

High Performance Research Computing

A Resource for Research and Discovery



TEXAS A&M
UNIVERSITY.

HPRC Short Course

Running Jupyter Notebooks on the Open On Demand Portal



DIVISION OF RESEARCH
TEXAS A&M UNIVERSITY

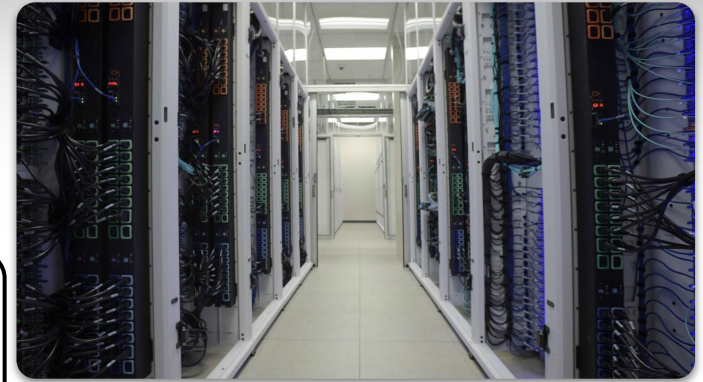


HPRC's Newest Cluster

Grace is a 925-node Intel cluster from Dell with an InfiniBand HDR-100 interconnect, A100 GPUs, RTX 6000 GPUs and T4 GPUs. There are 925 nodes based on the Intel Cascade Lake processor.

Grace Status: Testing and Early user onboarding

Grace
3TB Large Memory-80 cores/nodes
Other Login Nodes-48 cores/node



Login Nodes	5
384GB memory general compute nodes	800
GPU - A100 nodes with 384GB memory	100
GPU - RTX 6000 nodes with 384GB memory	9
GPU - T4 nodes with 384GB memory	8
3TB Large Memory	8

Available late Spring 2021

For more information:
<https://hprc.tamu.edu/wiki/Grace:Intro>

HPRC Portal

- HPRC Portal is an open source web platform through which users can access HPC clusters and services with a web browser
- Both ada and terra portal can be accessed through the landing page:
<https://portal.hprc.tamu.edu/>
- Key services provided:
 - Job submission and monitoring
 - File transfer and management
 - File editing
 - Shell access
 - Interactive applications

High Performance Research Computing

A Resource for Research and Discovery



TEXAS A&M
UNIVERSITY

TAMU HPRC OnDemand Homepage



[Ada OnDemand Portal](#)



[Terra OnDemand Portal](#)

Accessing the HPRC Portal

Access

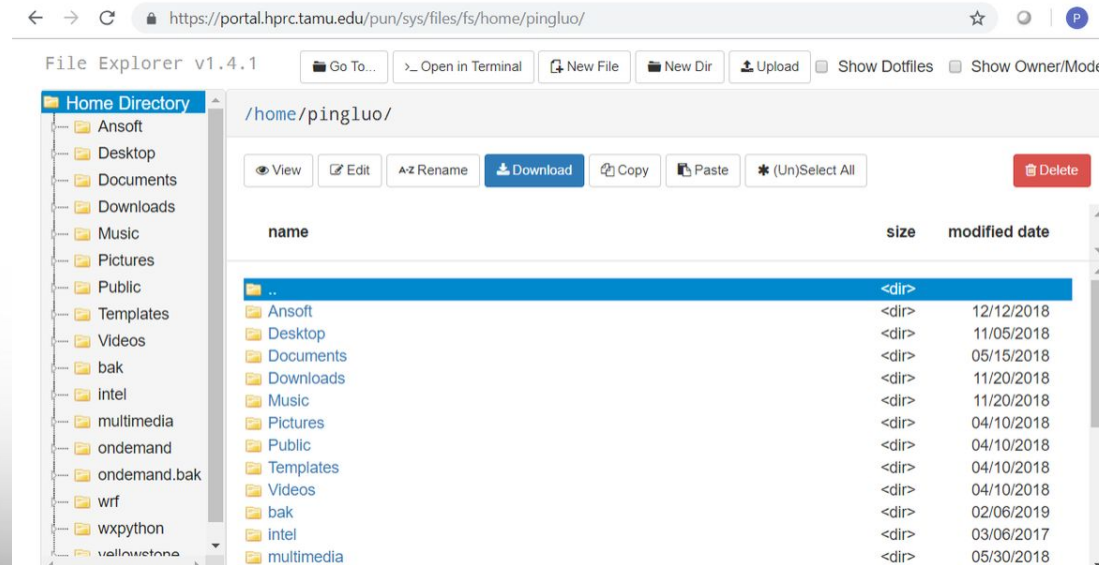
- On campus: <https://portal.hprc.tamu.edu/>
 - Off campus:
 - Set up and start VPN (Virtual Private Network): u.tamu.edu/VPnetwork
 - Then access the link: <https://portal.hprc.tamu.edu/>
 - *Two-Factor Authentication* enabled
- OnDemand user guide helps to navigate the easy-to-use interface and access the interactive applications:
<https://hprc.tamu.edu/wiki/SW:Portal>

hprc.tamu.edu/wiki/HPRC:Access

Navigating through the Portal

File Explorer

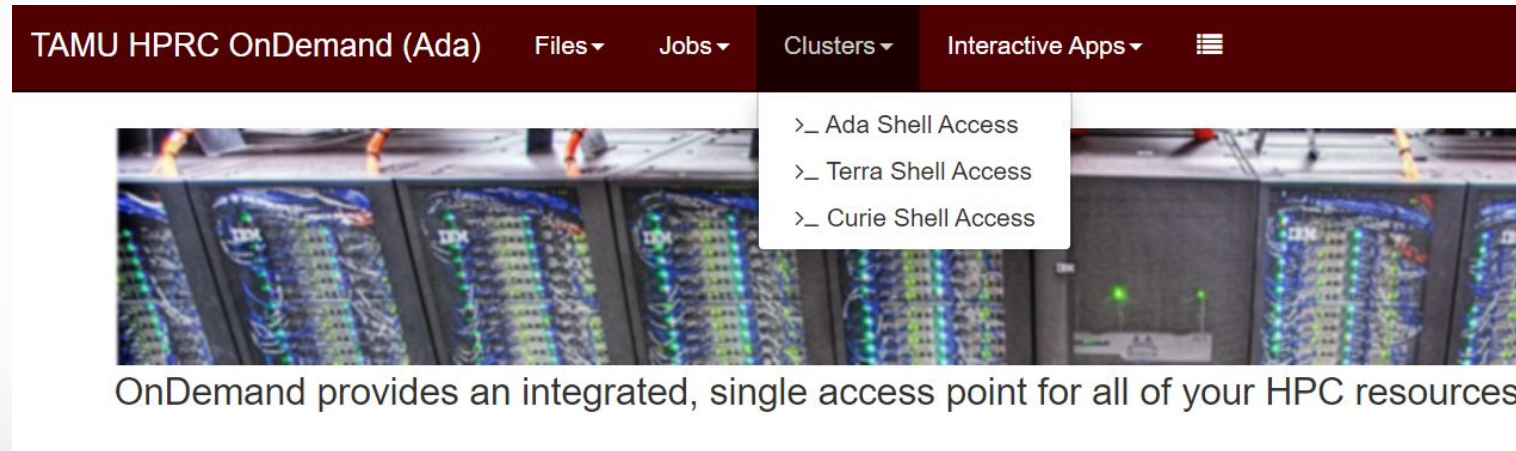
- Using the Files visual interface, a user can view a file explorer at either their home directory or scratch directory
- Files on portal can be used for viewing, editing and creating new files as well as directories on clusters



Navigating through the Portal

Cluster Shell

- Shell access to any of the three clusters is available from this drop down menu with one click
- Similar to ssh client such as Putty and MobaXterm using Netid and password



TAMU HPRC OnDemand (Ada) Files ▾ Jobs ▾ Clusters ▾ Interactive Apps ▾ ☰

- >_ Ada Shell Access
- >_ Terra Shell Access
- >_ Curie Shell Access

OnDemand provides an integrated, single access point for all of your HPC resources.

Navigating through the Portal

Jobs

- **Active Jobs:** Provides information regarding jobs running on the cluster, JobID, name, user, account, time used, queue, and status
- **Job Composure:** Provides template job scripts to create new jobs
Refer the wiki link for more details on job submission: https://hprc.tamu.edu/wiki/Ada:Batch_Processing_LSF

The screenshot displays the HPRC portal interface. At the top left, there is a '+ New Job' button. To its right is a 'Create Template' button. Below these are several action buttons: 'Edit Files', 'Job Options', 'Open Terminal', 'Submit', 'Stop', and 'Delete'. A search bar is located on the right side of the job list area.

The main job list is a table with the following columns: Created, Name, ID, Cluster, and Status. It contains three entries:

Created	Name	ID	Cluster	Status
March 16, 2020 11:02pm	serial	9550589	Ada	Completed
March 16, 2020 11:00pm	serial	9550588	Ada	Completed
February 10, 2020 1:28pm	serial			Not Submitted

At the bottom of the list, it says 'Showing 1 to 3 of 3 entries' and includes 'Previous', '1', and 'Next' navigation buttons.

On the right side, the 'Job Details' panel is visible. It includes fields for 'Job Name' (serial), 'Submit to' (Ada), 'Account' (Not specified), 'Script location' (/scratch/user/stutitrviedi1373/ondemand/data/sys/myjobs/projects/default/9), and 'Script name' (02-parallel.sh).

Navigating through the Portal

Interactive Apps

- Most common GUI software like MATLAB, Abaqus etc. along with servers like Jupyter Notebooks, RStudio can be directly launched by providing mentioned parameters.
- Required job parameters:
 - number of cores
 - wall time
 - memory
 - type of node
- If a software is not available, you can always run it within VNC

To load ABAQUS using VNC:

```
module load ABAQUS  
vglrun abaqus cae
```

- For more details on running VNC, refer:
<https://hprc.tamu.edu/wiki/Ada:Remote-Viz>

Launch Jupyter Notebook on HPRC Portal

Jupyter

- Jupyter interactive app on portal will launch Jupyter Notebook server on clusters
- Jupyter Notebook environment can be created using Python or Anaconda(module version specified) on HPRC Portal
- Note: Run the command “**showquota**” to check if available file limit >10,000 as conda and pip creates thousands of files
- Jupyter Notebook session data will be saved in user’s home directory by default. This can be modified by creating virtual environment under user’s scratch directory
- To launch Jupyter Notebooks, go to **Interactive Apps-> Server->Jupyter Notebooks**

Launch Jupyter Notebook on HPRC Portal

Jupyter

- Choose **Module**(Python/Anaconda version)
- Provide **number of hours** for running notebooks on clusters
- Specify **number of cores** [1-28] allocated on node on clusters
- Requested **total memory** (2-112GB)
 - If total mem from all cores is $\leq 54\text{G}$, the job will run on a 64G memory node
 - If total mem from all cores is $> 54\text{G}$, the job will run on a 128G memory node
- Provide **node type** (GPU/Any)
- Specify account details and email to receive a pop-up when session is ready to launch

Launch Jupyter Notebook on HPRC Portal

Optional Environment Activation-Python

- Existing virtual environments created on clusters can be used to launch Jupyter Notebooks. Alternately, Python or Anaconda module can be used to new create virtual environment
- For creating virtual environment using Python, create new directory under scratch

```
mkdir -p /scratch/user/mynetid/pip_envs
```
- Load Python module(select from the list of module available on portal). Create virtual environment named *my_notebook-python-3.6.6-foss-2018b*

```
module purge virtualenv
/scratch/user/mynetid/pip_envs/my_notebook-python-3.6.6-foss-2018b
```

Launch Jupyter Notebook on HPRC Portal

Optional Environment Activation-Python

- Activate virtual environment

Install notebook and python packages

```
source /scratch/user/myetid/pip_envs/my_notebook-python-3.6.6-foss-2018b/bin/activate  
pip install notebook  
pip install python_package_name
```

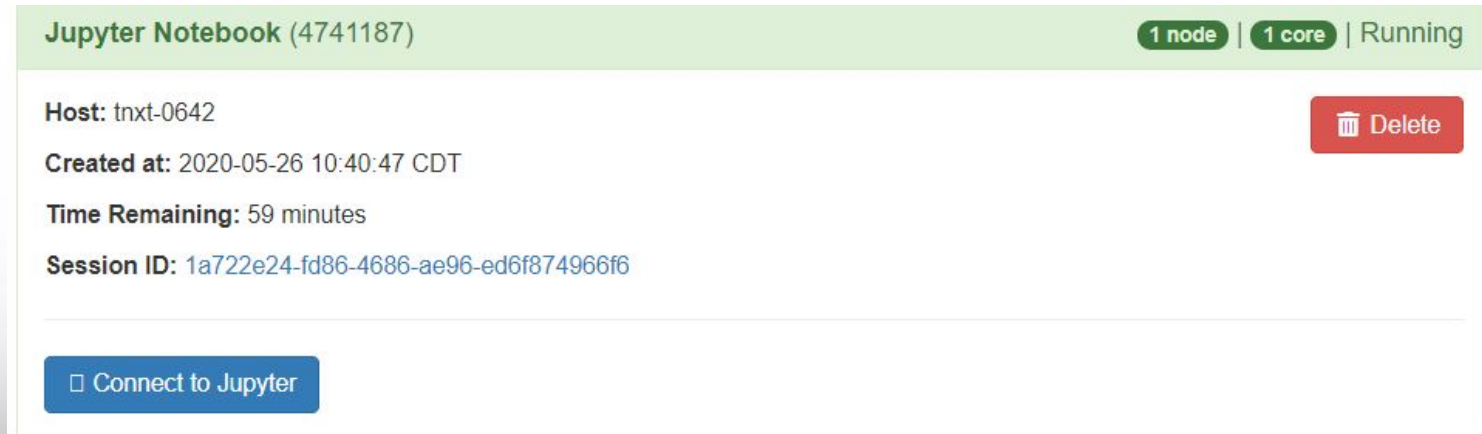
- Refer wiki link for more information:

https://hprc.tamu.edu/wiki/SW:Portal#Jupyter_Notebook

Launch Jupyter Notebook on HPRC Portal

Optional Environment Activation-Python

- Provide full path to to the activate command for your Python/3.6.6-foss-2018b environment in the "Optional Conda Environment to be activated"
- Jupyter notebook session with JobID (4741187 in this case) and session ID is active.
- Session ID can be used to trace logs



The screenshot shows a Jupyter Notebook session interface. At the top, a green header bar displays "Jupyter Notebook (4741187)" on the left and "1 node | 1 core | Running" on the right. Below the header, the session details are listed: "Host: tnx-0642", "Created at: 2020-05-26 10:40:47 CDT", "Time Remaining: 59 minutes", and "Session ID: 1a722e24-fd86-4686-ae96-ed6f874966f6". A red "Delete" button with a trash icon is located to the right of the session details. At the bottom left, there is a blue button labeled "Connect to Jupyter".

Launch Jupyter Notebook on HPRC Portal

Optional Environment Activation-Anaconda

- For creating virtual environment using Anaconda, create new directory under scratch and virtual environment my_notebook

```
module purge
module load Anaconda/3-5.0.0.1
conda create -n my_notebook
```

- Activate/Deactivate environment using the command mentioned below:
my_notebook-python-3.6.6-foss-2018b

```
Activate: source activate my_notebook
Deactivate: source deactivate
```

Launch Jupyter Notebook on HPRC Portal

Optional Environment Activation-Anaconda

- Activate virtual environment. Install notebook and conda packages

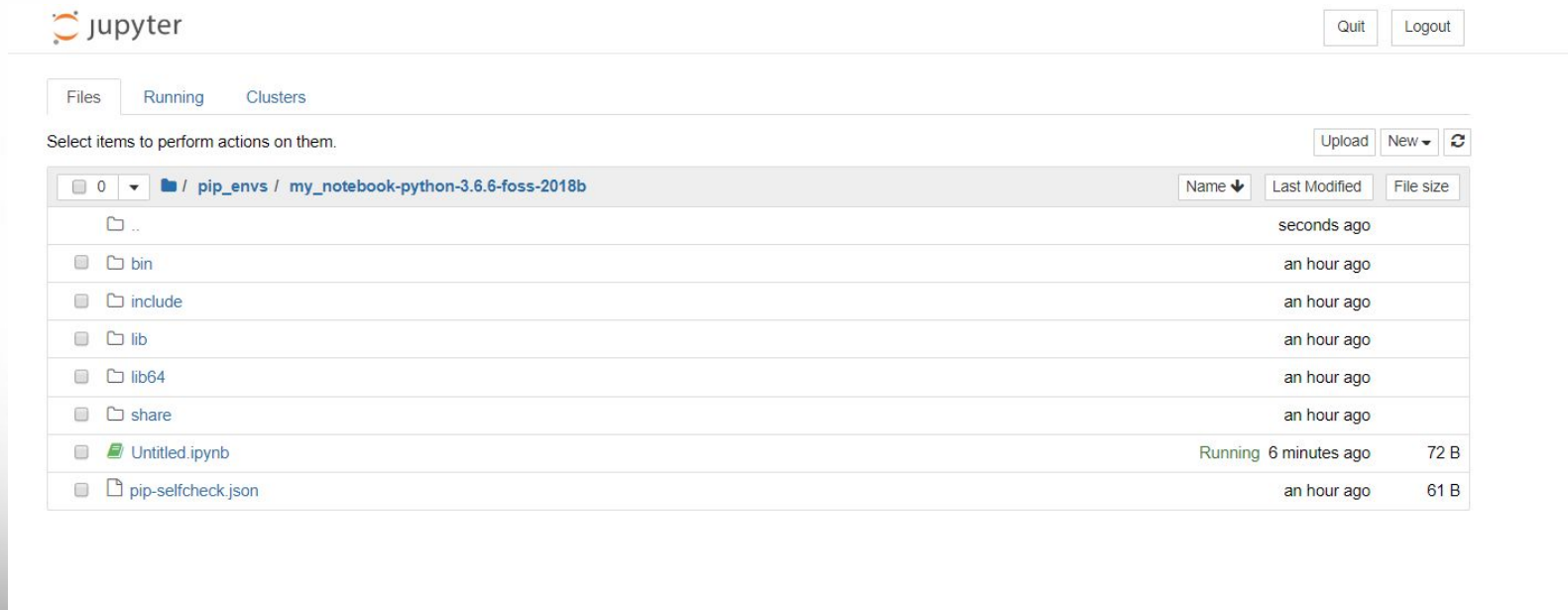
```
source activate my_notebook  
conda install -c conda-forge notebook  
conda install -c conda-forge package-name
```

- Refer wiki link for more information: https://hprc.tamu.edu/wiki/SW:Portal#Jupyter_Notebook

Using Jupyter Notebook on HPRC Portal

Using Jupyter Notebooks

- Default directory-user's home or directory pointing to virtual environment (optional)
- Create new file using New-> Python3
- If you are uploading existing .ipynb file, make sure the python version is similar



The screenshot displays the Jupyter Notebook interface. At the top left is the Jupyter logo. On the top right are 'Quit' and 'Logout' buttons. Below the logo are tabs for 'Files', 'Running', and 'Clusters'. A message says 'Select items to perform actions on them.' To the right of this message are 'Upload', 'New', and a refresh icon. The main area shows a file browser for the directory `pip_envs / my_notebook-python-3.6.6-foss-2018b`. The browser contains a table of files and folders:

	Name	Last Modified	File size
<input type="checkbox"/>	..	seconds ago	
<input type="checkbox"/>	bin	an hour ago	
<input type="checkbox"/>	include	an hour ago	
<input type="checkbox"/>	lib	an hour ago	
<input type="checkbox"/>	lib64	an hour ago	
<input type="checkbox"/>	share	an hour ago	
<input type="checkbox"/>	Untitled.ipynb	Running 6 minutes ago	72 B
<input type="checkbox"/>	pip-selfcheck.json	an hour ago	61 B

Using Jupyter Notebook on HPRC Portal

Using Jupyter Notebook-Basic Python

Exercise 1

- Create a new directory under pip_envs
- Create a new notebook inside it. Save the file.
- Print a message- Welcome to HPRC

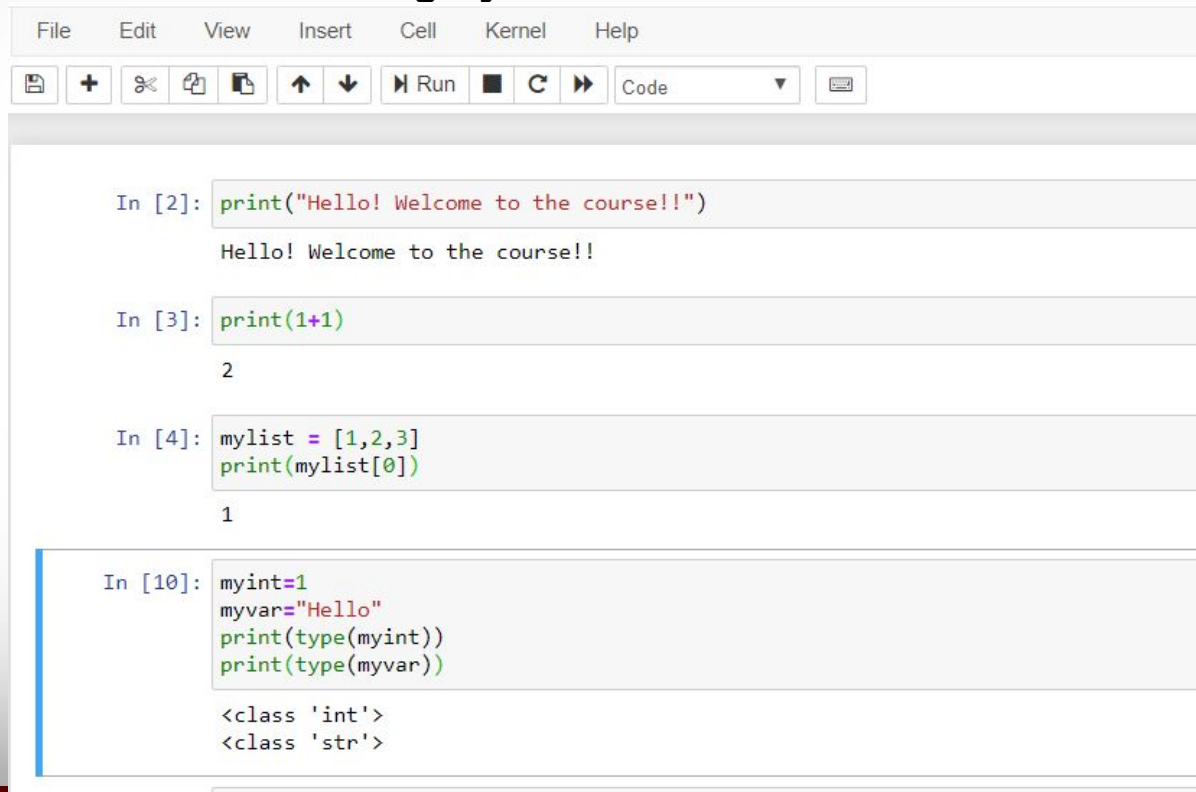
Using Jupyter Notebook on HPRC Portal

Jupyter Notebook- Familiarize the setup

- Jupyter server is up and running. Create notebook
New->Notebook-> Python
- Upload notebook(Python version must be same)
Upload->Select notebook
- File naming convention
All lowercase module names. Long module names can have words separated by underscores (really_long_module_name.py). CamelCase for class names
- Running the cells (**Shortcut: Ctrl + Enter**)

Using Jupyter Notebook on HPRC Portal

Jupyter Notebook- Running Python



The screenshot shows a Jupyter Notebook interface with a menu bar (File, Edit, View, Insert, Cell, Kernel, Help) and a toolbar with icons for file operations, navigation, and execution. Below the toolbar are four code cells, each with its input and output:

```
In [2]: print("Hello! Welcome to the course!!")
Hello! Welcome to the course!!
```

```
In [3]: print(1+1)
2
```

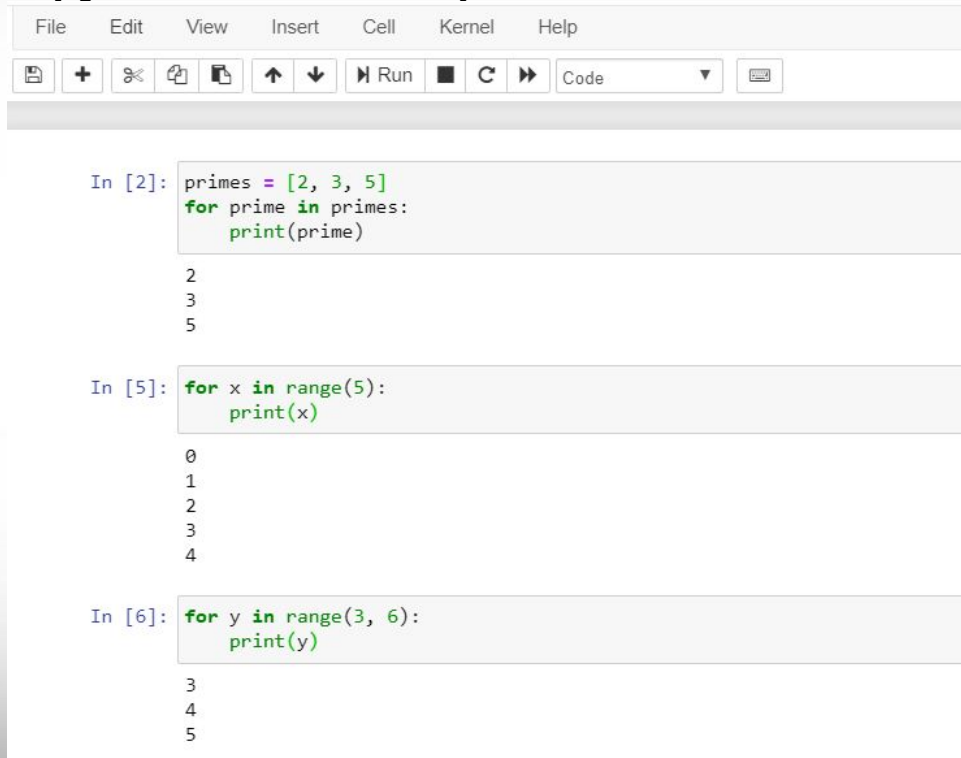
```
In [4]: mylist = [1,2,3]
print(mylist[0])
1
```

```
In [10]: myint=1
myvar="Hello"
print(type(myint))
print(type(myvar))

<class 'int'>
<class 'str'>
```

Using Jupyter Notebook on HPRC Portal

Using Jupyter Notebook-Loops



The screenshot shows a Jupyter Notebook interface with a menu bar (File, Edit, View, Insert, Cell, Kernel, Help) and a toolbar with icons for file operations, navigation, and execution. Below the toolbar are three code cells, each followed by its output.

```
In [2]: primes = [2, 3, 5]
        for prime in primes:
            print(prime)

2
3
5
```

```
In [5]: for x in range(5):
        print(x)

0
1
2
3
4
```

```
In [6]: for y in range(3, 6):
        print(y)

3
4
5
```

Using Jupyter Notebook on HPRC Portal

Jupyter Notebook- Familiarize the setup

- **Kernel Attributes**

- Restart

- Interrupt

- Change kernel (switch between various python versions)

- Shutdown

- List down all kernels available eg. Python 2, Anaconda 3

- Remove specific kernels

```
jupyter kernelspec list  
jupyter kernelspec remove <kernel-name>
```

Using Jupyter Notebook on HPRC Portal

Jupyter Notebook- iPython Shell Commands

- Print working directory- `!pwd`
- List files inside directory- `!ls`
- Change directory- `%cd newdir`
- Make directory- `%mkdir newdir`
- Copy file- `%cp filename.ipynb newdir/`
- Remove directory- `rm -r newdir`

Using Jupyter Notebook on HPRC Portal

Using Jupyter Notebook-Basic Python

Exercise 2

- Change the formatting as indicated below:
 - Bold
 - Italics
 - Header1 format
 - Include a link (<https://hprc.tamu.edu/>) next to the text

Using Jupyter Notebook on HPRC Portal

Jupyter Notebook- Markdown

- **Headers**

(Header 1, title) stands for html code `<h1>Header 1,title</h1>`

- **Line Break**

The line breaks after using `
` br tags and it is awesome

- **Inline Formatting**

Bold: ****HPRC**

Italics: **HPRC*

Horizontal Line: *******

- **Embed external link**

`Link to Google`

`[section title](#section-title)`

`[HPRC Link](https://hprc.tamu.edu/)`

Using Jupyter Notebook on HPRC Portal

Jupyter Notebook- Markdown

- Perform complex mathematical operations ('\$ math expression \$')

$$\sqrt{k}$$

```
In [ ]:  $\sqrt{k}$ 
```

- Embed Python Code

```
```Python  
str = "This is block level code"
print(str)
```
```

Using Jupyter Notebook on HPRC Portal

Using Jupyter Notebook-Introduction to Numpy

```
In [9]: # Create 2 new Lists height and weight
height = [1.87, 1.87, 1.82, 1.91, 1.90, 1.85]
weight = [81.65, 97.52, 95.25, 92.98, 86.18, 88.45]

# Import the numpy package as np
import numpy as np

# Create 2 numpy arrays from height and weight
np_height = np.array(height)
np_weight = np.array(weight)

# Calculate bmi
bmi = np_weight / np_height ** 2

# Print the result
print(bmi)
# For a boolean response
bmi > 25

# Print only those observations above 23
bmi[bmi > 25]

[23.34925219 27.88755755 28.75558507 25.48723993 23.87257618 25.84368152]
```

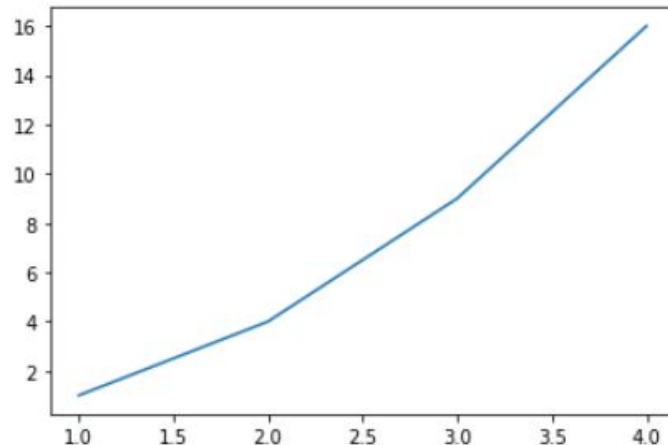
Out[9]: array([27.88755755, 28.75558507, 25.48723993, 25.84368152])

Using Jupyter Notebook on HPRC Portal

Using Jupyter Notebook-Introduction to Matplot

```
In [5]: %matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
plt.plot([1, 2, 3, 4], [1, 4, 9, 16])
```

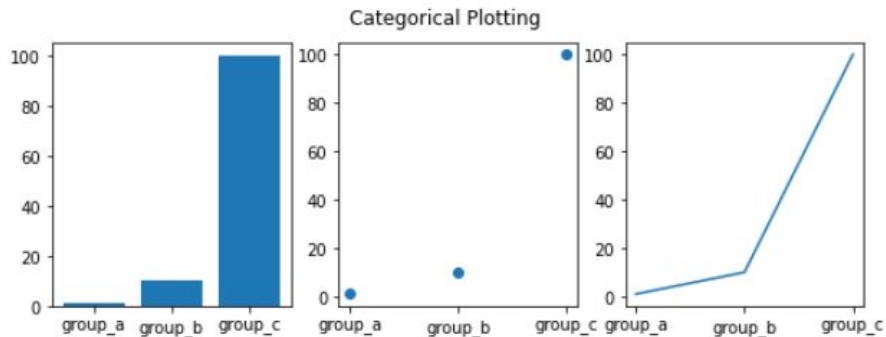
Out[5]: [



Using Jupyter Notebook on HPRC Portal

Using Jupyter Notebook-Introduction to Matplot

```
In [6]: names = ['group_a', 'group_b', 'group_c']  
values = [1, 10, 100]  
  
plt.figure(figsize=(9, 3))  
  
plt.subplot(131)  
plt.bar(names, values)  
plt.subplot(132)  
plt.scatter(names, values)  
plt.subplot(133)  
plt.plot(names, values)  
plt.suptitle('Categorical Plotting')  
plt.show()
```



Using Jupyter Notebook on HPRC Portal

Final Steps

- **Logging Out:**

To properly log out the portal, follow the below mentioned steps:

- log out the portal by clicking 'Log out' from the top navigation bar
- close the browser to completely terminate the session

- **Clean up:**

The portal stores temporary files for interactive apps in *\$SCRATCH/ondemand/data/sys/dashboard*.

- Use the below mentioned command on cluster to clean up after completing the simulations on Jupyter Notebooks.

```
rm -rf $SCRATCH/ondemand/data/sys/dashboard/batch_connect/sys/*
```

- **Saving the files:**

Notebooks will be saved in the default directory (home or virtual environment)

Continued Learning

[Intro to HPRC Video Tutorial Series](#)

[HPRC's Wiki Page](#)