

# Step-by-Step Guide: Running Jupyter Notebooks Online using Openscapes 2I2C

## Useful links :

<https://openscapes.2i2c.cloud/>

<https://github.com/ghrcdaac/ghrc-playground>

<https://nasa-openscapes.github.io/earthdata-cloud-cookbook/>

Haven't filled the request form yet? Please fill out the form with your GitHub username information [JupyterHub request form](#) to access Openscapes 2I2C JupyterHub

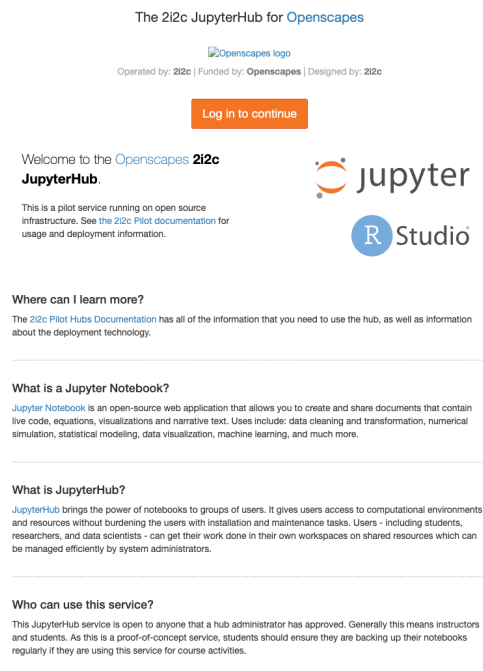
## Getting Started:

### Step 1: Logging into JupyterHub


Go to the Openscapes 2I2C page by using the URL:

<https://openscapes.2i2c.cloud/>

If this is your first time using this link, a login prompt will be shown as the page loads



The 2I2c JupyterHub for Openscapes


 Openscapes logo


Operated by: 2I2c | Funded by: Openscapes | Designed by: 2I2c

[Log in to continue](#)

Welcome to the **Openscapes 2I2c JupyterHub**.

This is a pilot service running on open source infrastructure. See [the 2I2c Pilot documentation](#) for usage and deployment information.

 **jupyter**

 **R Studio**

**Where can I learn more?**

The [2I2c Pilot Hubs Documentation](#) has all of the information that you need to use the hub, as well as information about the deployment technology.

---

**What is a Jupyter Notebook?**

**Jupyter Notebook** is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

---

**What is JupyterHub?**

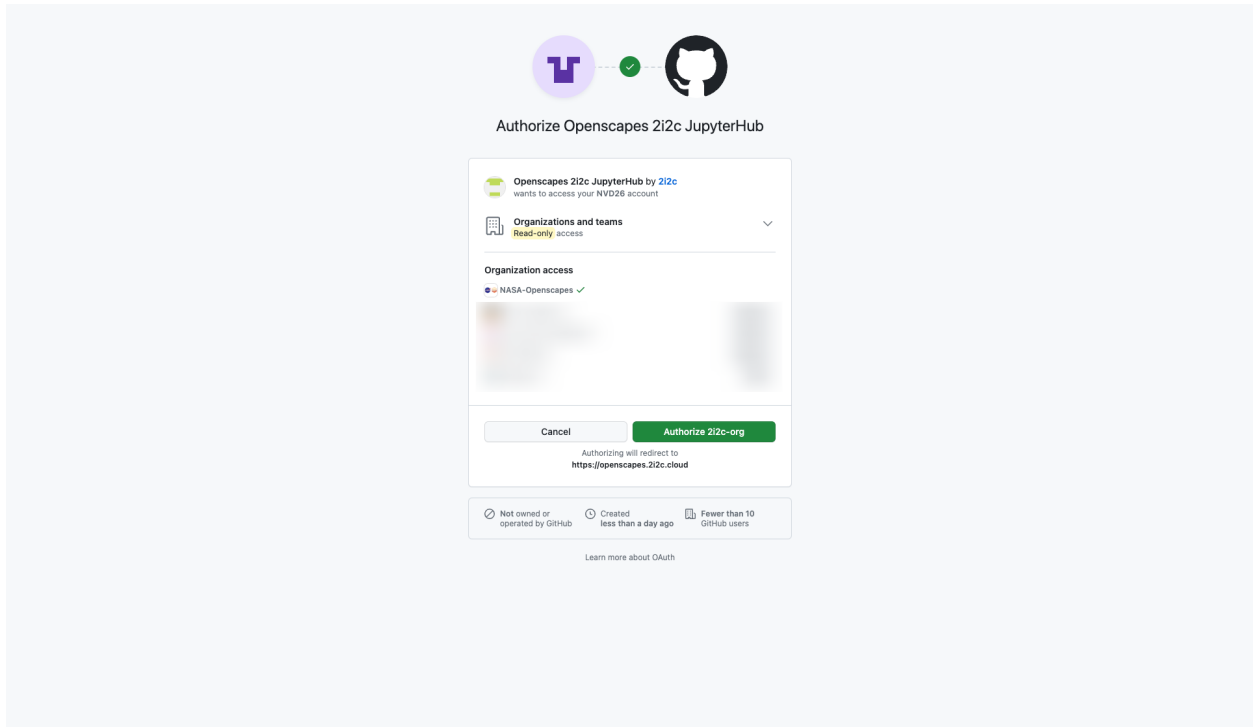
**JupyterHub** brings the power of notebooks to groups of users. It gives users access to computational environments and resources without burdening the users with installation and maintenance tasks. Users - including students, researchers, and data scientists - can get their work done in their own workspaces on shared resources which can be managed efficiently by system administrators.

---

**Who can use this service?**

This JupyterHub service is open to anyone that a hub administrator has approved. Generally this means instructors and students. As this is a proof-of-concept service, students should ensure they are backing up their notebooks regularly if they are using this service for course activities.

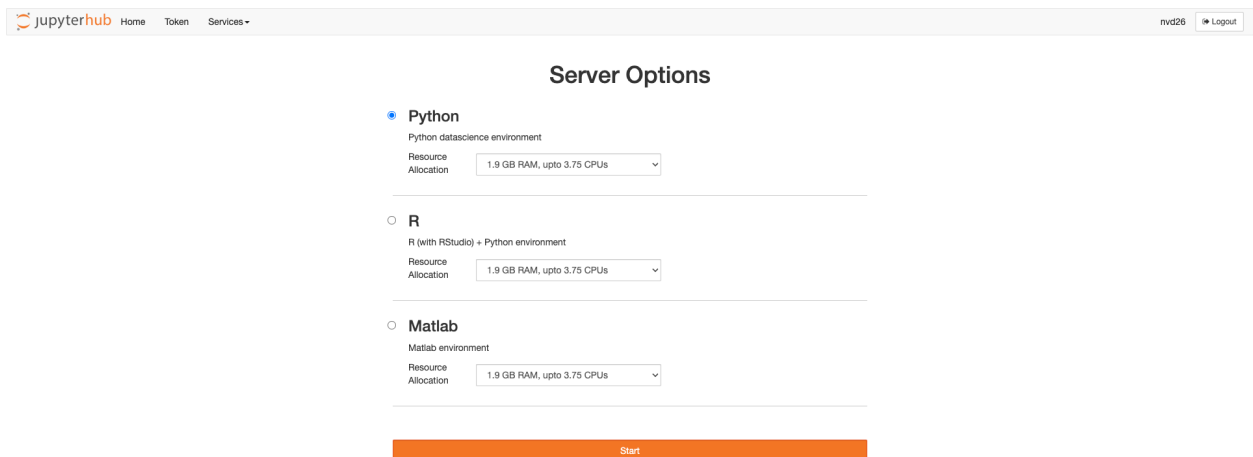
Click “Log in to continue”. You will be prompted to a GitHub authorization page



Click “Authorize 2i2c-org” button to allow OpenScapes 2i2C to authorize your GitHub user information (for login)

## Step 2: Selecting instance type and resource allocation

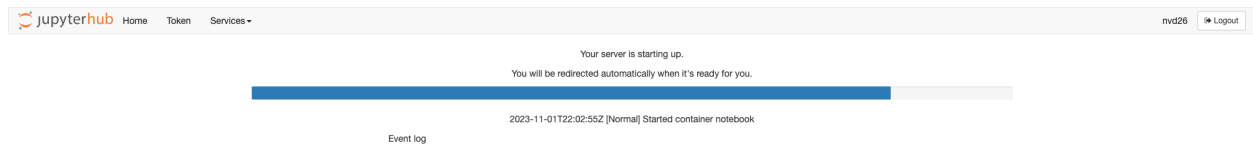
After a successful login, You will be redirected to choosing the server instance and resource allocation



For our demo, we'll be using "Python" environment. Most of our notebooks run on default resources, for memory/CPU intensive programs, allocate more resources.

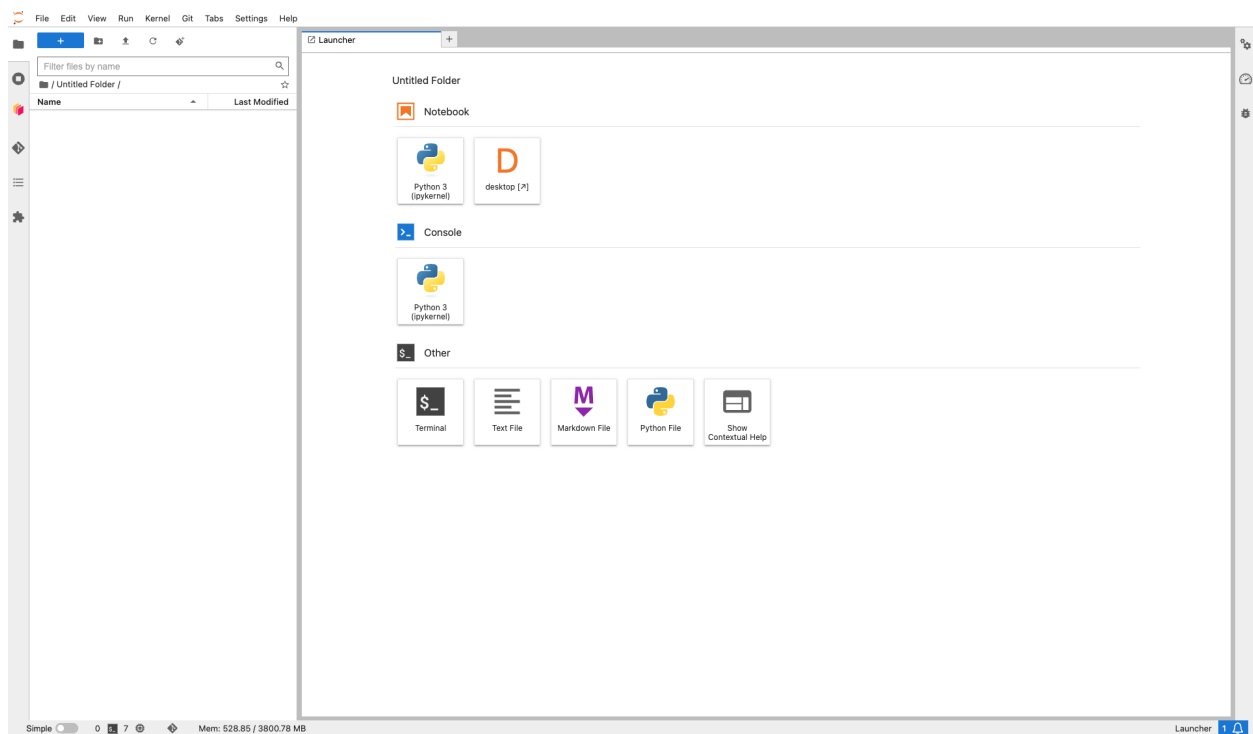
Note: One of our notebooks use "3.7GB RAM, upto 3.75 CPUs" (Option 2 on the list).

Once you made the selection, Click start to proceed to the next step.



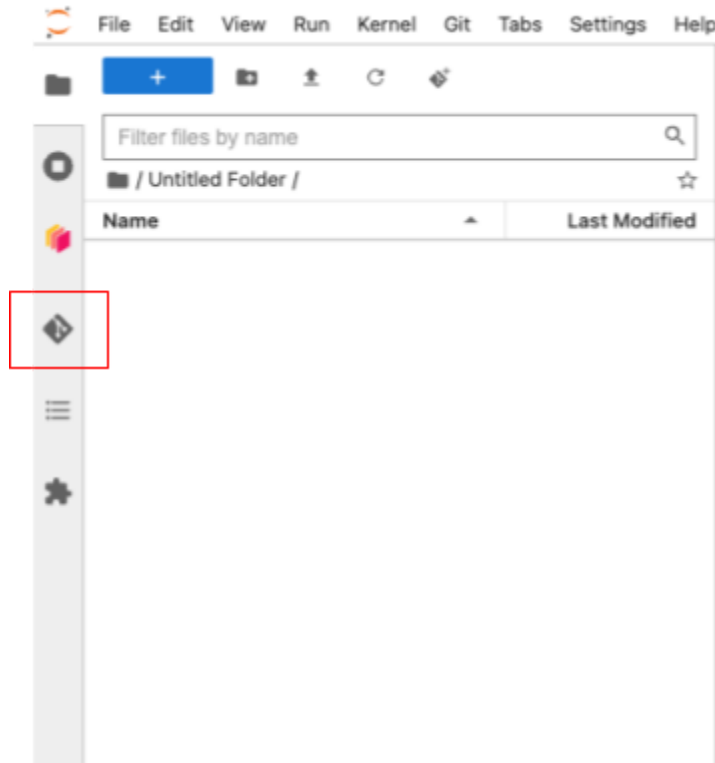
You can see the server starting up.

### Step 3: Get ready to code!

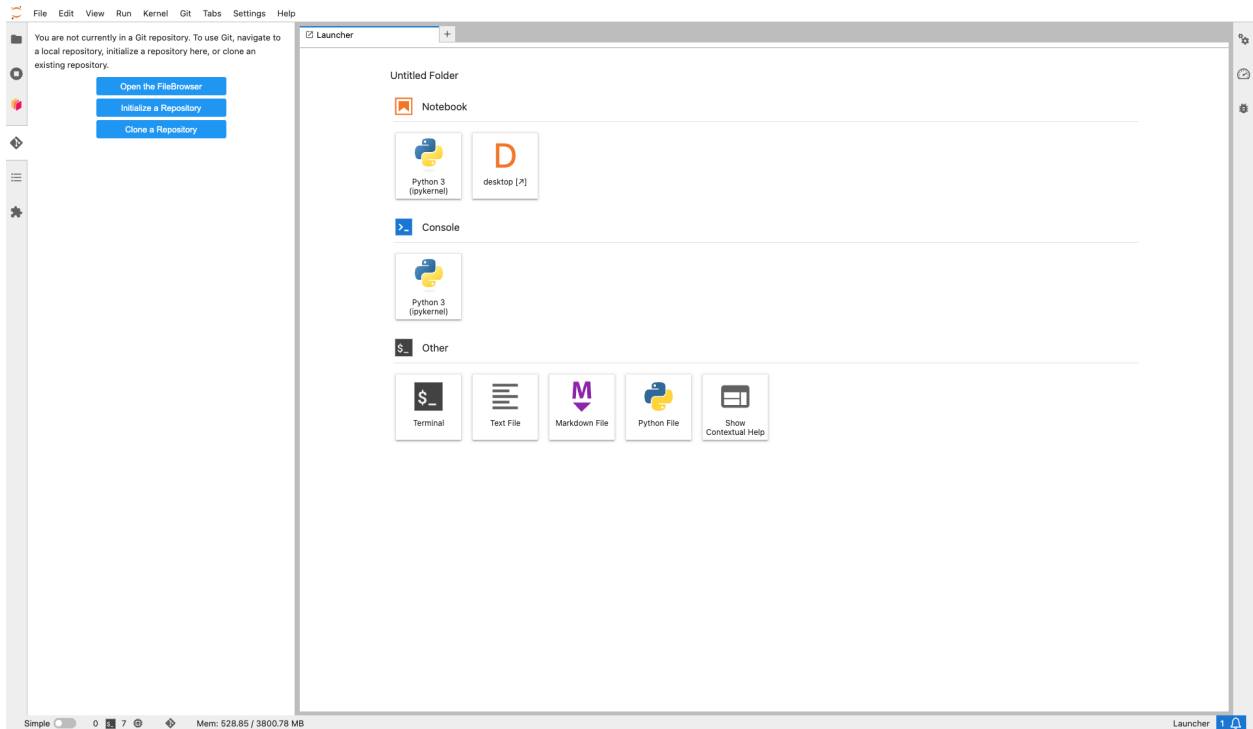


The home page will look like the above image. You can create directories, use the Terminal, and start working on the Python code.

Our demo uses GHRC's jupyter notebooks, We will be cloning the project from GitHub. 2I2C hub comes with the Git extension,



Click the third icon on the left sidebar to open Git Extension.

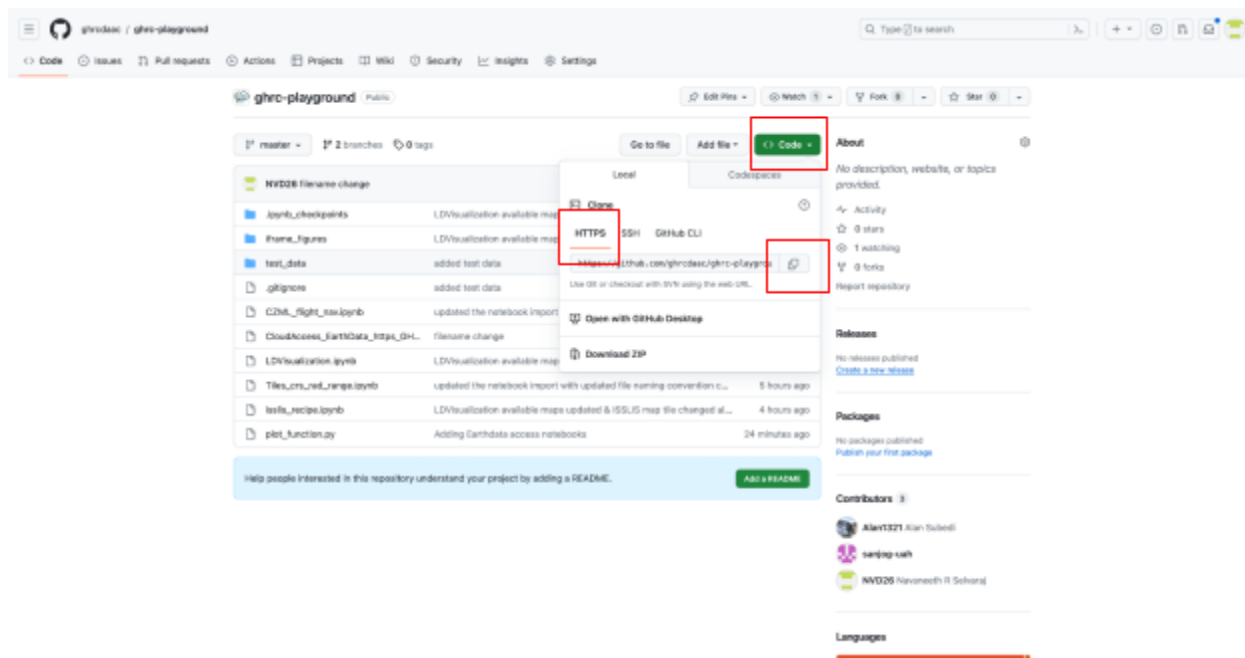


## Step 4: Cloning the GIT project

GHRCs data recipes are hosted in a public GitHub repository

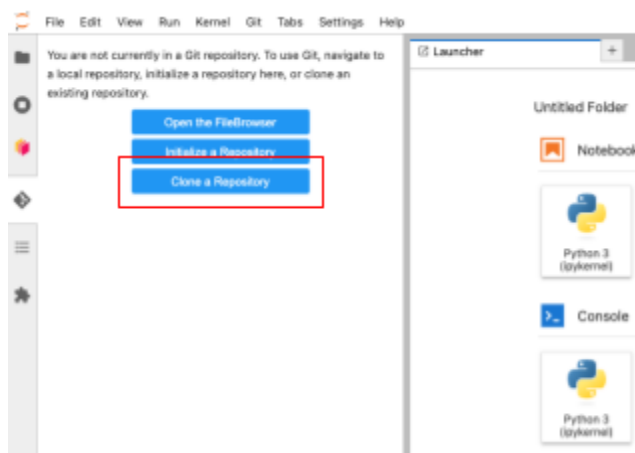
<https://github.com/ghrcdaac/ghrc-playground>

Click on the above link, you will be redirected to the GitHub page.

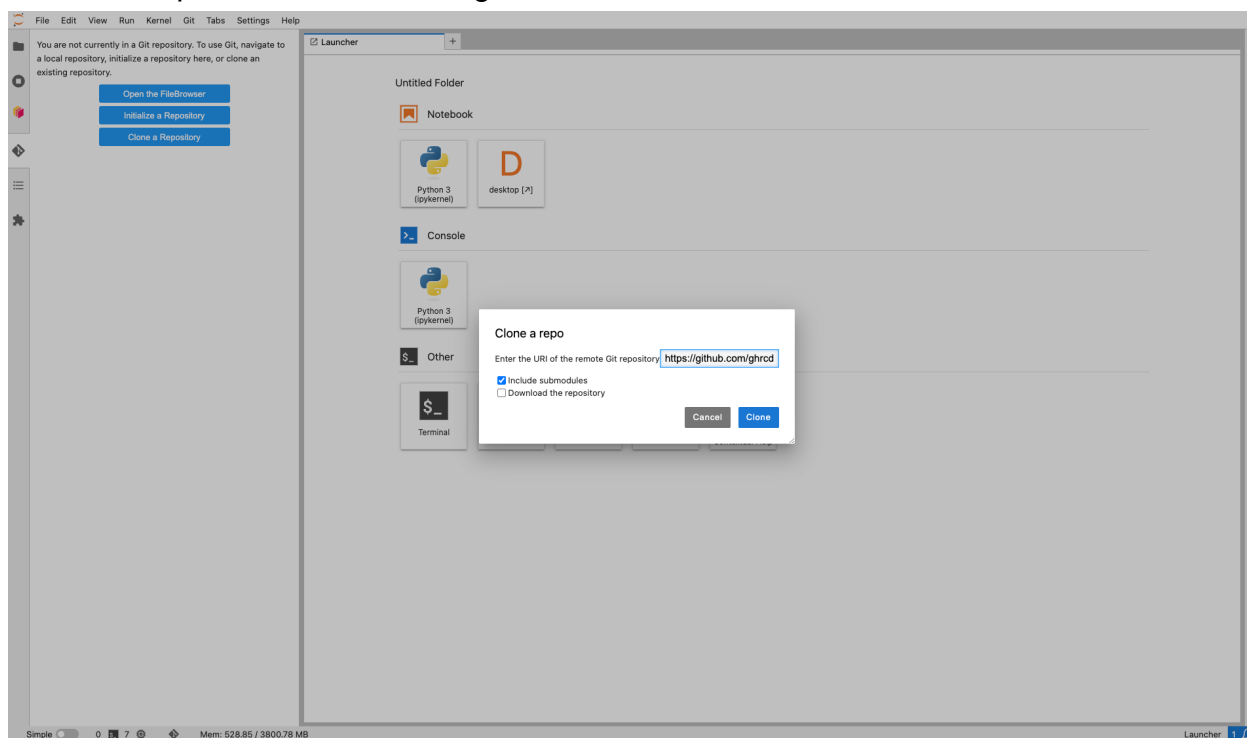


Select “Code” button, then click HTTPS, and then copy the URL.

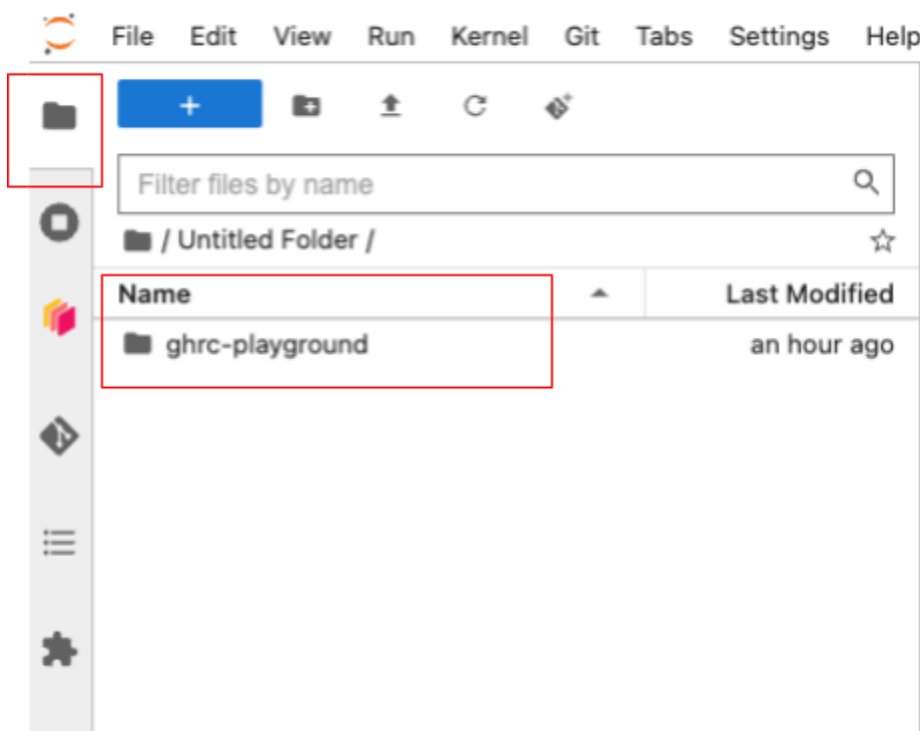
We’ll be heading back to 2I2C page and click “Clone a Repository” button



Enter the copied URL in the dialog box and click “Clone”

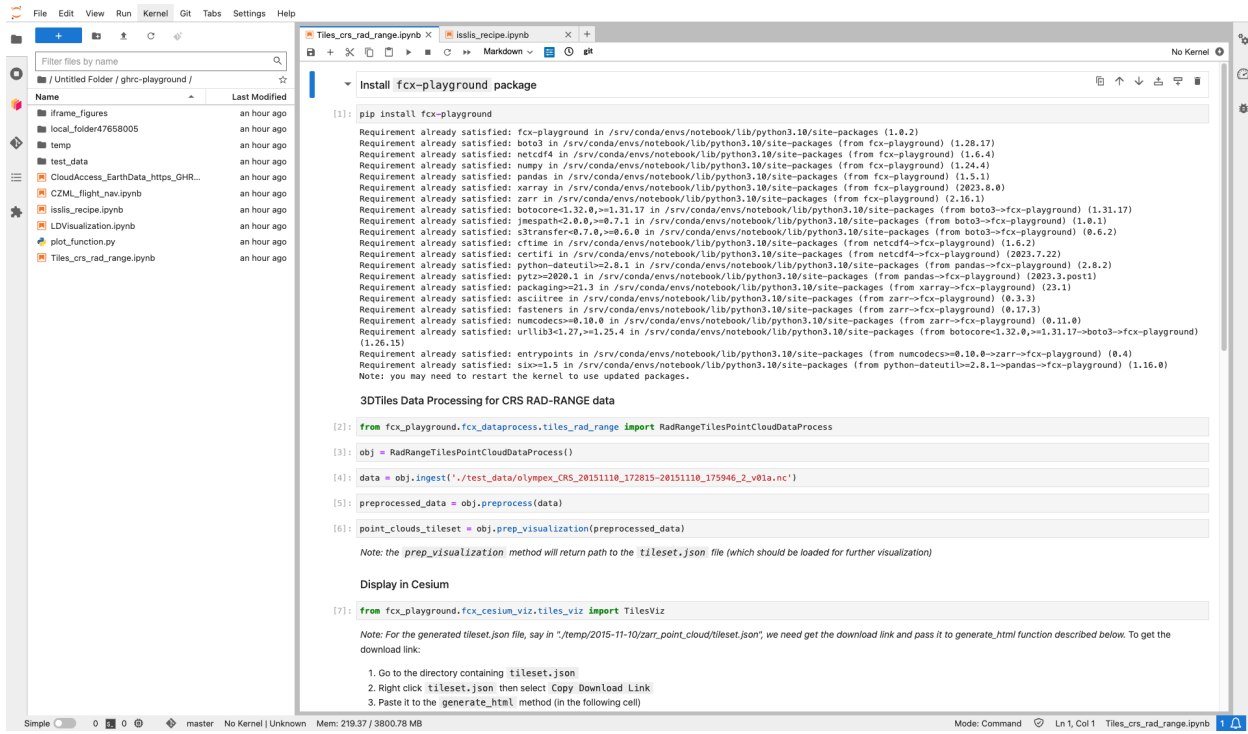


The repository will be cloned into your instance, and you should be able to see in on the left side file explorer

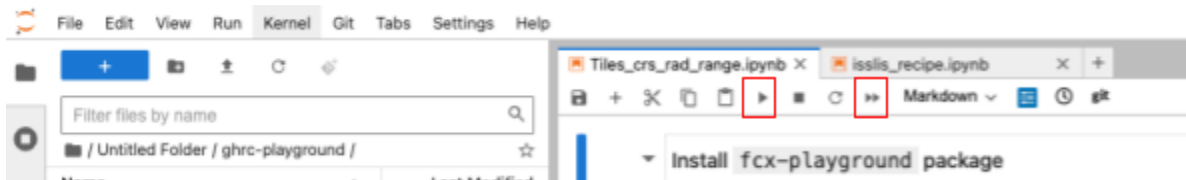


## Step 5: A closer look at the notebooks

Double click on the “ghrc-playground” directory

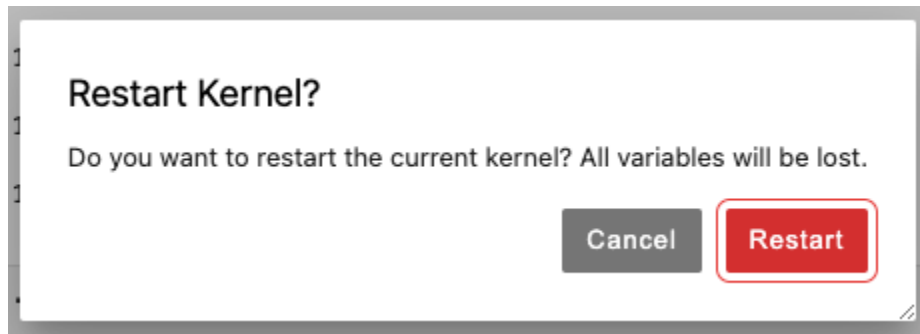


You now have access to 5 notebooks. Open any notebook by double clicking on the filename



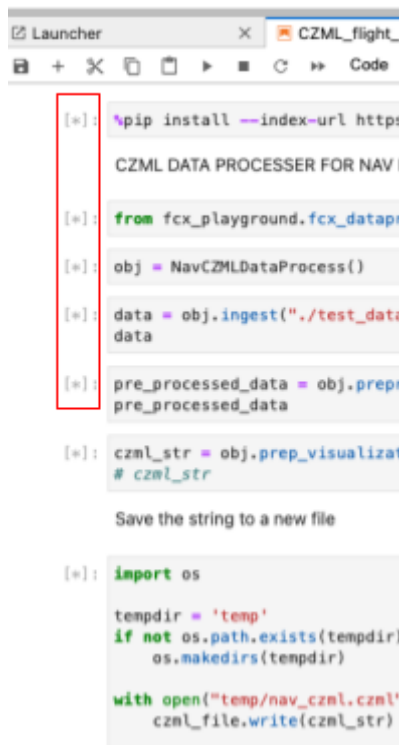
Click on the “Play” button to run the code row by row. If you would like to run all the code at once, Click on the “Play All” (Double play/Forward) button.

If you click “Run All” a dialog box will prompt you to Restart the Kernel



Click Restart, it will run all the code fields and waits for you in the field, if a user entry is required.

If the field is still executing the code, you can see the “\*” on the left side of the row.

A screenshot of a code editor window. The window has two tabs: "Launcher" and "CZML\_flight\_". Below the tabs are icons for file operations and a "Code" button. The main area shows a code cell with the following Python code:

```
[*]: %pip install --index-url https
CZML DATA PROCESSER FOR NAVI
[*]: from fcx_playground.fcx_datapi
[*]: obj = NavCZMLDataProcess()
[*]: data = obj.ingest("./test_data
data
[*]: pre_processed_data = obj.prepr
pre_processed_data
[*]: czml_str = obj.prep_visualizat
# czml_str
Save the string to a new file
[*]: import os
tempdir = 'temp'
if not os.path.exists(tempdir)
os.makedirs(tempdir)
with open("temp/nav_czml.czml"
czml_file.write(czml_str)
```

The first line of code is highlighted with a red box, and an asterisk (\*) is visible to its left.

And the \* will change back to the row numbers as the code finish processing.



The screenshot shows a Jupyter Notebook window titled "Tiles\_crs\_rad\_range.ipynb". The notebook content is as follows:

```
Install fcx-playgr  
[1]: pip install fcx-play  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
Requirement already  
(1.26.15)  
Requirement already  
Requirement already  
Note: you may need t  
  
3DTiles Data Proces  
[2]: from fcx_playground.  
[3]: obj = RadRangeTilesP  
[4]: data = obj.ingest('.  
[5]: preprocessed_data =  
[6]: point_clouds_tileset  
  
Note: the prep_visua
```

Success!, you have executed the python code on the JupyterHub Online. Now, its your opportunity to try different code or adjust the parameters of the existing code to explore, learn, and develop new applications.

## Notes:

- If you open multiple Iframe/HTML viewer notebooks. Make sure you close the notebooks you are not working on. As the HTML page is loaded, the JupyterHub doesn't know where to load the viewer. You may see unexpected results (if multiple notebooks are open).
- Make sure you copy the path of the generated file and paste it on the field to view the visualization.