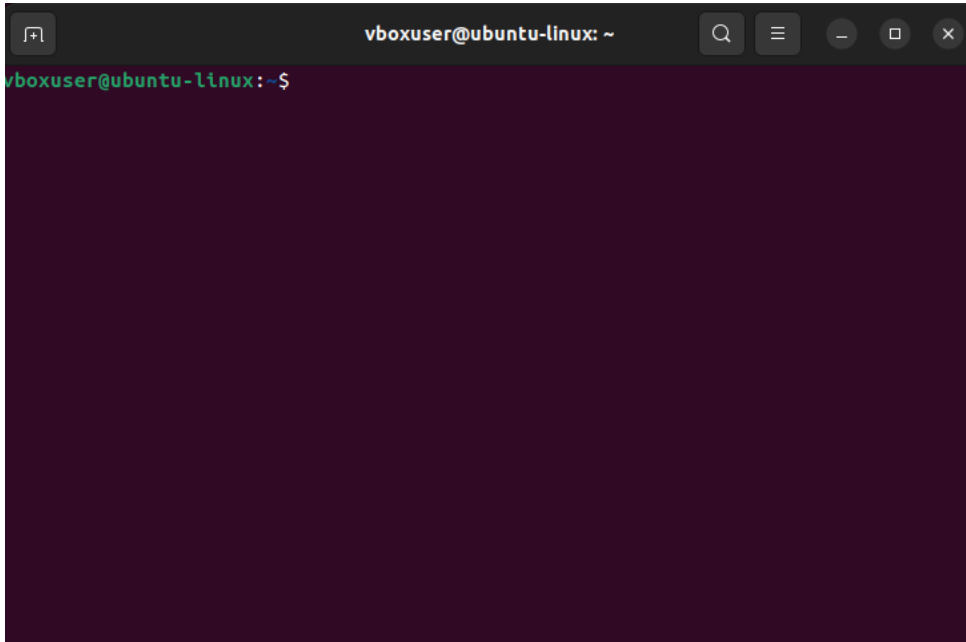


Miniconda for Ubuntu Linux Python Geospatial Installation

Miniconda Install	2
Update conda	4
Installing <i>JupyterLab</i>	5
Create <i>geospatial</i> environment (installing packages required for the <i>Introduction to Geospatial Raster and Vector Data with Python</i> Lesson)	6
Activating the <i>geospatial</i> environment and launching <i>JupyterLab</i>	8
Installing code and data files for the <i>Programming with Python</i> Lesson	9
Launching the <i>Programming with Python or Introduction to Geospatial Raster and Vector Data with Python</i> Lesson Software Carpentry lessons	11

Miniconda Install

1. Miniconda download: <https://docs.conda.io/en/latest/miniconda.html>
2. Miniconda3 Linux 64-bit from the Latest Miniconda Installer Links section: https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
3. Bring up a terminal window using the shortcut, `Ctrl-Alt-T`:



4. In the terminal, change to the Downloads directory by typing:
`cd ~/Downloads`
5. Run the Miniconda install script by typing:
`bash Miniconda3-latest-Linux-x86_64.sh`
6. Press `ENTER` to continue.
7. Read through the license agreement and accept by typing `'yes'`
8. Press `ENTER` to confirm the location.
9. Type `'yes'` to initialize your Miniconda3 sessions by running the `'conda init'` command.
10. Close the terminal window and open a new terminal using `Ctrl-Alt-T`.

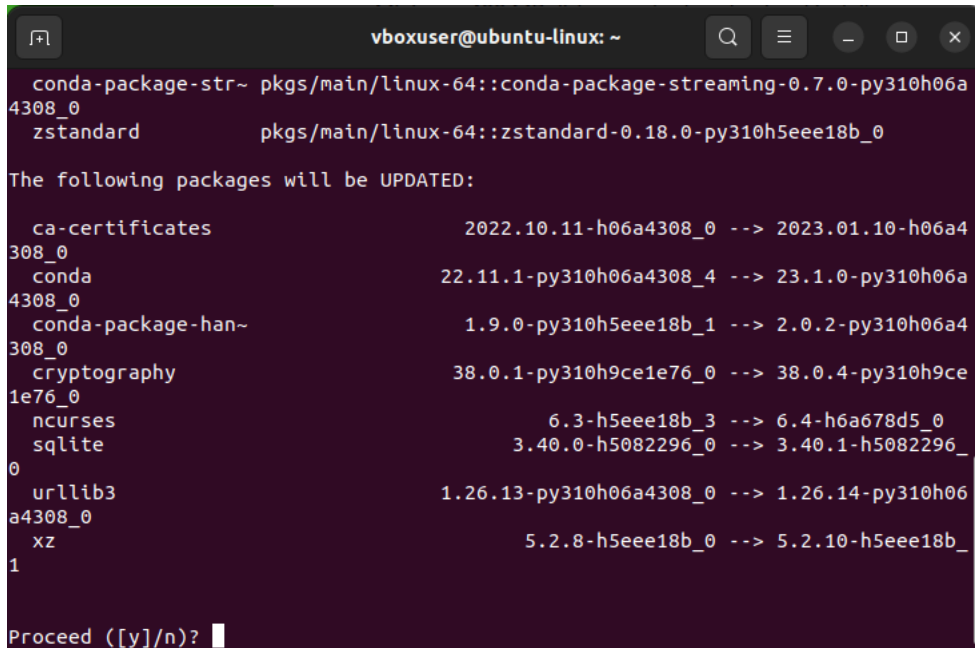
11. Initialize conda by typing:

```
conda init
```

12. Notice that the terminal prompt has changed to prepended **(base)**.

Update conda

1. At the terminal prompt, type:
`conda update conda`
2. The `conda update` command will identify packages for conda that need to be updated. Type 'y' to proceed with the package updates.

A terminal window titled 'vboxuser@ubuntu-linux: ~' showing the output of the 'conda update conda' command. The terminal lists several packages to be updated, including ca-certificates, conda, conda-package-handling, cryptography, ncurses, sqlite, urllib3, and xz, along with their current and target versions. The prompt 'Proceed ([y]/n)?' is visible at the bottom.

```
vboxuser@ubuntu-linux: ~
conda-package-str~ pkgs/main/linux-64::conda-package-streaming-0.7.0-py310h06a
4308_0
zstandard          pkgs/main/linux-64::zstandard-0.18.0-py310h5eee18b_0

The following packages will be UPDATED:

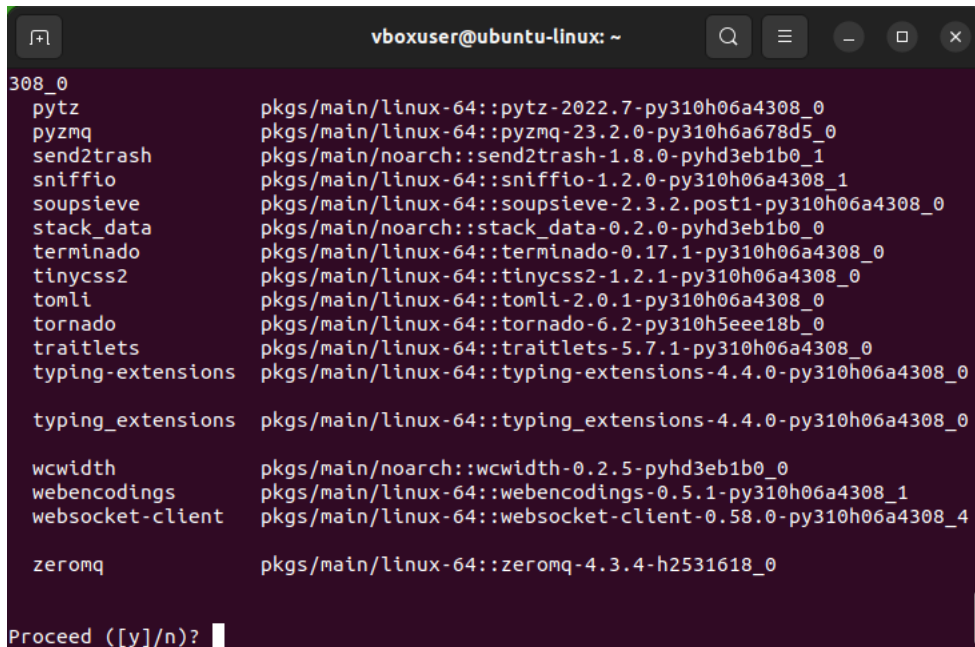
  ca-certificates                2022.10.11-h06a4308_0 --> 2023.01.10-h06a4
308_0
  conda                         22.11.1-py310h06a4308_4 --> 23.1.0-py310h06a
4308_0
  conda-package-handling        1.9.0-py310h5eee18b_1 --> 2.0.2-py310h06a4
308_0
  cryptography                   38.0.1-py310h9ce1e76_0 --> 38.0.4-py310h9ce
1e76_0
  ncurses                       6.3-h5eee18b_3 --> 6.4-h6a678d5_0
  sqlite                        3.40.0-h5082296_0 --> 3.40.1-h5082296_
0
  urllib3                       1.26.13-py310h06a4308_0 --> 1.26.14-py310h06
a4308_0
  xz                            5.2.8-h5eee18b_0 --> 5.2.10-h5eee18b_
1

Proceed ([y]/n)?
```

3. Type 'y' to proceed.

Installing *JupyterLab*

1. In the Terminal prompt, type:
`conda install jupyterlab`
2. Type 'y' to proceed with the *JupyterLab* installation.



A terminal window titled 'vboxuser@ubuntu-linux: ~' showing the output of the command 'conda install jupyterlab'. The output lists various packages to be installed, including pytz, pyzmq, send2trash, sniffio, soupsieve, stack_data, terminado, tinycss2, tomli, tornado, traitlets, typing-extensions, typing_extensions, wcwidth, webencodings, websocket-client, and zeromq. Each package is followed by its source and version information. At the bottom, the prompt 'Proceed ([y]/n)?' is shown with a cursor.

```
vboxuser@ubuntu-linux: ~
308_0
pytz                pkgs/main/linux-64::pytz-2022.7-py310h06a4308_0
pyzmq               pkgs/main/linux-64::pyzmq-23.2.0-py310h6a678d5_0
send2trash          pkgs/main/noarch::send2trash-1.8.0-pyhd3eb1b0_1
sniffio             pkgs/main/linux-64::sniffio-1.2.0-py310h06a4308_1
soupsieve           pkgs/main/linux-64::soupsieve-2.3.2.post1-py310h06a4308_0
stack_data          pkgs/main/noarch::stack_data-0.2.0-pyhd3eb1b0_0
terminado           pkgs/main/linux-64::terminado-0.17.1-py310h06a4308_0
tinycss2            pkgs/main/linux-64::tinycss2-1.2.1-py310h06a4308_0
tomli               pkgs/main/linux-64::tomli-2.0.1-py310h06a4308_0
tornado             pkgs/main/linux-64::tornado-6.2-py310h5eee18b_0
traitlets           pkgs/main/linux-64::traitlets-5.7.1-py310h06a4308_0
typing-extensions   pkgs/main/linux-64::typing-extensions-4.4.0-py310h06a4308_0

typing_extensions   pkgs/main/linux-64::typing_extensions-4.4.0-py310h06a4308_0

wcwidth             pkgs/main/noarch::wcwidth-0.2.5-pyhd3eb1b0_0
webencodings        pkgs/main/linux-64::webencodings-0.5.1-py310h06a4308_1
websocket-client     pkgs/main/linux-64::websocket-client-0.58.0-py310h06a4308_4

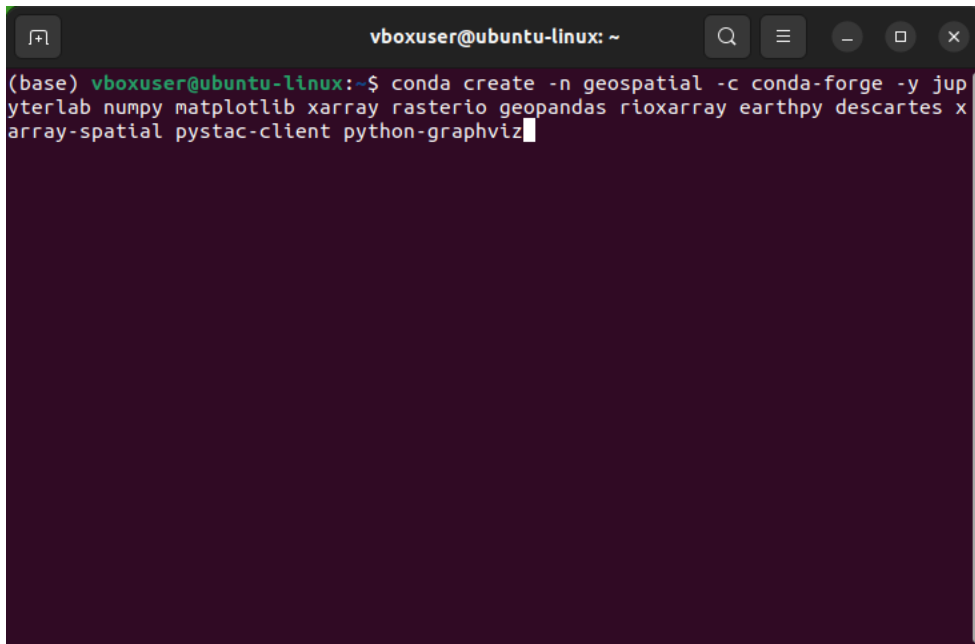
zeromq              pkgs/main/linux-64::zeromq-4.3.4-h2531618_0

Proceed ([y]/n)?
```

Create *geospatial* environment (installing packages required for the *Introduction to Geospatial Raster and Vector Data with Python* Lesson)

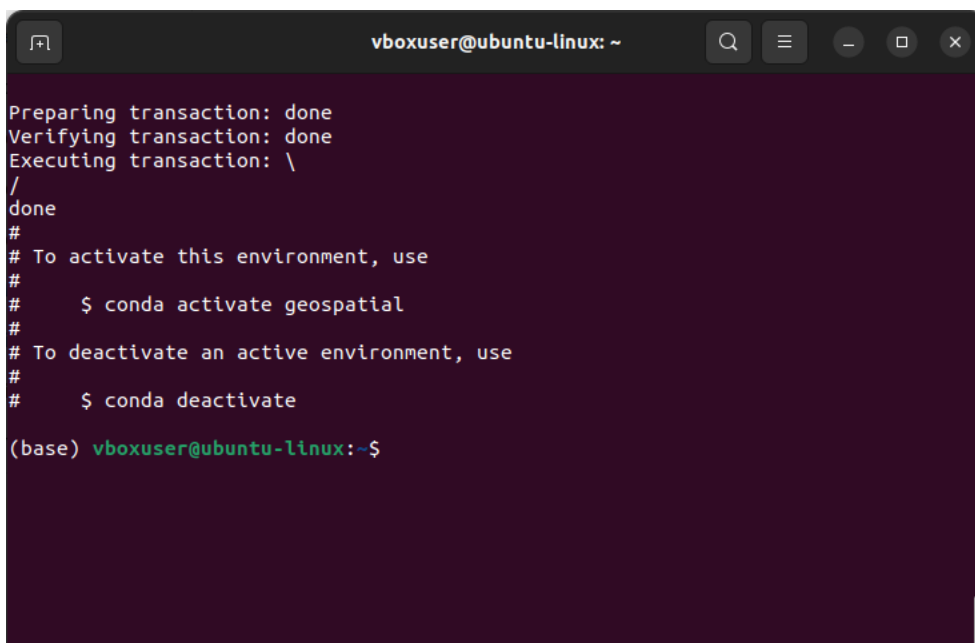
1. In the terminal prompt, type (this should be all on one line):

```
conda create -n geospatial -c conda-forge -y jupyterlab numpy  
matplotlib xarray rasterio geopandas rioxarray earthpy descartes  
xarray-spatial pystac-client python-graphviz
```

A terminal window titled 'vboxuser@ubuntu-linux: ~' with search, menu, and window control icons in the title bar. The terminal shows the command: `(base) vboxuser@ubuntu-linux:~$ conda create -n geospatial -c conda-forge -y jupyterlab numpy matplotlib xarray rasterio geopandas rioxarray earthpy descartes xarray-spatial pystac-client python-graphviz` followed by a cursor. The terminal background is dark purple.

```
(base) vboxuser@ubuntu-linux:~$ conda create -n geospatial -c conda-forge -y jup  
yterlab numpy matplotlib xarray rasterio geopandas rioxarray earthpy descartes x  
array-spatial pystac-client python-graphviz
```

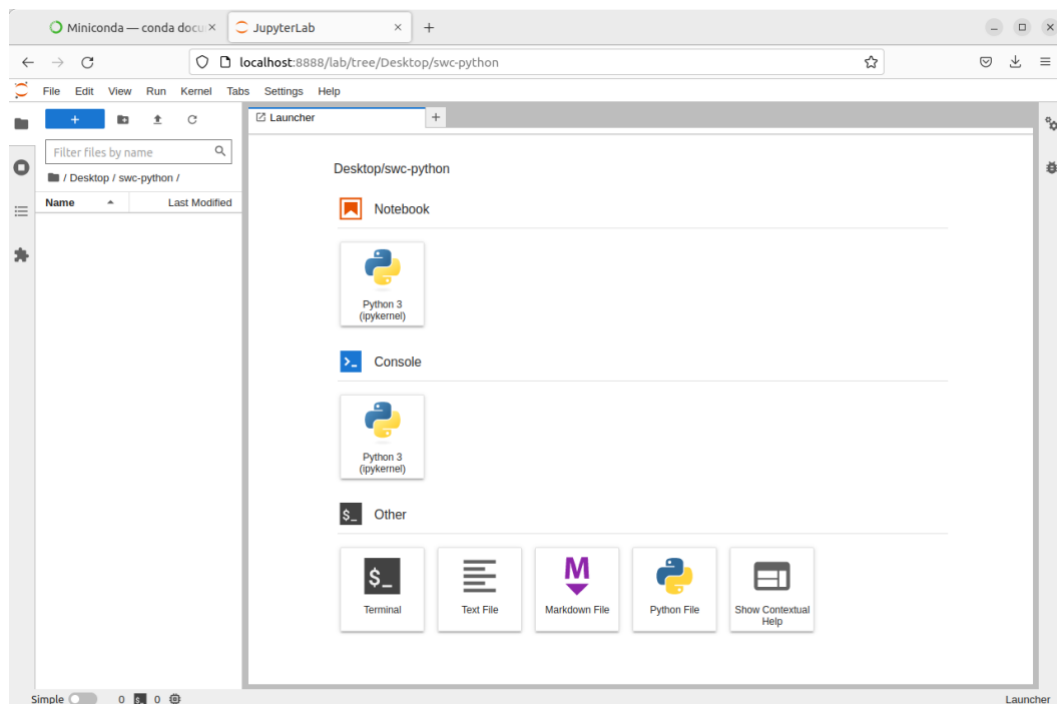
2. The conda will evaluate the current environment and determine which packages need to be installed. *This may take more than a few minutes.*
3. After installation, you will be presented with instructions on how to activate and deactivate the *geospatial* environment:



```
vboxuser@ubuntu-linux: ~  
Preparing transaction: done  
Verifying transaction: done  
Executing transaction: \  
/  
done  
#  
# To activate this environment, use  
#  
#   $ conda activate geospatial  
#  
# To deactivate an active environment, use  
#  
#   $ conda deactivate  
  
(base) vboxuser@ubuntu-linux:~$
```

Activating the *geospatial* environment and launching *JupyterLab*

1. In the Terminal prompt, type:
`conda activate geospatial`
2. You will be returned to the Terminal prompt, but notice that the prompt is prepended with **(geospatial)**
3. Create the configuration file from the terminal by typing:
`jupyter notebook --generate-config`
4. Edit the configuration file `~/.jupyter/jupyter_notebook_config.py` and set:
`c.NotebookApp.use_redirect_file = False`
5. Launch *JupyterLab* from the Terminal prompt by typing:
`jupyter lab`
6. This will launch *JupyterLab* in a browser window:



7. Click on the `Python 3` icon under the *Notebook* section to start an interactive *Jupyter Notebook* session.
8. You are now ready to proceed with the [Programming with Python](#) or [Introduction to Geospatial Raster and Vector Data with Python](#) Software Carpentry Lessons.

Installing code and data files for the *Programming with Python* Lesson

1. In the Terminal prompt, change the directory to your Desktop by typing:
`cd ~/Desktop`
2. Download `python-novice-inflammation-data.zip` and `python-novice-inflammation-code.zip`:

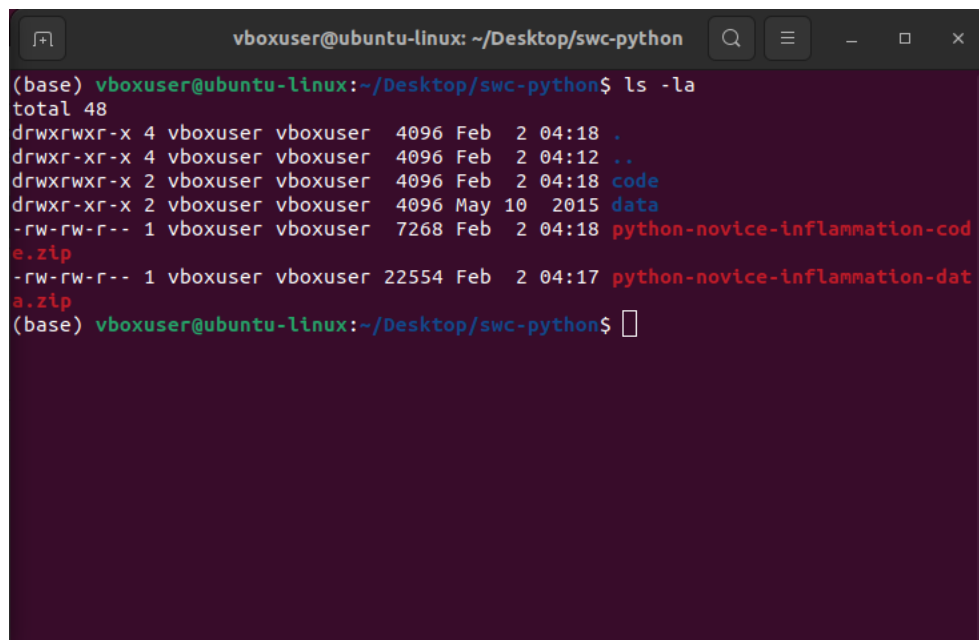
<https://swcarpentry.github.io/python-novice-inflammation/data/python-novice-inflammation-data.zip>
<https://swcarpentry.github.io/python-novice-inflammation/code/python-novice-inflammation-code.zip>

3. Create a folder called `swc-python` on your Desktop.
4. Move downloaded files to `swc-python`.

Unzip the files.

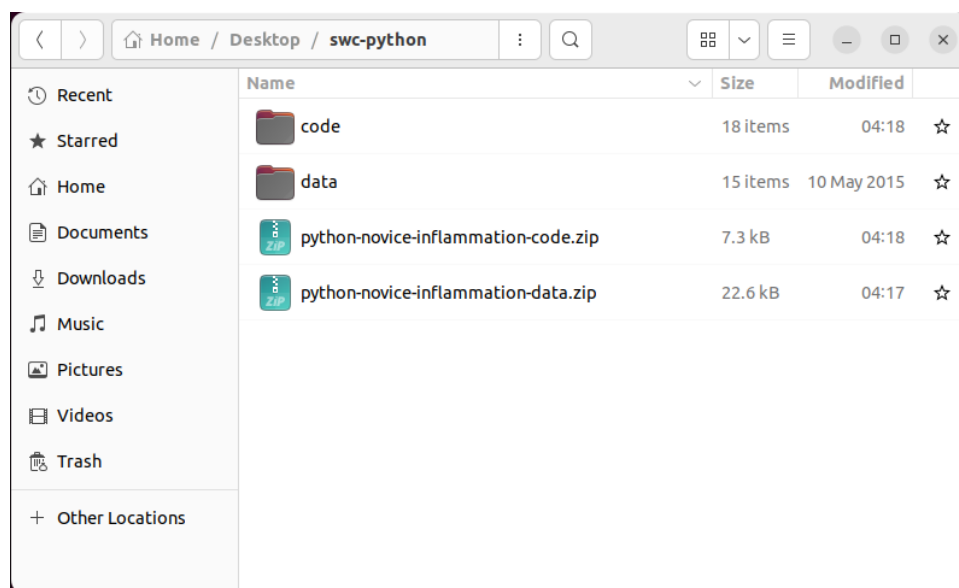
You should see two folders, `data` and `code`, in the `swc-python` directory on your Desktop.

5. In the Terminal prompt, change to the working directory by typing:
`cd swc-python`
6. In the Terminal prompt. list directory to verify directory structure is correct by typing:
`ls`

A terminal window titled 'vboxuser@ubuntu-linux: ~/Desktop/swc-python' showing the output of the 'ls -la' command. The output lists the current directory and its contents, including 'code' and 'data' folders, and two zip files: 'python-novice-inflammation-code.zip' and 'python-novice-inflammation-data.zip'.

```
(base) vboxuser@ubuntu-linux:~/Desktop/swc-python$ ls -la
total 48
drwxrwxr-x 4 vboxuser vboxuser 4096 Feb  2 04:18 .
drwxr-xr-x 4 vboxuser vboxuser 4096 Feb  2 04:12 ..
drwxrwxr-x 2 vboxuser vboxuser 4096 Feb  2 04:18 code
drwxr-xr-x 2 vboxuser vboxuser 4096 May 10 2015 data
-rw-rw-r-- 1 vboxuser vboxuser 7268 Feb  2 04:18 python-novice-inflammation-code.zip
-rw-rw-r-- 1 vboxuser vboxuser 22554 Feb  2 04:17 python-novice-inflammation-data.zip
(base) vboxuser@ubuntu-linux:~/Desktop/swc-python$
```

7. You can also verify the directory structure in an Ubuntu Files window:



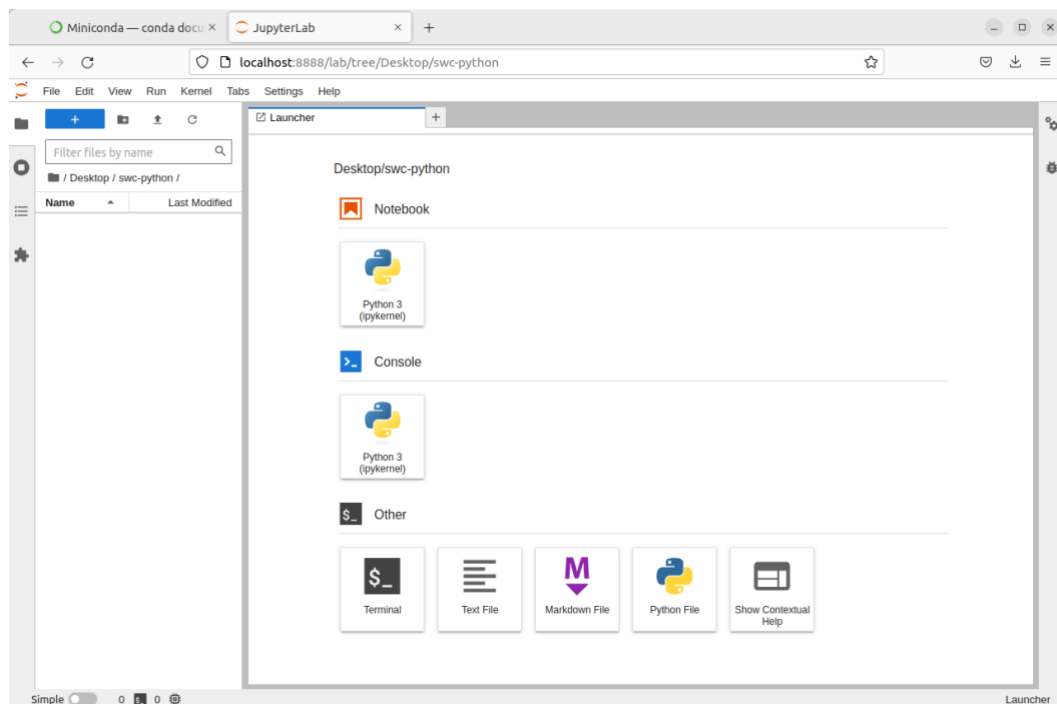
Launching the *Programming with Python or Introduction to Geospatial Raster and Vector Data with Python* Lesson Software Carpentry lessons

1. Ensure that you have downloaded the Python code and data required for this lesson. Refer to the section on [Installing code and data files for the *Programming with Python* Lesson](#) for installation instructions.
2. In the Terminal prompt, change the directory to your Desktop by typing:

```
cd ~/Desktop/swc-python
```
3. In the Terminal prompt, type:

```
conda activate geospatial
```
4. You will be returned to the Terminal prompt, but notice that the prompt is prepended with **(geospatial)**
5. Launch *JupyterLab* from the Terminal prompt by typing:

```
jupyter lab
```
6. This will launch *JupyterLab* in a browser window:



7. Click on the Python 3 icon under the *Notebook* section to start an interactive *Jupyter Notebook* session.
8. You are now ready to proceed with the [Programming with Python](#) or [Introduction to Geospatial Raster and Vector Data with Python](#) Software Carpentry Lessons.