In this video, we will create and publish a Jupyter Notebook tool. The first part of the tool will focus on creating a ipynb file containing the python code for creating different widgets. In the second part, we will discuss the tool creation in MyGeoHub and then adding the files to tool repository and publishing the tool.

As mentioned in the previous module, it is preferred to use MyGeoHub account login. After you login, try to terminate all unused sessions.

And, this will be the User Interface that can be viewed using the Appmode in Jupyter Notebook.

**Tool Creation**

Now, open you favorite browser and login to MyGeoHub Account.

Remember to terminate any unused Jupyter Notebook sessions using “My Sessions” module in the Dashboard or the Landing Page.

On the top-ribbon, click on Research and navigate to Cyber Training for FAIR Science. Then, on the left pane, click on Courses and open the course with title FAIR Climate and Water Science.

On the right side, click on the Go to Course button to see the different tutorials available within this course.

Scroll down to the bottom and click on Publication 2 Module. You can see a file with instructions and ipynb file. After obtaining a copy of PDF file containing instructions, click on the ipynb file and it redirect to Jupyter Notebook.

Now, you can observe that the file is in read only mode. There are two ways to remove the read-only mode:- Either close and reopen the file or just hit the refresh button.

Before that, let’s save the file with a shorter file name “toolcreation” and move this file in to a folder “Pub2”.

Now hit the refresh button to remove the read-only mode and you can see the earlier read-only symbols disappearing.

On top left, click on mygeohub and launch the Jupyter Notebook tool. Navigate to courses folder.

In the FAIR Science folder, create a sub-folder “Pub2”. And move the ipynb file to the newly created folder by inputting the path.

Go to the new sub-folder and open the moved ipynb file and let’s complete the different cells.

Other than the packages mentioned, we need to use ipywidgets package to use the textbox and button widgets. Let us complete the cell for importing the ipywidgets package.

The next two cells contains two different methods to include a sample image for the User Interface in the appmode. Click on the second cell and specify the URL to sample image which can be copied from the above comment. Also, this cell is in code format, so change it from Code cell to a Markdown cell using the dropdown at top menu.

If you run the cell without changing to Markdown Cell, you will get an error.

You can see the sample image getting loaded.

The third cell contains another method where you can use a code cell (instead of markdown cell) using IPython.display module.

The next cell contains two definition blocks for obtaining the daily discharge data from USGS (using hydrofunctions package) and plotting the daily discharge hydrograph. The two definitions are already given in the code.

Let us complete the next cell which will contain three textbox widgets for prompting the user to input the station number and starting and ending date of the analysis period. We will add a placeholder and description. Also, will call them using display function to view in the appmode. Let us type-in the code.

It is easier to make typos, but make sure to correct it later or when you get an error.

Once the above cell completed, we need to create a button widget which will obtain the data based on the USGS Station number and analysis period. We will initially create a button widget and then link the button widget with the two earlier defined functions so that when the button is clicked it will execute both the functions and display the results. Let us type-in the code.

We will include a description, button\_style, tool tip and an icon for the button. Finally we will add a on\_click method to enable the button to execute a set of code when it is clicked as they are stateless.

Run the cell and let’s move on to next cell. Here, we will again define two functions for calculating the exceedance probability and plotting flow duration curve (in linear-linear scale). The two definitions are already given in the ipynb file.

The next cell contains the code for creating a second button and linking it to the two functions (ExceedProb\_func and FlowDurCurveLinLin\_func) that were defined in the previous cell.

Run the cell. In the last cell, we will complete the code to display the two buttons in two different tabs (using a tab container widget) so that it appears in an organized manner and creates better use of space. Let us type-in the code.

Run the code once it is completed. Again, if you get any error, make sure to correct it.

To see the UI mode, click on Appmode on the top menu bar. It will convert ipynb file into web application. In this mode, all codes are hidden, but markdown cells are shown after executing.

Let us some sample inputs for the three textboxes to check the results. We will use the USGS station number for Wabash River at Lafayette and a six month time period in 2017.

Further, the two “plot” buttons in two different tabs will generate the Discharge Hydrograph and Flow Duration Curve for the sample inputs.

This completes the first part of the tutorial. Click on Edit App to return to ipynb file. Save the changes. Now, go to File and click on “Close and Halt” to shutdown the ipynb file. Close the tab. You can see the book symbol has changed from green to black in the folder directory.

Terminate the session using the button on the top right.

**Tool Publication**

Now, in the MyGeoHub page, go to Resources and click on Tools Tools. In the new page, click on Start a new Tool option.

Add a unique tool name, a title and a description for your new tool. I had previously used the same tool name and title which lead to conflicts. So try avoiding previously used tool name and title to register the tool. This may take a little time to figure out. A quick way is to add a number or first three characters of your name.

Change the Publishing Option to “Publish as a Jupyter Notebook” and leave all others as default.

Click on the Register button at the bottom. Oops. I used same title before. So I am adding a number 2 as a quick fix. This also did not work. So I am going to change the title a bit.

Ok, let me add a number 2 at the end.

Finally we have registered the tool.

Launch Jupyter Notebook Home Directory which can be accessed using My Tools modules available in Dashboard.

Now, launch the terminal window in Jupyter Notebook Home Directory by navigating to New (on the top right) and then clicking on Terminal option in the dropdown.

Download the tool code/files created by MyGeoHub while registering the tool to your local home directory using “svn” command

Once the command is executed, a new folder will be created in the home directory with different files and folders. For this tutorial, we had chosen the unique name as hydfdctool2 in the earlier step.

Open the newly created folder. The bin folder (inside the newly created folder) is assigned location for the python code, data folder is for any relevant data, and middleware contains the invoke script file.

Now the bin folder is empty.

Navigate to the folder containing the code from the first part. Create a copy of ipynb file (using Duplicate button in the folder and rename the file to toolcode.

Move the file to bin folder using Move button and then input the pathname to the bin folder.

Next, we have to change the script in invoke file in middleware sub-folder. Go to “middleware” sub-folder in tool folder and change the sample code in “invoke” file as the shown in the instruction pdf.

Now, save the invoke file.

At this point, we will add the changes to MyGeoHub Subversion Repository by using svn add … and svn commit … commands. First, in the terminal window, change the directory to the tool folder using the command “cd ” as shown below.

Using the change directory command, folder has changed from home directory to tool folder.

We have to upload the newly added files to remote repository using “svn add ”

We will use svn commit to save the changes as a version with a commit message “initial commit”. You need to enter the password (MyGeoHub Account) for changes to take place.

There are few more steps to complete the tutorials. But, the following steps should not be done while you are doing this tutorial for trial purposes. It should be done when you are about to publish your final version of the tool. Otherwise, admin will receive a lot of review requests at a time.

Ok, let’s move ahead. Next, you have to request for review by the admin to publish your tool. Navigate to the tool status page: (https://mygeohub.org/tools//status) or navigate to Dashboard > My Drafts > and then your tool. In my case, it is hydfdctool2.

In the status page, request for review by admin by clicking on “My code is committed, working, and ready to be installed”. Normally, it will take three business days for the admin to approve the request.

By clicking on the edit button at the top, you can add a contributor to this tool. Remember to save after any edits.

Now you have completed the tutorial on tool creation and publication. Thank you.