

**PIXEL PROJECT**

**Prepared for**

StJohn Russel

VP Engineering, TooMuchWifi

**By**

Mpetha S. Motaung

Software Engineer, TooMuchWifi

14.12.2024

# Abstract

# Introduction

Before you get started here are the prerequisites:

* Understand virtual environments
* Understand package managers
* Understand cmd commands
* Understand Github Desktop, Github and Gitversion control

## Objectives (Initial Assumptions)

* Create Django project that can handle image uploads.
* The uploaded images have to be processed to extract center most pixel and hex value.
* Both uploaded image and hex value pair have to be viewable by the user (As a gallery of photos and their hex values).
* The uploaded images and their hex values must be displayed using Django’s template system.
* The web application must be able to handle errors for non-image file uploads and image processing fail cases.
* Minimum memory usage when extracting center pixels without loading the entire image (Save the url of the image as text in the database.)

## Requirements

# Setup & Project Configuration

Download and install Anaconda for package management and virtual environment. (used for sourcing, building, and deploying data science and AI initiatives)

\*You may use Anaconda or any other virtual environment platform (e.g. virtual env, pyenv, pipenv) but I prefer Anaconda\*

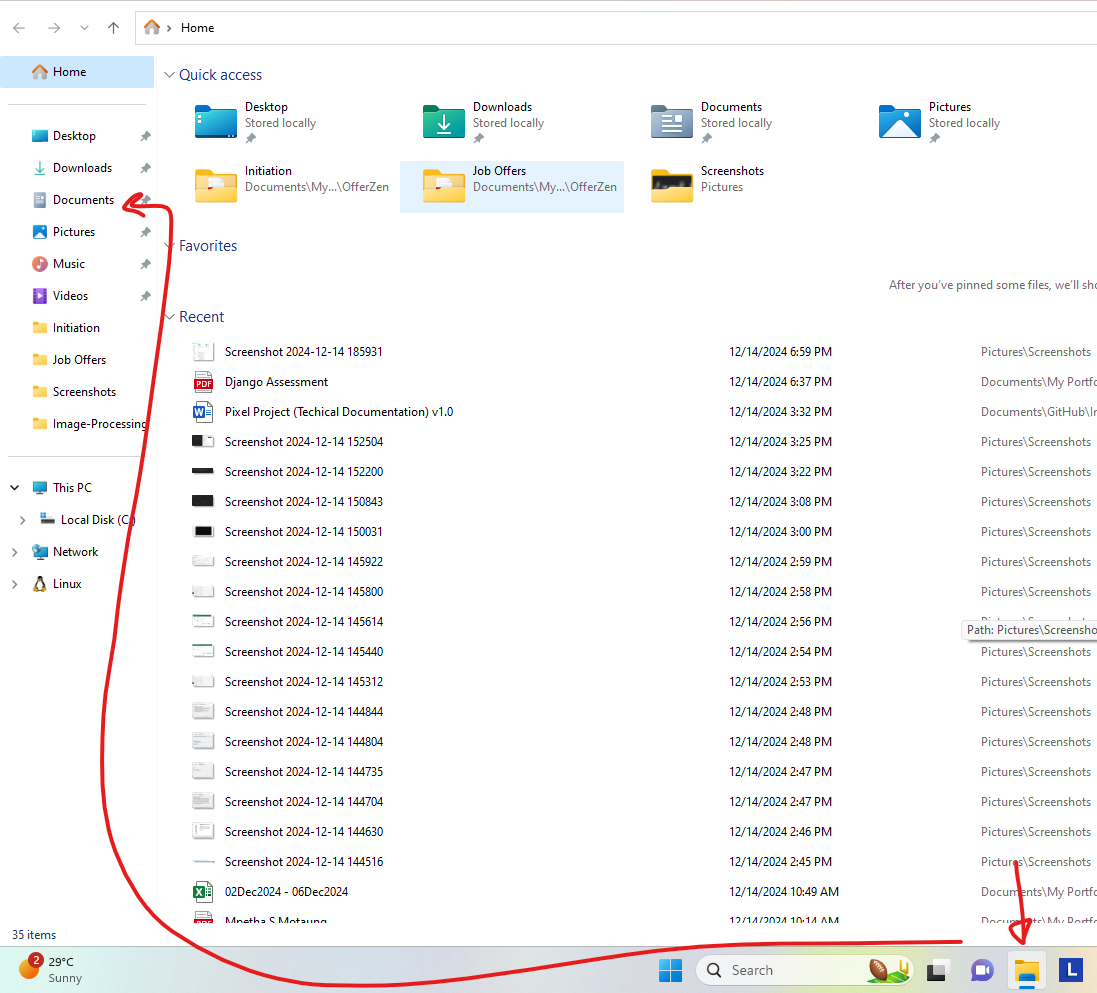
After Downloading <https://www.anaconda.com/download/success> and installing Anaconda. There are 2 methods of creating a virtual env

### Creating or choosing Project location

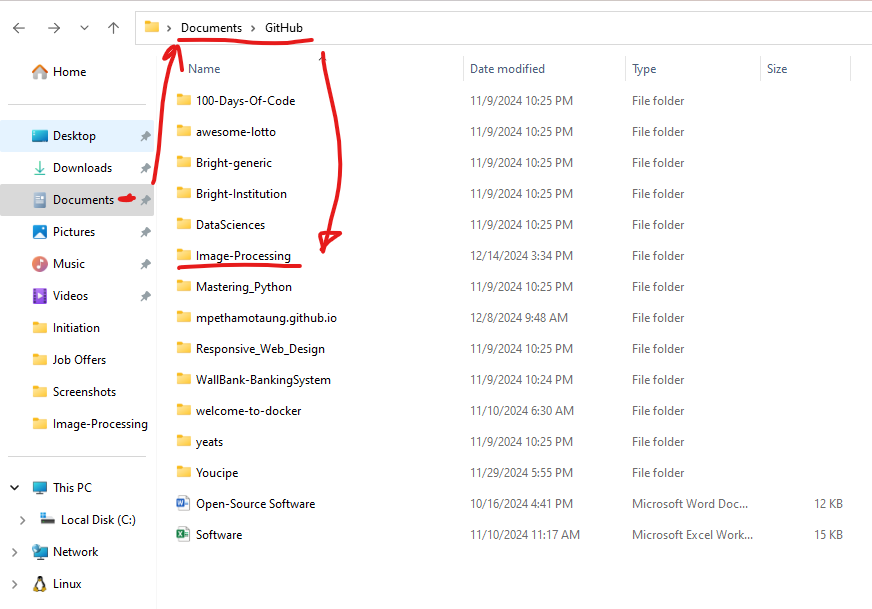
You can either create a folder using the Windows Graphic User Interface (GUI) or Command Line Interface (CLI). We must create a folder anywhere in file explorer but I suggest using Document/Github/(project name) to ensure that all your projects are organized. [[1](#_References)]

#### Method 1 (Windows GUI)

1. Open File Explorer (double left-click)



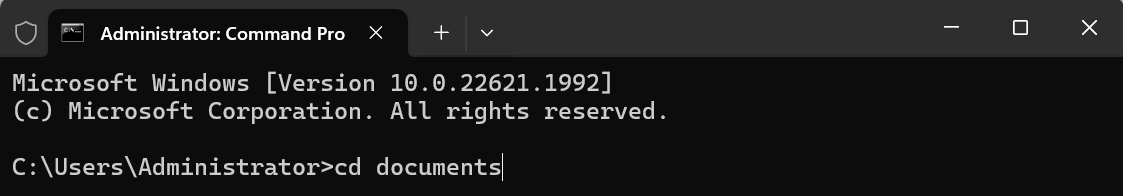
1. Navigate to documents, create a Github folder(if it doesn’t already exist). Within Github folder create project folder ‘Image Processing’ (Documents/Github/Image-Processing)



This is the folder that we will be executing the Django project from

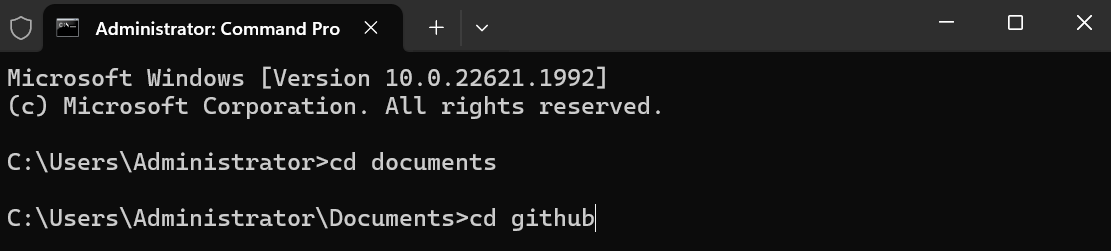
#### Method 2 (CMD)

1. Open CMD, then navigate to Documents/Github. Create the Github folder if it doesn’t already exist.



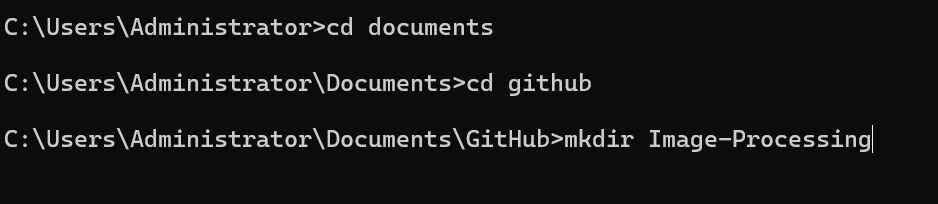
To navigate into documents cd documents, then hit enter

1. Navigate into github folder



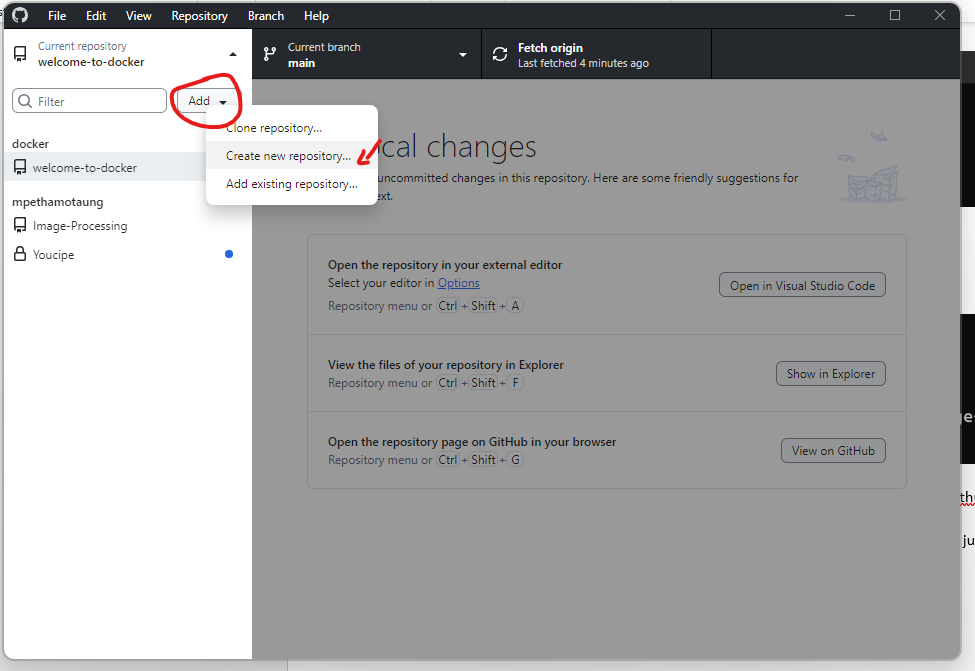
To navigate into Github folder in documents cd Github, then hit enter

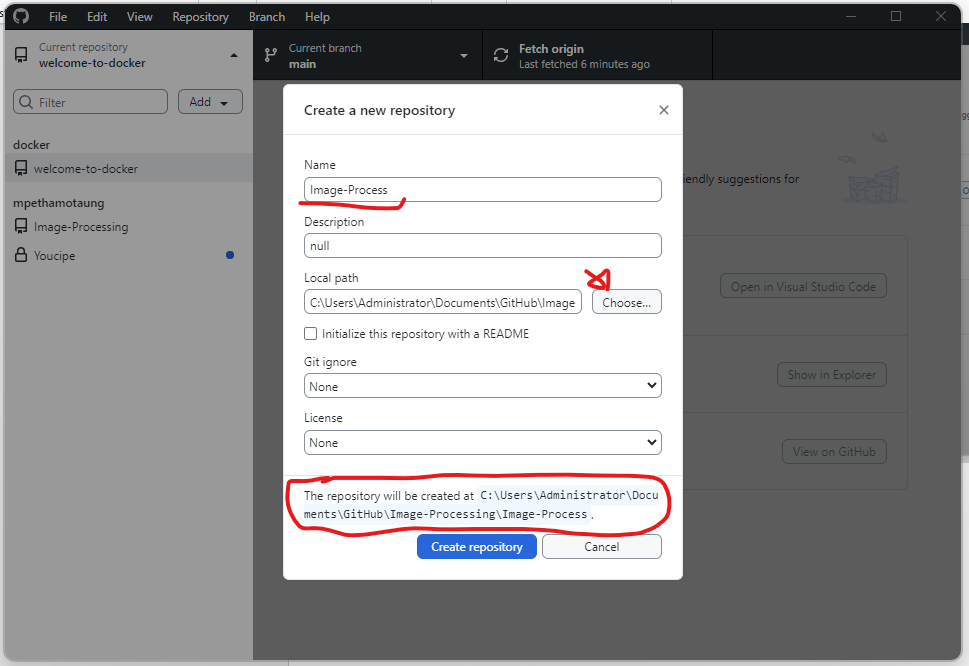
1. Create project folder for our Django project



This will create a folder for our Django project but it is still not connected to Github or any repository on Github

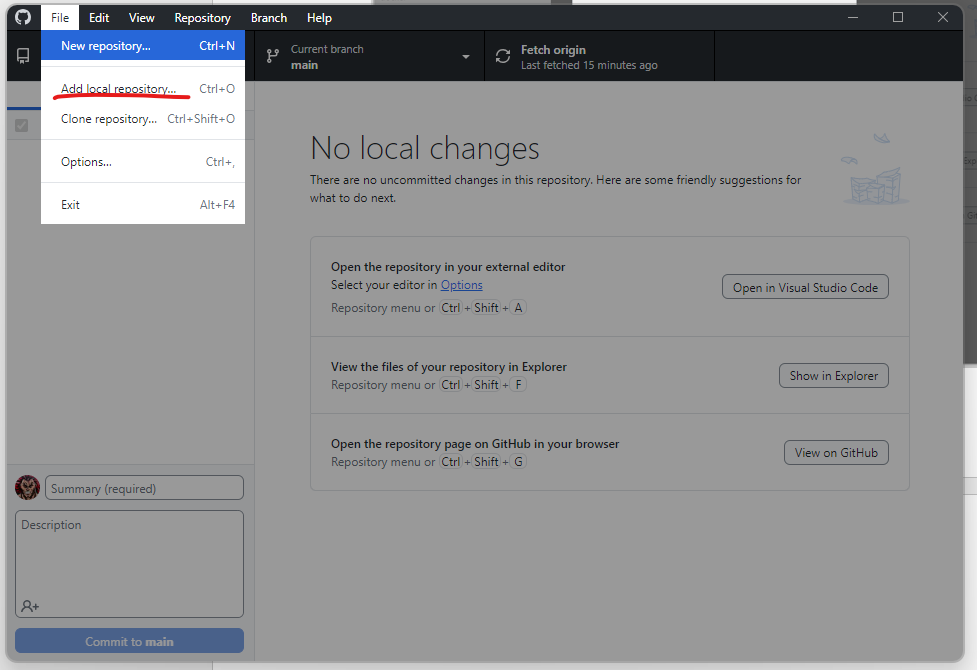
1. Next open Github desktop, create a new repository and link it to the folder we just created



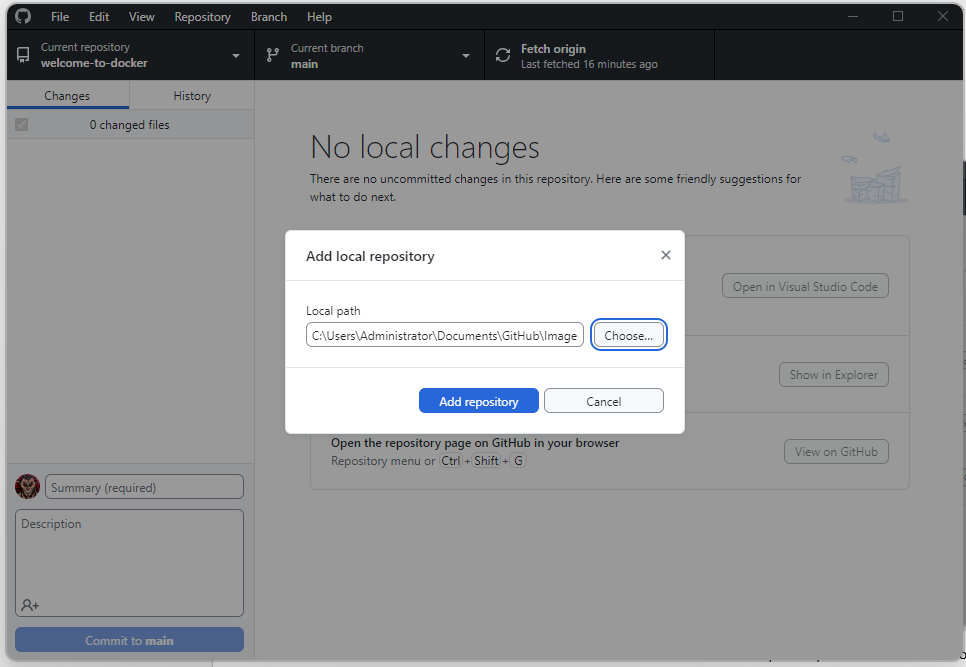


OR

Open Github Desktop, Go to File> Add local repository



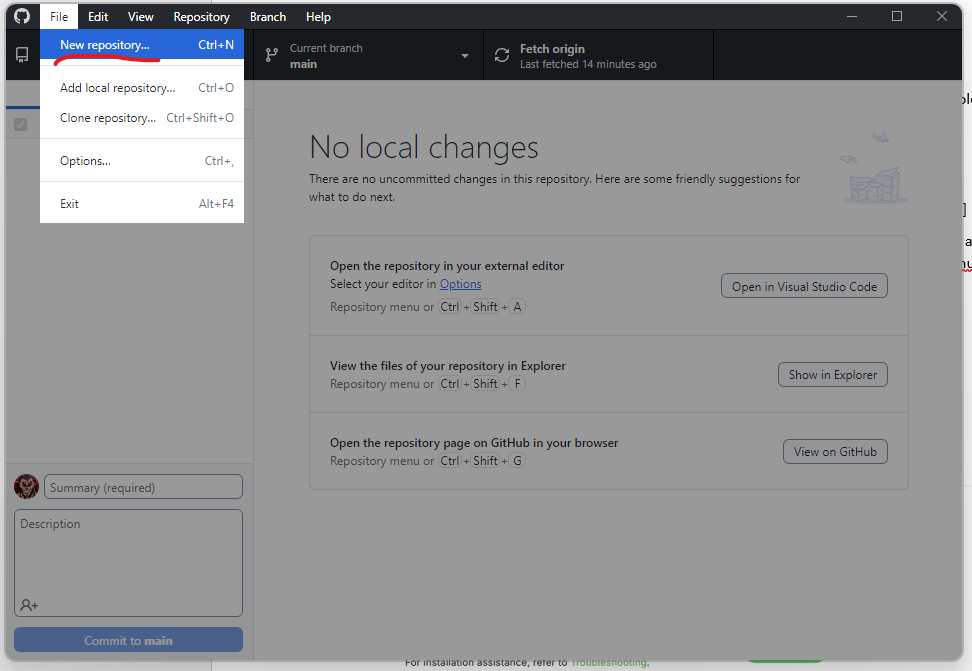
Point Github to the local repository/folder that you want to add to Github Desktop. Then publish to Github.



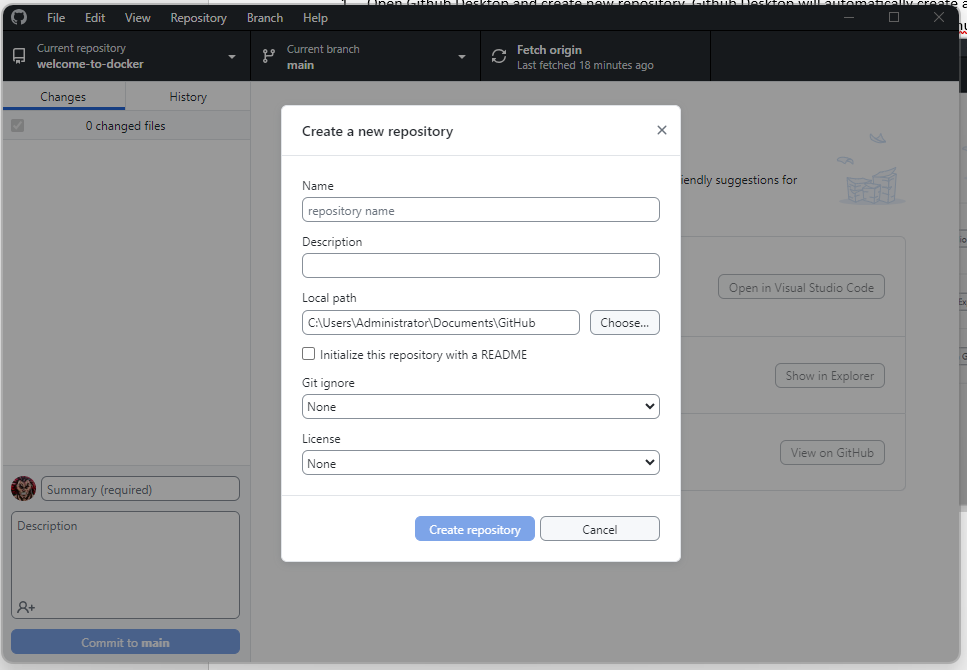
#### Alternatively (To save time)

\*Note\* You must have Github Desktop installed for you to proceed with this alternative method [[2](#_References)]

1. Open Github Desktop and create new repository. Github Desktop will automatically create a project folder on your local machine that will be used through and IDE to interact with Github.



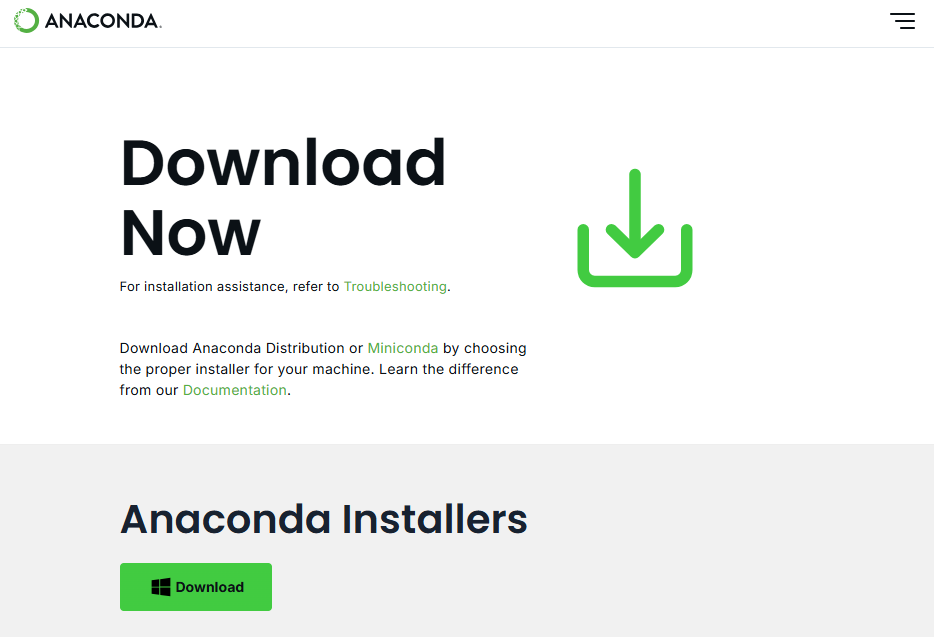
1. Give your repository a name, this will also automatically be the name of the folder that Github will create in your chosen path. Read up on Github Desktop, how to create and publish a repository [[3](#_References)]



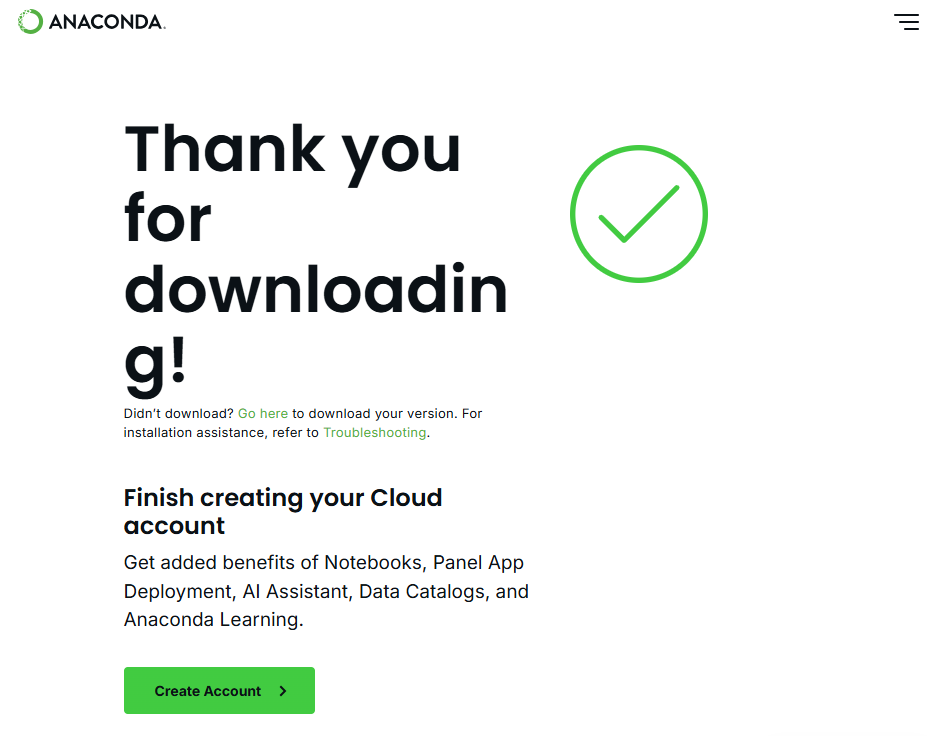
Now that our folder has been created and we have created a new repository and linked it to our folder we can move to the next step

### Downloading and Installing Anaconda

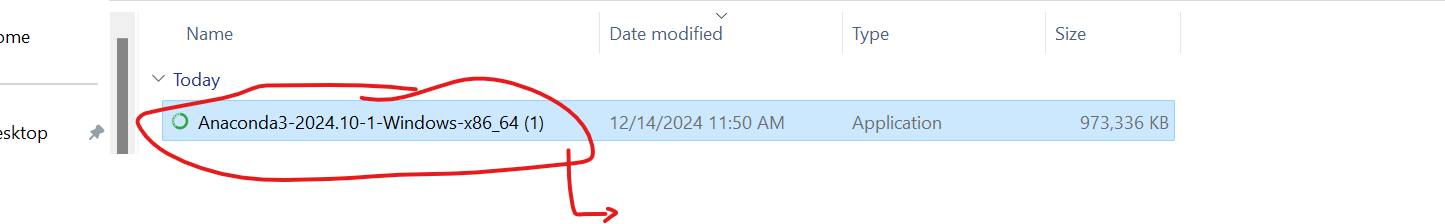
1. Click on the Download button

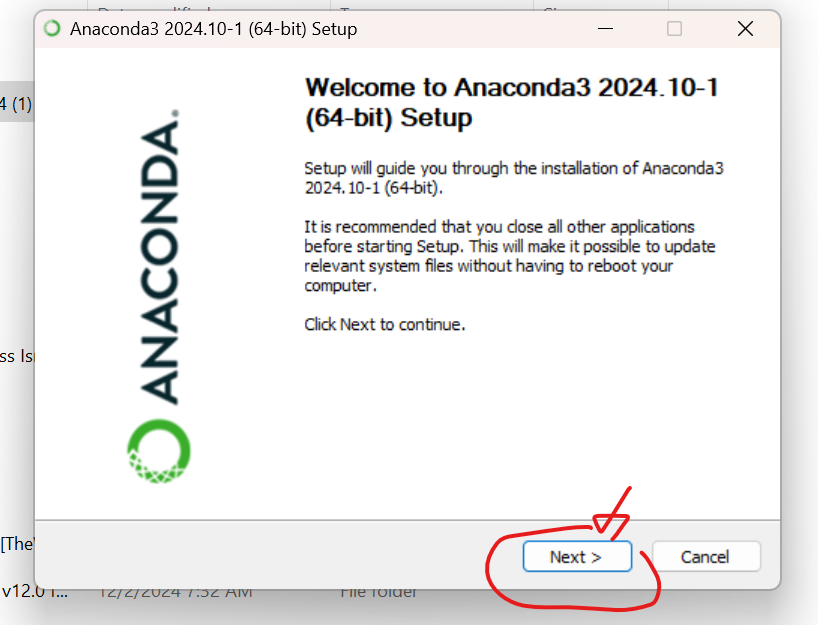


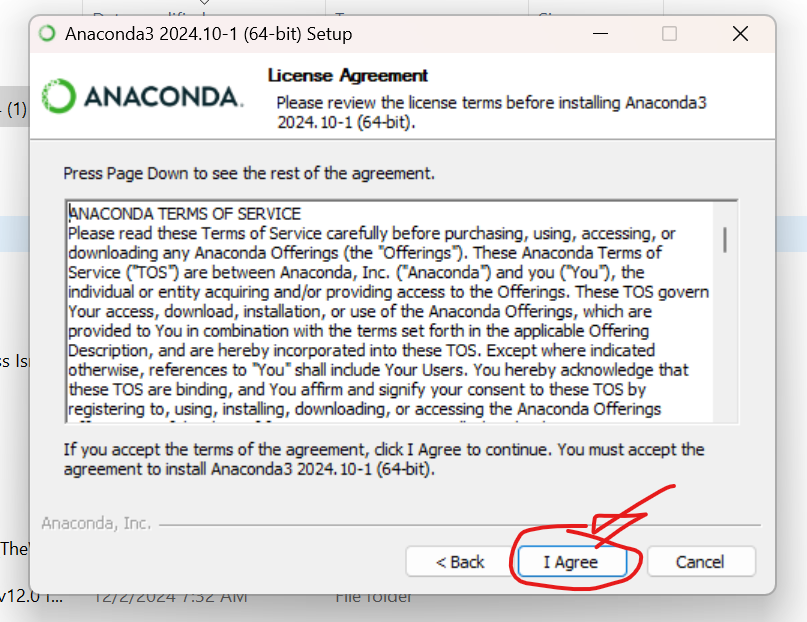
1. You’ll be redirected to this page and your download should start

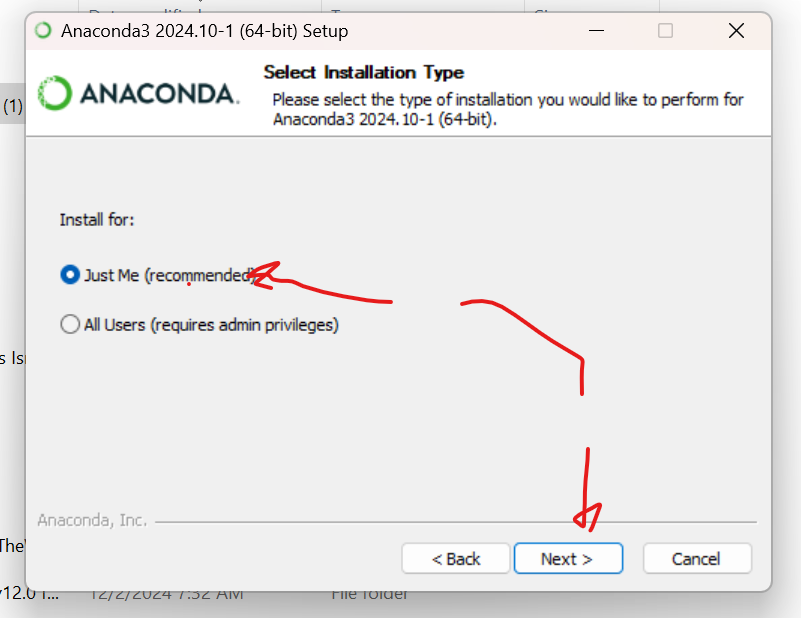


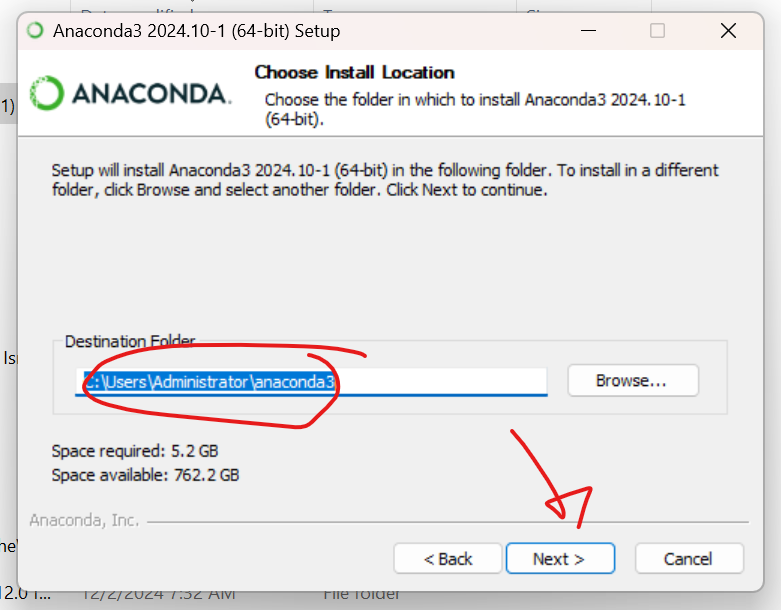
1. Once the file has finished downloading (/downloads), double click on the file and install Anaconda

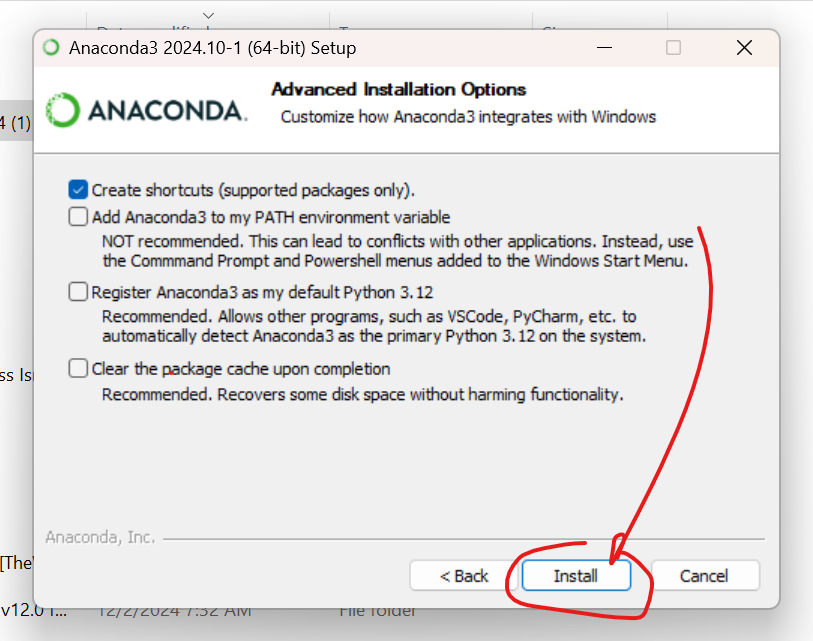






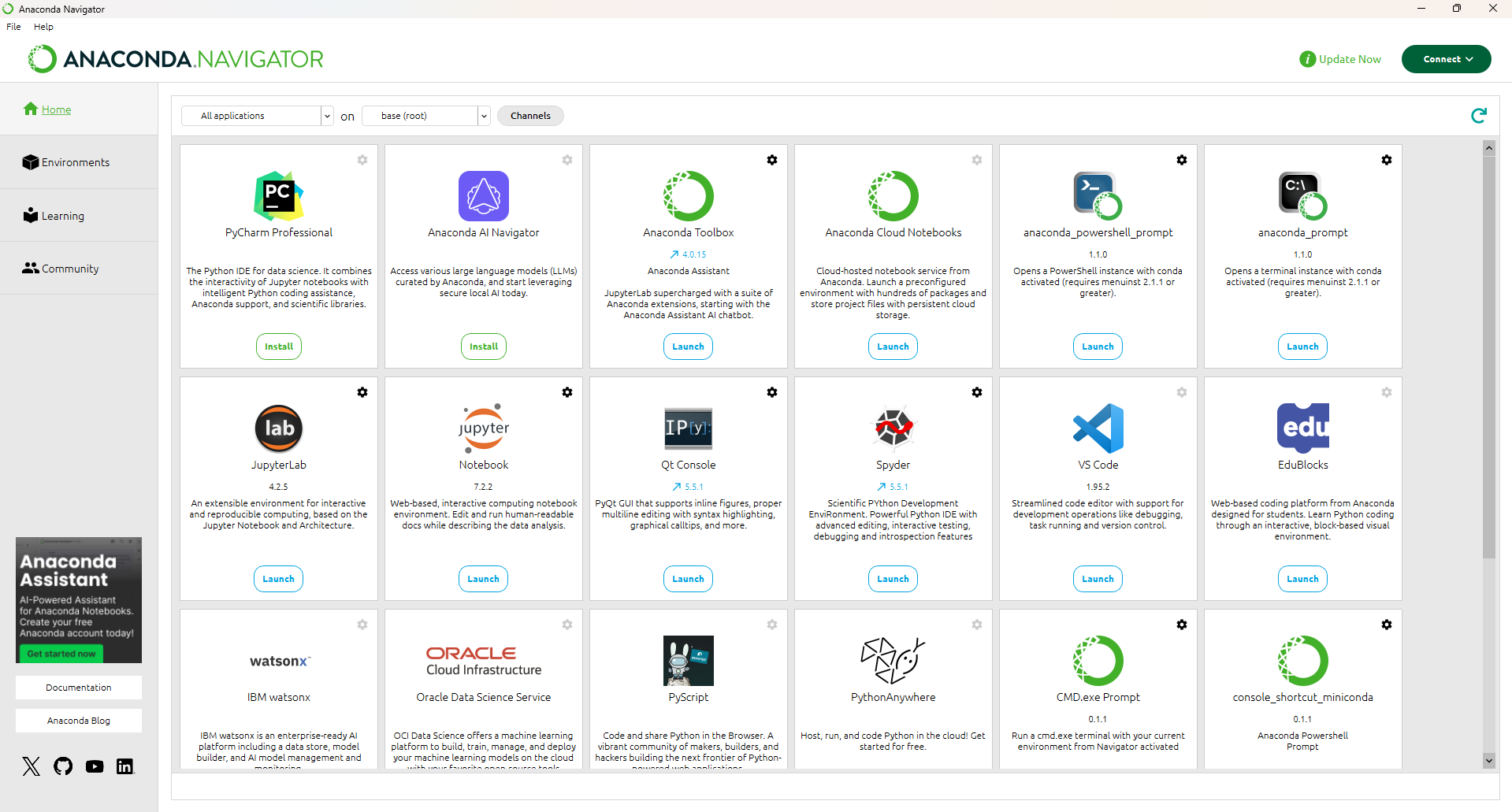






1. Now that you have downloaded and installed Anaconda. Let’s launch it for the first time

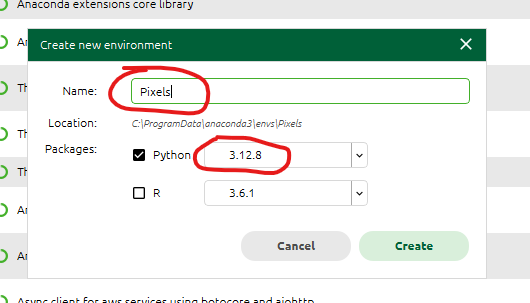
\*If you have never used Anaconda, I suggest you watch a few short tutorials (YouTube or on the Anaconda website). It’s quite user friendly\*



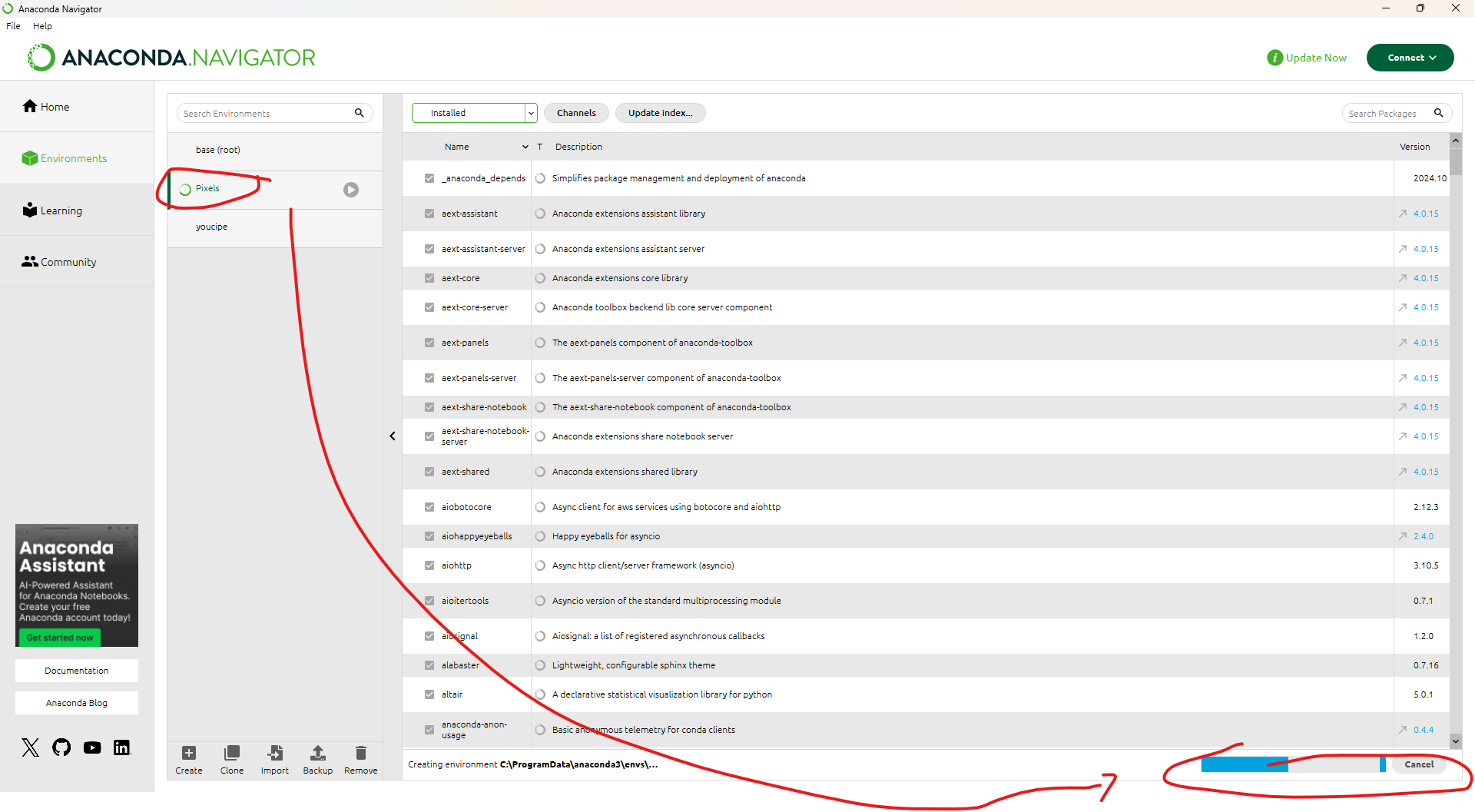
### Creating a virtual environment - Method 1 (GUI)

1. Open Anaconda
2. On the sidebar on the left click>environments
3. Then click create

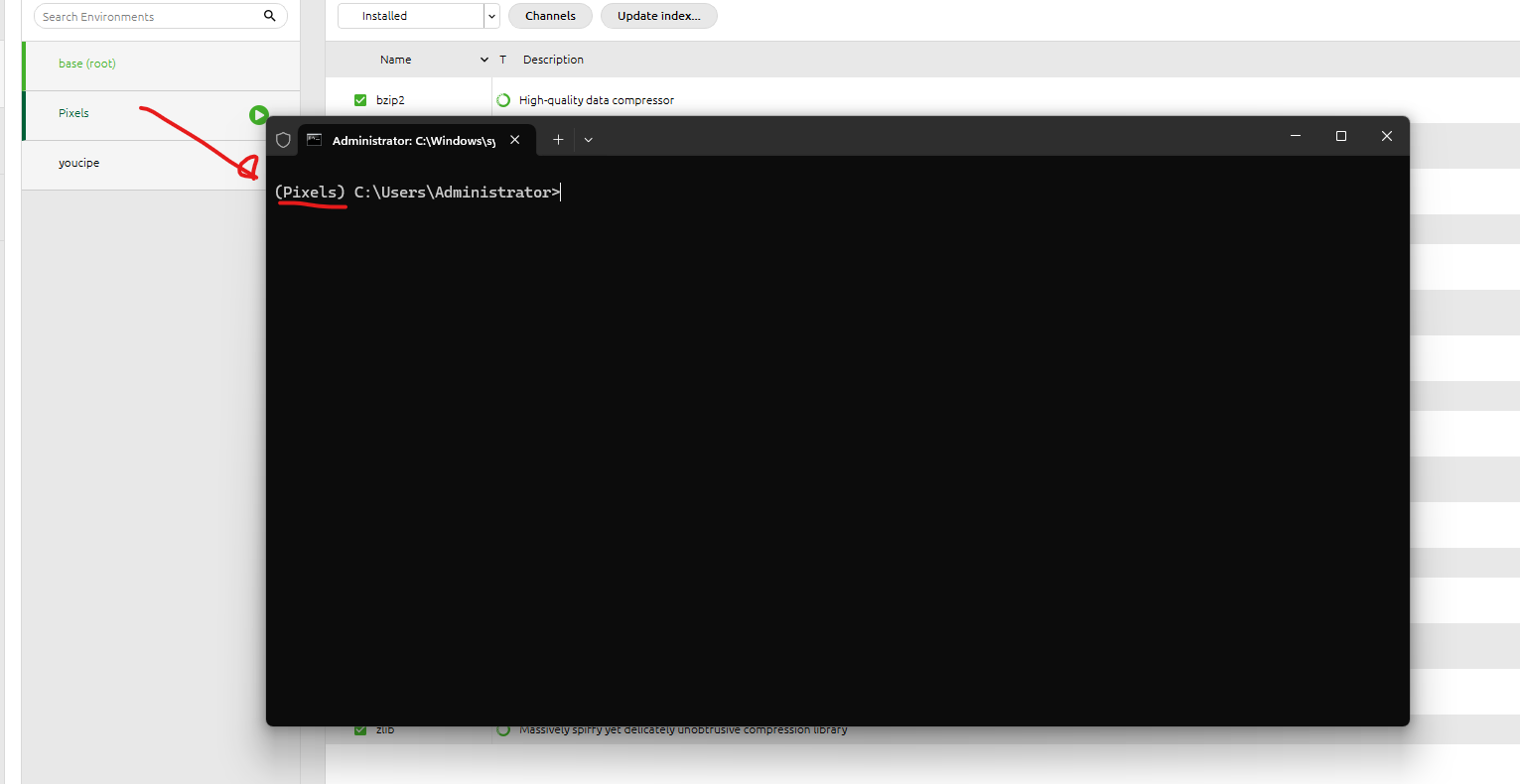
* Name your environment according to the project that you’re working on so that you can remember it. In this case I will name this one ‘Pixels’



1. Anaconda will now create a virtual environment with the python version and the name that you chose. It should take 2-5 minutes

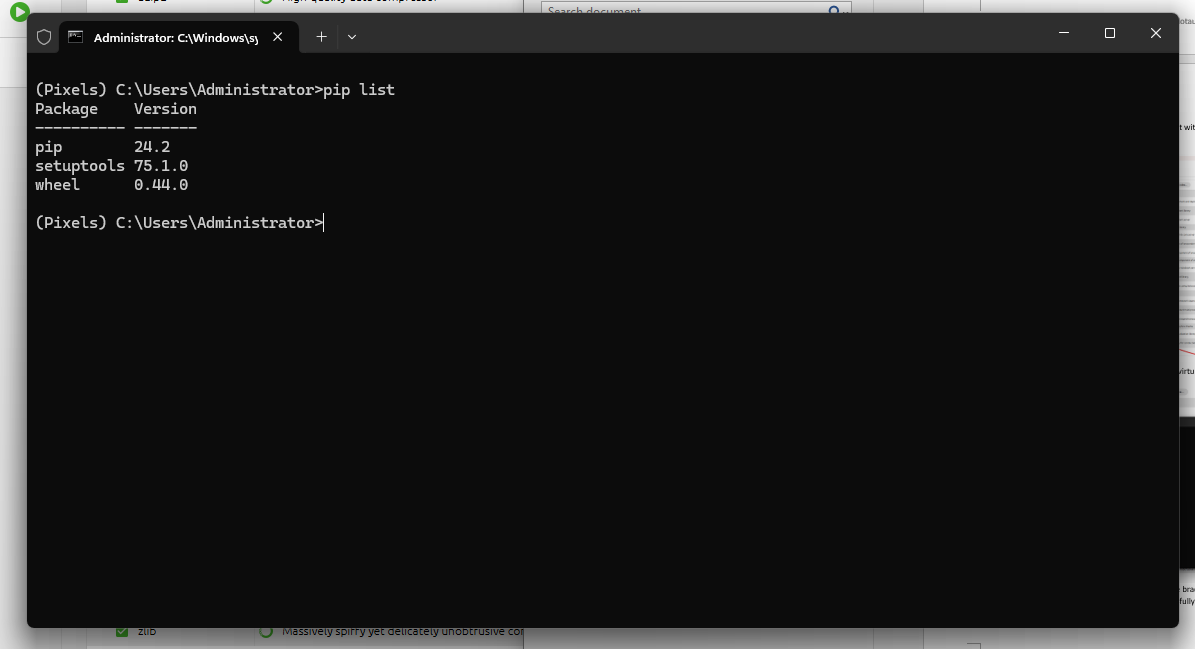


1. Next you can hit the play button to check your virtual environment. If it is created successfully, you should see the following:



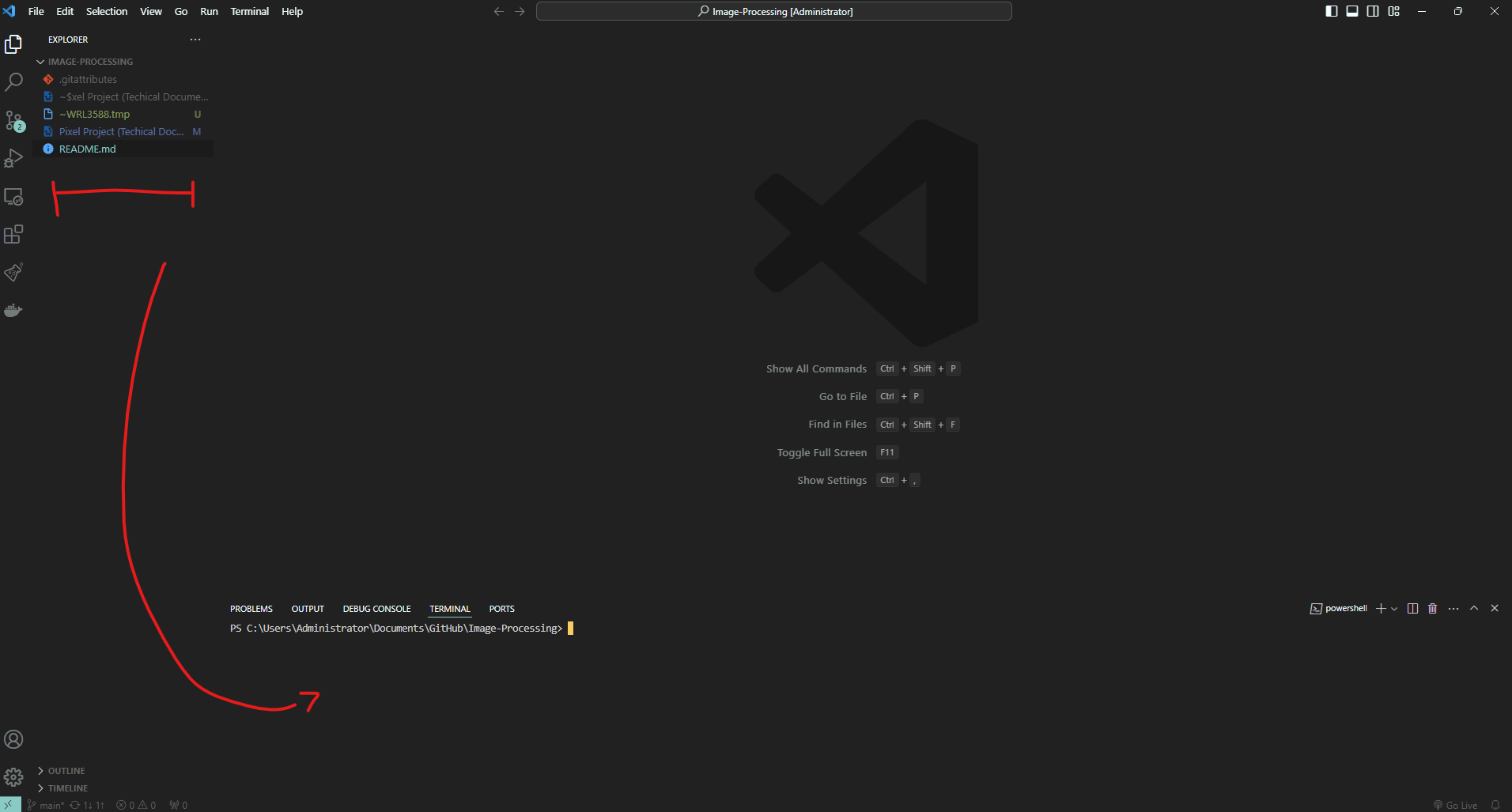
The name of our virtual environment inside the brackets confirm that our virtual environment was created successfully and is running successfully.

1. Type pip list in the terminal to see all the installed packages in the Pixels virtual environment. Since we do not have any additionally installed. We will only see the default packages that are built in(default) with every environment creation using pip

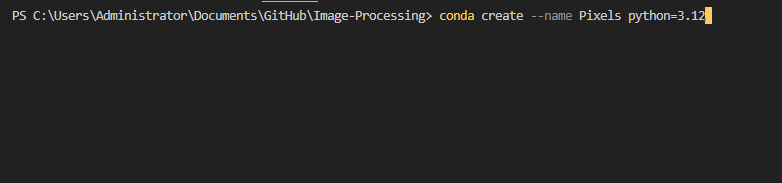


### Creating a virtual environment - Method 2 (CMD)

1. You can follow along with CommandPromt(CMD) or open the terminal window in an IDE of your choice. In this case, I’m going to use my preferred IDE, which is VSCODE

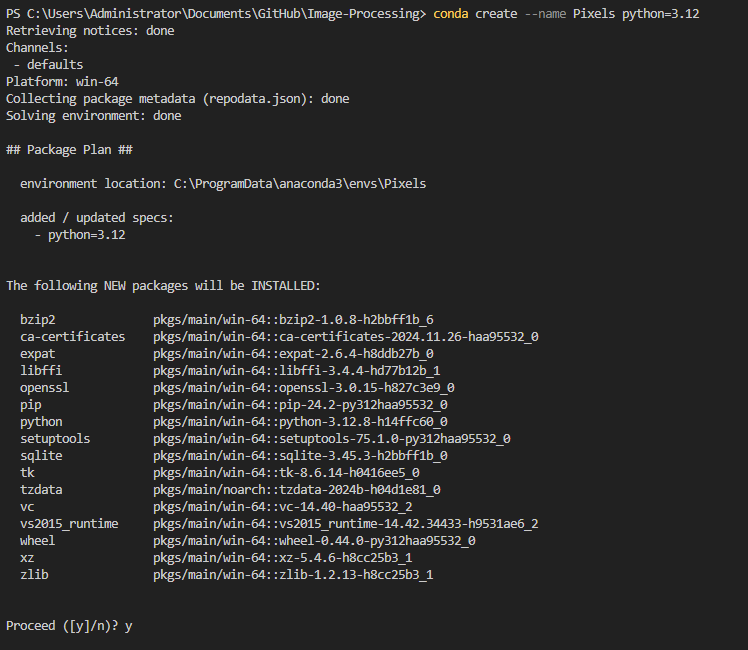


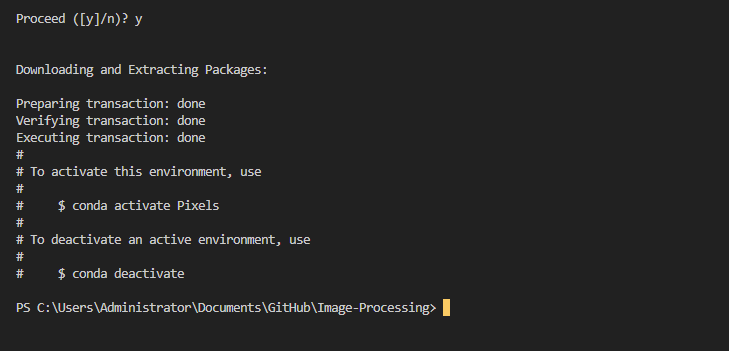
1. Navigate to the terminal and type the following command conda create –name (name of your environment) python=3.12



You can choose any name for your environment but to keep things simple and aligned, go with Pixels. Additionally, you can choose any python version to install on your virtual environment (i.e, python=3.8, python=2.7). Knowing this will be vital so that you can test your software in different versions of python to ensure consistency and to test for issues.

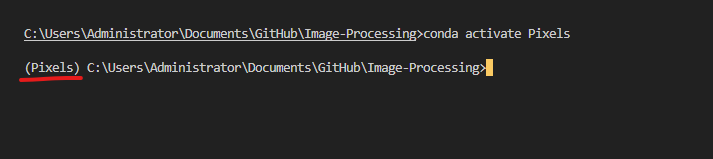
1. Conda will now create a virtual environment





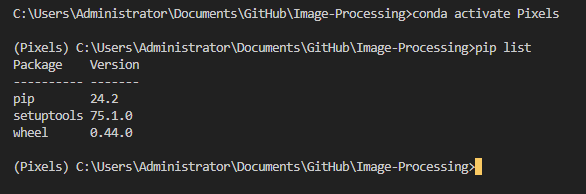
Choose ‘y’ to proceed to install the suggested packages in your environment.

1. Now that we have created our virtual environment let’s try to activate it and test if it has been successfully created. Run the following command conda activate Pixels



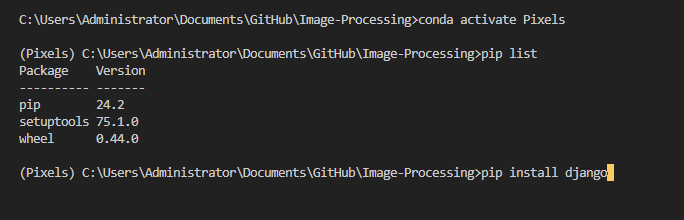
If you followed all the steps correctly, you should see (Pixels) which is showing that your virtual environment has been created successfully.

1. When you run the command pip list, you should see the default installed packages when a virtual env is initiated through Anaconda



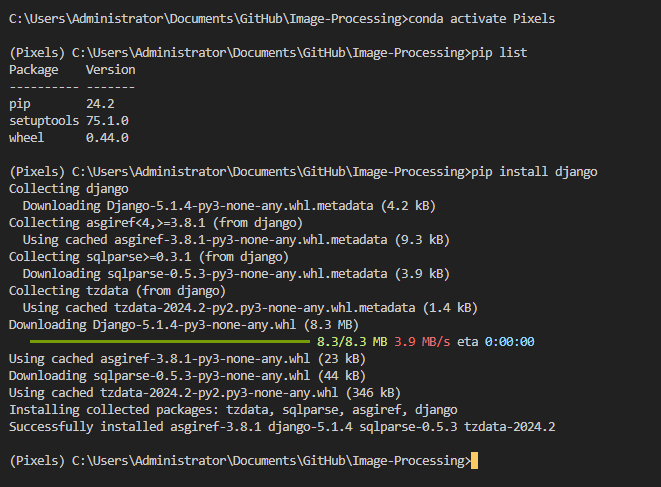
### Install Django

Now let’s install Django into our virtual environment. Run the following command pip install Django (\*note\*: if you do not specify the Django version, pip will install the latest Django version)



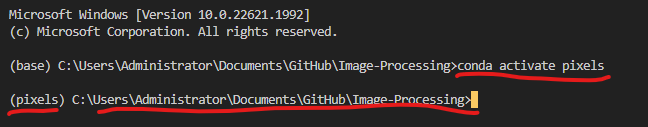
\*Ensure that when working on your project, that your virtual environment is active before running any commands or trying to install packages. Remember we do not want to modify our system packages thus we created a virtual environment. \*

If the above command runs successfully, you should see the following



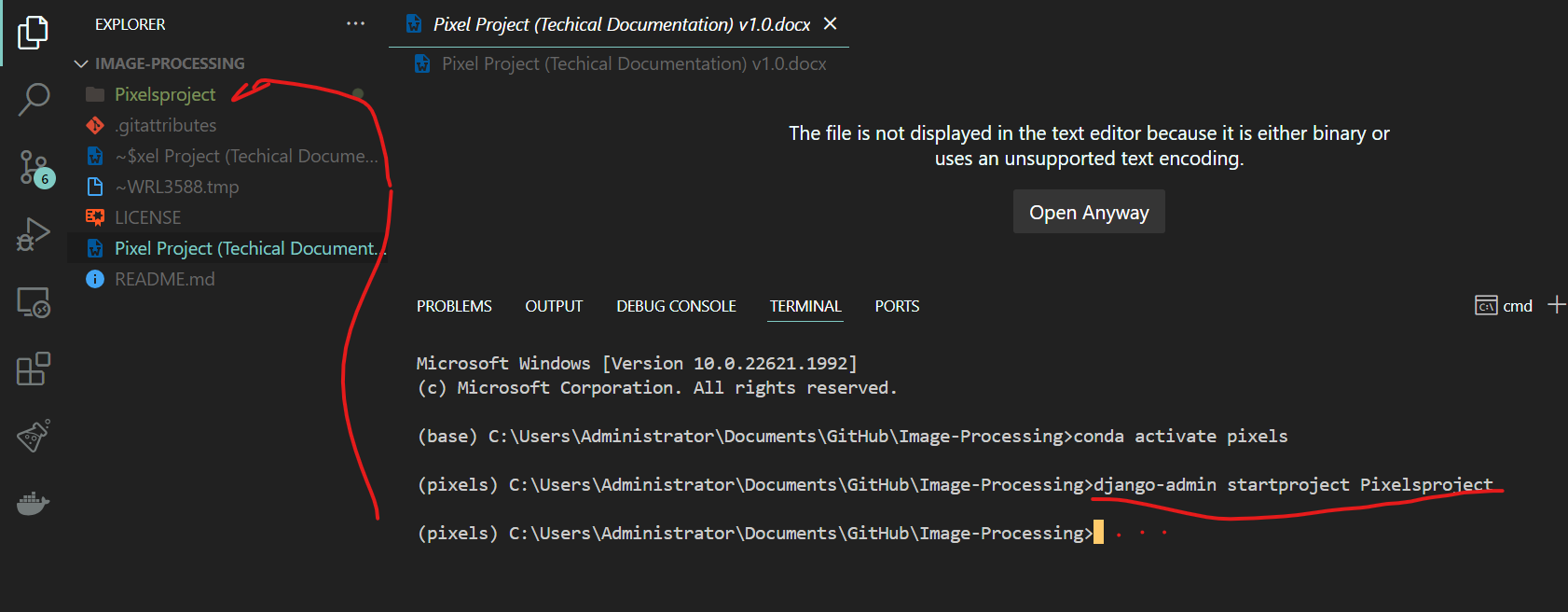
### Create Django Project

Now before we dive into it, let’s make sure that you’re in the correct directory. In my case I’m in Visual Studio code CMD terminal



1. Create Django project [[4](#_References)]

Run command django-admin startproject Pixelsproject



# Model Creation

# Image Upload & Processing

# Bonus Points

## Frontend Enhancement

## Error Handling

## Efficiency

# Evaluation Criteria

## Functionality

## Code Quality

## Efficiency

## Error Handling

## Bonus

# Architecture

# API

# Conclusion

# Appendix A

## References

[1] Creating directories through CLI

<https://www.ibm.com/docs/en/aix/7.1?topic=directories-creating-mkdir-command>

[2] Installing Github Desktop

<https://docs.github.com/en/desktop/installing-and-authenticating-to-github-desktop/installing-github-desktop>

[3] Creating a repository using Github Desktop

<https://docs.github.com/en/desktop/overview/creating-your-first-repository-using-github-desktop>

[4] Create Django project

https://docs.djangoproject.com/en/5.1/intro/tutorial01/

[ ] How to create a conda virtual environment

https://saturncloud.io/blog/how-to-create-a-conda-environment-with-a-specific-python-version/

[ ] Image Field

<https://docs.djangoproject.com/en/5.1/ref/forms/fields/>

[ ] Bind Uploaded files to form

<https://docs.djangoproject.com/en/5.1/ref/forms/api/#binding-uploaded-files>

[ ] Pillow (needed for Image Field usage)

<https://pypi.org/project/pillow/>

# Appendix B

## Figures