

Pytorch-Wildlife Installation instructions



1. Install Anaconda



[Platform](#) [Solutions](#) [Pricing](#) [Partners](#) [Resources](#) [Company](#)

[Sign In](#)


[Contact Sales](#)

Anaconda Distribution

Free Download

Everything you need to get started in data science on your workstation.

- ✓ Free distribution install
- ✓ Thousands of the most fundamental DS, AI, and ML packages
- ✓ Manage packages and environments from desktop application
- ✓ Deploy across hardware and software platforms

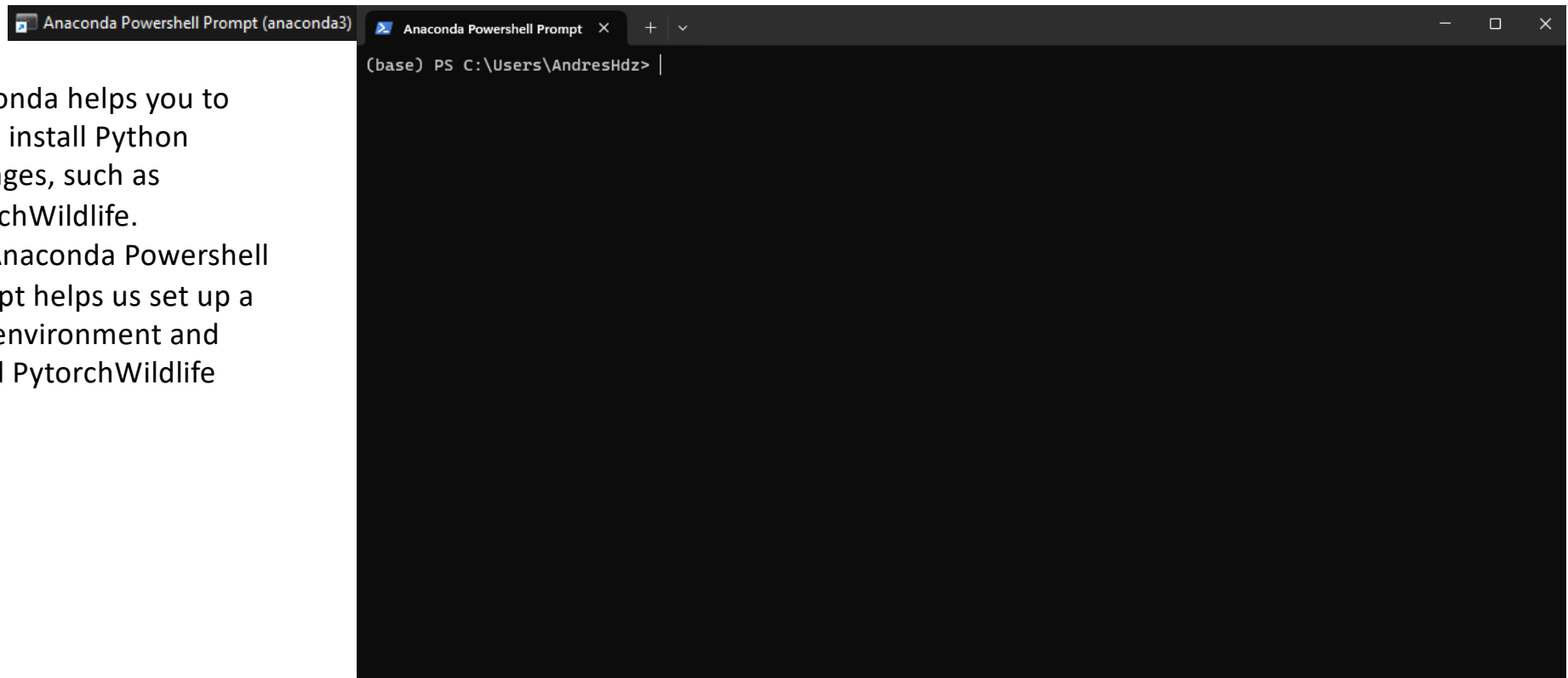
 **Download**

Get Additional Installers



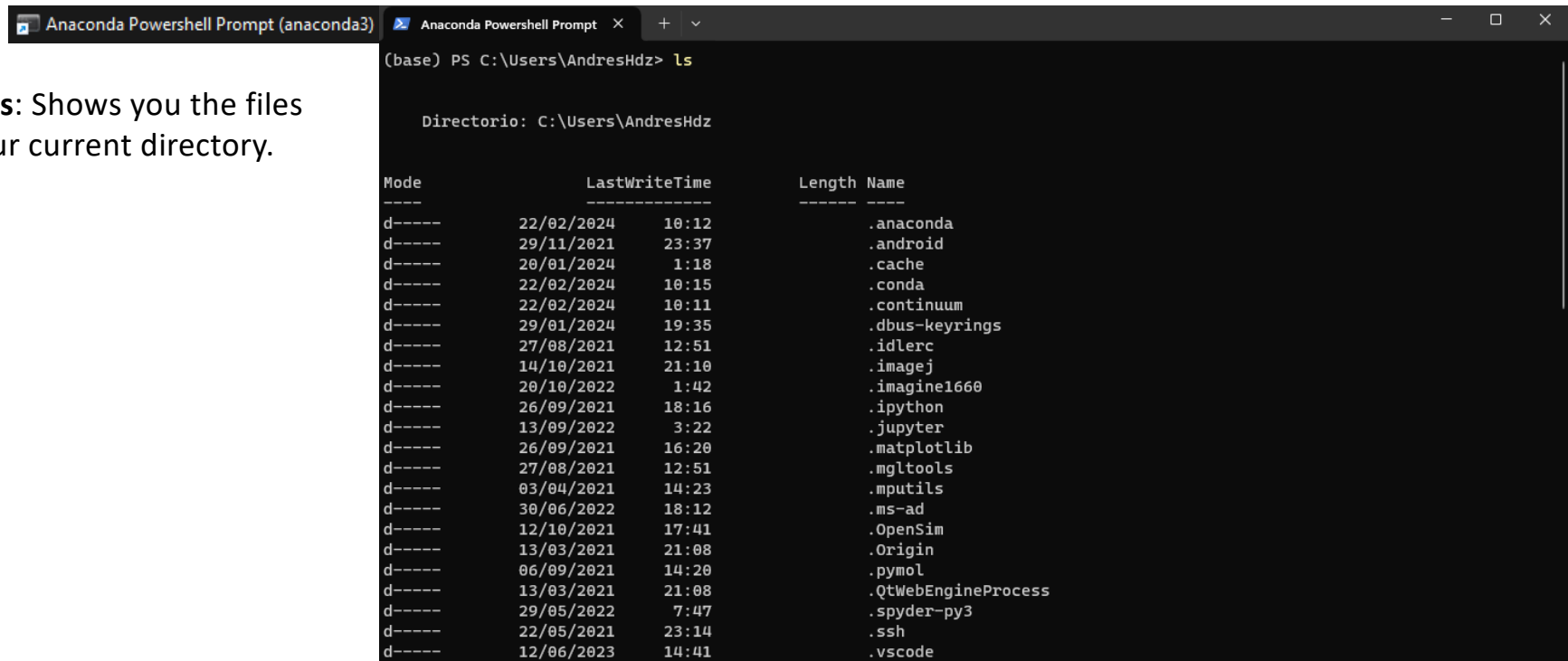
2. Open the powershell prompt

- Anaconda helps you to easily install Python packages, such as PytorchWildlife.
- The Anaconda Powershell Prompt helps us set up a new environment and install PytorchWildlife



3.1 Navigate to your destination folder

- (list) **ls**: Shows you the files in your current directory.



The screenshot shows an Anaconda PowerShell Prompt window with the following content:

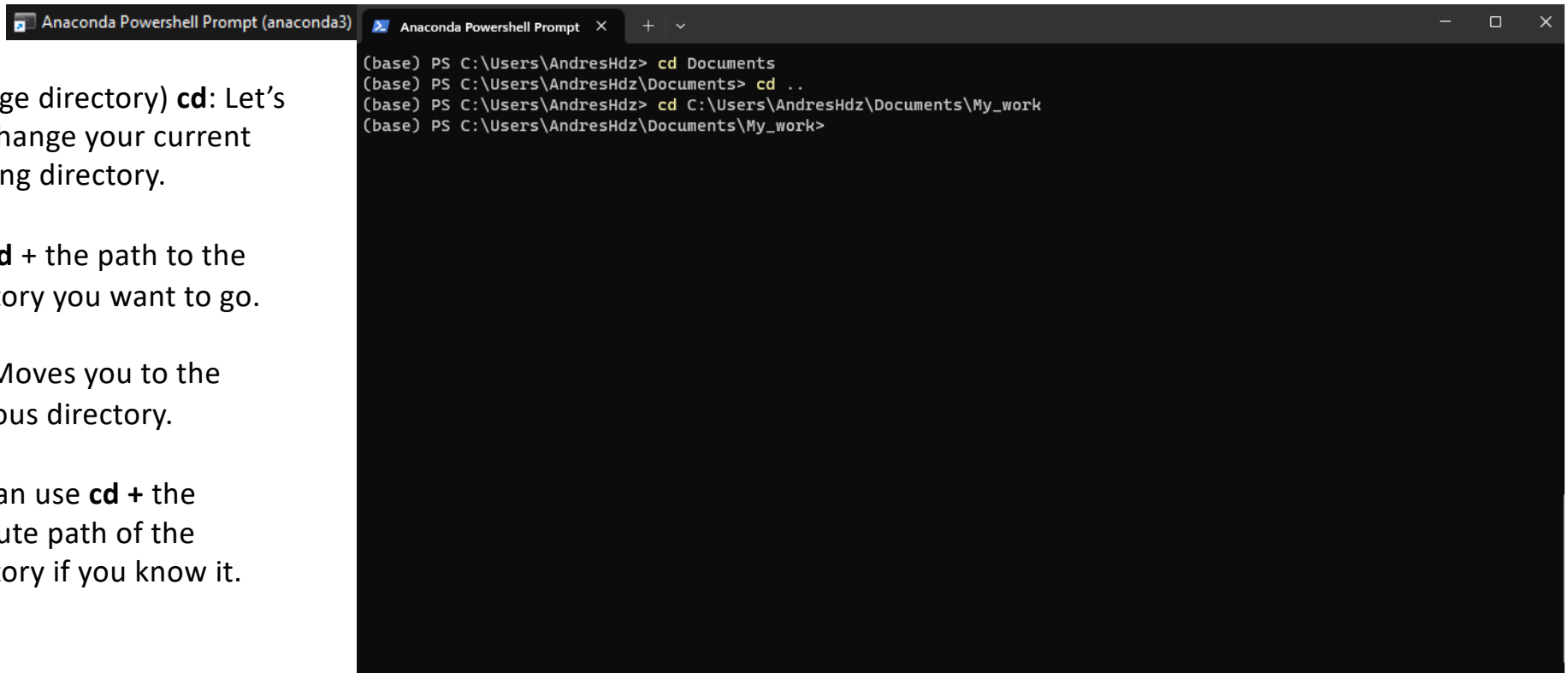
```
(base) PS C:\Users\AndresHdz> ls
```

Directorio: C:\Users\AndresHdz

Mode	LastWriteTime	Length	Name
d----	22/02/2024 10:12		.anaconda
d----	29/11/2021 23:37		.android
d----	20/01/2024 1:18		.cache
d----	22/02/2024 10:15		.conda
d----	22/02/2024 10:11		.continuum
d----	29/01/2024 19:35		.dbus-keyrings
d----	27/08/2021 12:51		.idlerc
d----	14/10/2021 21:10		.imagej
d----	20/10/2022 1:42		.imagine1660
d----	26/09/2021 18:16		.ipython
d----	13/09/2022 3:22		.jupyter
d----	26/09/2021 16:20		.matplotlib
d----	27/08/2021 12:51		.mgltools
d----	03/04/2021 14:23		.mputils
d----	30/06/2022 18:12		.ms-ad
d----	12/10/2021 17:41		.OpenSim
d----	13/03/2021 21:08		.Origin
d----	06/09/2021 14:20		.pymol
d----	13/03/2021 21:08		.QtWebEngineProcess
d----	29/05/2022 7:47		.spyder-py3
d----	22/05/2021 23:14		.ssh
d----	12/06/2023 14:41		.vscode

3.2 Navigate to your destination folder

- (change directory) **cd**: Let's you change your current working directory.
- Use **cd** + the path to the directory you want to go.
- **cd ..** Moves you to the previous directory.
- You can use **cd** + the absolute path of the directory if you know it.

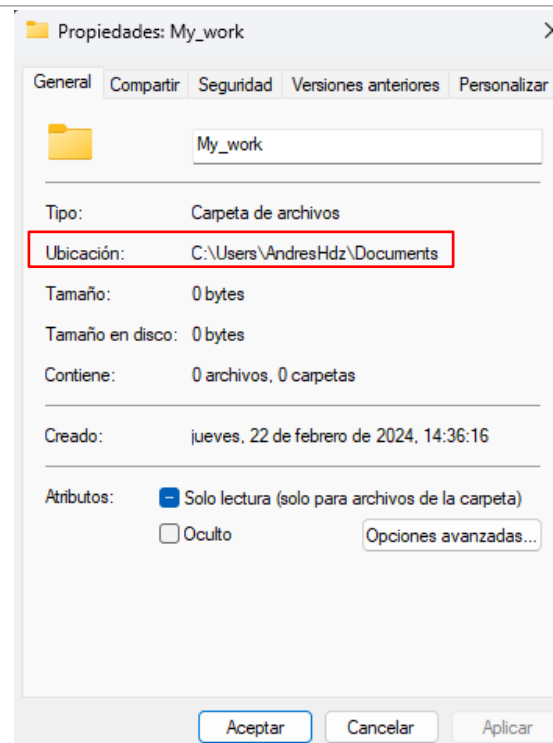


```
Anaconda PowerShell Prompt (anaconda3) Anaconda PowerShell Prompt X + -
```

```
(base) PS C:\Users\AndresHdz> cd Documents
(base) PS C:\Users\AndresHdz\Documents> cd ..
(base) PS C:\Users\AndresHdz> cd C:\Users\AndresHdz\Documents\My_work
(base) PS C:\Users\AndresHdz\Documents\My_work>
```

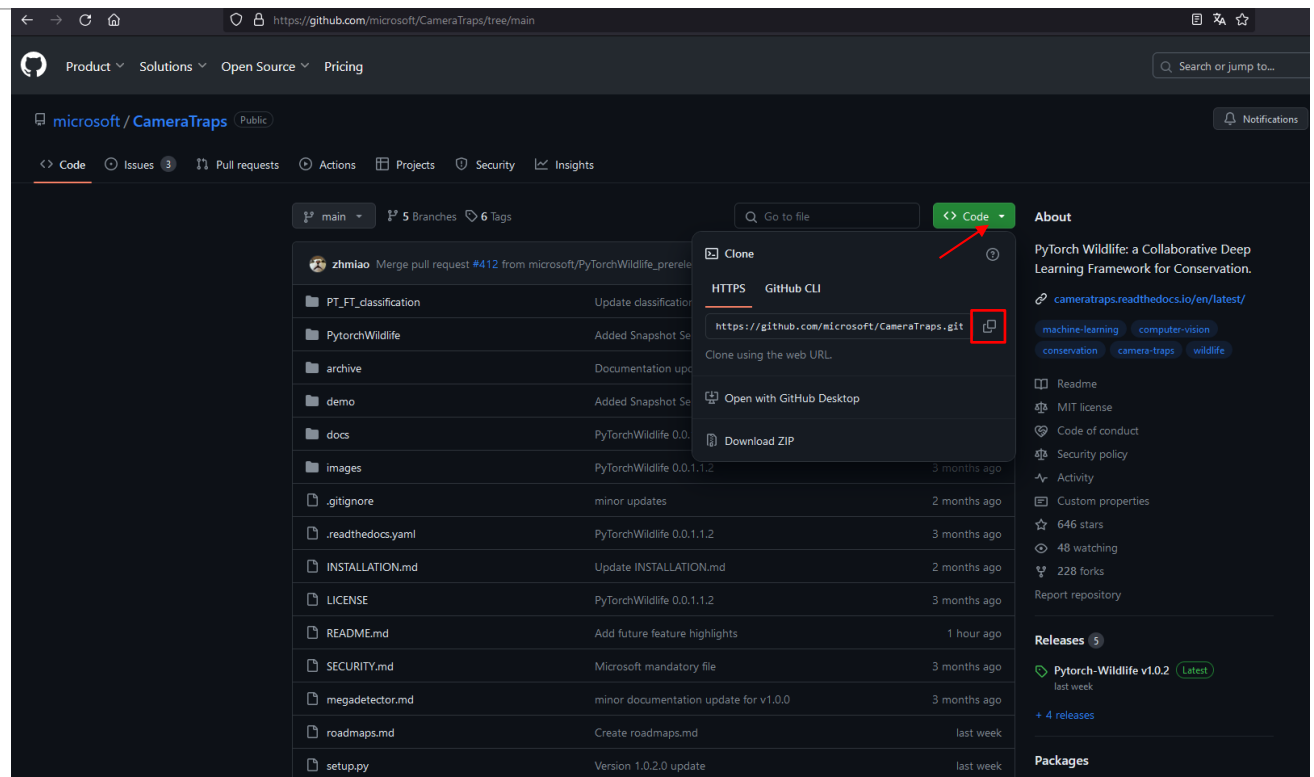
3.3 Navigate to your destination folder

- If you want to know the absolute path of the folder that you want to use. Right-click and click **properties**. It will show you the path.



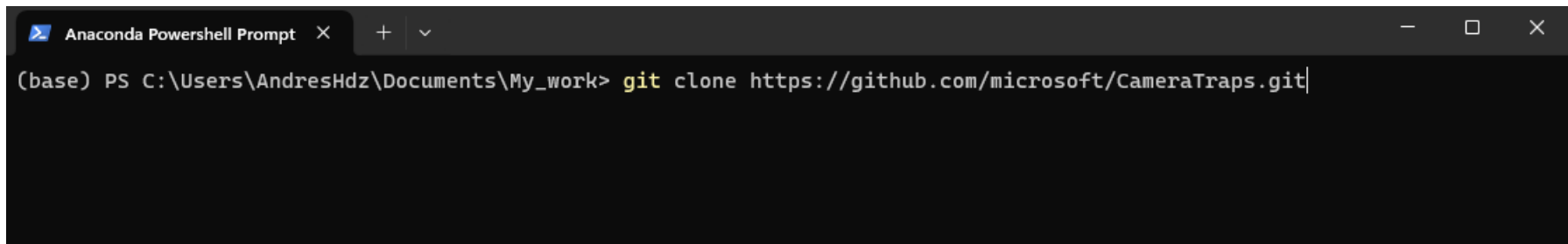
4.1 Clone the CameraTraps repo

- Go to github.com/microsoft/CameraTraps/tree/main and copy the link.



4.2 Clone the CameraTraps repo

- The PytorchWildlife GitHub repo contains a set of installation instructions, we will be following them.
<https://github.com/microsoft/CameraTraps/blob/main/INSTALLATION.md>

A screenshot of an Anaconda Powershell Prompt window. The window has a title bar with the text "Anaconda Powershell Prompt" and standard window controls (minimize, maximize, close). The command prompt shows the current directory as "C:\Users\AndresHdz\Documents\My_work" and the command "git clone https://github.com/microsoft/CameraTraps.git" being entered. The prompt is "(base) PS C:\Users\AndresHdz\Documents\My_work>".

```
(base) PS C:\Users\AndresHdz\Documents\My_work> git clone https://github.com/microsoft/CameraTraps.git|
```

Clone the repository by typing **git clone** <https://github.com/microsoft/CameraTraps.git> in the Anaconda Powershell Prompt

4.3 Clone the CameraTraps repo

Once cloned, you should have a **CameraTraps** directory, you can check if it is there by typing **ls**.



```
Anaconda PowerShell Prompt X + v
(base) PS C:\Users\AndresHdz\Documents> cd CameraTraps
(base) PS C:\Users\AndresHdz\Documents\CameraTraps>
```

Go to the CameraTraps directory by typing **cd CameraTraps**

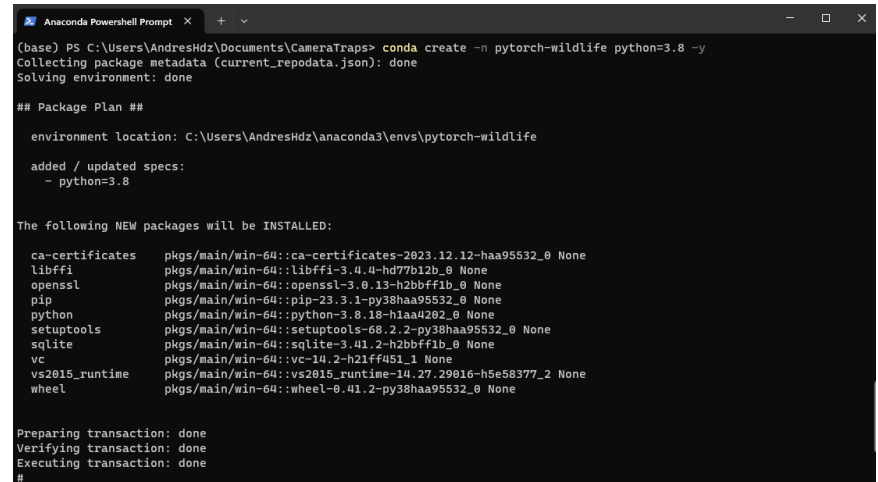
5.1 Create a new conda environment

Following PytorchWildlife's installation instructions, we are going to create a conda environment. Run the following code in your Anaconda Powershell Prompt:

```
conda create -n pytorch-wildlife python=3.8 -y
conda activate pytorch-wildlife
```

```
(base) PS C:\Users\AndresHdz\Documents\CameraTraps> conda activate pytorch-wildlife
(pytorch-wildlife) PS C:\Users\AndresHdz\Documents\CameraTraps>
```

You know you have set up the environment correctly once the **(base)** changes to **(pytorch-wildlife)**.



```

Anaconda Powershell Prompt
(base) PS C:\Users\AndresHdz\Documents\CameraTraps> conda create -n pytorch-wildlife python=3.8 -y
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: C:\Users\AndresHdz\anaconda3\envs\pytorch-wildlife

added / updated specs:
  - python=3.8

The following NEW packages will be INSTALLED:

ca-certificates pkgs/main/win-64::ca-certificates-2023.12.12-haa95532_0 None
libffi           pkgs/main/win-64::libffi-3.4.4-hd77b12b_0 None
openssl         pkgs/main/win-64::openssl-3.0.13-h2bbff1b_0 None
pip             pkgs/main/win-64::pip-23.3.1-py38haa95532_0 None
python          pkgs/main/win-64::python-3.8.18-h1aa4202_0 None
setuptools      pkgs/main/win-64::setuptools-68.2.2-py38haa95532_0 None
sqlite          pkgs/main/win-64::sqlite-3.41.2-h2bbff1b_0 None
vc              pkgs/main/win-64::vc-14.2-h21ff451_1 None
vs2015_runtime  pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2 None
wheel           pkgs/main/win-64::wheel-0.41.2-py38haa95532_0 None

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
```

6. Install Pytorch-Wildlife

We have focused on making Pytorch-Wildlife easy to install, you only need to run one command to set it up!

pip install pytorch-wildlife

```
(pytorch-wildlife) PS C:\Users\AndresHdz\Documents\CameraTraps> pip install PytorchWildlife
```

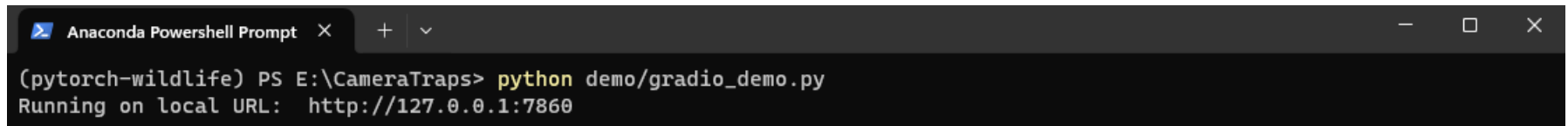
Once it is done installing, it will tell you that the installation is complete, **it may take several minutes.**

```
Using cached oauthlib-3.2.2-py3-none-any.whl (131 kB)
Using cached pyasn1-0.5.1-py2.py3-none-any.whl (86 kB)
Installing collected packages: wcwidth, pytz, pydub, pure-eval, pickleshare, ffmpeg, backcall, zipp, websockets, urllib3,
tzdata, typing-extensions, traitlets, toolz, tomlkit, termcolor, tensorboard-data-server, sniffio, six, shellingham, se
mantic-version, rpsd-py, pyyaml, python-multipart, pyrsistent, pygments, pyasn1, psutil, protocol, prompt-toolkit, plogu
l-resolve-name, Pillow, parso, packaging, orjson, oauthlib, numpy, mdurl, markupsafe, kiwisolver, idna, hii, grpcio, fss
pec, fonttools, filelock, executing, exceptiongroup, decorator, cyclo, colorama, charset-normalizer, certifi, cachetool
s, attrs, aiofiles, absl-py, werkzeug, tqdm, torch, scipy, rsa, requests, referencing, python-dateutil, pydantic-core, p
yasn1-modules, opencv-python-headless, opencv-python, matplotlib-inline, markdown-it-py, Jinja2, jedi, importlib-resourc
es, importlib-metadata, httpcore, fire, contourpy, click, asttokens, anyio, annotated-types, uvicorn, typer, torchvision
, torchaudio, thop, starlette, stack-data, rich, requests-oauthlib, pydantic, pandas, matplotlib, markdown, jsonschema-s
pecifications, huggingface-hub, https, google-auth, supervision, seaborn, jsonschema, lpython, gradio-client, google-aut
h-oauthlib, fastapi, tensorboard, altair, ultralytics-yolov5, gradio, PytorchWildlife
Successfully installed Pillow-10.1.0 PytorchWildlife-1.0.1.1 absl-py-2.1.0 aiofiles-23.2.1 altair-5.2.0 annotated-types-
0.6.0 anyio-3.7.0 asttokens-2.1.1 attrs-23.2.0 backcall-0.2.0 cachetools-5.3.2 certifi-2024.2.2 charset-normalizer-3.3.2
click-8.1.7 colorama-0.4.6 contourpy-1.1.1 cyclo-0.12.1 decorator-5.1.1 exceptiongroup-1.2.0 executing-2.0.1 fastapi-0
.109.2 ffmpeg-0.3.2 filelock-3.13.1 fire-0.5.0 fonttools-4.49.0 fsspec-2024.2.0 google-auth-2.28.1 google-auth-oauthlib-1
.0.6 gradio-0.6.0 gradio-client-0.7.1 grpcio-1.62.0 hii-0.10.0 httpcore-1.0.4 httpx-0.27.0 huggingface-hub-0.20.3 idna-3
.6 importlib-metadata-7.0.1 importlib-resources-6.1.1 ipython-8.12.3 jedi-0.19.1 Jinja2-3.1.3 jsonschema-4.21.1 jsonsche
ma-specifications-2023.12.1 kiwisolver-1.4.5 markdown-3.5.2 markdown-it-py-3.0.0 markupsafe-2.1.5 matplotlib-3.7.5 matpl
otlib-inline-0.1.6 mdurl-0.1.2 numpy-1.24.4 oauthlib-3.2.2 opencv-python-4.9.0.80 opencv-python-headless-4.9.0.80 orjson
-3.9.10 packaging-23.2 pandas-2.0.3 parso-0.8.3 pickleshare-0.7.5 plogu-l-resolve-name-1.3.10 prompt-toolkit-3.0.43 prot
ocol-0.1.2 psutil-5.9.8 pure-eval-0.2.2 pyasn1-0.5.1 pyasn1-modules-0.3.0 pydantic-2.6.1 pydantic-core-2.16.2 pydub-0.2
5.1 pygments-2.17.2 pyrsistent-3.1.1 python-dateutil-2.8.2 python-multipart-0.0.9 pytz-2024.1 pyyaml-6.0.1 referencing-0
.33.0 requests-2.31.0 requests-oauthlib-1.3.1 rich-13.7.0 rpsd-py-0.18.0 rsa-4.9.0 scipy-1.10.1 seaborn-0.13.2 semantic-ver
sion-2.10.0 shellingham-1.5.4 six-1.16.0 sniffio-1.3.0 stack-data-0.6.3 starlette-0.36.3 supervision-0.16.0 tensorboard-
2.14.0 tensorboard-data-server-0.7.2 termcolor-2.4.0 thop-0.1.1.post2209072238 tomlkit-0.12.0 toolz-0.12.1 torch-2.0.1
torchaudio-0.10.1 torchvision-0.11.2 tqdm-4.66.1 traitlets-5.14.1 typer-0.9.0 typing-extensions-4.9.0 tzdata-2024.1 ultr
alytics-yolov5-6.1.1 urllib3-2.2.1 uvicorn-0.27.1 wcwidth-0.2.13 websockets-11.0.3 werkzeug-3.0.1 zipp-3.17.0
(pytorch-wildlife) PS C:\Users\AndresHdz\Documents\CameraTraps>
```

7.1 Start using our demo!

PytorchWildlife comes with a set of instruction for people with and without technical knowledge! If you want to load our user interface, run the following command (Make sure your path is in the CameraTraps folder):

python demo/gradio_demo.py



```
Anaconda Powershell Prompt x + v
(pytorch-wildlife) PS E:\CameraTraps> python demo/gradio_demo.py
Running on local URL: http://127.0.0.1:7860
```

Once loaded, the Anaconda Powershell Prompt Will show a URL to access the demo, copy and paste it in your web browser without closing the Powershell.



7.2 Start using our demo!

Once you paste the URL to your web browser, you can start using the demo!
Please remember that this demo uses the processing capabilities from your local computer.

