

setting up a conda environment

Contents

- anaconda navigator
- anaconda command prompt
- notes and references

Note

This page provides instructions for creating a new **conda** environment using **Anaconda Navigator**, or using the **Anaconda Command Prompt**.

You only need to choose **ONE** of these methods for setting up the environment.

Danger

You should have already opened a [GitHub](#) account, installed [git](#) and [GitHub Desktop](#), [forked](#) the EGM722 repository, [cloned](#) the repository to your computer, and installed [Anaconda Navigator](#).

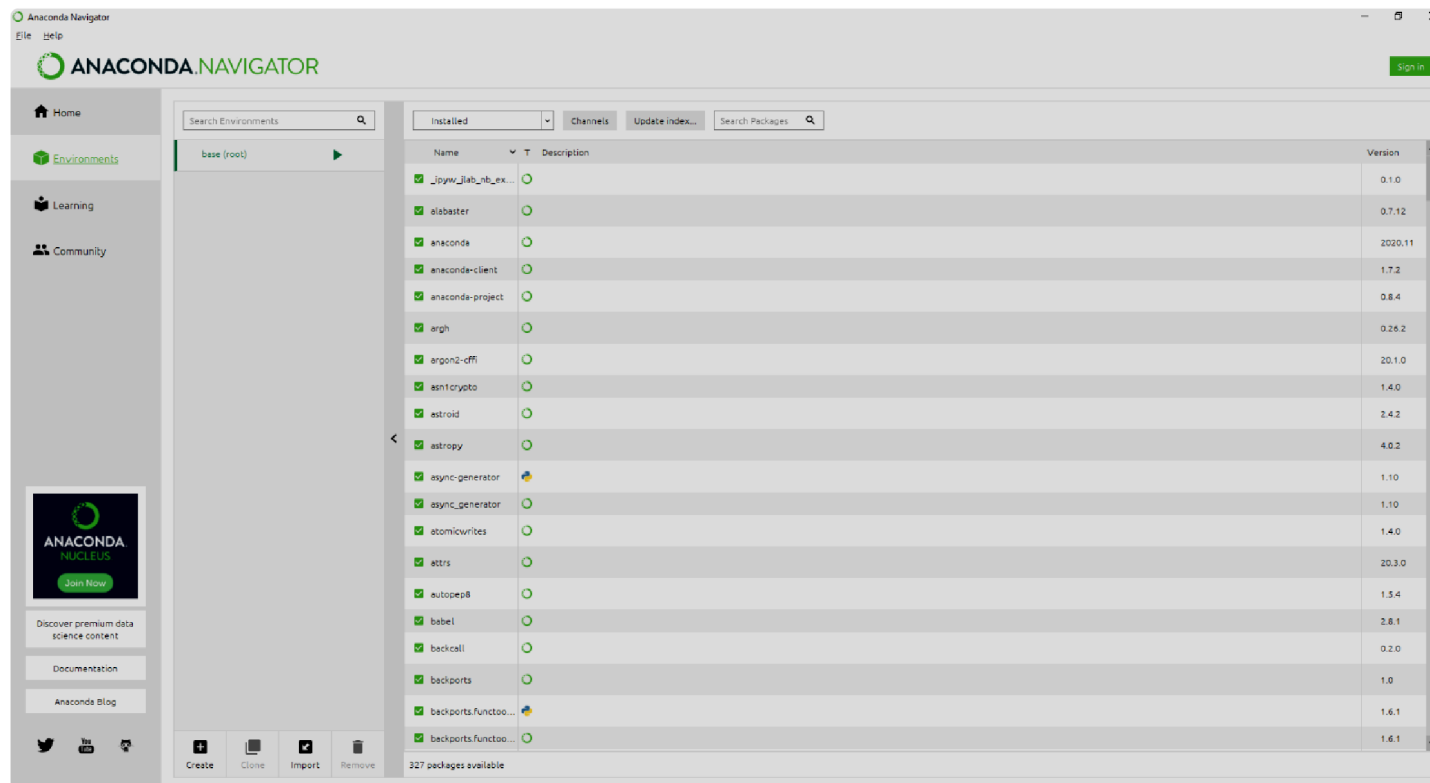
If you haven't done all of these steps, please do so now before continuing.

[Skip to main content](#)

anaconda navigator

creating an environment

From the **Start** menu, open **Anaconda Navigator**. When it opens, click on the **Environments** tab on the left-hand side of the screen. You should see the following:



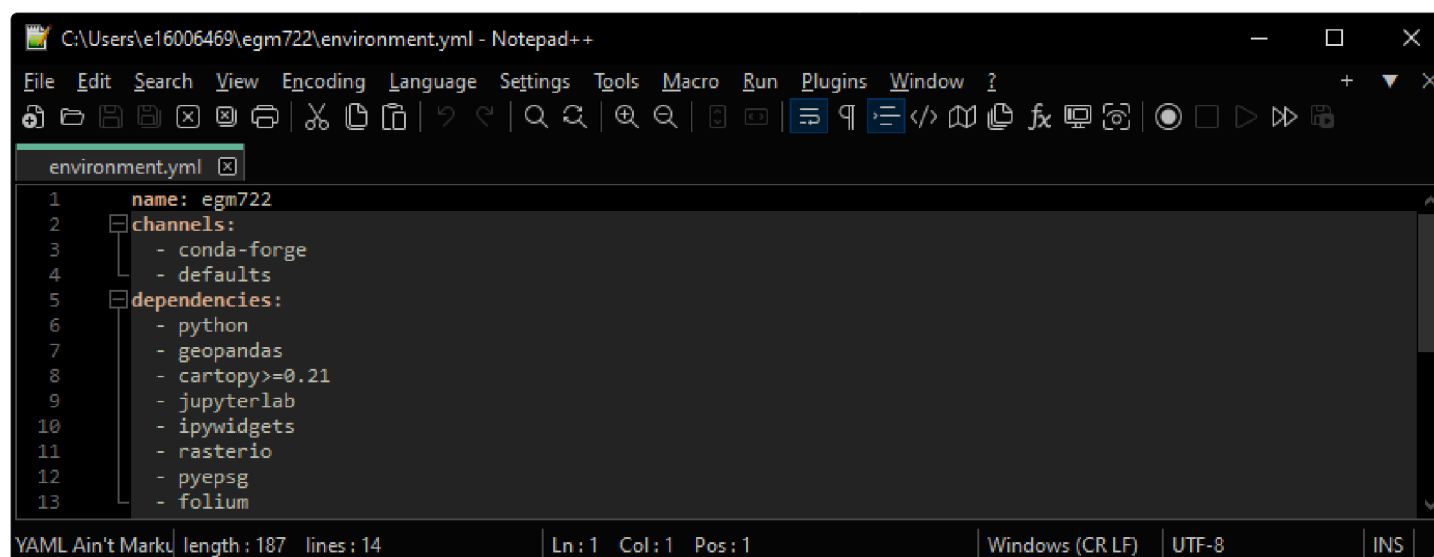
As mentioned above, **conda** is a package management system. We can use **conda** to create different programming environments, which will enable us to keep track of and manage the specific versions of python packages that we use for this

[Skip to main content](#)

module. It also enables us to easily share our environment specifications across different computers, so that we can be sure that any processing steps that we do are consistent and reproducible.

We will be creating a new environment using the **environment.yml** file provided in the git repository. A **.yml** file is one of the ways that we can easily duplicate environments using **conda** or other package management systems – it provides a list of the packages for the package management system to find and install.

Open the **environment.yml** file in a text editor (**NB – this means NotePad, Notepad++ or something similar, NOT MS Word!**). It should look something like this (the exact list of dependencies may differ somewhat):

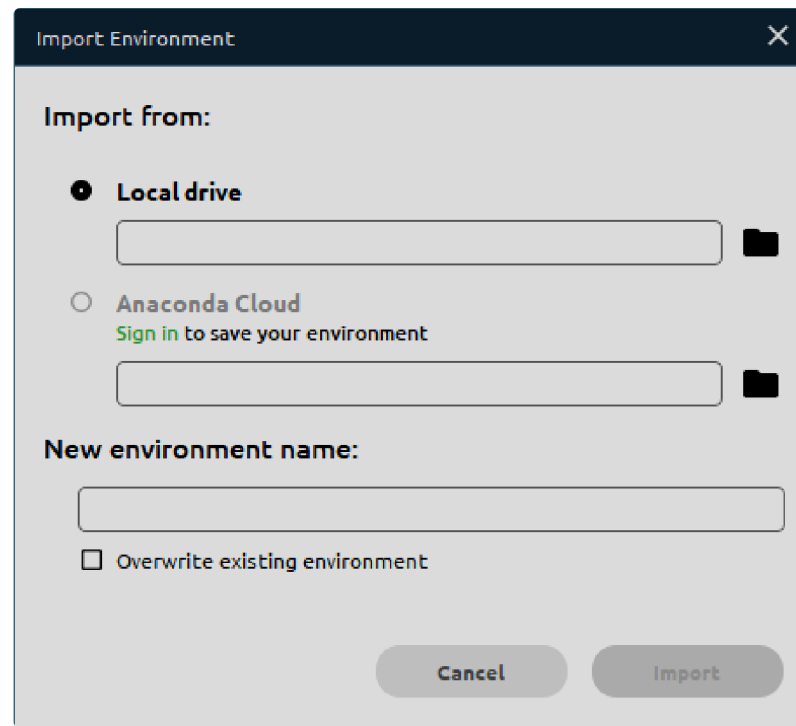


```
1  name: egm722
2  channels:
3    - conda-forge
4    - defaults
5  dependencies:
6    - python
7    - geopandas
8    - cartopy>=0.21
9    - jupyterlab
10   - ipywidgets
11   - rasterio
12   - pyepsg
13   - folium
```

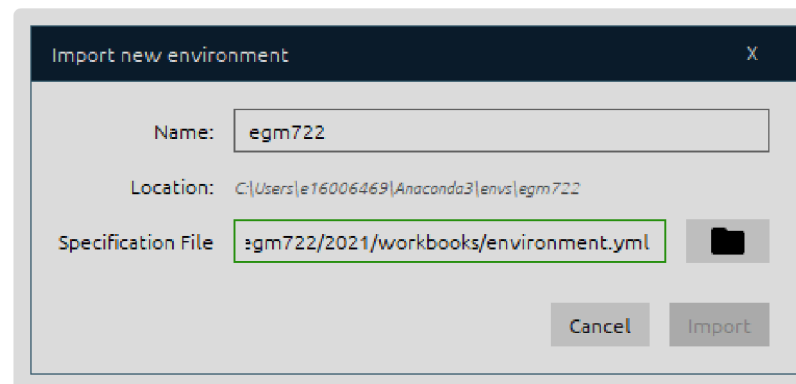
Here, you can see the name of the environment (egm722), the **channels** to install packages from, listed in order of preference, and the **dependencies**, or required packages.

From the **Anaconda Navigator** window, click on the **Import** button at the bottom of the **Environments** tab. An import window will open:

[Skip to main content](#)



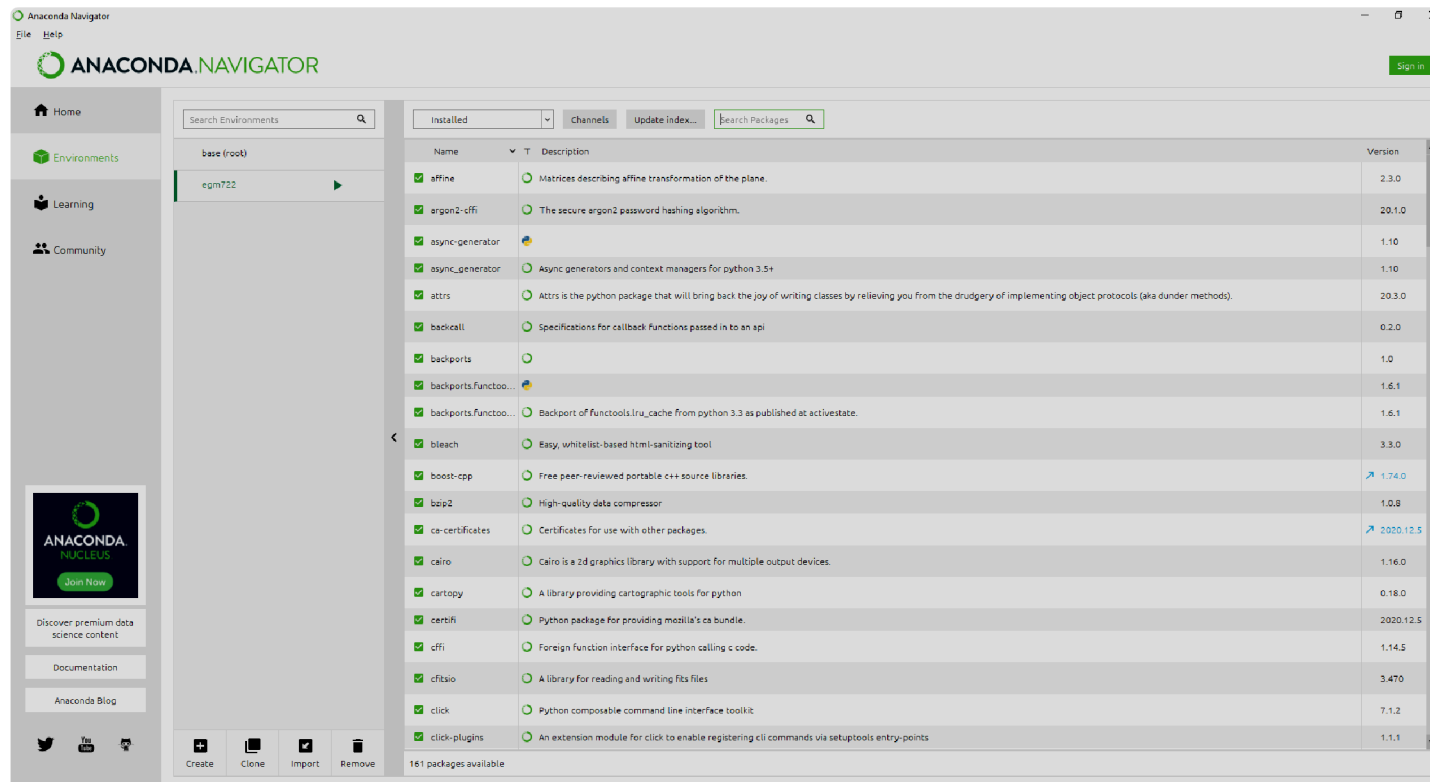
Click the folder icon to navigate to the **environment.yml** file, select it and click **Open**. The window should now look like this:



The **Name** field may not automatically populate based on the **.yml** file, so be sure to check that it is `egm722`. The **Location** will

[Skip to main content](#)

Click **Import** to start setting up the environment. Depending on your connection speed and computer's specifications, this may take some time. When it finishes, you should see the following screen:

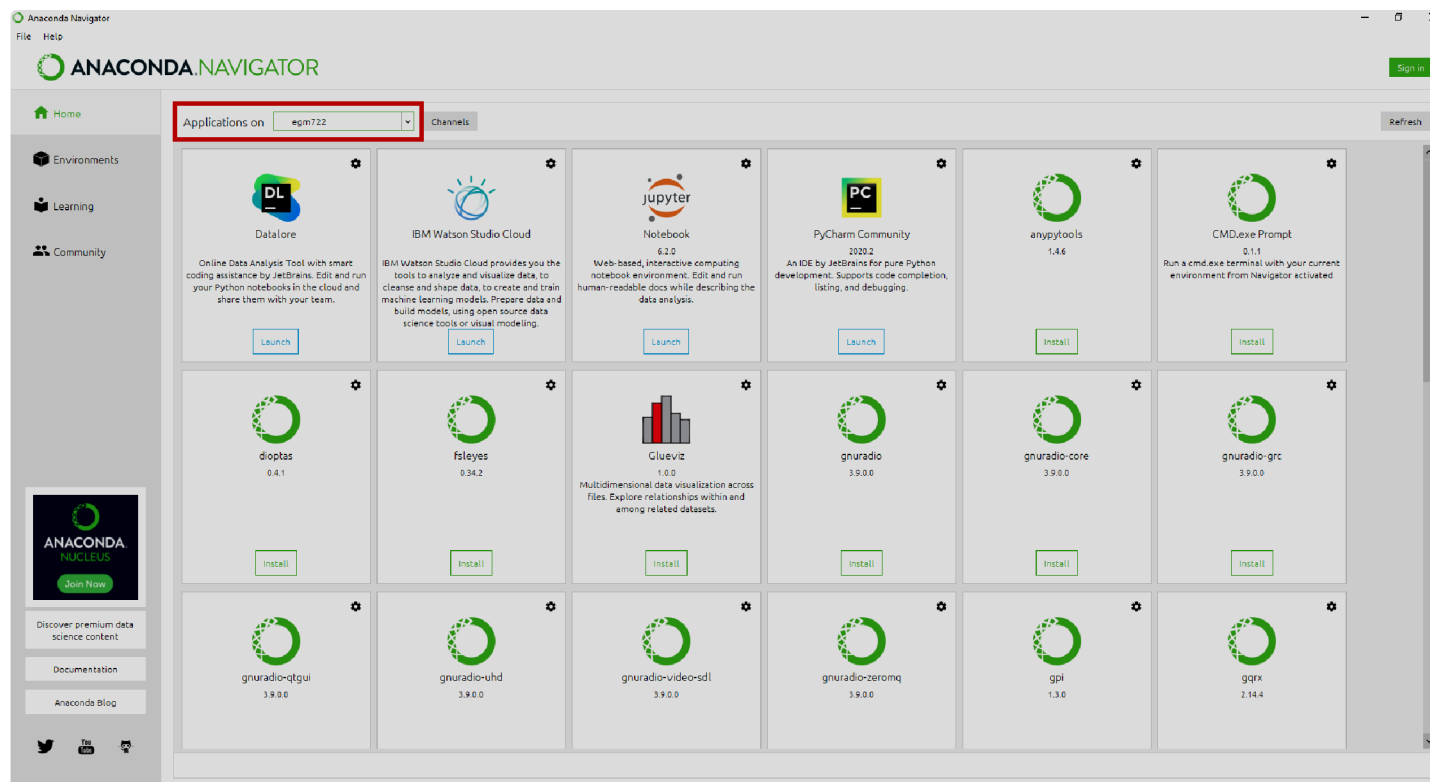


Note how many packages were actually installed – 161, from the 5 shown in the original `.yaml` file.^[1]

This is because each of those 5 packages have additional dependencies which have to be installed as well. Fortunately, almost all of this is done automatically – we don't have to worry about tracking down each individual dependency on our own.

changing environments

[Skip to main content](#)



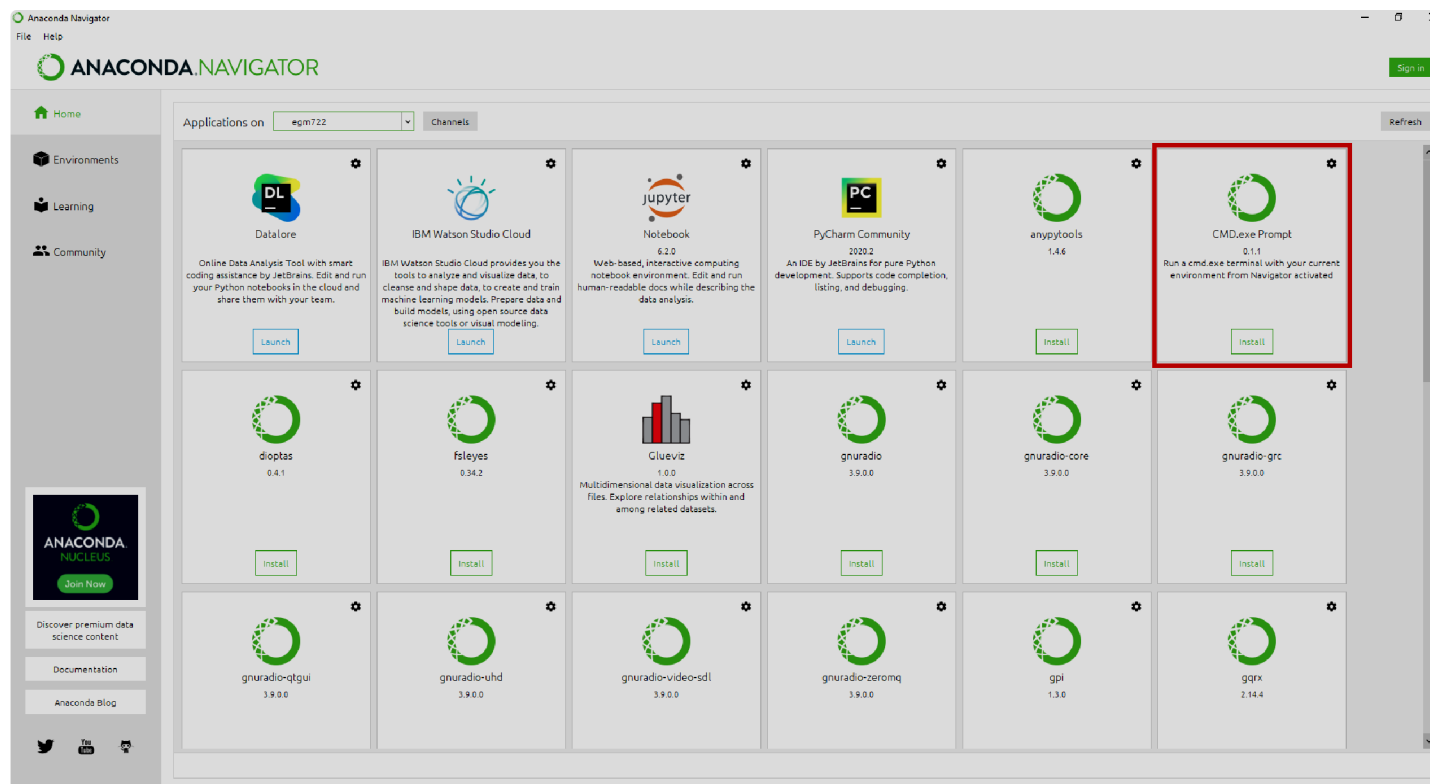
You should see that you have two different environments when you open the dropdown menu next to **Applications on**. Whenever you want to switch environments in **Anaconda Navigator**, you can select the environment from this menu.

You should see that when you change between the `base` and `egm722` environments, the list of installed applications changes – this is because we have not installed the same applications in the different environments.

setting up the command prompt

The final step covered in this manual is to install the **CMD.exe Prompt** for this environment:

[Skip to main content](#)



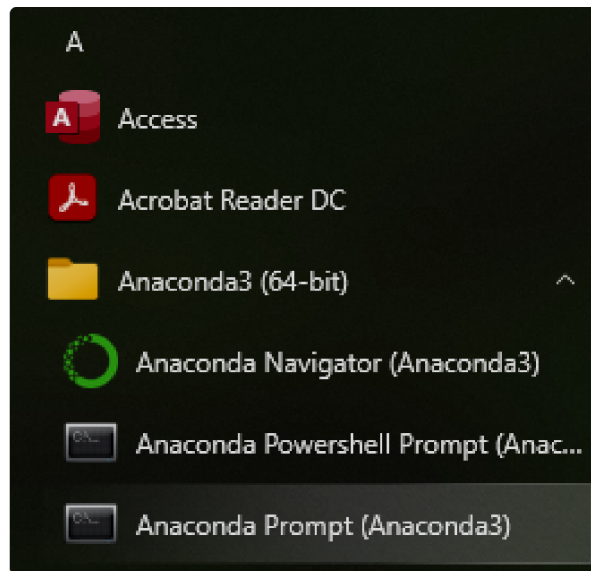
While not strictly required, this will enable you to directly launch a Windows Command Prompt with your **egm722** environment already loaded.

If you run the **Anaconda Prompt** from the **Start Menu**, it will automatically load the default (`base`) environment, and you will need to switch environments using the `conda activate` command when you want to use your **egm722** environment.

Once you have the conda environment set up, you are ready to move on to configuring **jupyter**.

anaconda command prompt

[Skip to main content](#)



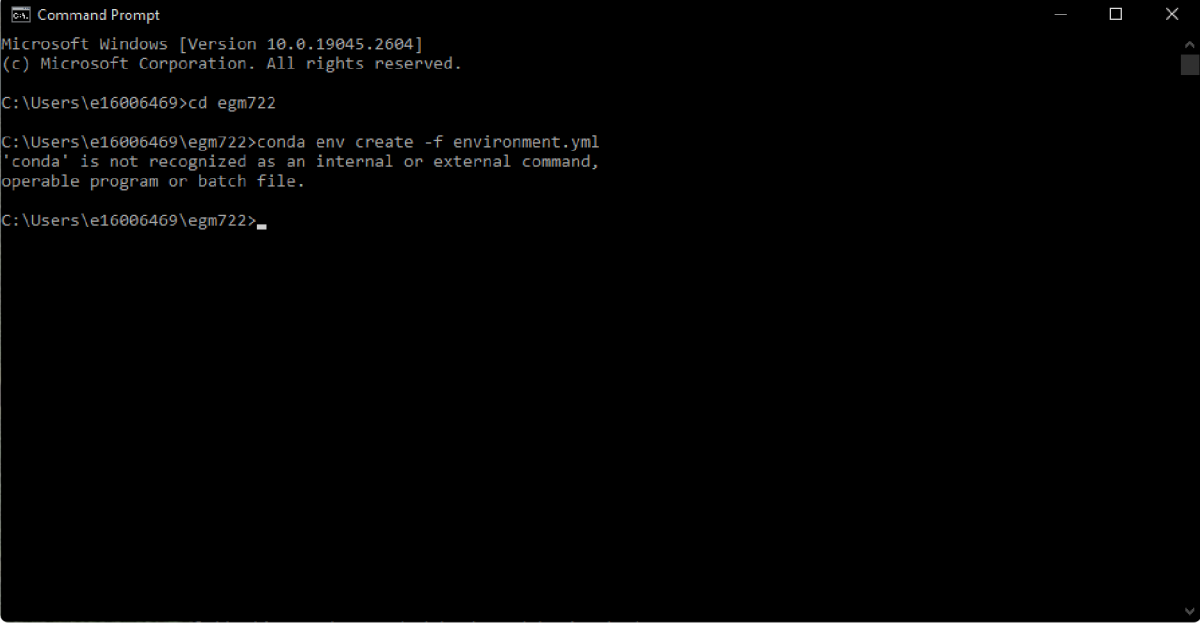
When you open the command prompt, you should see `(base)` next to the prompt:

```
(base) C:\Users\bob>
```

[Skip to main content](#)

Warning

The **Anaconda Command Prompt** automatically loads **conda** so that you can use it; if you don't see `(base)`, it likely means that you have opened the normal **Command Prompt**, and so you will see the following error when you try to run a command with **conda**:



```
Command Prompt
Microsoft Windows [Version 10.0.19045.2604]
(c) Microsoft Corporation. All rights reserved.

C:\Users\e16006469>cd egm722

C:\Users\e16006469>conda env create -f environment.yml
'conda' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\e16006469>conda env create -f environment.yml
```

Next, navigate to where you have cloned the repository using `cd`:

```
cd c:\Users\bob\projects\egm722
```

If you're in the right place, you should see all of the files from the repository when you enter the `dir` command:

[Skip to main content](#)

```
Anaconda Prompt (Anaconda3) - conda env create -f environment.yml

(base) C:\Users\e16006469>cd egm722

(base) C:\Users\e16006469\egm722>dir
Volume in drive C is OS
Volume Serial Number is F098-DD8D

Directory of C:\Users\e16006469\egm722

26/02/2023  09:35    <DIR>        .
26/02/2023  09:35    <DIR>        ..
29/09/2022  08:55                27 .gitignore
26/02/2023  09:35                155 environment.yml
29/09/2022  08:55            19,051 LICENSE
29/09/2022  08:55            3,352 README.md
26/02/2023  09:35    <DIR>        Week1
               4 File(s)        22,585 bytes
               3 Dir(s)  634,741,927,936 bytes free

(base) C:\Users\e16006469\egm722>
```

Now, enter the following command:

```
conda env create -f environment.yml
```

This tells **conda** to create a new environment, using the “recipe” provided in `environment.yml`. You should (eventually) see something like the following:

[Skip to main content](#)

```
Anaconda Prompt (Anaconda3) - conda env create -f environment.yml

(base) C:\Users\e16006469>cd egm722

(base) C:\Users\e16006469\egm722>dir
Volume in drive C is OS
Volume Serial Number is F098-DD8D

Directory of C:\Users\e16006469\egm722

26/02/2023  09:35    <DIR>        .
26/02/2023  09:35    <DIR>        ..
29/09/2022  08:55             27 .gitignore
26/02/2023  09:35             155 environment.yml
29/09/2022  08:55          19,051 LICENSE
29/09/2022  08:55           3,352 README.md
26/02/2023  09:35    <DIR>        Week1
               4 File(s)        22,585 bytes
               3 Dir(s)  634,741,927,936 bytes free

(base) C:\Users\e16006469\egm722>conda env create -f environment.yml
Collecting package metadata (repodata.json): /
```

This step may take some time^[2], but if all goes well, you will eventually see the following message:

[Skip to main content](#)

```

Anaconda Prompt (Anaconda3)
markupsafe-2.1.2      | 25 KB | ##### | 100%
win_inet_pton-1.1.0  | 8 KB  | ##### | 100%
cryptography-39.0.1 | 981 KB| ##### | 100%
jupyter_events-0.6.3 | 75 KB | ##### | 100%
pyyaml-6.0           | 153 KB| ##### | 100%
pyepsg-0.4.0         | 20 KB | ##### | 100%
pygments-2.14.0      | 805 KB| ##### | 100%
matplotlib-base-3.7.1| 6.4 MB| ##### | 100%
nbconvert-core-7.2.9 | 195 KB| ##### | 100%
bleach-6.0.0         | 128 KB| ##### | 100%
libhwloc-2.9.0       | 2.4 MB| ##### | 100%
importlib_metadata-6 | 9 KB  | ##### | 100%
stack_data-0.6.2     | 26 KB | ##### | 100%
gdal-3.6.2           | 1.3 MB| ##### | 100%
jpeg-9e              | 283 KB| ##### | 100%
nbclassic-0.5.2      | 5.4 MB| ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate egm722
#
# To deactivate an active environment, use
#
#     $ conda deactivate
#
(base) C:\Users\e16006469\egm722>

```

And that's it. Once you have done this, you may want to open **Anaconda Navigator** and set up a new command prompt link for your environment, following the instructions in [setting up the command prompt](#).

Once you have the conda environment set up, you are ready to move on to configuring **jupyter**.

notes and references

- [1] The exact number of packages may differ - the point is that we have only told conda to install a few of these, and conda has figured out the rest.

[Skip to main content](#)

Conda can be notoriously slow, which is why there has been an effort to increase the speed/efficiency of the package “solver”: [mamba](#). I am happy to help you get set up with mamba, but I am unsure about how well this works with **Anaconda Navigator**.

< Previous
[setting up conda/anaconda](#)

Next >
[configuring jupyter](#)