Wolfram Mathematica with JupyterLab

Alan Humphrey[*] - Tuan Pham[†]
Last revision: Sep 3, 2025

Contents

What to expect? 2
Installation on Windows 3
  Install Python 3
  Install JupyterLab 4
  Install Wolfram Engine 5
  Install WolframLanguageForJupyter 7
Installation on MacOS 8
  Install Python 8
  Install JupyterLab 11
  Install Wolfram Engine 11
  Install WolframLanguageForJupyter 13
Post-installation 13
  Start and terminate a session 13
  Create shortcut icon (optional) 15
  Upgrade Wolfram Engine (optional) 15
Overcoming some challenges 16
  Copy and Paste 16
  Export a report 16
  Interact with graphs 16
  Safari: kernel not connecting or Interact@ showing nothing 18

[*] Eastern Oregon University
[†] Brigham Young University-Hawaii

What to expect?

1) Mathematica is a powerful mathematical software developed by the Wolfram Research company. Many universities purchase institutional licenses so that their students and employees can use Mathematica for free during their study program or employment. However, Wolfram releases Wolfram Engine, the powerful computing kernel of Mathematica, for free for software development purposes. The missing piece is the so-called Wolfram Notebook, a user interface through which users can enter codes and interact with outputs. The purpose of this guide is to help you install on your personal computer the Wolfram Engine and integrate it into JupyterLab, an open-source software serving as an interface for Mathematica (and many other programing languages). The guide is written for Windows and MacOS. To minimize possible technical problems, it is recommended that you follow the installation order shown here. The last section of this guide addresses some common issues when using Mathematica on JupyterLab and how to overcome them.

2) Once you have finished all the installations as guided below, you will be able to work on an interface called Jupyter notebook as if on the native Mathematica notebook. To open JupyterLab, go to the terminal and enter the command:

   jupyter-lab

   (Don't worry! You won't be working on a black command-line screen. A user-friendly interface will pop up after this command.) Mathematica syntaxes must be used, including:
   - pressing Enter to start a new input line,
   - pressing the combination Shift + Enter to execute an input cell,
   - pressing the combination Alt + . to abort an execution.

3) The following video was created to give you a better idea on how to use Mathematica on JupyterLab and create a lab report: https://youtu.be/vftZmbMq7yU

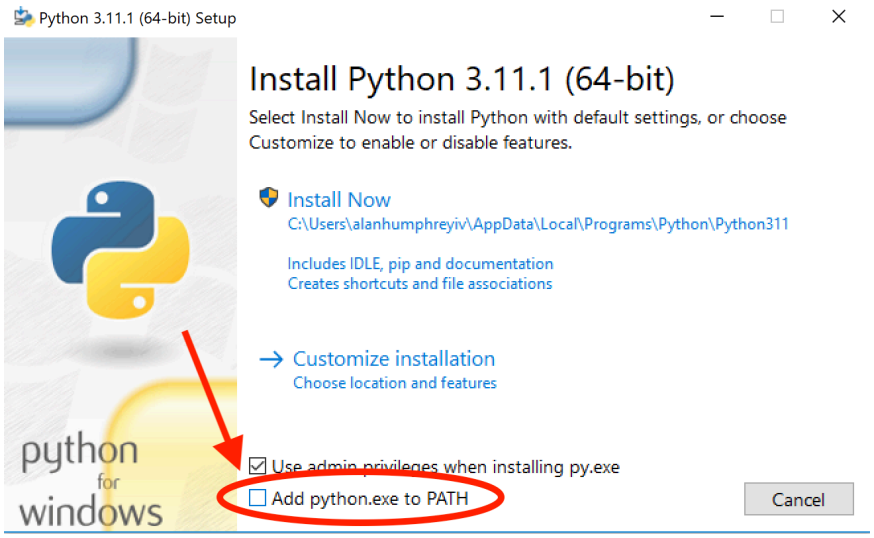
4) If you prefer to use Mathematica on Wolfram Cloud instead of JupyterLab, the following video will show you how to make a lab report: https://youtu.be/5wsfG80oD1g

Installation on Windows

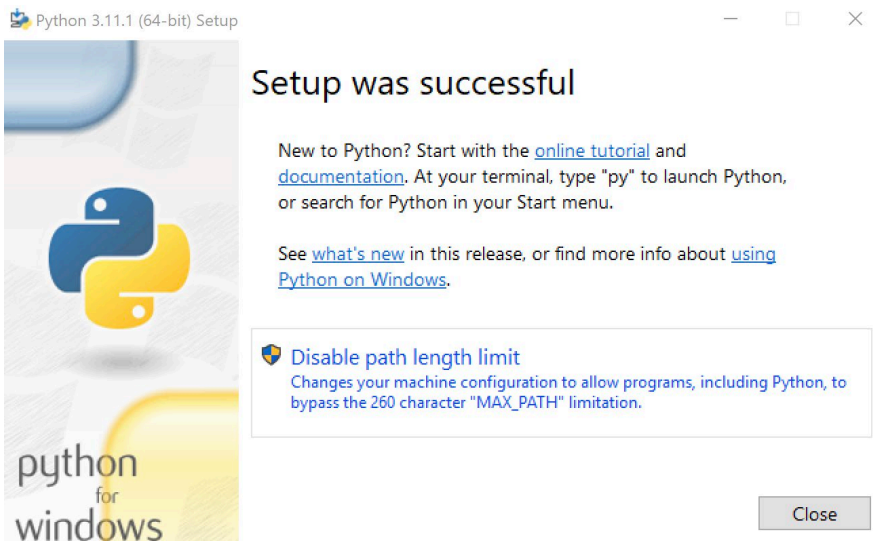
Install Python

1) Download the latest version of Python from https://www.python.org/downloads/

2) Open the .exe file that was downloaded. Check "Add python.exe to PATH". Then press Install Now and continue the installation.



3) Before closing the installer, make sure to click on "Disable PATH length limit" and agree to the changes.

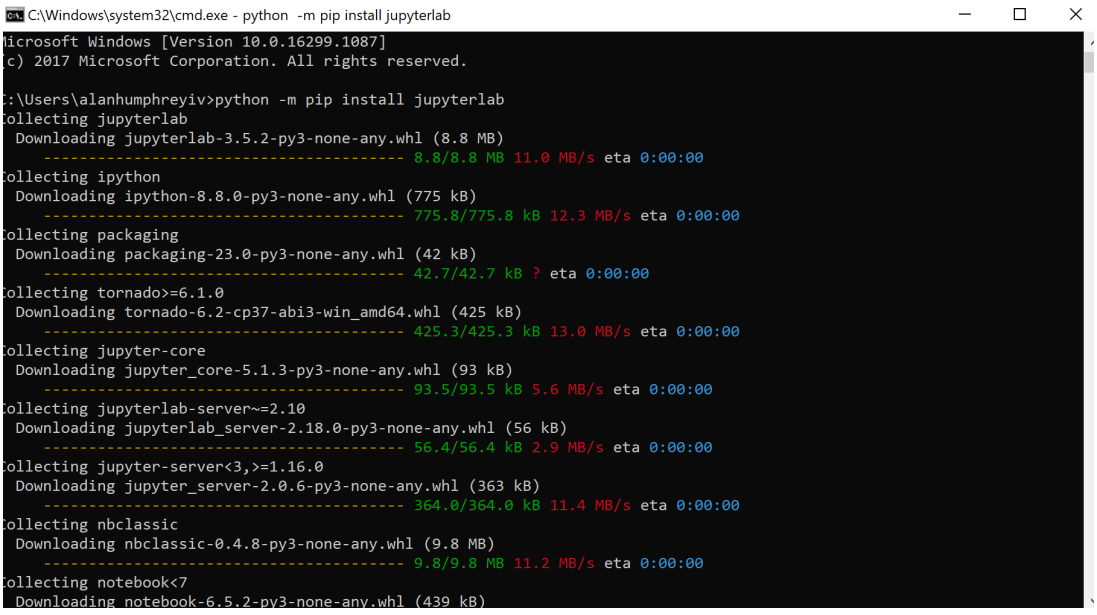


Install JupyterLab

After installing Python, the next step is to install JupyterLab.

1) Open your Command Prompt: an easy way is to press **Win + R**, type **cmd**, then press Enter.
2) Type in the following command[‡] and press Enter:

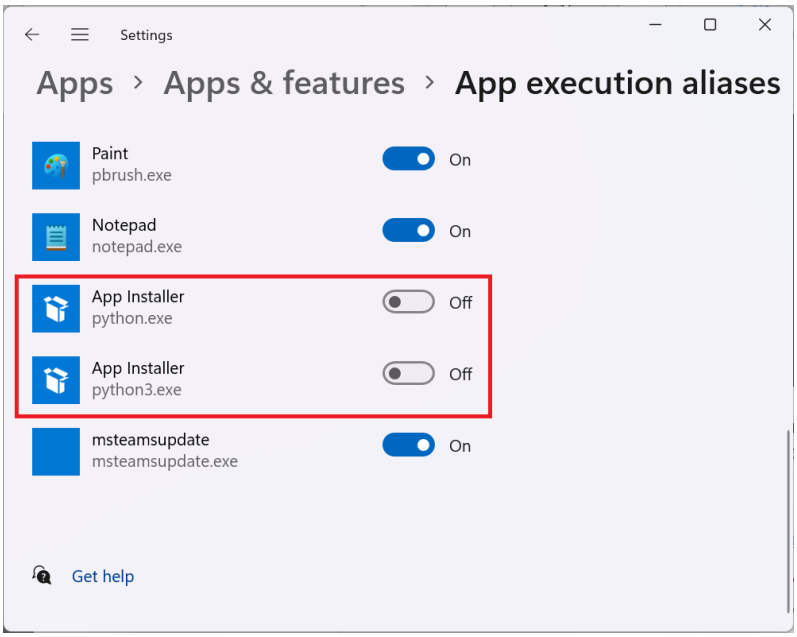
python -m pip install jupyterlab



Note: If you receive a message "Python not found…", then go to Start and type "Manage App Execution Aliases". Go to it and turn off Python (see the screenshot below). You

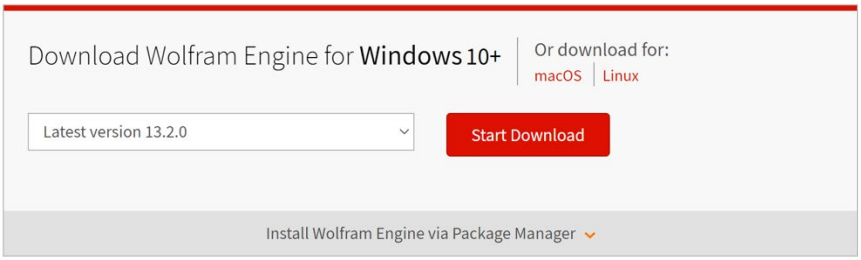
[‡] See below if you receive a message "Python not found…"

might need to restart your computer for the change to take effect. Then go back to the terminal and try again the command: python -m pip install jupyterlab

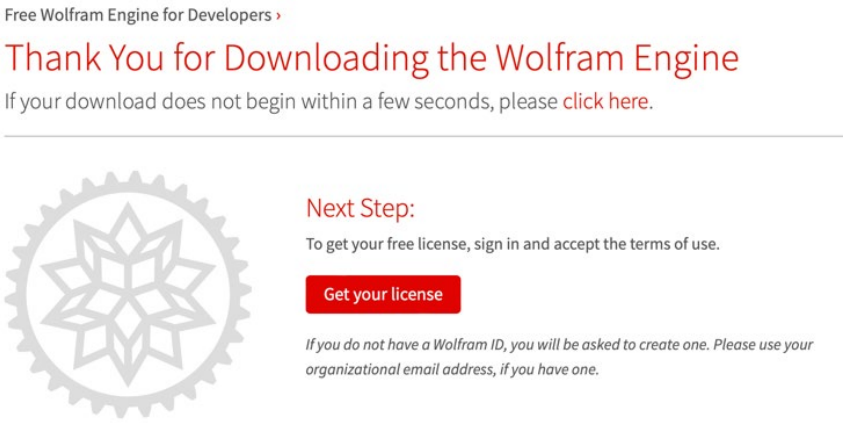


Install Wolfram Engine

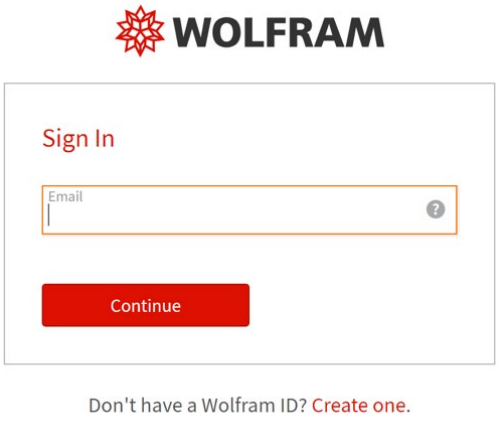
1) Download Wolfram Engine from https://www.wolfram.com/engine/



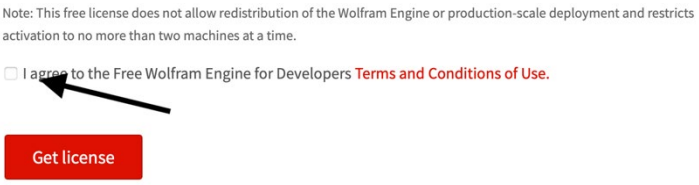
2) Click on Start Download. Then click on "Get your license".



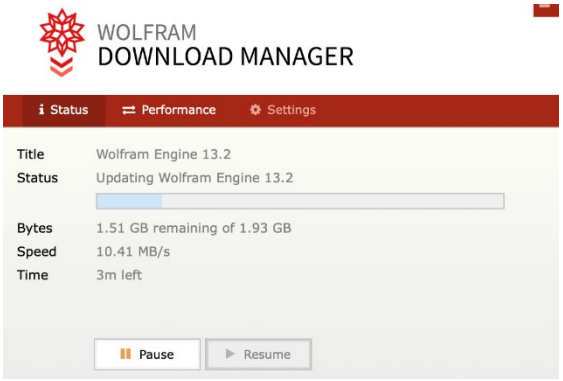
If you have not had a Wolfram account yet, create a free account.



3) Accept the Terms and Conditions. Then click on “Get License”.



4) After the download is finished, open the .exe file that was downloaded from Wolfram and let the installation run.



5) Normally, a terminal will automatically be opened once the installation is finished. But if it does not, then open the terminal and type

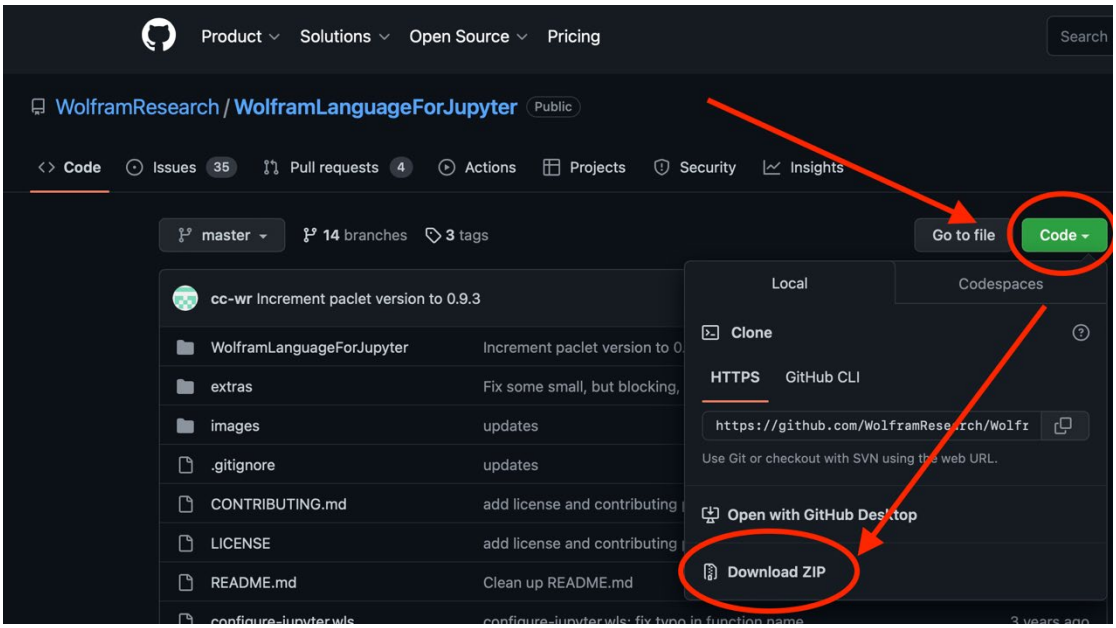
wolframscript

6) Activate your free license by entering your Wolfram ID and password, which are the email address and password you used to create a Mathematica account. Note that nothing is shown on the screen as you type your password.

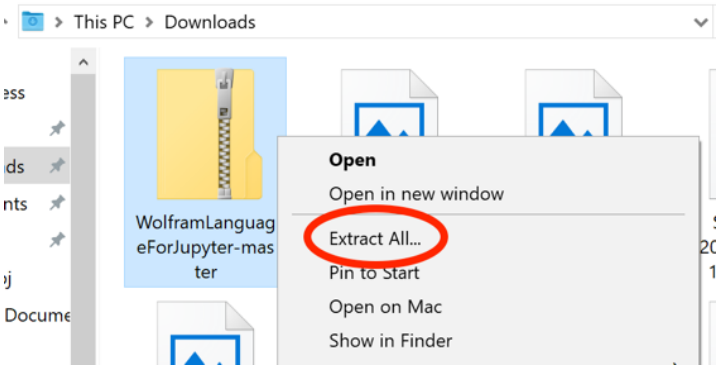
```
The Wolfram Engine requires one-time activation on this computer.

Visit https://wolfram.com/engine/free-license to get your free license.

Wolfram ID: alanghumphreyiv@gmail.com
Password:
Wolfram Engine activated. See https://www.wolfram.com/wolframscript/ for more information.
Wolfram Language 13.2.0 Engine for Microsoft Windows (64-bit)
Copyright 1988-2022 Wolfram Research, Inc.

In[1]:= _
```

Install WolframLanguageForJupyter

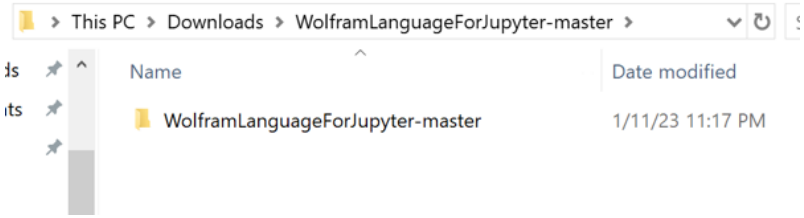
1) Go to https://github.com/WolframResearch/WolframLanguageForJupyter
2) Click on the green button "Code". From the dropdown, choose "Download ZIP".



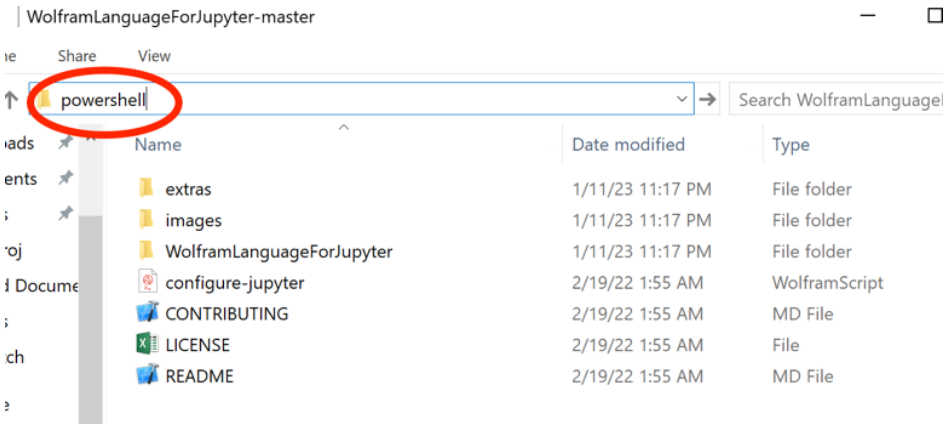
3) After the download is finished, right click on the ZIP folder, choose "Extract All", and press Enter. Move your newly extracted folder to a safe place in your computer.



4) Go into the newly extracted folder.



5) Type in “powershell” in the address bar of the opened folder and press Enter.



6) Now enter this command in the powershell that was just opened and press Enter twice.

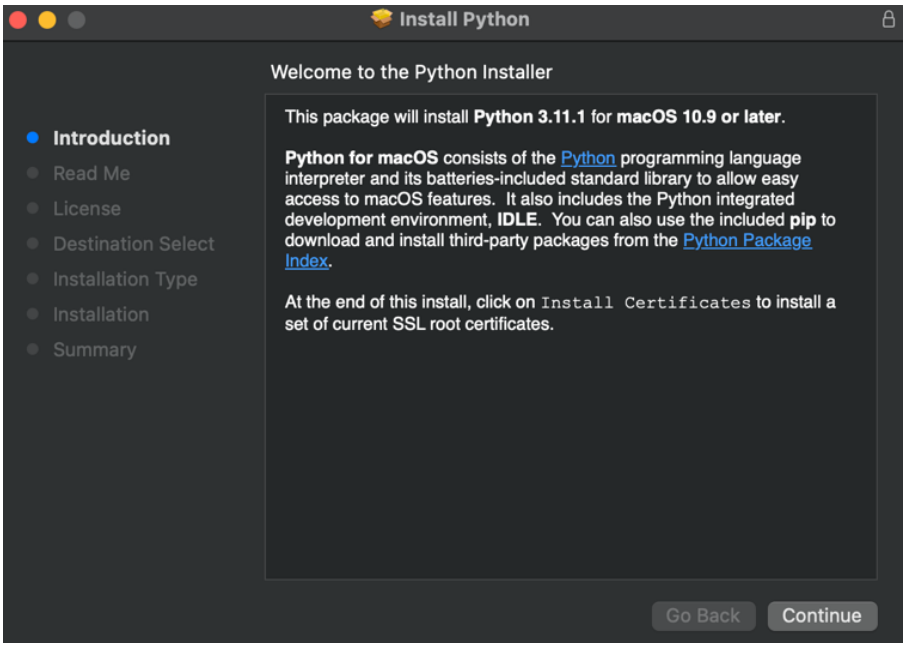
./configure-jupyter.wls add

Installation on MacOS

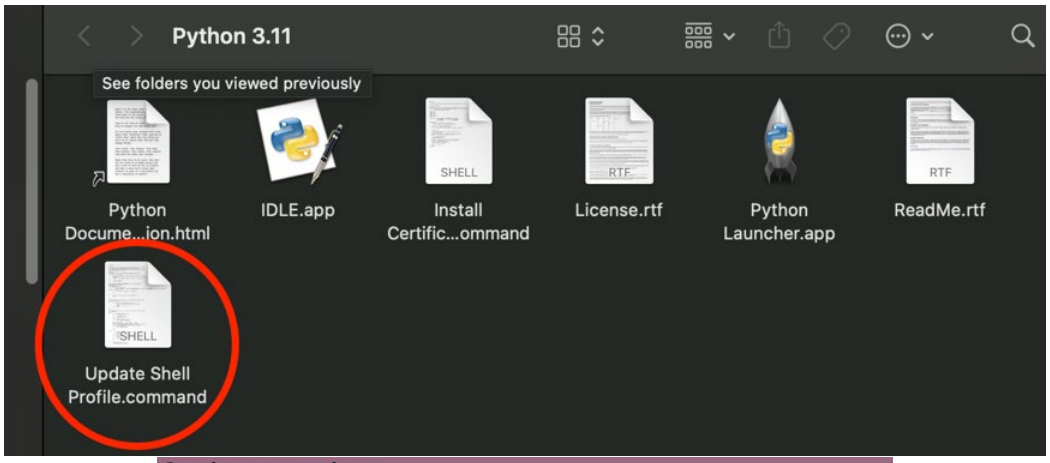
Install Python

1. Download Python from https://www.python.org/downloads/

2. Open the .pkg file that was downloaded and go through the Installation prompts. All default options should be fine.

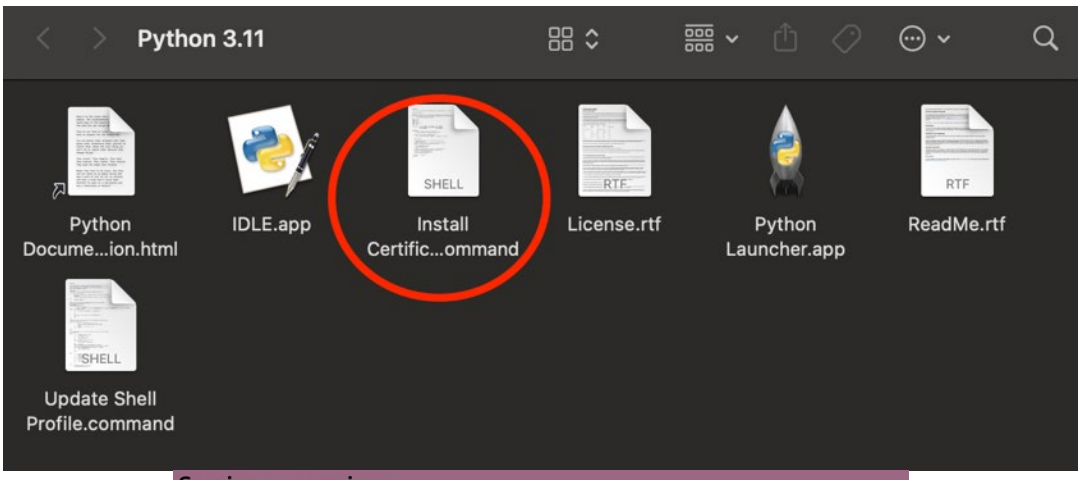


3. In the folder that is opened, double click “Update Shell Profile.command”. This will open the terminal and should finish with “[Process completed]”.




```
Saving session...
...copying shared history...
...saving history...truncating history files...
...completed.

[Process completed]
```

4. Now double click “Install Certificates.command”. This should open the terminal and finish with “[Process completed]”.



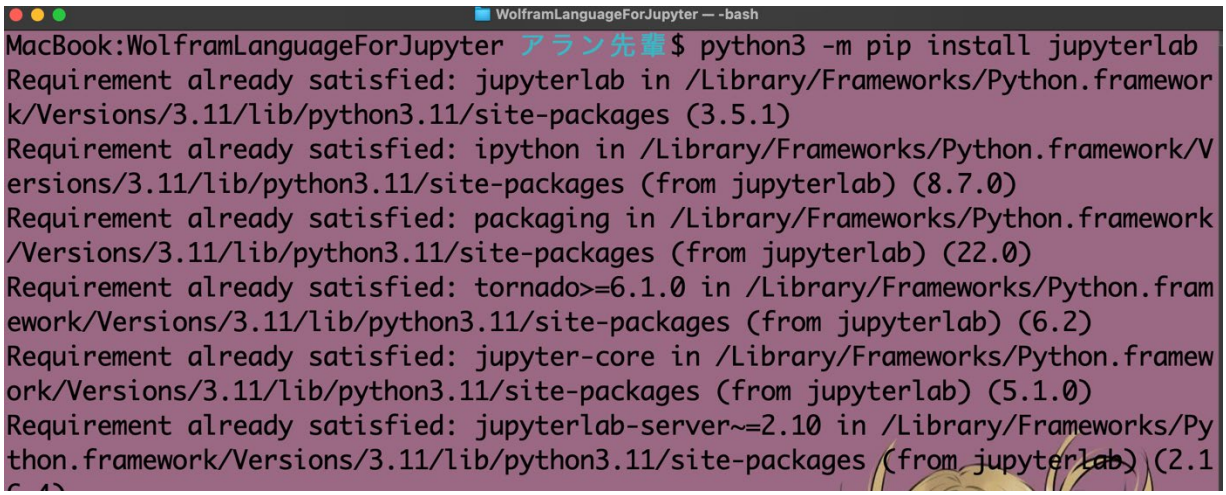

```
Saving session...
...copying shared history...
...saving history...truncating history files...
...completed.

[Process completed]
```

5. Python should be installed. You can now exit out of the terminals and the finder window, and move the installer to the trash.

Install JupyterLab

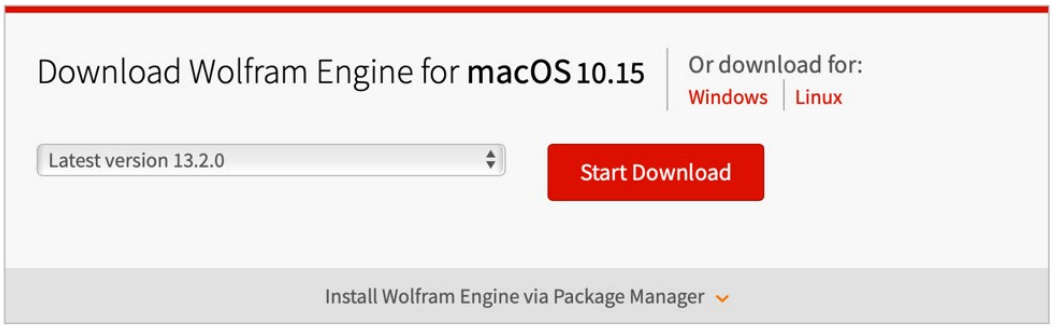
Open the terminal and run the following command:

python3 -m pip install jupyterlab

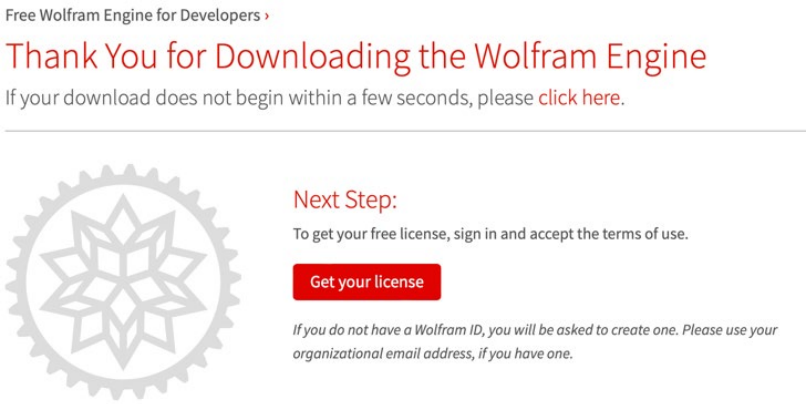


Install Wolfram Engine

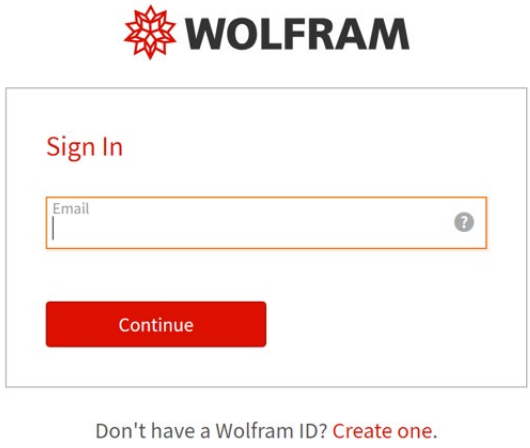
1. Download Wolfram Engine from https://www.wolfram.com/engine/



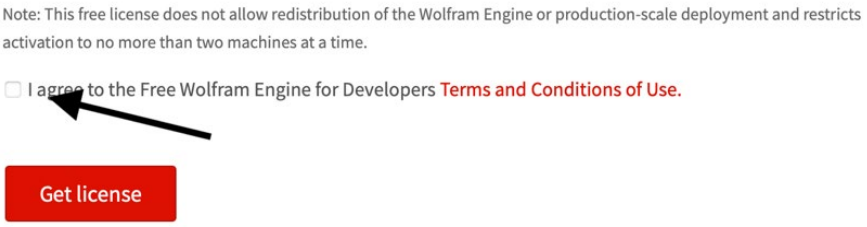
2. Click on Start Download. Then click on “Get your license”.



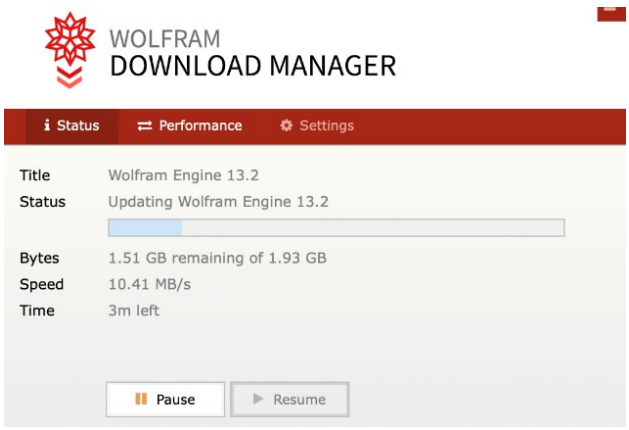
If you have not had a Wolfram account yet, create a free account.



3. Accept the Terms and Conditions check box and click “Get License”.



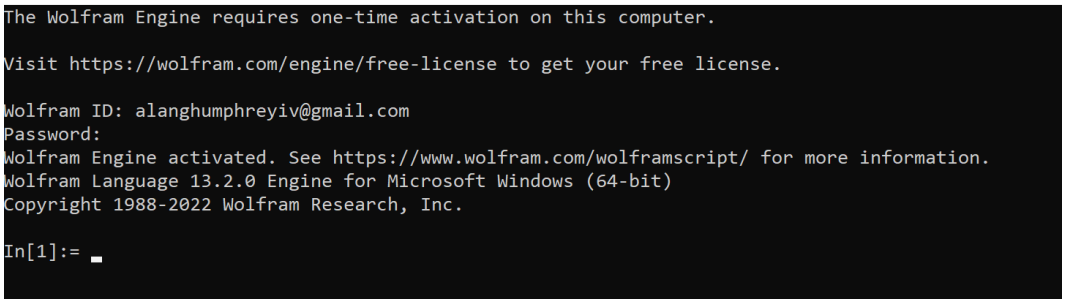
4. Now open the .dmg file that was downloaded from Wolfram and let the installation run.



5. Open the terminal and type

wolframscript

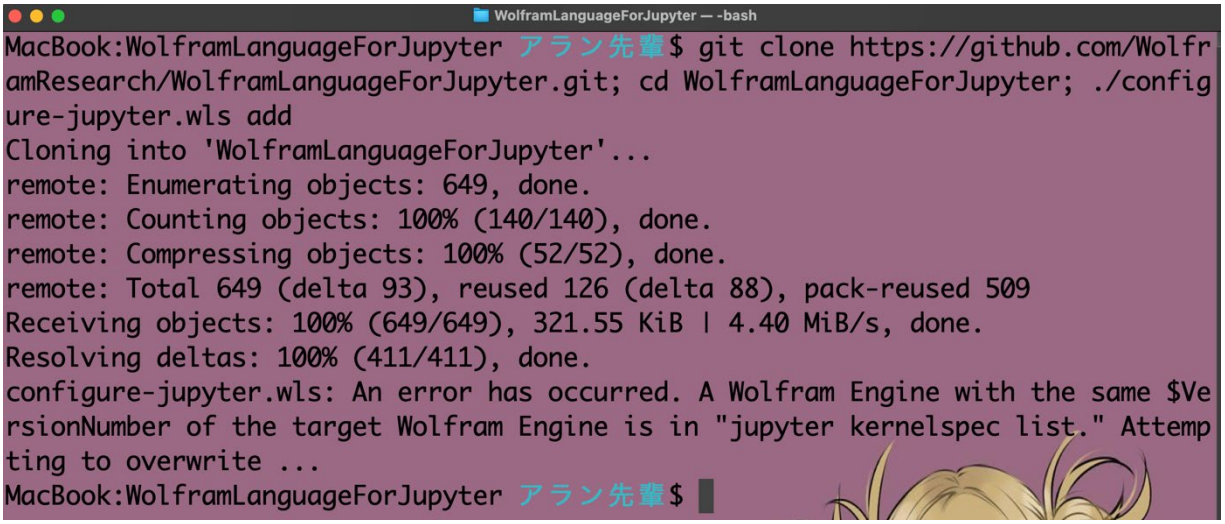
6. Now activate your free license by entering in your Wolfram ID and password, which are the email and password you used to create a Mathematica account. Note that nothing is shown on the screen as you type your password.



Install WolframLanguageForJupyter

Open the terminal and enter this command:

```
git clone https://github.com/WolframResearch/WolframLanguageForJupyter.git; cd WolframLanguageForJupyter; ./configure-jupyter.wls add
```



Post-installation

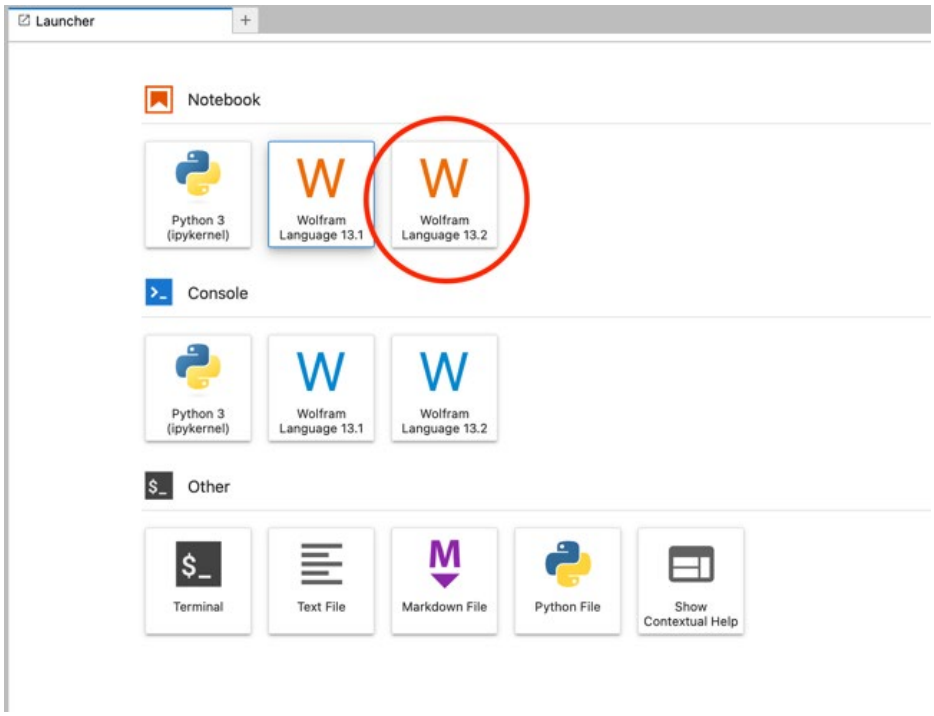
Start and terminate a session

Once the installation is complete, you can open JupyterLab by going to the terminal and enter the command below:

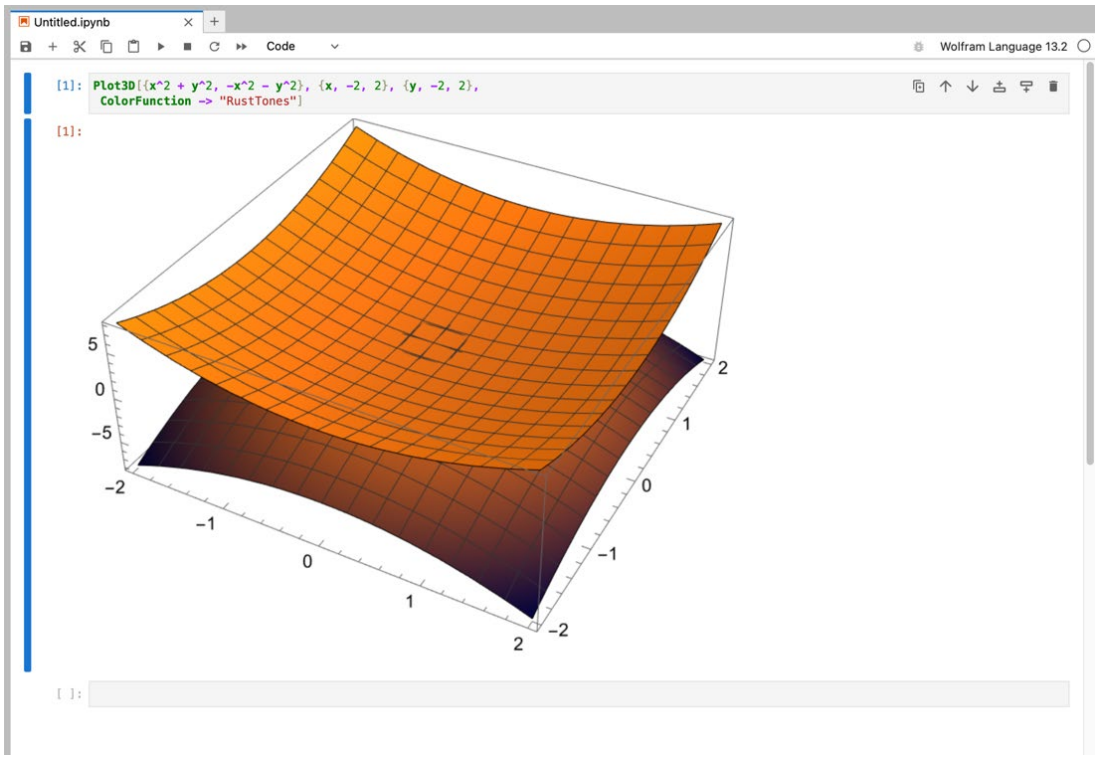
```
jupyter-lab
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS Microsoft.PowerShell.Core\FileSystem::\\Mac\Home\Downloads\WolframLanguageForJupyter-master\WolframLanguageForJupyter
-master> ./configure-jupyter.wls add
PS Microsoft.PowerShell.Core\FileSystem::\\Mac\Home\Downloads\WolframLanguageForJupyter-master\WolframLanguageForJupyter
-master> jupyter-lab
[I 2023-01-11 23:30:12.888 ServerApp] jupyter_server_terminals | extension was successfully linked.
[I 2023-01-11 23:30:12.888 ServerApp] jupyterlab | extension was successfully linked.
[I 2023-01-11 23:30:12.904 ServerApp] nbclassic | extension was successfully linked.
[I 2023-01-11 23:30:12.904 ServerApp] Writing Jupyter server cookie secret to C:\Users\alanhumphreyiv\AppData\Roaming\ju
pyter\runtime\jupyter_cookie_secret
[I 2023-01-11 23:30:13.325 ServerApp] notebook_shim | extension was successfully linked.
[I 2023-01-11 23:30:13.747 ServerApp] notebook_shim | extension was successfully loaded.
[I 2023-01-11 23:30:13.747 ServerApp] jupyter_server_terminals | extension was successfully loaded.
[I 2023-01-11 23:30:13.747 LabApp] JupyterLab extension loaded from C:\Users\alanhumphreyiv\AppData\Local\Programs\Pytho
n\Python311\Lib\site-packages\jupyterlab
[I 2023-01-11 23:30:13.747 LabApp] JupyterLab application directory is C:\Users\alanhumphreyiv\AppData\Local\Programs\Py
```

You will see terminal running and then opening a webpage for JupyterLab. From here, click on the Wolfram Language notebook to start a session.



Anything you type in the Jupyter notebook is automatically saved.



It is important to know that closing the browser tab does not terminate the JupyterLab session. You cannot start a new session if the old session is still open in the terminal. To fully exit JupyterLab, press the combination Ctrl + C (or Command + C for Mac) inside the terminal. Sometimes, you might need to press the combination a few times to get the terminal to respond.

Create shortcut icon (optional)

If you want to have a JupyterLab desktop icon to click on (instead of going to the terminal to type jupyter-lab) every time you want to start it, you can do so as follows (on Windows).

1) Open a Text Editor. This could be Notepad, Notepad++, …
2) Type jupyter-lab and save the file as a .bat file. Make sure the file's name is something like filename.bat and not filename.txt.bat.
3) Put this file in a safe folder.
4) Right click on the file in its folder, select Sent to → Desktop. That will create a shortcut of this file on the Desktop. Of course, you can change this shortcut's icon to anything you wish.

Upgrade Wolfram Engine (optional)

From time to time, Wolfram will release a new version of Mathematica. That is when a new version of Wolfram Engine is available on https://www.wolfram.com/engine/. You can download and install the new engine. (If you have Wolfram Mathematica software on your computer and already upgraded it, then you don't need to download the new engine.) Then repeat the last step of the installation guide, which is as follows:

On Windows: (the first 3 steps might not be necessary)

1) Go to the folder WolframLanguageForJupyter-master (which already exists on your computer as in the installation guide above)

2) On the address line, type powershell
3) Then run the command ./configure-jupyter.wls add
4) Go to the command line (Run cmd) and type jupyter kernelspec list. You will see a list of kernels available on JupyterLab.
5) If the latest version of Mathematica is not found in the list, then follow the path where the older version is kept. It might look something like:
   C:\Users\pntua\AppData\Roaming\jupyter\kernels\wolframlanguage13.1
6) Then change the folder's name to the updated version, such as wolframlanguage13.3.
7) Open this folder and then open the file kernel.json. Change all the old names (ending with 13.1) to new names (ending with 13.3). Check on your local computer to make sure that all the paths in this file are valid.

On Mac: go to the terminal and enter the command

cd WolframLanguageForJupyter; ./configure-jupyter.wls add

Overcoming some challenges

Using Mathematica on JupyterLab is not without challenges. Below are some common issues and easy ways to overcome them.

Copy and Paste

Some outputs on JupyterLab may not be copiable by Ctrl+C or right click. For example, the command

Expand[(1+x)^10]

returns an output as an image of the expanded polynomial. To copy it (for example, in order to use it as an input in the next command), enter the command CopyToClipboard[%] or change the original command to CopyToClipboard[Expand[(1+x)^10]]. The percentage symbol '%' means the last output. Now you can paste the output to anywhere you want.

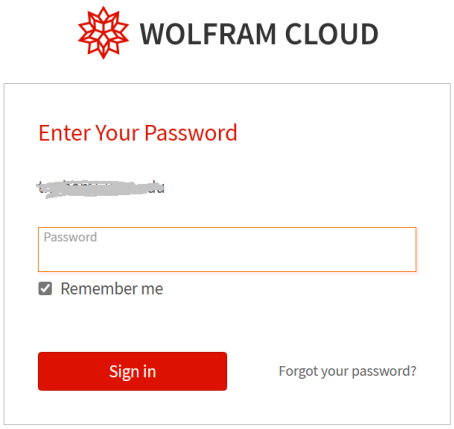
Export a report

When you export/print your JupyterLab as a pdf file, all LaTeX formula you type in your report will be rendered as raw LaTeX code. To overcome this issue, you first export your notebook to an HTML file by going to *File* → *Save and Export Notebook As* → *HTML*. Then open the HTML file that has been created and print that webpage into a pdf file. Its quality is just as good as the HTML file.

Interact with graphs

On JupyterLab, in order to use the interactive features of Mathematica, such as dragging or rotating 3D graphs with your mouse, do the simple one-time procedure below:

1) First, make sure that you are signed in on Wolfram Cloud by going to https://www.wolframcloud.com. If you are not signed in yet, then sign in with your Wolfram ID (email address) and password. Check the box "Remember me" so that you don't have to do it again in the future.



2) Second, make sure that your JupyterLab is connected to the same Wolfram account you are signed in with on the web browser. This is done by executing the following command in JupyterLab:

CloudConnect[“your-WolframID”, “your-password”]

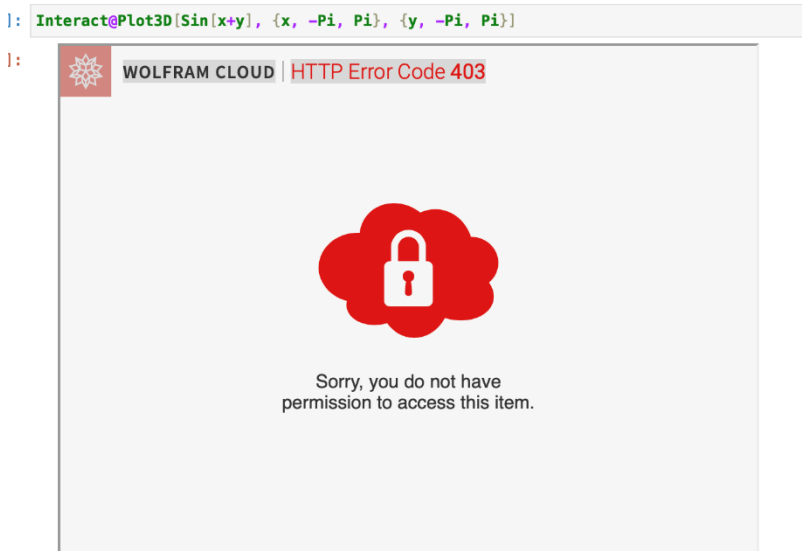
From now on, you can use any command that has interactive features in Mathematica by adding the prefix **Interact@** to that command. For example, the command

Interact@Plot3D[Sin[x+y], {x, -Pi, Pi}, {y, -Pi, Pi}]

will let you drag and rotate the graph. Without the prefix **Interact@**, the graph is still shown on Jupyter notebook but just not interactable.

In fact, the prefix **Interact@** can be added to *any command*. Every time you use it, you are asking the Wolfram server to do the computation for you. And that counts toward the monthly credit limit on your Wolfram account. For the basic (free) plan, you have about 8 minutes of computation time. Only use this prefix when you need to animate graphics. Any command without **Interact@** is executed by your local computer and does not count toward your monthly credit limit.

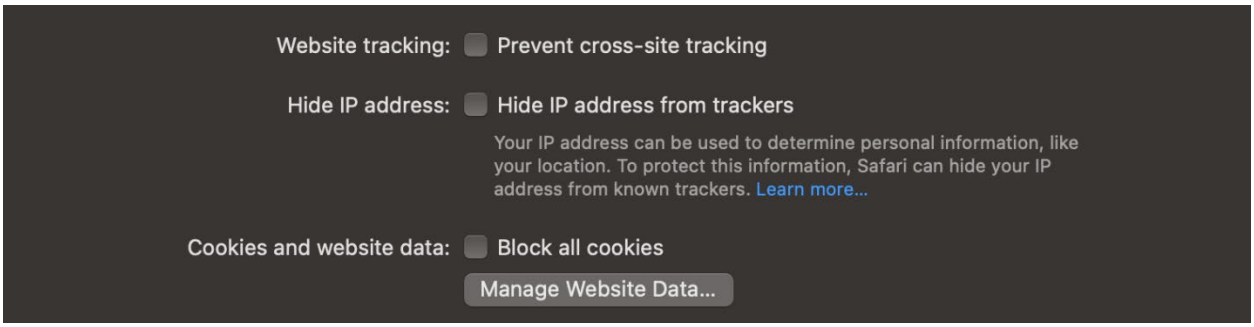
Note that if the Wolfram kernel of your JupyterLab is not connected to the same Wolfram ID as the one you use to log in on wolframcloud.com then the prefix **Interact@** will give you an “HTTP Error Code 403” as in the picture below.



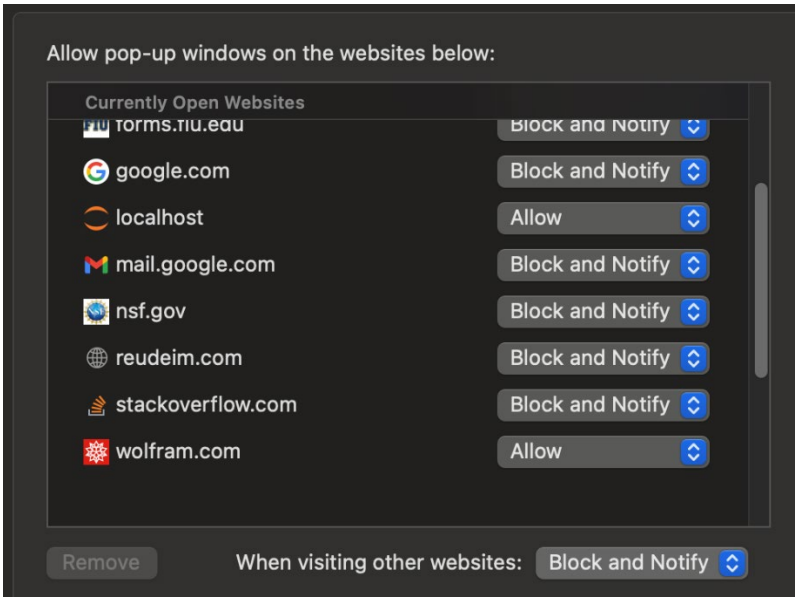
Safari: kernel not connecting or Interact@ showing nothing

On Safari (MacOS), you may encounter the issue that either the kernel does not load or **Interact@** shows a blank screen. You can fix this issue by following the below steps:

1) Under Privacy tab in Safari settings, uncheck the boxes.



2) Under Websites tab in Safari settings, make sure to allow pop-up windows for localhost and wolfram.com.



Safari may change the Setting back to default after several uses. If the problem with loading occurs again in the future, please follow the above instructions again to adjust the Setting.