



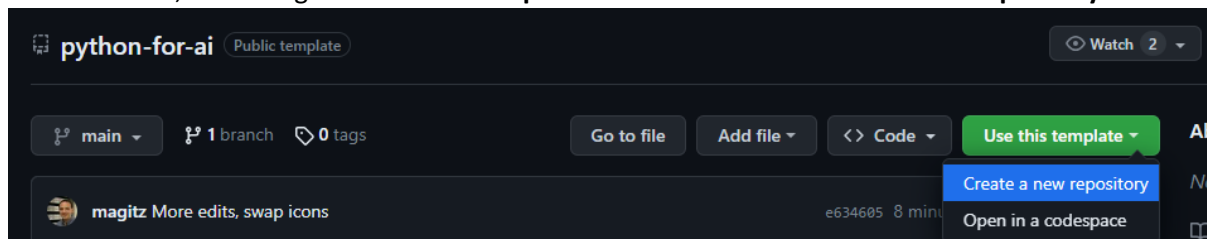
## Python for AI Jupyter Notebooks

### Step 1: Create your repository from the template

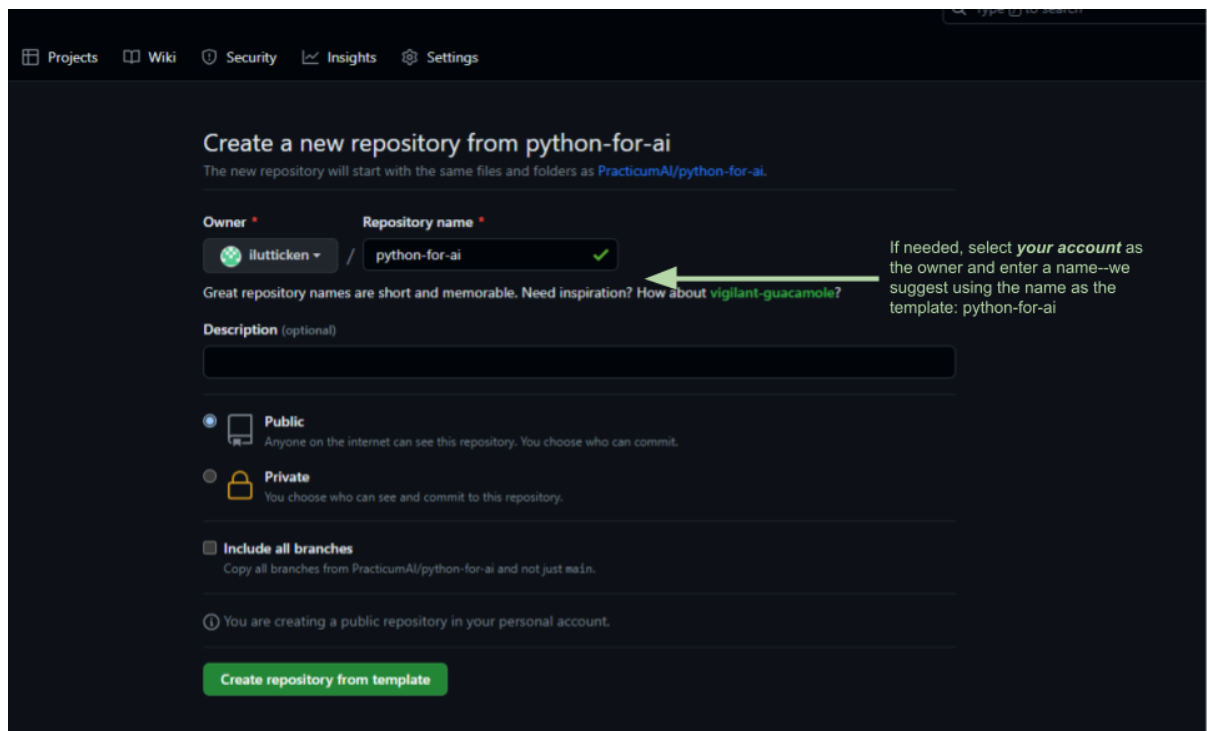
Most of the *Practicum AI* exercises will start with you using one of our template repositories to create your own repository. This allows you to get the latest version of our exercises and put a copy in your own space where you can work on the exercises using git and GitHub.

**Note:** You will need a GitHub.com account for this. You can create your account at <https://github.com/>. Click the **Sign up** button.

- The template for this module is at: <https://github.com/PracticumAI/python-for-ai>
- From that site, click the green **Use this template** button and select **Create a new repository**.



- On the new window, you may need to select the owner. **Select your GitHub account**. Then type a name for the new repository. We suggest using the same name as the template: **python-for-ai** in this case.



- Click the **Create repository from template** button.



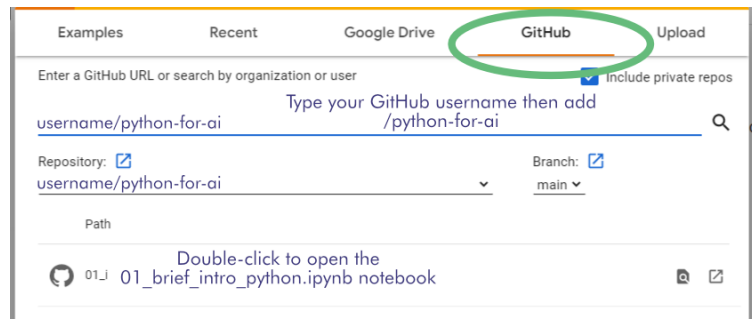
- This will now create your own copy of the repository. This is your own playground for you to work in. GitHub will show where you generated the repository from so you can always get back to the source, but this is yours to use.
- Now you are ready to setup a Jupyter session using one of the two options in step 2.

## Step 2:

### Option 1: Launch Notebooks in Google Colab

**Note:** University of Florida associated Google accounts may not work with Google Colab, you may need to use a personal Google account.

- Once you have a Google account, login at: <https://colab.research.google.com/>
- Now we are ready to open the notebook from the GitHub repository you created above.
- From the **File Menu** select **Open Notebook**.
- 
- Click on the GitHub tab and then type your GitHub username followed by the name of the repository: e.g. username/python-for-ai
- Then double-click on the **01\_brief\_intro\_python.ipynb** notebook to open it.
- Read through the notebook and complete the exercises.



### Option 2: Launch Jupyter on HiPerGator

Using HiPerGator requires that you have a HiPerGator account.

If you have an account, you will want to make sure you are part of a group that has access to at least 2 NCUs, 1 GPU, and 1 TB of Blue storage. Details of these resources are reviewed in the [HiPerGator New User Account training](#).

Launching a Jupyter Notebook session on HiPerGator involves specifying the resources that your session will use. As outlined in the account training, the HiPerGator scheduler needs information on four key elements:

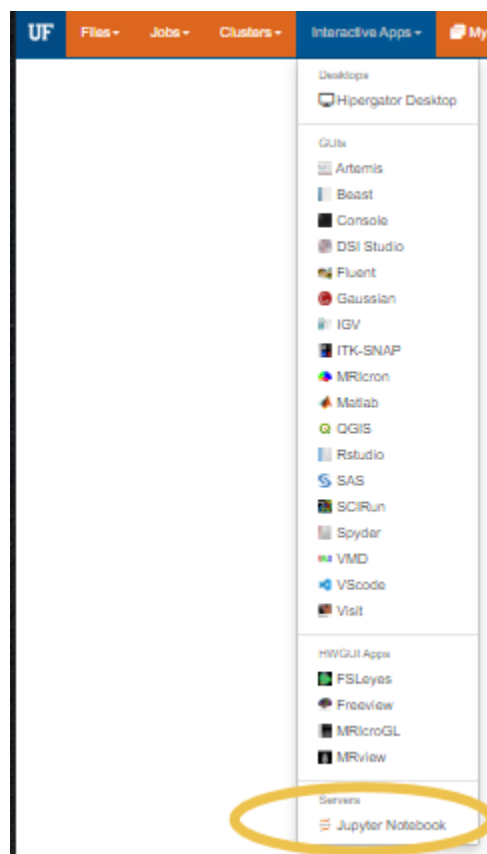
1. **How many CPUs** you want: For the Practicum AI trainings, 2 should be sufficient.
2. **How much memory** you want: For the Practicum AI trainings, 15GB should be sufficient.

3. **How many GPUs** you want: For the Practicum AI trainings, 1 should be sufficient.
4. **How long** you want the resources: For the Practicum AI trainings, 2 hours should be sufficient for most exercises.

We will use the web interface Open OnDemand (OOD) to launch and connect to Jupyter sessions. OOD provides an easy interface to provide the resource request, submit the job to the scheduler, and connect to the Jupyter server once it starts.

## Steps to launch a Jupyter Notebook session

- Once you have a HiPerGator account, login at: <https://ood.rc.ufl.edu/>
- Click on the **Interactive Apps** menu and scroll all the way to the bottom and select **Jupyter Notebook**
- The next screen allows you to request resources to run a job on HiPerGator.
- See the image for reference, but the suggested resource request for most of the *Practicum AI* courses is:
  - **Number of CPUs:** 1 or 2
  - **Maximum Memory:** 15GB
  - **SLURM Account and QOS:** Group name as needed.
  - **Time Requested:** 2
  - **Cluster Partition:** gpu
  - **Generic Resource Request:** gpu:a100:1





Number of CPU cores requested per MPI task (--cpus-per-task, -p)  
 Enter the number of CPU cores needed. One or two in most cases.

Number of cores per MPI task requested for application, (default = 1).

Maximum memory requested for this job in Gigabytes (--mem, -m)  
 Enter the amount of memory needed. 15GB in most cases.

Maximum amount of memory to be used by the job (blank/default = 600M per CPU). If you are using advanced memory options in **Additional SLURM Options** below, then leave this blank.

SLURM Account (--account, -A)  
 If needed, you can enter a secondary group name in the Account and QoS boxes.

Enter an alternative account if required. (default = primary investor)

QoS (Required if custom Account is set, --qos, -q)

Enter an alternative QoS, required if alternative account is entered above. Please note, if you use the burst qos (-b), your jobs may take longer to start. (default = primary investor)

Time Requested for this job in hours (--time, -t)  
 Enter the amount of time needed. Two hours in most cases.

Time in hours requested from SLURM for this job to run. (default = 1) Please note, interactive gpu partition jobs will be limited to 72 hours.

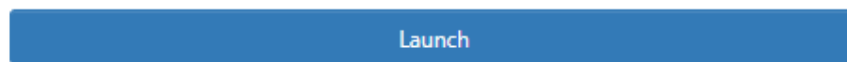
Cluster partition (--partition, -p)  
 Select gpu from the dropdown.

Select a specific cluster partition for job. (default = first available compute partition)

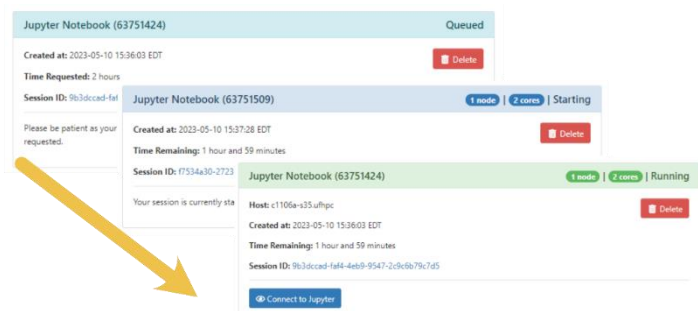
Generic Resource Request (--gres).  
 Enter the GPU resources needed. gpu:a100:1 in most cases.

This is the Generic resource request string to request GPU resources. See also [https://help.rc.ufl.edu/doc/GPU\\_Access](https://help.rc.ufl.edu/doc/GPU_Access)

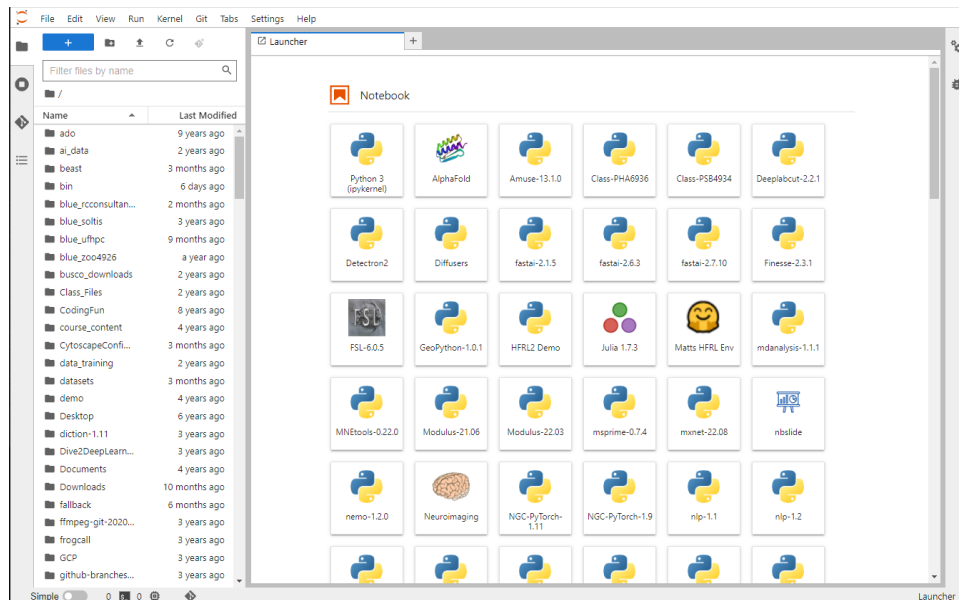
- After entering the information, click the Launch button



- On the next page, your job will have a card. The card will start with a light blue boarder with “Queued” in the upper right. That means your job has been submitted to the scheduler and is waiting for the scheduler to find resources (or for them to be available in your group’s allocation). Once the resources are available, the card will change to a dark blue boarder with “Starting” and finally, when you job is ready, it will have green boarder, say “Running” and have a button to click to “Connect to Jupyter”.

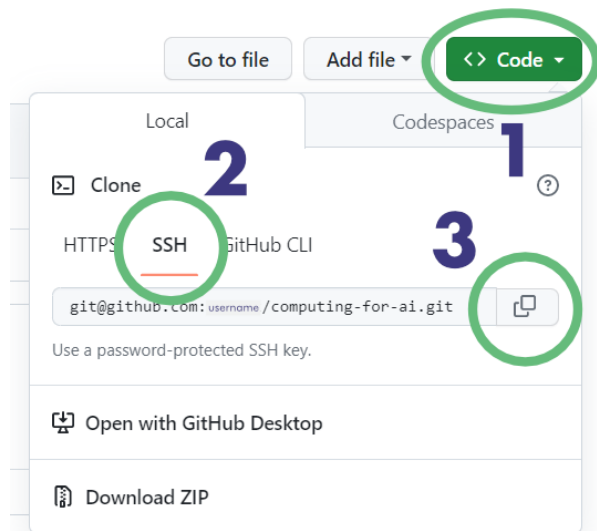


- Clicking the Connect to Jupyter button will open a new tab and should open with a window like what is pictured below—on future connections, it will normally open with the documents you had opened last time.



## Cloning your repository onto the HPC system

- In a different tab, go to your GitHub repository for this course. It should be at <https://github.com/> then your GitHub username, then the repository name you provided, e.g. `python-for-ai`.
- Click on the following three buttons:
  1. The Green “<> Code” button
  2. The SSH button
  3. The copy button to copy the URI (Universal Resource Identifier—a more general term than the more familiar URL which is just for links).





- Return to your Jupyter tab and click on the Git tab on the far left of the Jupyter screen (3<sup>rd</sup> one down), then Clone a Repository button.
- In the pop-up window, paste the URI you copied from your GitHub repository above.
- Click back on the File Browser tab (top tab on the left under the Jupyter logo). You should now have a folder with the name of your repository (python-for-ai). Double-click to open that folder.
- **Open the 01\_brief-intro-python.ipynb** notebook.
- Read through the notebook and complete the exercises.

