

Jupyter Lab

What is a Jupyter Lab?

Project Jupyter describes JupyterLab as a next generation web-based user interfaces for all products under the Jupyter ecosystem. It enables you to work seamlessly with Lab, editors and terminals in an extensible manner.

Some of the notable features of JupyterLab are discussed below:

- Code Console acts as scratchpad for running code interactively. It has full support for rich output and can be linked to a Lab kernel to log Lab activity.
- Any text file (Markdown, Python, R, LaTeX, etc.) can be run interactively in any Jupyter kernel.
- Lab cell output can be shown into its own tab, or along with the Lab, enabling simple dashboards with interactive controls backed by a kernel.
- Live editing of document reflects in other viewers such as editors or consoles. It is possible to have live preview of Markdown, Delimiter-separated Values, or Vega/Vega-Lite documents.

JupyterLab can manage many file formats (images, CSV, JSON, Markdown, PDF etc.). It also displays rich output in these formats. JupyterLab provides customizable keyboard shortcuts uses key maps from several well-known text editors.

What is the Jupyter Lab App?

As a server-client application, the Jupyter Lab App allows you to edit and run your Labs via a web browser. The application can be executed on a PC without Internet access, or it can be installed on a remote server, where you can access it through the Internet.

Advantages of a Jupyter Lab

- Lab has the ability to re-run individual code snippets, and it provides you the flexibility of modifying them before re-running.
- You can deploy a Jupyter Lab on a remote server and access it from your local web browser.
- You can insert notes and documentation to your code in a Jupyter Lab in various formats like markdown, latex, and HTML.
- Headings in a Jupyter Lab are also supported, which is particularly useful when you want to provide heading & sub-heading to signify various important parts of your codebase. It helps improve the readers' experience.
- Use a Jupyter Lab as cells.
- Ability to share a Jupyter Lab in various formats like .py, .html, .md, .pdf, etc.
- For machine learning folks, Jupyter Lab is helpful since it allows for Data Visualization. Jupyter Lab supports visualizations and enables you to render graphs and charts. These are generated from codes with the help of modules like Matplotlib, Plotly, or Bokeh.

Jupyter Lab

How to Install Jupyter Lab Using PIP

1. Make sure that you have the latest versions of Python and PIP.

```
python3 --version
```

```
Python 3.8.10
```

```
pip3 --version
```

```
pip 20.0.2 from /usr/lib/python3/dist-packages/pip (python 3.8)
```

2. From the above output, you can see that we have Python3 version 3.8.10 and pip version 20.2.2. You upgrade to the latest version by typing:

```
sudo apt update && sudo apt upgrade python3
```

```
python3 -m pip install -U pip
```

3. Once you have pip, you can just run the below command to install Jupyter Lab.

```
sudo apt install nodejs -y
```

```
pip3 install jupyterlab
```

4. You may need to add the user-level bin directory to your PATH environment variable in order to launch Jupyter Lab. You can achieve this by typing:

```
export PATH="$HOME/.local/bin:$PATH"
```

5. First, let us generate a configuration file for Jupyter Lab:

```
jupyter-lab --generate-config
```

```
Writing default config to: /home/user/.jupyter/jupyter_lab_config.py
```

6. Set a password for accessing your Jupyter Lab instance remotely:

```
jupyter-lab password
```

```
Enter password:
```

```
Verify password:
```

```
[JupyterPasswordApp] Wrote hashed password to /home/user/.jupyter/jupyter_server_config.json
```

Jupyter Lab

Launching Jupyter Lab

1. To launch your Jupyter Lab, simply type the below command!

jupyter lab

```
[I 2021-10-11 15:00:48.627 ServerApp] jupyterlab | extension was successfully linked.  
[I 2021-10-11 15:00:48.696 LabApp] JupyterLab extension loaded from /home/cisco/.local/lib/python3.8/site-packages/jupyterlab  
[I 2021-10-11 15:00:48.697 LabApp] JupyterLab application directory is /home/cisco/.local/share/jupyter/lab  
[I 2021-10-11 15:00:48.701 ServerApp] jupyterlab | extension was successfully loaded.  
[I 2021-10-11 15:00:48.702 ServerApp] Serving notebooks from local directory: /mnt/c/Users/scott  
[I 2021-10-11 15:00:48.702 ServerApp] Jupyter Server 1.11.1 is running at:  
[I 2021-10-11 15:00:48.703 ServerApp] http://localhost:8888/lab  
[I 2021-10-11 15:00:48.703 ServerApp] or http://127.0.0.1:8888/lab  
[I 2021-10-11 15:00:48.703 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
```

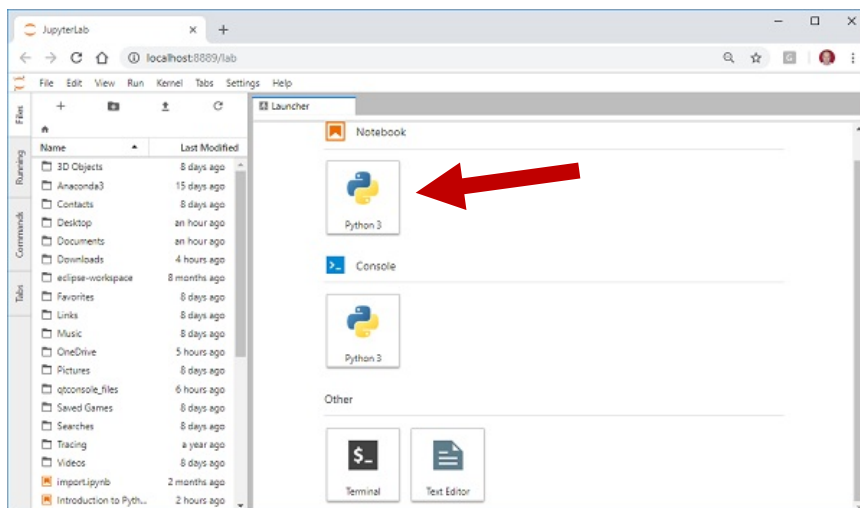
2. Once you do this, the Jupyter Lab should open up the below tab for you inside your web browser:

Note that the Jupyter Lab will open using this URL:

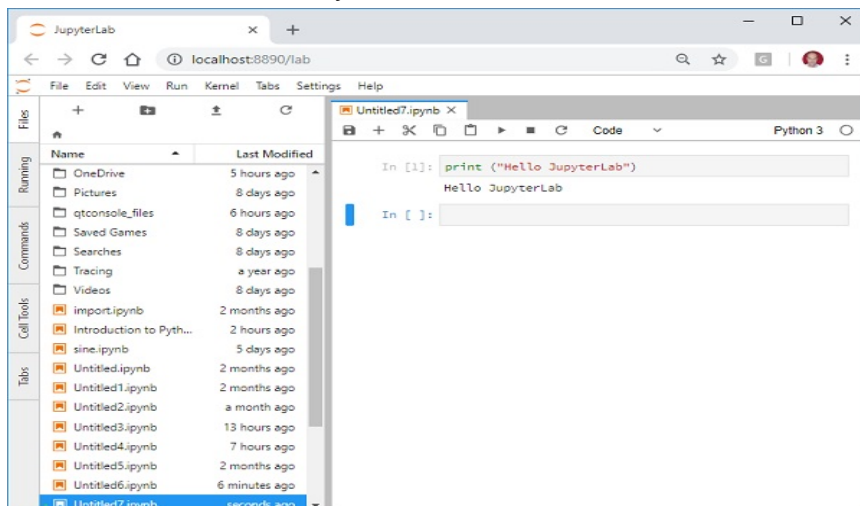
<http://localhost:8888/lab>

However, the port number 8888 might change if a process is using it, or a second Jupyter Lab session runs in the background.

In some cases, the Jupyter Lab might not open up on its own, in which case you can simply copy one of the URLs that are generated when you run the command jupyter-lab.



3. To start a new Lab, click the desired kernel. In the above screenshot, the first kernel is **Python3** kernel. Click it to start a Python Lab.



Jupyter Lab

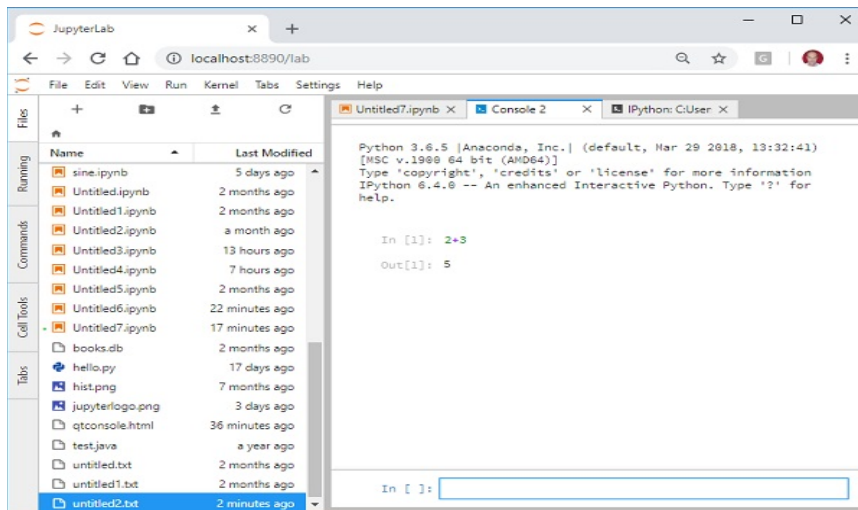
Menu Bar

The menu bar at the top of window has these default menus:

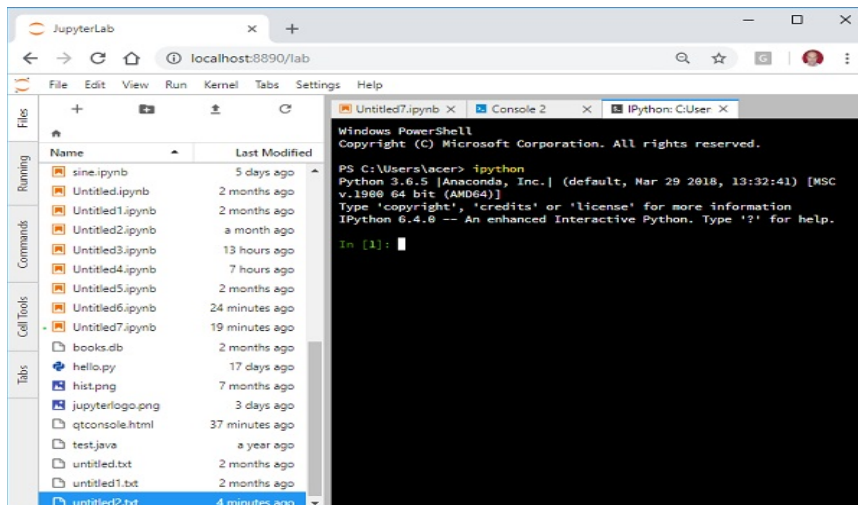
- **File** – Actions related to files and directories.
- **Edit** – Actions related to editing documents and other activities.
- **View** – Actions that alter the appearance of JupyterLab.
- **Run** – Actions for running code in different activities such as Labs and code consoles.
- **Kernel** – Actions for managing kernels, which are separate processes for running code.
- **Tabs** – A list of the open documents and activities in the dock panel.
- **Settings** – Common settings and an advanced settings editor.
- **Help** – A list of JupyterLab and kernel help links.

The left sidebar shows buttons for starting a new launcher, adding a folder, uploading file and refresh file list. The right pane is the main working area where Lab, console and terminals are shown in tabbed view.

To start a new console, click **+** symbol in the left side bar to open a new launcher and then click the console option. The console will open in new tab on the right pane.



Note: The input cell is at the bottom, but when it is run, the cell and its corresponding output cell appears in upper part of console tab.



Jupyter Lab

Adding Heading and Notes in Jupyter Lab

In the drop-down menu shown below, you have four options:

- **Code** – In the code cell, you write the Python code.
- **Markdown** – In this cell type, you can add headings, notes, documentation, etc.
- **Raw NBConvert** – This cell type can be used to render different code formats into HTML or LaTeX. This information is stored in the Lab metadata and converted appropriately.

4. Use a **markdown** example for adding headings and bullet points:



5. Click the Run icon to see the results:



Sharing a Jupyter Lab

This is one of the great and cool features of a Jupyter Lab. Go to the **Files** menu, and you will see an **Export Notebook As** option where you can download the Jupyter Lab in nine different ways.

You can also use **Download** to download the entire **.ipynb** or **.py** file:



Installing a Python Module in Jupyter Lab

Installing a Python module inside Jupyter Lab is as easy as installing it from the terminal. All you need to do is use an **!** (exclamation) mark at the start of the command:

