- 1. Install Unity 2022.3.4f1
 - a. This is the version the examples use and the one I have tested
- 2. Install Anaconda
 - a. Free Download | Anaconda
 - b. We use anaconda as it provides a way to install python 3.10.13
- 3. Make an environment with Python 3.10.13
 - a. Open Anaconda navigator
 - b. Click on "environments" in the left side
 - c. Click on "create" at the bottom
 - d. Name your environment and choose the python version(here 3.10.13)

Create new environment					
Name: Location:	New environ	iment name			
Packages:	Python	3.11.7	~		
	R	2.7.18			
		3.5.6			
		3.6.13		Create	
		3.7.16			
		3.8.18			
		3.9.18			
		3.10.13			
		✓ 3.11.7			
		3.12.1			

e. After having created your environment, you need to run it. You run your environment by pressing the green arrow next to your environment.

base (root)	
MLSTUFF	0

- f. Here you choose "Open Terminal"
- 4. Use git to clone mlagents to a folder
 - a. Git might not be installed by default
 - i. To install git we use Navigator
 - ii. Set Navigator to show all (or not installed) and not just installed

(Search Environments Q	Not installed Channels Update index
base (root)	Installed Description
MLSTUFF	Updatable Dpen-source file archiver primarily used to compress files
	Selected -zip is a file archiver with a high compression ratio.
	All
	□ _go_select O The golang select package.
	ipyw_jlab_nb_ex O A configuration metapackage for enabling anaconda-bundled jupyter extensions
	Libarchive_static O A static build of libarchive containing only conda-related parts
	□ _libgcc_mutex O Mutex for libgcc and libgcc-ng
	Low_priority O Metapackage to lower the priority of a package using track_features
	mutex_mxnet O Mutex package to pin a variant of mxnet conda package

- iii. Search for git and choose the one just named "git"
- b. It doesn't matter which folder you clone git to, I'll just clone mine to C:/IT
- c. Use the command "cd" followed by the path to where you want mlagents to be cloned to

Distributed version control system

- i. In my situation it would be cd C:/IT
- d. Use the following command to clone mlagents into the folder you just navigated to (using cd)

"git clone --branch release_21 <u>https://github.com/Unity-Technologies/ml-agents.git</u>"

- 5. Now we install PyTorch
 - a. We use the PyTorch version that mlagents recommends (1.13.1)
 - b. Here is a link to 1.13.1 Previous PyTorch Versions | PyTorch
 - c. Now we find the download command under "wheel" not under "conda"
 - d. Then we choose the version we use

🗖 git

- i. For me is cuda, but if you're on a laptop without a GPU you'll need "CPU only"
- ii. You can check if you have cuda by running "nvcc –version" in cmd (this also tells you the version of cuda you have)
- iii. If you use cuda you need to download the correct version of cuda as PyTorch 1.13.1 only supports 11.6 and 11.7

Wheel

OSX

pip install torch==1.13.1 torchvision==0.14.1 torchaudio==0.13.1

Linux and Windows

```
# ROCM 5.2 (Linux only)
pip install torch==1.13.1+rocm5.2 torchvision==0.14.1+rocm5.2 torchaudio==0.13.1 --extra-index-url https://c
# CUDA 11.6
pip install torch==1.13.1+cu116 torchvision==0.14.1+cu116 torchaudio==0.13.1 --extra-index-url https://down1
# CUDA 11.7
pip install torch==1.13.1+cu117 torchvision==0.14.1+cu117 torchaudio==0.13.1 --extra-index-url https://down1
# CPU only
pip install torch==1.13.1+cpu torchvision==0.14.1+cpu torchaudio==0.13.1 --extra-index-url https://download.
```

- 6. Now we install all the packages and dependencies that mlagents uses
 - a. Go into your mlagents folder (the one you cloned)

📕 .git	23/01/2024 21.52	File folder
📙 .github	23/01/2024 21.51	File folder
📙 .yamato	23/01/2024 21.51	File folder
🔄 colab	23/01/2024 21.51	File folder
om.unity.ml-agents	23/01/2024 21.51	File folder
com.unity.ml-agents.extensions	23/01/2024 21.51	File folder
📊 config	23/01/2024 21.51	File folder
- DevProject	23/01/2024 21.51	File folder
docs	23/01/2024 21.51	File folder
localized_docs	23/01/2024 21.51	File folder
nl-agents	23/01/2024 21.52	File folder
nl-agents-envs	23/01/2024 21.52	File folder
nl-agents-plugin-examples	23/01/2024 21.52	File folder
nl-agents-trainer-plugin	23/01/2024 21.52	File folder
Project	23/01/2024 21.51	File folder
protobuf-definitions	23/01/2024 21.52	File folder
unity-volume	23/01/2024 21.52	File folder
utile	22/01/2024 21 52	File folder

b. Find these 2 folders

- c. Go into ml-agents-envs and edit setup.py
- d. Change the specified version of NumPy and Python

```
zip_safe=False,
    install_requires=[
        "cloudpickle",
        "grpcio>=1.11.0,<=1.48.2",
        "Pillow>=4.2.1",
        "protobuf>=3.6,<3.20",
        "pyyaml>=3.1.0",
        "gym>=0.21.0",
        "pettingzoo==1.15.0",
        "numpy>=1.23.3",
        "filelock>=3.4.0",
    ],
    python_requires=">=3.10.1,<=3.10.13",
    # TODO: Remove this once mypy stops having spurious setuptools issues.
    cmdclass={"verify": VerifyVersionCommand}, # type: ignore
)
```

- e. The end of setup.py should look like this after you've changed it
- f. Now we go to ml-agents (not envs) and edit that setup.py

```
g. Here we just change the python version to 3.10.13
    python_requires=">=3.10.1,<=3.10.13",
    entry_points={
        "console_scripts": [
        "mlagents-learn=mlagents.trainers.learn:main",
        "mlagents-run-experiment=mlagents.trainers.run_experiment:main",
        "mlagents-push-to-hf=mlagents.utils.push_to_hf:main",
        "mlagents-load-from-hf=mlagents.utils.load from hf:main",</pre>
```

```
],
# Plugins - each plugin type should have an entry here for the default behavior
ML_AGENTS_STATS_WRITER: [
```

```
"default=mlagents.plugins.stats_writer:get_default_stats_writers"
```

```
],
ML_AGENTS_TRAINER_TYPE: [
    "default=mlagents.plugins.trainer_type:get_default_trainer_types"
  ],
},
# TODO: Remove this once mypy stops having spurious setuptools issues.
cmdclass={"verify": VerifyVersionCommand}, # type: ignore
```

```
)
```

- 7. Now we cd into the cloned mlagents folder
 - a. Run these 2 commands
 - i. python -m pip install ./ml-agents-envs
 - ii. python -m pip install ./ml-agents
- 8. Now we can create a project or use the example projects
- 9. To use the examples you go to Unity Hub and click "add"
 - a. Now navigate to the cloned mlagents folder and find the folder "project"
 - b. Now you go to the package manager in Unity (under window) and install MLAgents (Remember to set it to show Unity Registry and not just In Project)
- 10. Now you can follow guides as it is now installed