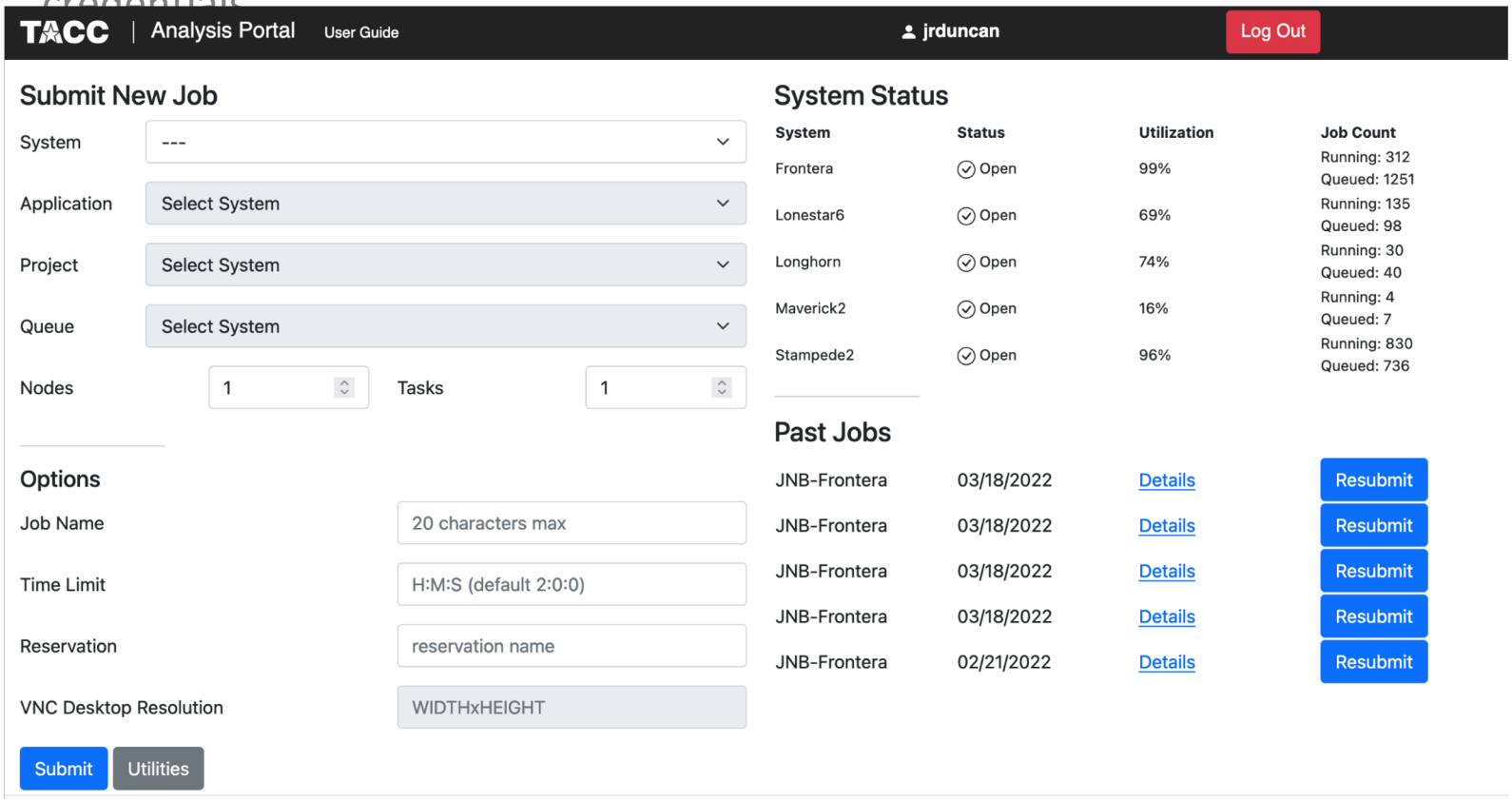
- TACC Analysis Portal (TAP) provides access to interactive sessions on TACC machines
- Allows user to utilize web-based interactions on TACC computers
  - e.g. Jupyter notebook, Rstudio
- tap.tacc.utexas.edu



Jupyter notebook: Accessing shell

- TACC Analysis Portal:

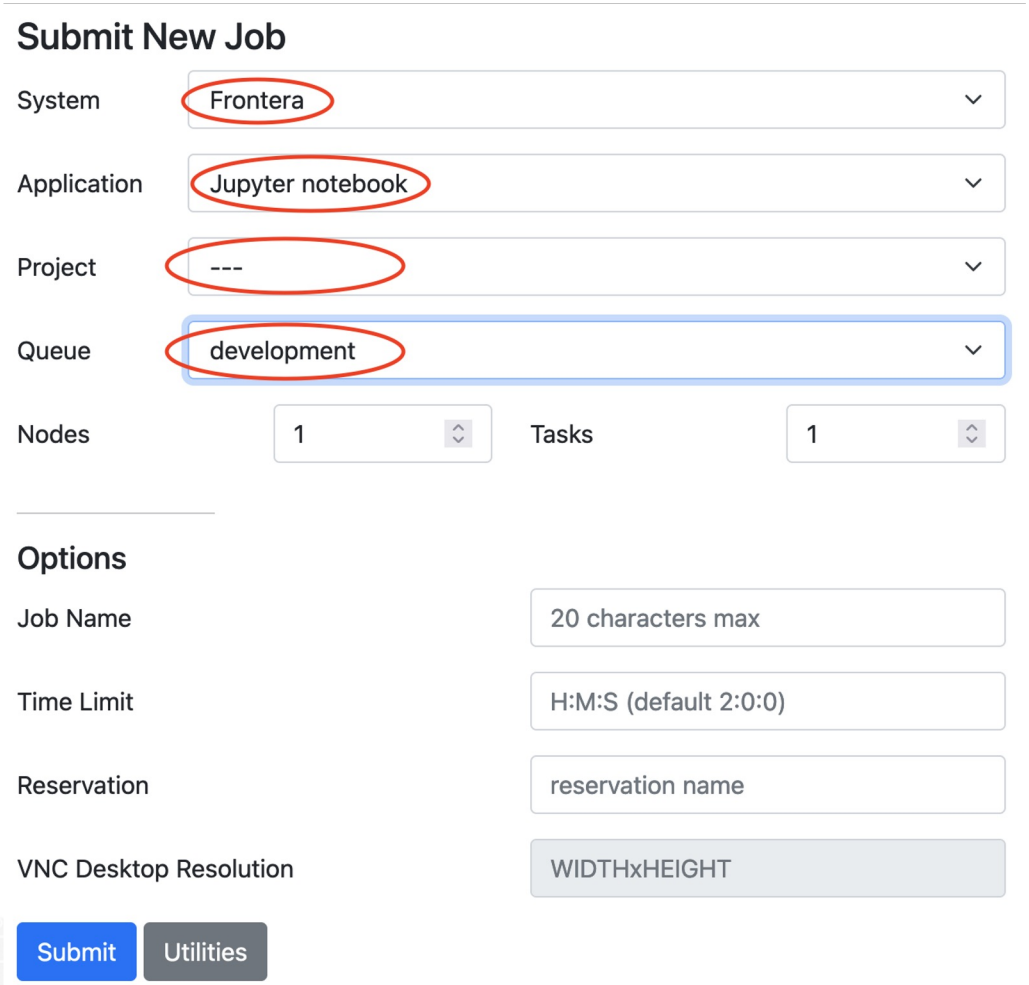
  Go to https://tap.tacc.utexas.edu; Login with your training account credentials



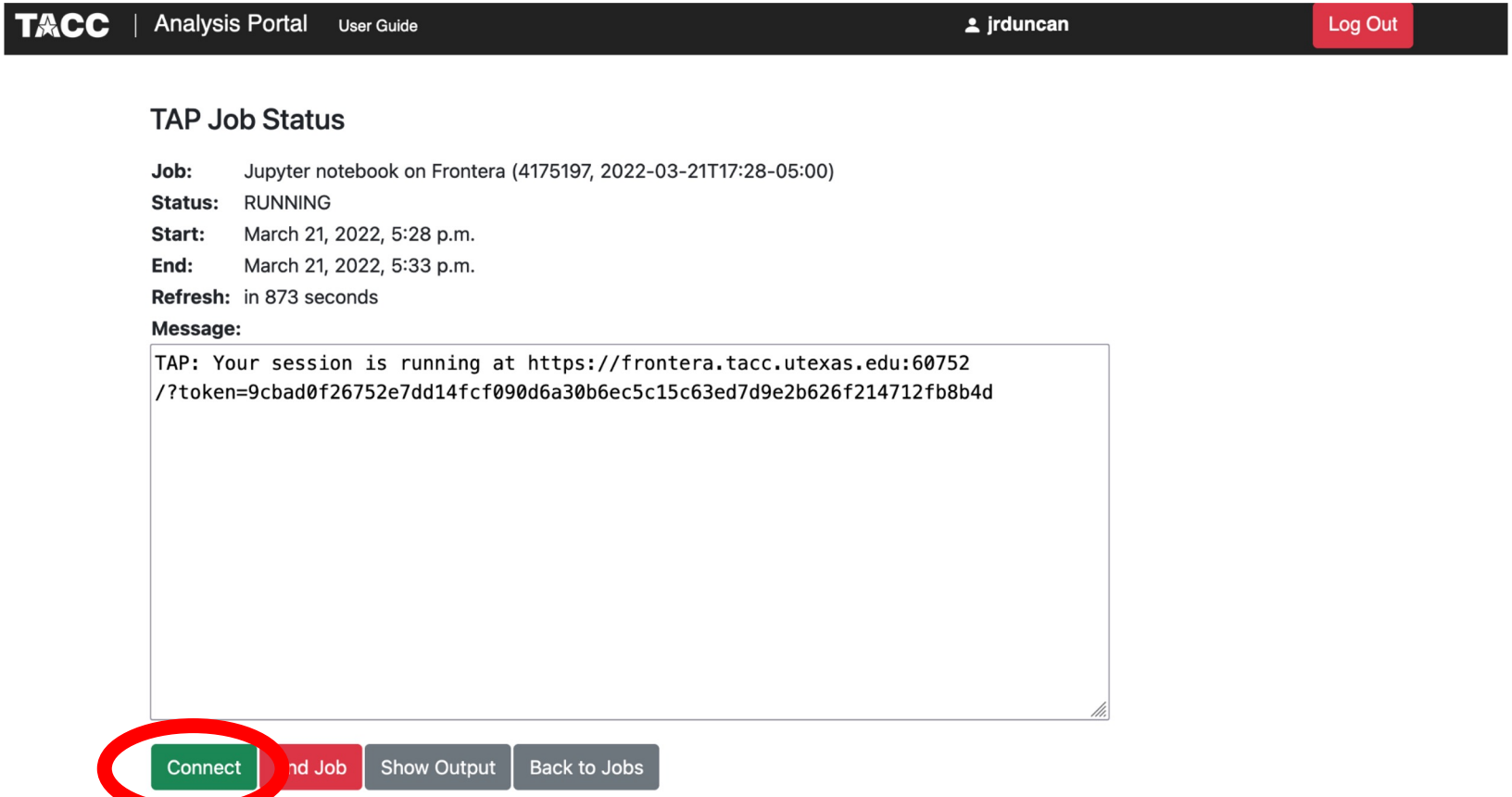
Jupyter notebook: Accessing shell

- TACC Analysis Portal:

  Select the following options

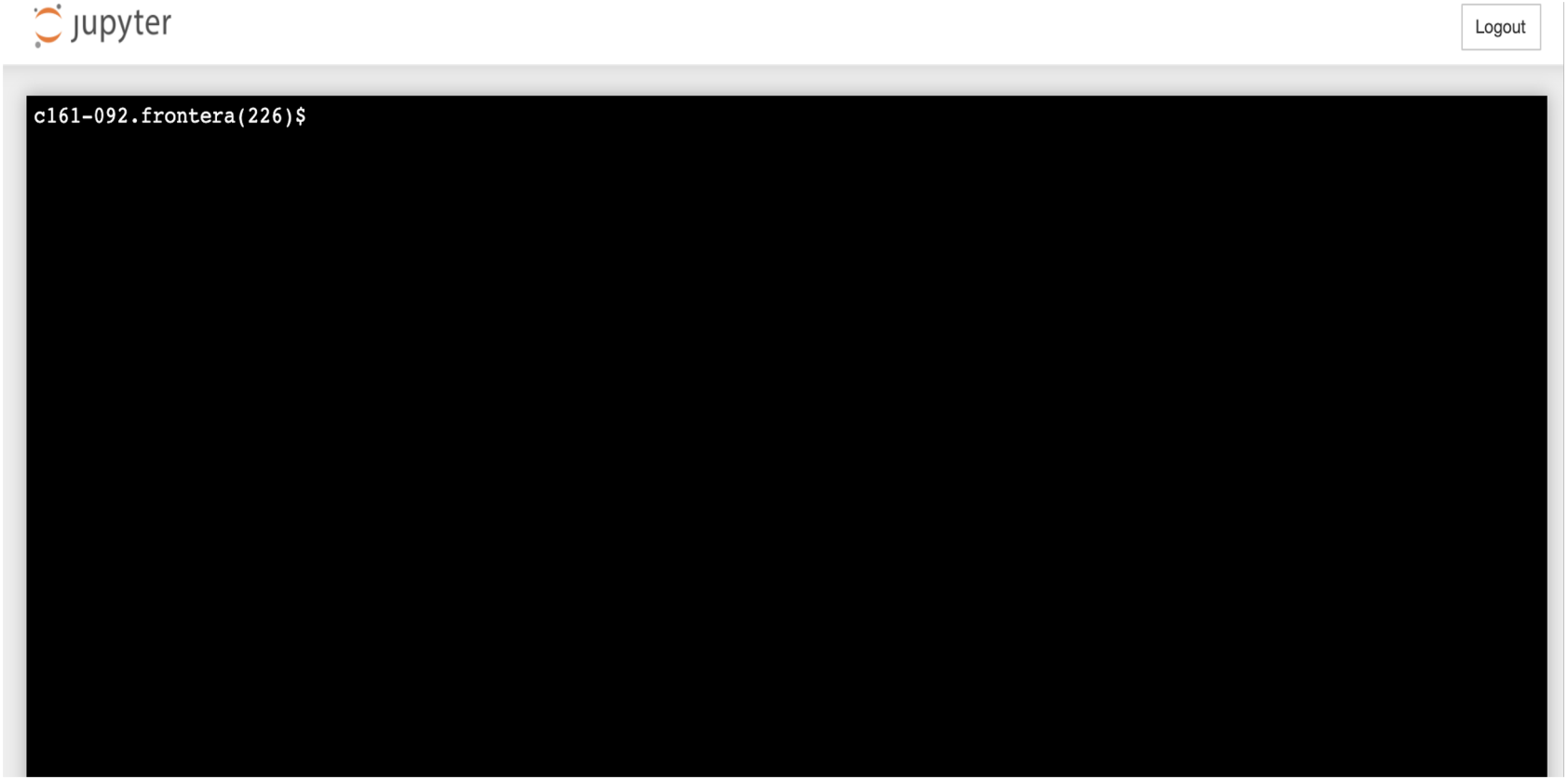


Jupyter notebook: Accessing shell



Jupyter notebook: Accessing shell

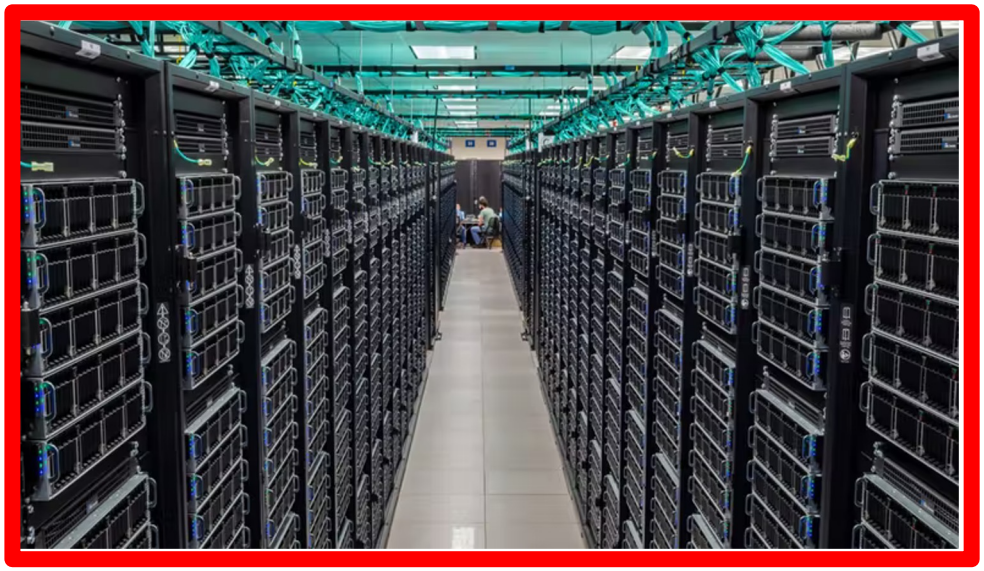
Jupyter notebook: Accessing shell (4)



Resources on TACC Machines

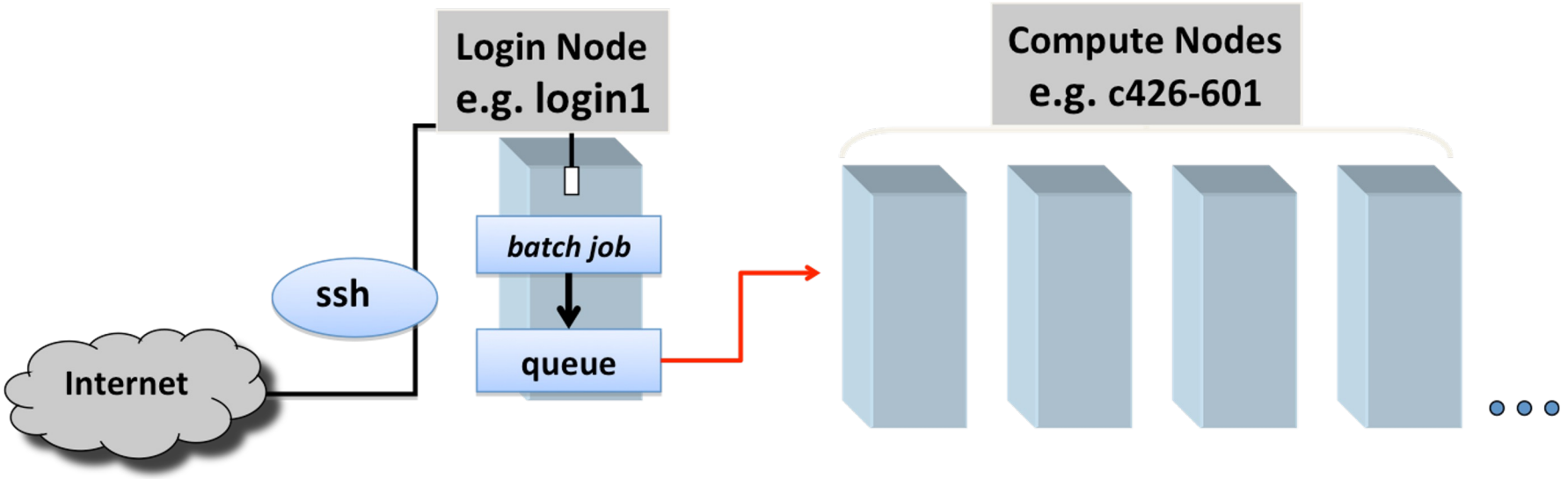
Personal Laptop

Remote Access:
ssh
Analysis Portal



- Login versus Compute nodes
- File systems

Login vs Compute Nodes



Login Nodes

- Shared by multiple users
- Not meant for CPU or memory intensive jobs
- Meant for managing file and compiling code

Compute Nodes

- Where research computation occurs
- Used for sbatch or interactive session to access compute nodes

Accessing Compute Nodes

Ways to access compute nodes

- submit a batch job via the sbatch command
  - job waits in queue until resources are available
- activate an interactive session via **idev**
  - idev
    - Allows you to get a compute node for testing purposes
    - -m option lets you specify time in minutes
      - idev –m 30

Partitions

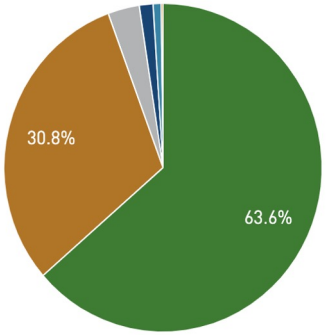
- Compute nodes are divided into different pools called queues
  - For example here are a few queues on Frontera
    - development (interactive sessions)
    - small (one or two node jobs)
    - normal (>=3 node jobs)
    - rtx (gpu)
    - rtx-dev (interactive sessions with gpus)
    - nvdimm (large memory nodes)
  - Refer to the user guide for the queue and accounting details

File systems

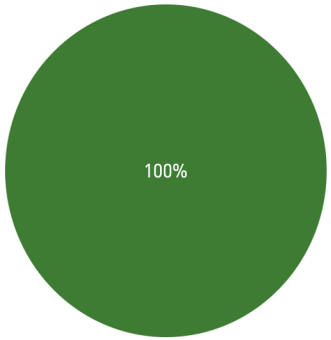
- Three (3) main LUSTRE file systems
  - $HOME
    - Quota (25G)
    - Backed up
    - Used for: small scripts, environment settings, other routine file management task
  - $WORK/$WORK2
    - Quota (1TB)
    - Not Backed up
    - Used For: Intermediate staging/preparation of files
      - Move files in/out of $SCRATCH
    - Part of one large Global file system, available on all TACC systems
  - $SCRATCH
    - No Quota
    - Not Backed up, subject to purge
    - Used for: High I/O bandwidth jobs
- Refer to Frontera User Guide
  - https://frontera-portal.tacc.utexas.edu/user-guide/files/

Linux/Shell

- TACC machines use the Linux operating system (OS)
  - Linux is dominant OS in HPC
  - Rocky Linux/CentOS
- The Shell provides an interface between the user and the linux operating system



June 2005



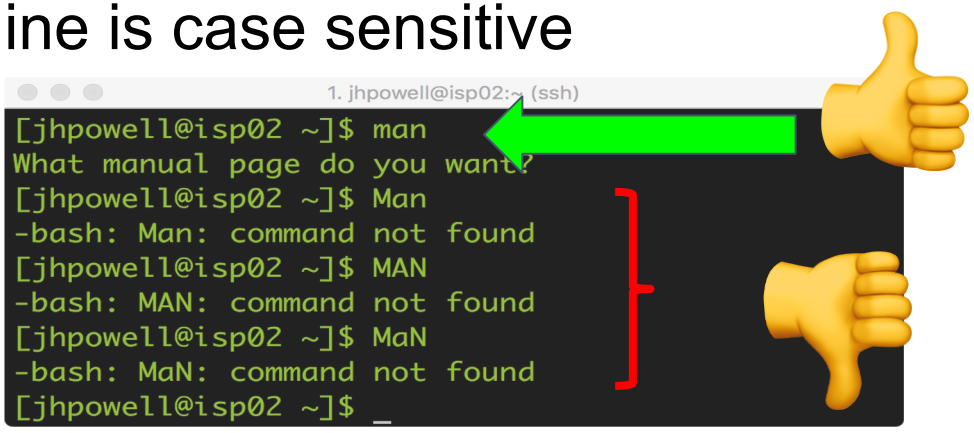
June 2019

bash

- Bourne Again Shell (bash)
  - Most common shell type on Linux systems
  - Combines features from several other older shell types
    - C shell (csh)
    - Bourne shell
    - Korn shell (ksh)
- Auto completion of commands
- Bash command line is case sensitive

Example:
The "**man**" command

man ≠ Man ≠ MAN



Files and File Tree

/ - the root or base directory

/bin - non-essential binaries (applications)

/sbin - binaries essential to the system

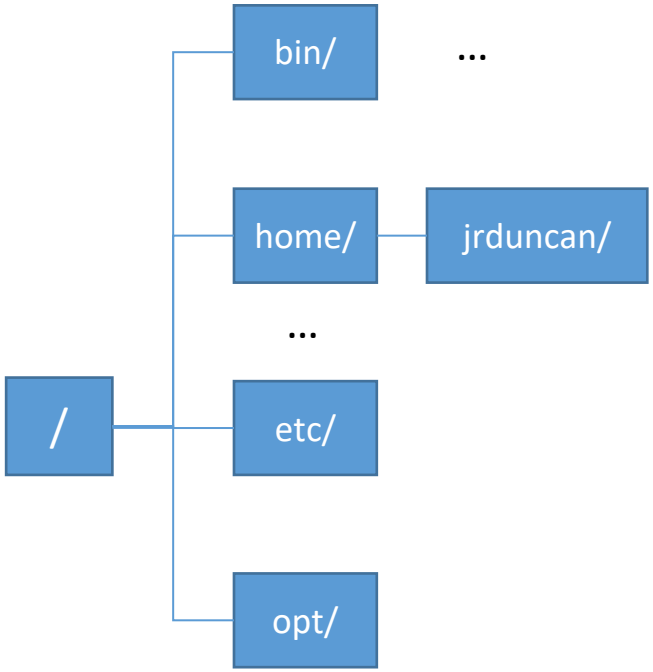
/dev - contains all folders for devices

/etc - contains all configuration files

/home - user specific folder

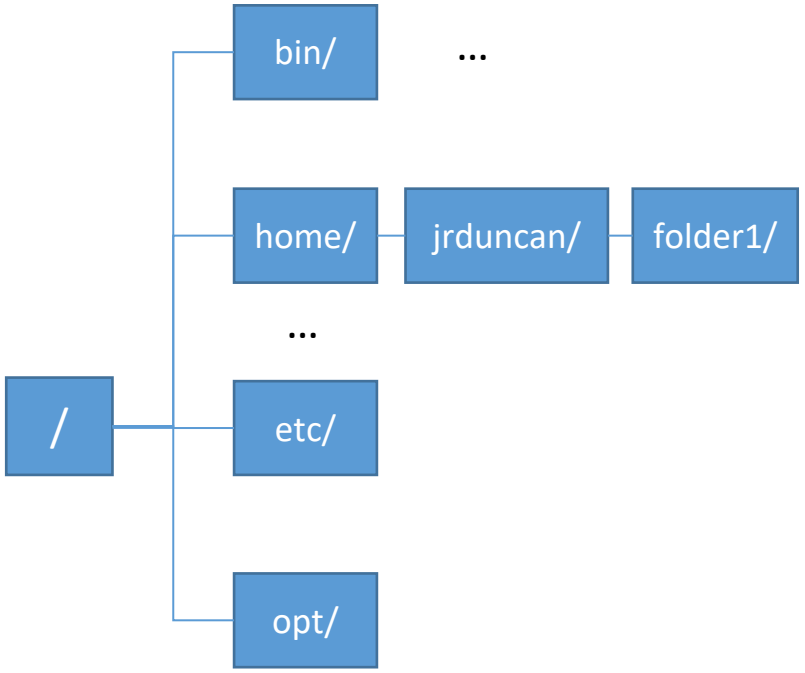
/opt - holds software add-on packages

/var - holds spooling data



Files and File Tree

- Absolute Path
  - Path relative to top level root “/”
  - “/home/jrduncan/folder1”
- Relative Path
  - Path relative to current working directory
    - i.e where you are working now
  - “./folder1”
- “.” references current folder
- “..” references parent folder



Bash – basic commands

`command -options arg1 arg2 …`

For help: `command -h`

Command	Function

`man <command name>`
examples: `man man`
`man cp`
	Shows documentation for the command from (usually) authors of the tool
	* Type “q” to get out of man page

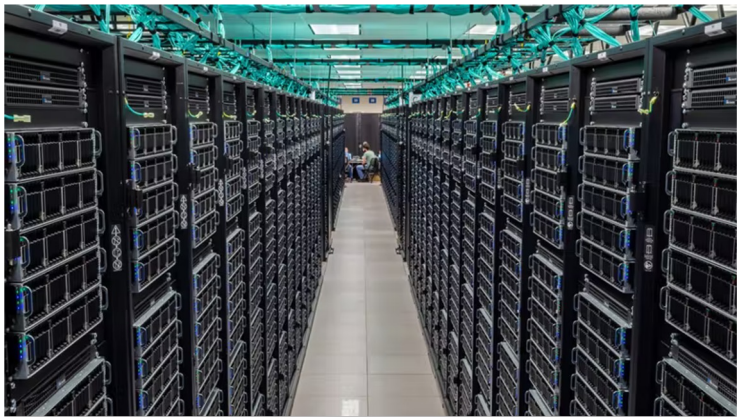
`hostname`	Shows hostname of machine
`whoami`	Shows your user name
`pwd`	Shows absolute path of your present working directory
`ls <directory name>` examples: `ls -al ./`	Shows contents of directory (if mentioned) or present working directory

Bash – basic commands (2)

Command	Function
`mkdir <name>`	Creates a new directory with mentioned name
`cd <name>` example: `cd ~`	Changes working directory to name mentioned
`touch <file name>`	Creates new file if not present
`cp <File1> <File2>`	Makes a copy of File2 and names it File2
`cp –r <directory1> <directory2>`	Makes a recursive copy of directory1 and names it directory2
`mv <File1> <File2>`	Moves File1 to location given by File2

File Transfer Mechanisms

- Copying files from your local machine to TACC system
- Mac/OSX or Linux
  - Open Terminal window
  - Use scp (Secure Shell Copy)
    - Copies over ssh connection
    - -r option for recursive



```
scp <local file name> user@frontera.tacc.utexas.edu:
```

```
scp  -r <local folder name> user@frontera.tacc.utexas.edu:
```

File Transfer Mechanisms (2)

- Copying files from your local machine to TACC system
- Windows
  - Download WinSCP (https://winscp.net)
  - Double click installer
  - Choose “Explorer” Option
  - Hostname: `frontera.tacc.utexas.edu`
  - Username: `username`
  - TACC Code: <Use Code from your credentials handout>
- Drag and Drop Files/Folders to copy to Frontera

Command Line Editing

- Often will be required
  - Working with code
  - Editing configuration files
  - Writing Bash Scripts
    - TACC Systems: Writing job submission scripts
- Vim
  - Vi Improved
- Emacs
- Nano
  - Simple, Easy to use

module

- On TACC Systems the module system helps setup or teardown useful packages that are supported by TACC, on the fly
- To find more information on a module
  - `module spider package-name`
- To load a module
  - `module load package-name`
  - `module load package-name/version.number`
- Documentation
  - https://lmod.readthedocs.io/en/latest/010_user.html

Module commands

- Additional Commands
  - `module unload package-name`
  - `module swap package-name-1 package-name-2`
  - `module list`
  - `module save`
    - `module save collection-name`
  - `module describe collection-name`
  - `module savelist`
  - `module restore`
    - `module restore collection-name`

Exercise

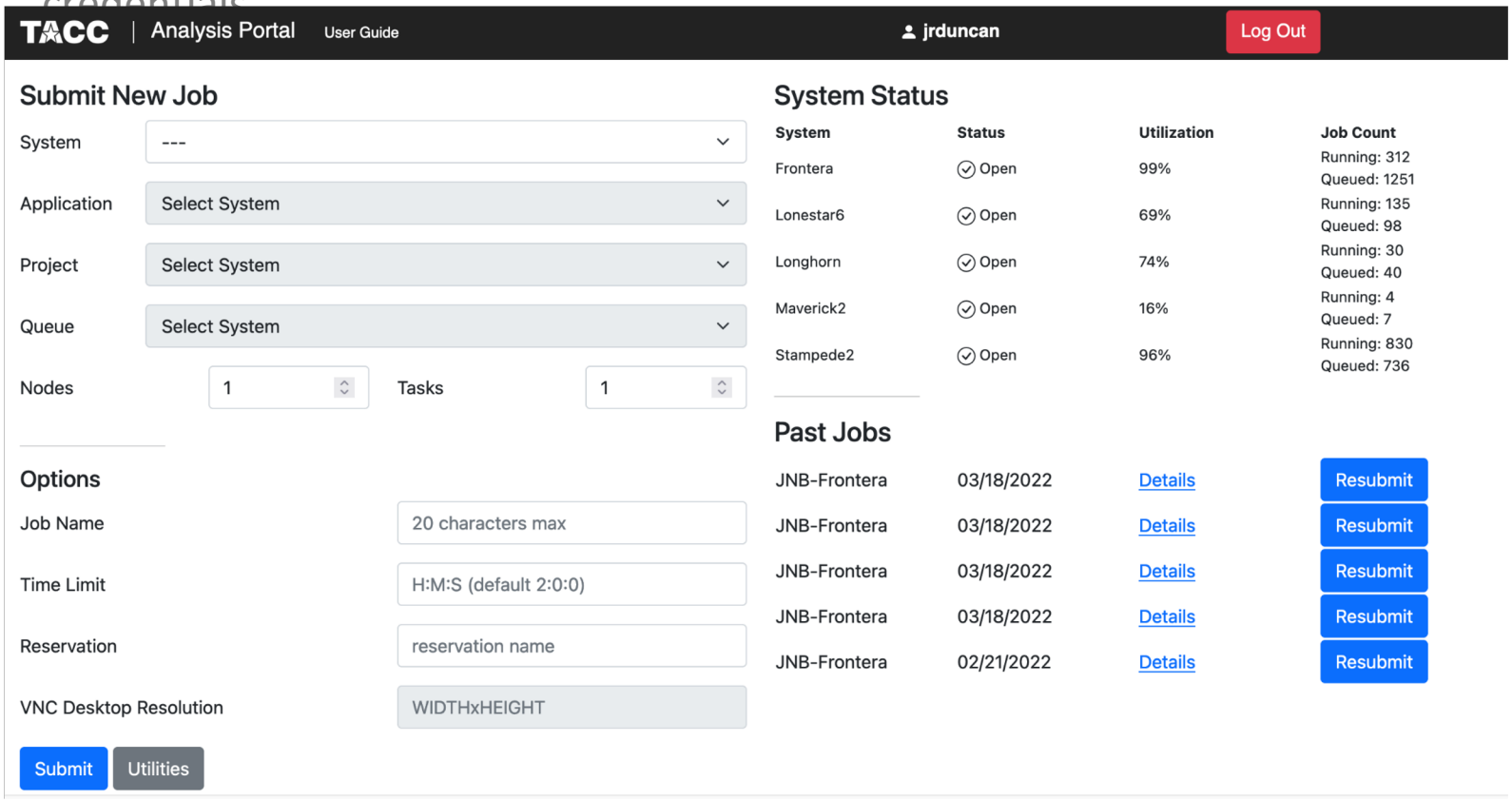
In this exercise you will:

- access a shell via a jupyter notebook
- copy files needed for next python introduction session

Jupyter notebook: Accessing shell

- TACC Visualization Portal:

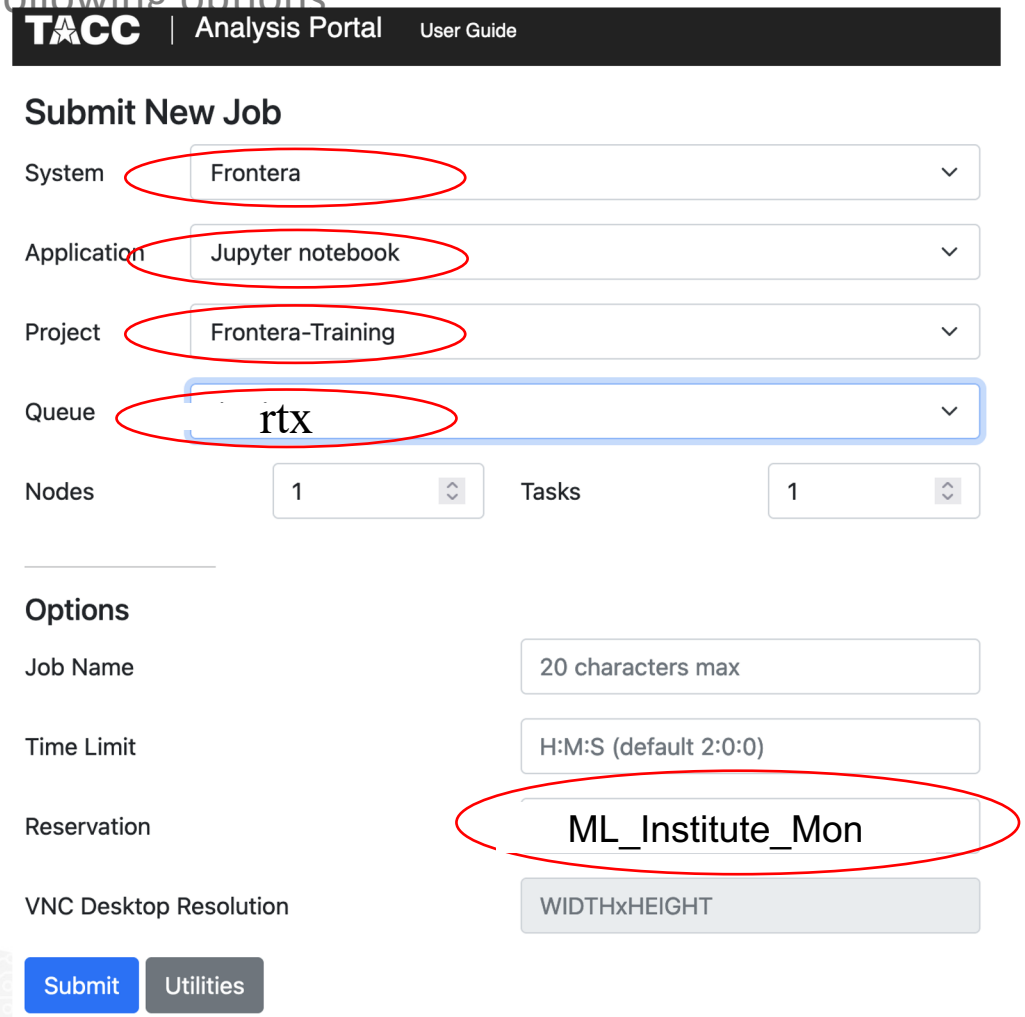
  Go to https://tap.tacc.utexas.edu; Login with your training account credentials



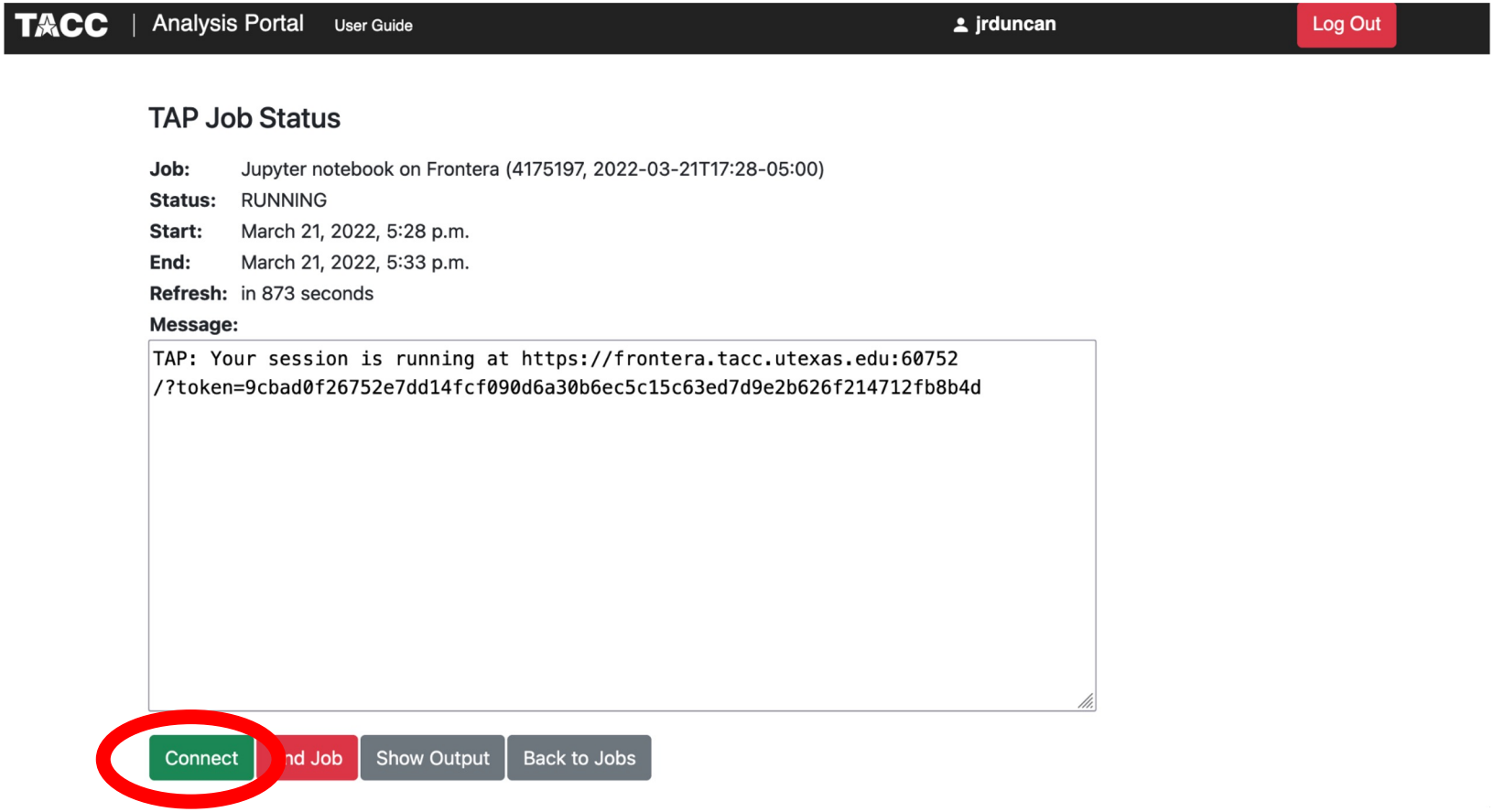
Jupyter notebook: Accessing shell

- TACC Visualization Portal:

  Select the following options

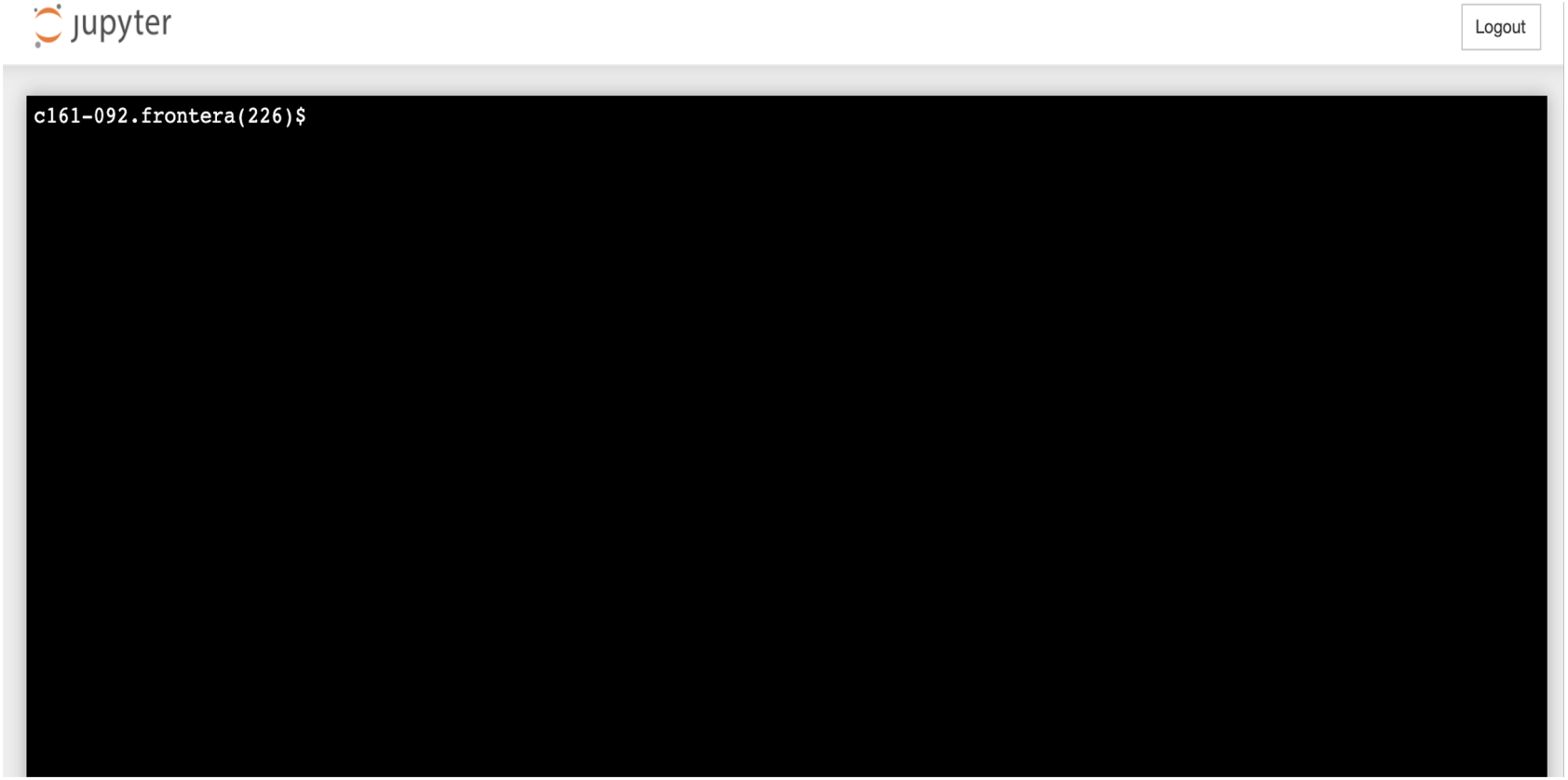


Jupyter notebook: Accessing shell



Jupyter notebook: Accessing shell (3)

Jupyter notebook: Accessing shell (4)



Bash commands

If you are new to linux, try out some of the following commands:

pwd

ls

cd

Setup Environment

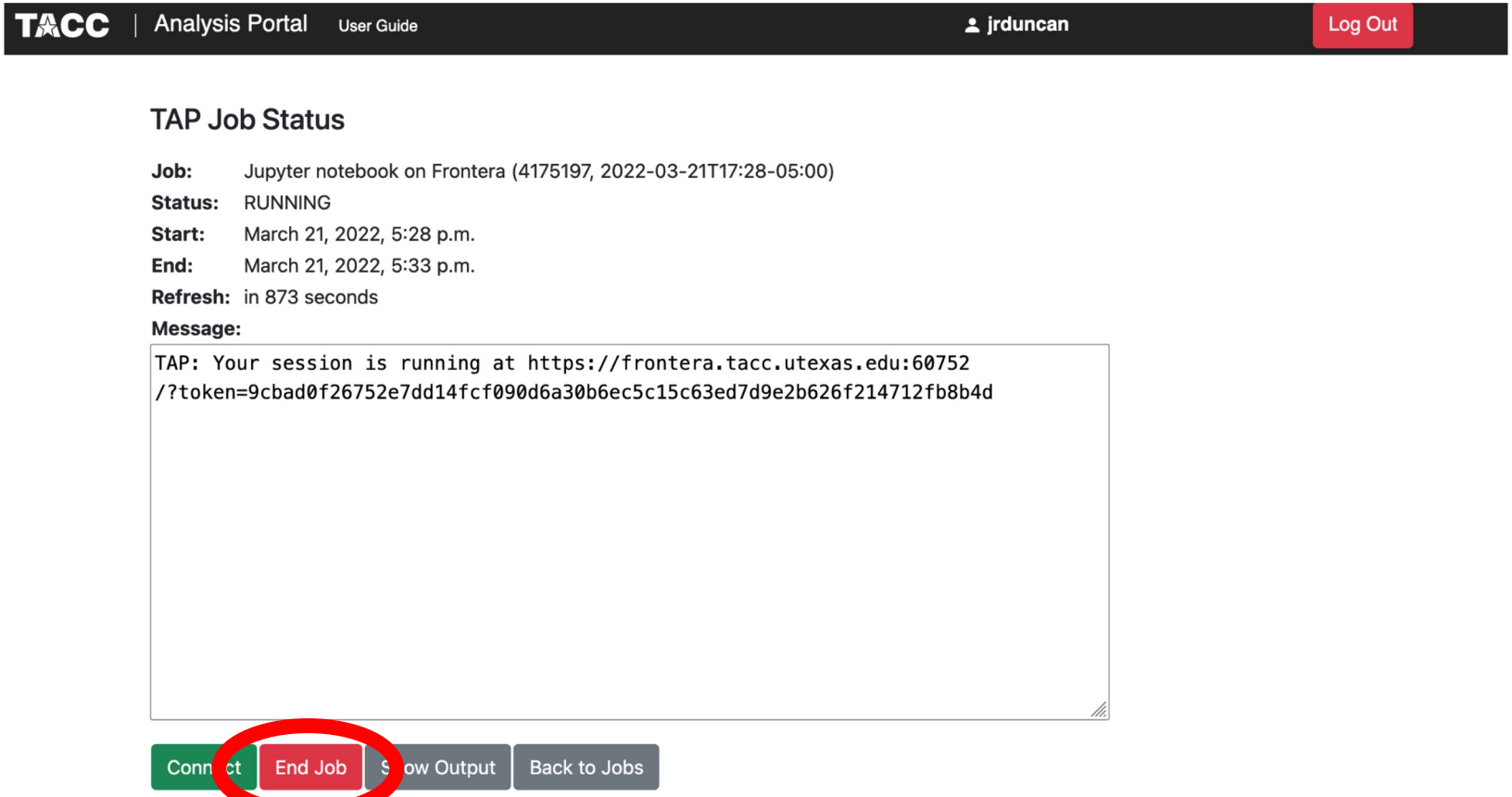
Run the following command to setup your environment and copy materials for lectures

/scratch1/01596/jrduncan/ml_institute_setup/install

- Installs the container with the right Python and libraries for the training days
- Copies code for lectures into your home folder
  - **ml_institute_summer_24**

- Close your jupyter notebook session and relaunch

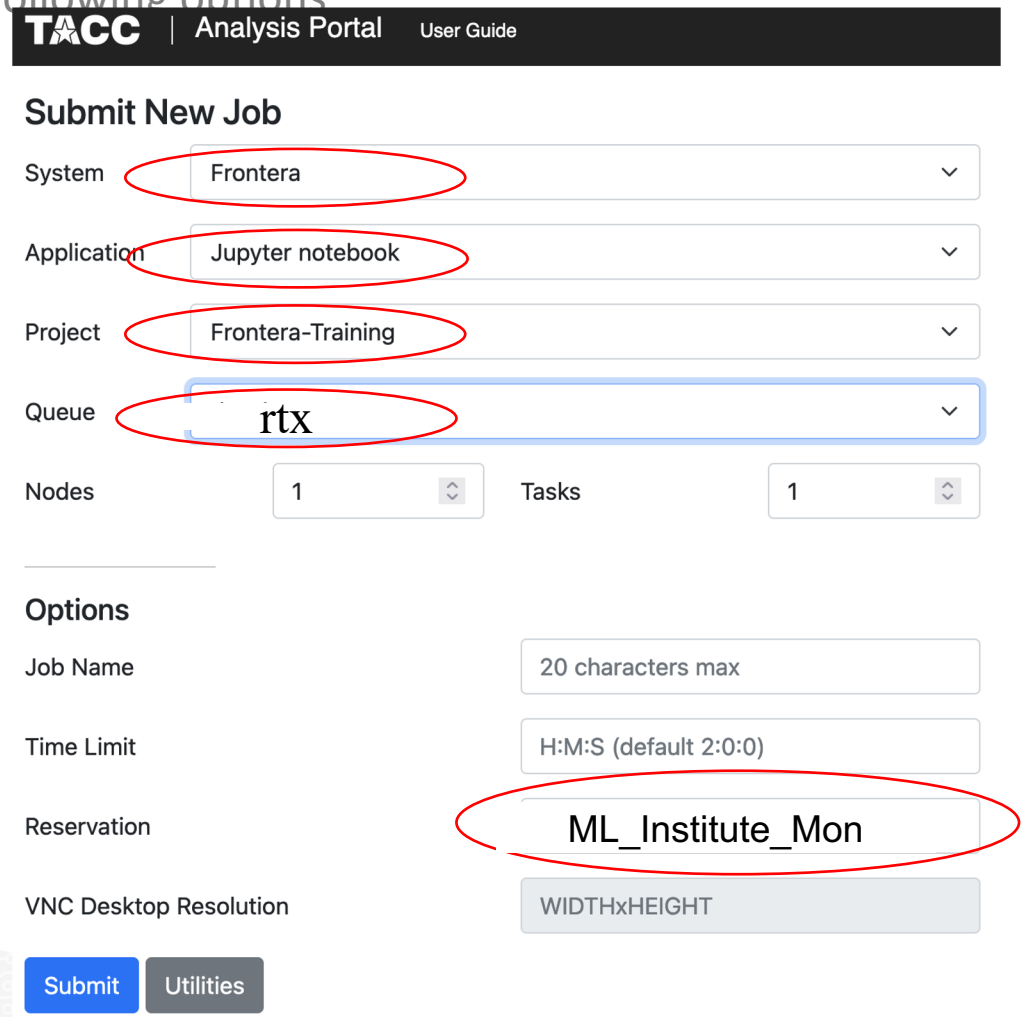
Jupyter notebook: Accessing shell



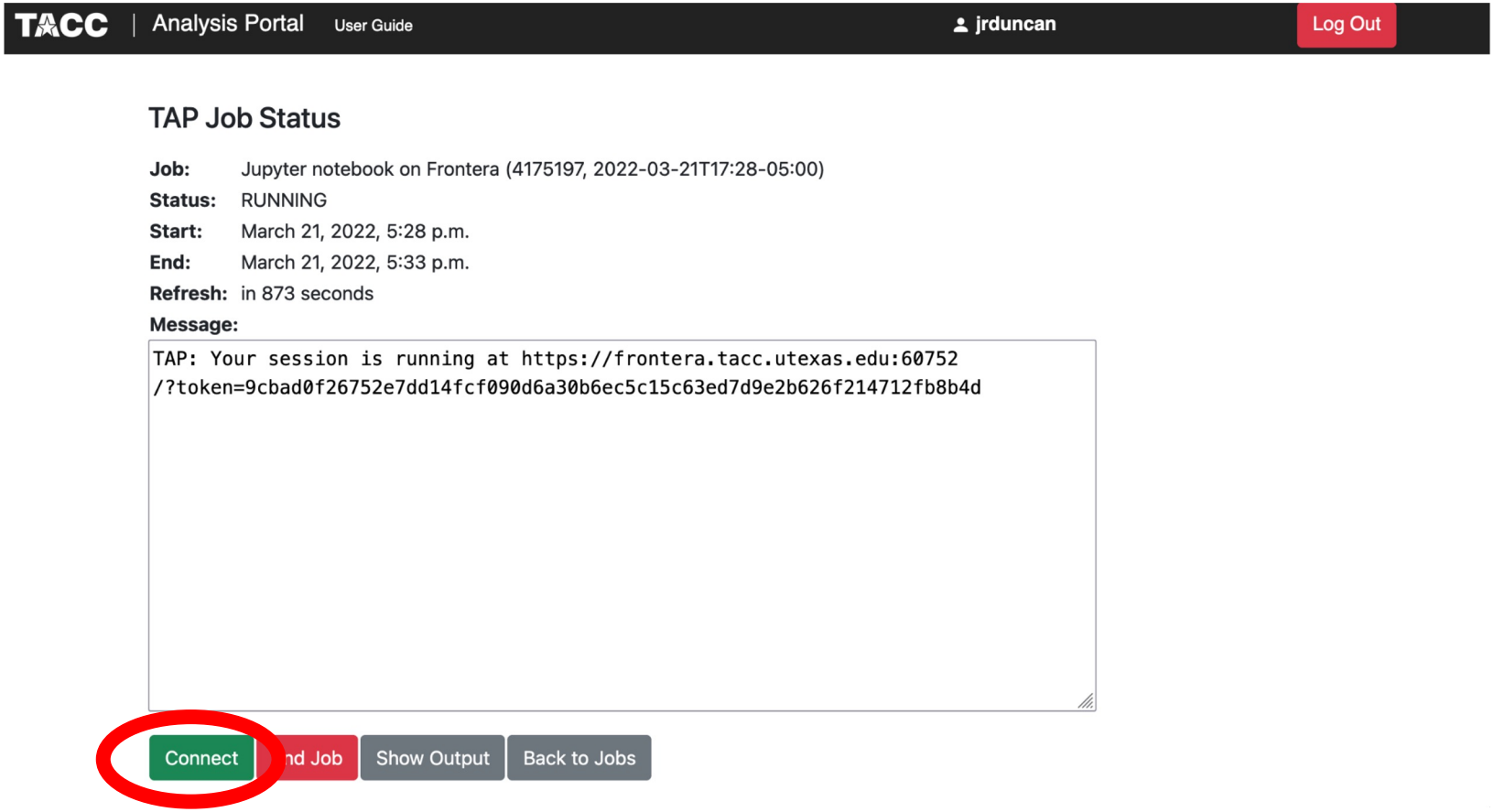
Jupyter notebook: Accessing shell

- TACC Visualization Portal:

  Select the following options



Jupyter notebook: Accessing shell



Jupyter Notebook: Open Existing File

- Click on the **day1** folder
- Click on the file **Python Introduction.ipynb**
- A jupyter notebook should launch with the material for the next lecture

Thanks

- Questions
- Contact: agupta@tacc.utexas.edu
- Any Issues on TACC Systems
  - Open a ticket a **http://consult.tacc.utexas.edu**