Week 1: Introduction to the JupyterHub environment

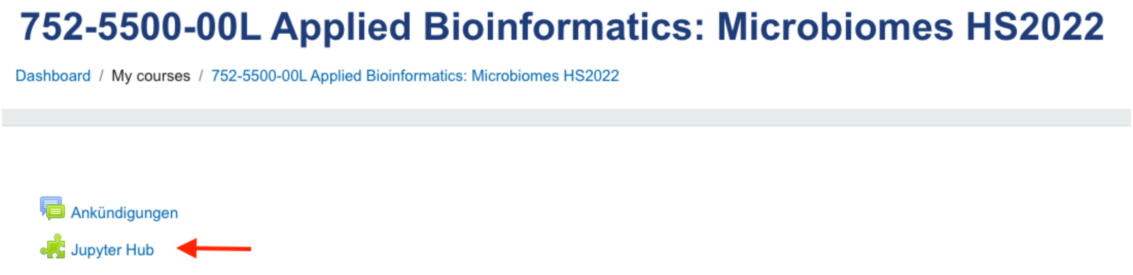
Welcome to the first week of the Applied Bioinformatics course! During this exercise we will familiarize ourselves with the JupyterHub computational environment that we will be using throughout this course. JupyterHub allows us to work with Jupyter notebooks where we will be running all the code, and provides several extensions to make our work easier (more on that later).

> **Why JupyterHub?** Jupyter notebooks are interactive computational notebooks that allow you to organize code and text together in a single notebook that is accessible via a web browser interface. JupyterHub is a multi-user server for Jupyter Notebooks. In other words, JupyterHub allows us to launch interactive notebooks and exercises in a classroom setting, both for interactive exercises and evaluations as well as collaborate group projects. Jupyter Notebook is widely used for scientific computing (and otherwise), and we will use it to organize our code and projects in Applied Bioinformatics. After this course you will understand why it was voted among the top 10 most influential computer codes that transformed science.

Accessing JupyterHub

To access JupyterHub follow the steps described below:

1. On the Moodle platform, navigate to our course.
2. On the dashboard you should see an option named *JupyterHub* (see figure 1) - click to launch in a new window/tab. You may see a page with a blue button saying "Start My Server" - click to continue (it may take a short while to launch the server when you do it for the first time).



752-5500-00L Applied Bioinformatics: Microbiomes HS2022

Dashboard / My courses / 752-5500-00L Applied Bioinformatics: Microbiomes HS2022

Figure 1: Moodle dashboard

3. Once your server launches, you should see the interface similar to the one shown below (figure 2).

   On the left you will see the directory structure of your workspace (this is where your files and folders will live - right now it is empty). On the right you should see the *Launcher*. Launcher is a special tab that lets you open different kinds of working environments:

   - Jupyter notebooks (with pre-selected kernels; more on that later)
   - Python console (with pre-selected kernels) - here you can interact with Python directly
   - Terminal (aka *command line*) - we will not be interacting with it much, but may sometimes need it to e.g. create new Python virtual environments
   - Other tools (feel free to interact with any/all of those and try them out!)

   On the *very* left you will also see a narrow toolbar: this is where some extensions have their place, e.g. the *Git* extension (third icon from the top) - more on that later.

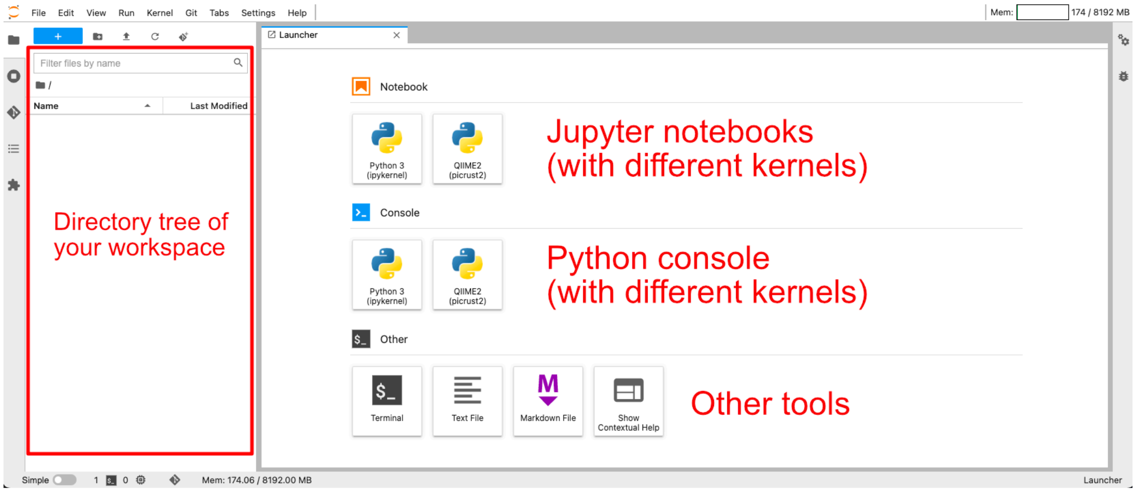


Figure 2: JupyterHub landing page

Tip: If you ever close the *Launcher* tab and can't find it anymore, just click on the big blue button on top of the directory tree. You can also go to the *File* menu on the very top and select *New Launcher* or simply hover over *New* and directly select what you want to create.

4. Create a new Jupyter notebook. Jupyter notebook is a kind of "computational notebook", where you can execute code while embedding markdown text and images to add structure and more verbose content. You will hear more details about notebooks in the next week's exercise - here we will just set up the stage. There are (at least) two ways to create a new notebook in the JupyterHub interface:

   a) **from the *Launcher*:** in the *Notebook* section, click on the leftmost icon to open a notebook with our default kernel (**Python 3 (ipykernel)**)

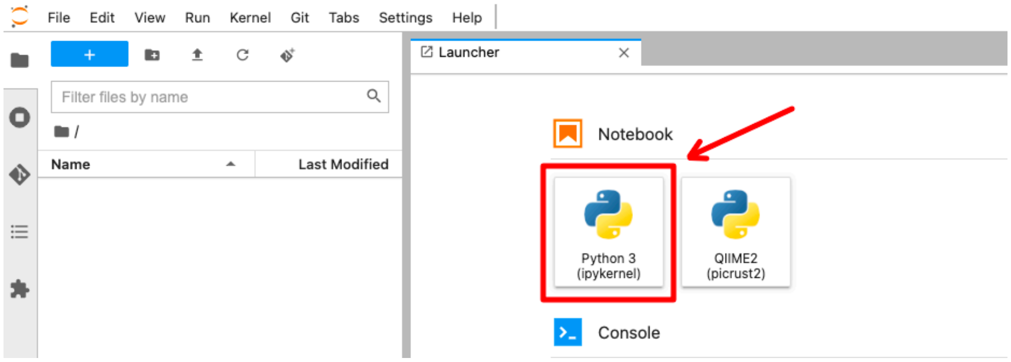


Figure 3: Launching a new Jupyter notebook

   b) **from the *File* menu:** go to *New* -> *Notebook*. When prompted, select the **Python 3 (ipykernel)** kernel

Note on kernels: You should think of kernels as different runtime environments that can be used in Jupyter notebooks. Each environment can have different packages/tools installed and you can switch between them inside the notebook (more on that later). They do not even need to always be based on Python - we can also have kernels based on other languages like R or Julia. In this class we will focus on Python-based kernels though. Our default kernel will be **Python 3 (ipykernel)** - it has QIIME 2 v2022.2 already preinstalled and preconfigured with all the plugins/tools required in this class.

That's it! You should now see a new tab with an empty notebook, waiting to be filled. You're ready to start coding! If you want to read more about how to use this environment, feel free to check out the official guide (there is a nice, short video demo).

Reconnecting to an already running environment

It may (and will) happen that you start some analysis in the JupyterHub environment and leave it or lose connection. When that happens, simply go to Moodle and use the JupyterHub link there to reconnect. You should then see two buttons (figure 4):

- "Stop My Server"
- "My Server"

Do **not** click on the red "Stop My Server" button! This will interrupt all the running kernels and you will lose all the unsaved work! Instead, click on "My Server" to reconnect to your environment.

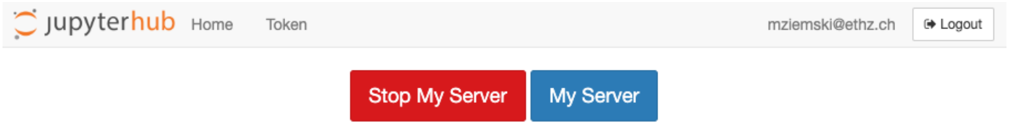


Figure 4: Reconnecting to JupyterHub

Note: your server will be stopped automatically if it is inactive for a while (around 4 hours). In that case you will only see one option to reconnect.

Moodle assignments

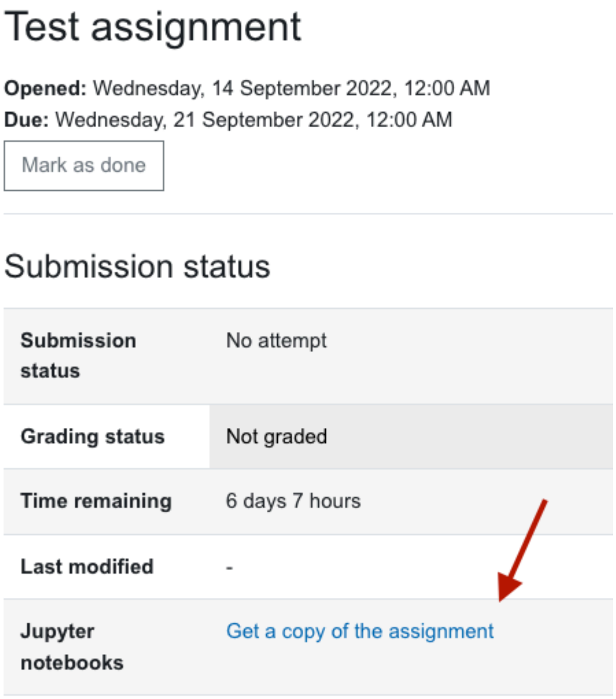
Your JupyterHub workspace is tightly integrated with your Moodle account. Most, if not all, of the assignments in this course can be easily cloned into your workspace and re-submitted back to Moodle for grading. See below for instructions on how to copy an assignment and then submit it.

To clone an assignment into your JupyterHub workspace

1. Before tackling your first assignment, create a new folder `Assignments` in the home directory of your JupyterHub workspace where you will store all the assignments.
2. In Moodle, go to the assignment you want to copy. On the page of the assignment click on the `Get a copy of the assignment` link (see figure 5).
3. On the next page, select a directory you created in step 1 - this is where the current assignment will be copied (see figure 6). Click on the `Copy assignment` button.
4. Head back to your JupyterHub workspace and navigate to the *Assignments* directory - you should now see a fresh copy of the assignment.

To submit an assignment for grading

1. Within the assignment's directory create a new folder where you will place the notebook(s) to be submitted. You can name it e.g.: `submission`.
2. Copy the files (notebooks) you want to submit to the respective directory that you created in step 1.
3. In Moodle, on the assignment's page, click on the `Add submission` button.



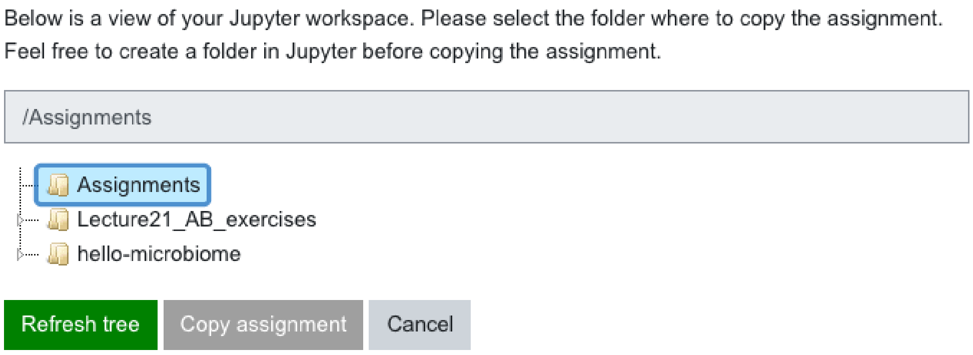
Test assignment

Opened: Wednesday, 14 September 2022, 12:00 AM
Due: Wednesday, 21 September 2022, 12:00 AM

Submission status

Submission status	No attempt
Grading status	Not graded
Time remaining	6 days 7 hours
Last modified	-
Jupyter notebooks	Get a copy of the assignment

Figure 5: Cloning an assignment - part 1



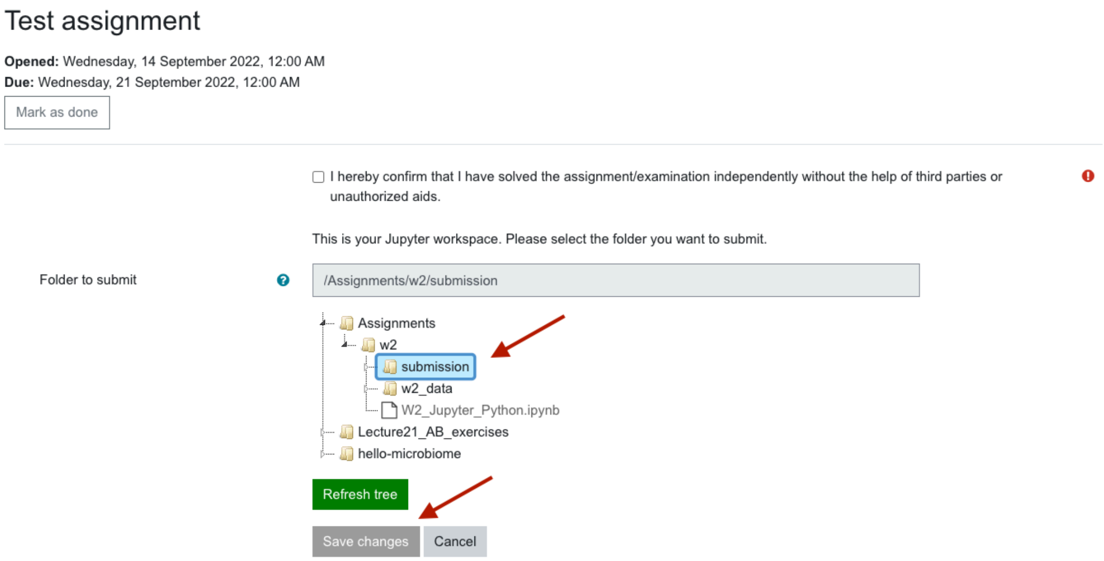
Below is a view of your Jupyter workspace. Please select the folder where to copy the assignment. Feel free to create a folder in Jupyter before copying the assignment.

/Assignments

Figure 6: Cloning an assignment - part 2

4. From the directory tree, select the submission folder containing the files you want to submit (see figure 7). Don't forget to tick the required field after reading its contents.
5. Click on the `Save changes` button - congratulations! You have submitted your Jupyter assignment!

Important: The files you want to submit will always need to be located in a separate directory - it is not possible to submit individual files but directories. Please **do not** upload the entire folder which you cloned originally from the Moodle assignment as it may contain a lot of additional data.



Test assignment

Opened: Wednesday, 14 September 2022, 12:00 AM
Due: Wednesday, 21 September 2022, 12:00 AM

☐ I hereby confirm that I have solved the assignment/examination independently without the help of third parties or unauthorized aids.

This is your Jupyter workspace. Please select the folder you want to submit.

Folder to submit: /Assignments/w2/submission

Figure 7: Submitting an assignment

JupyterHub FAQ

1. **I messed up my default QIIME 2 conda environment - what should I do?**

   Close the JupyterHub tab and try to reconnect from Moodle. On the page with the red/blue buttons select the red one to stop your server and then the blue one to re-create it. The default QIIME 2 environment will be restored. Any other environments which you created will not be deleted. None of the files you keep in your workspace will be deleted. Any unsaved work may be lost though.

2. **I am getting an error message when removing an empty folder.**

   The folder most likely contains hidden files. To delete such folders you need to open the *Terminal* (from *Launcher* or the *File* menu) and execute:

   ```
   rm -r <path to the directory>
   ```

   where you replace the "path to the directory" with your actual folder path.

3. **I am not able to install a Python package due to permissions being denied.**

   In some, rather rare, cases, installation of a Python package may require additional, admin permissions. If you encounter this issue, please talk to us directly.