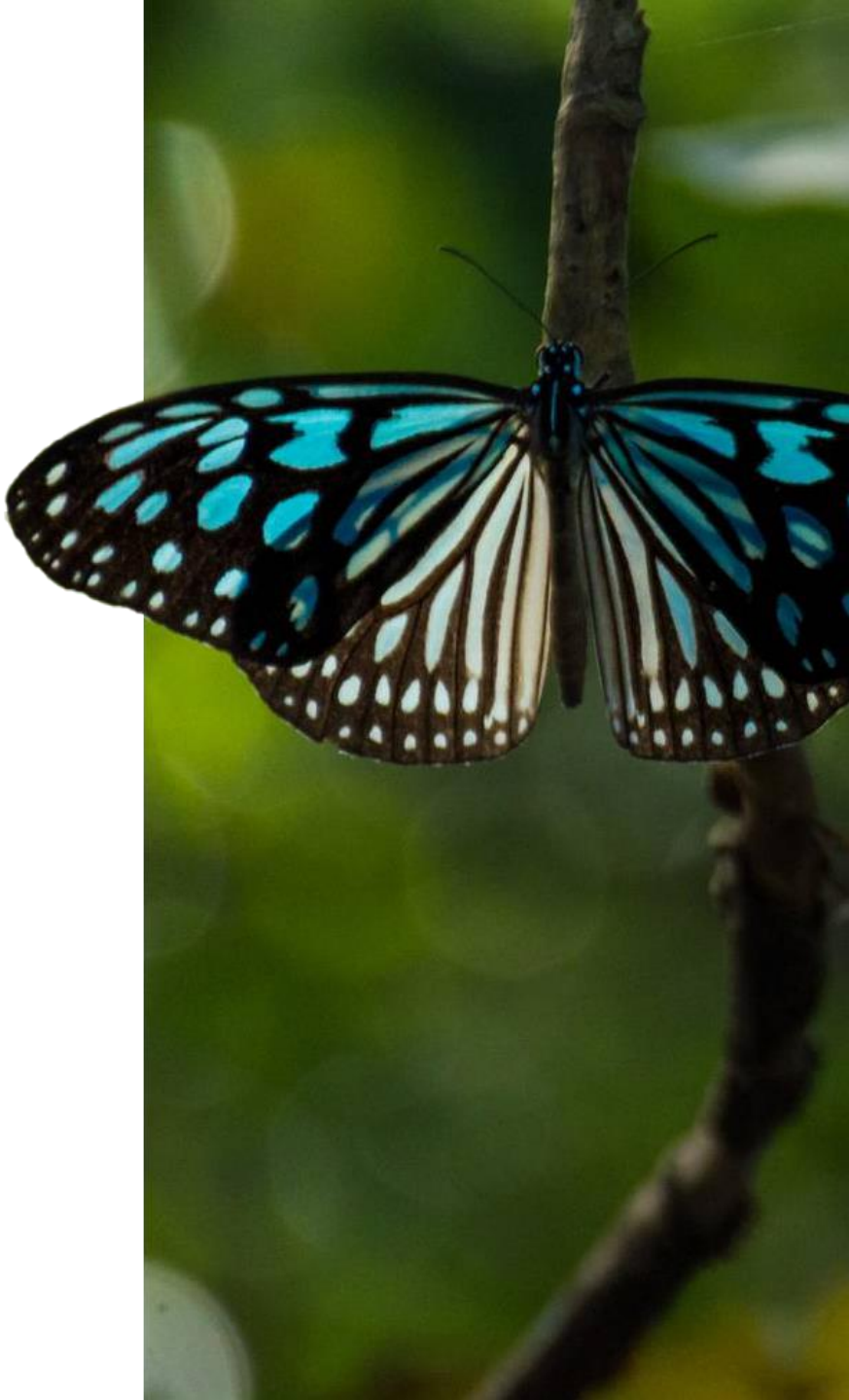
EVERYTHING YOU NEED TO KNOW TO TAKE THE FREE COURSE, CHATGPT PROMPT ENGINEERING FOR DEVELOPERS

VERSION 1.0, JULY 2023

A great way to learn about ChatGPT and Large Language Models – and you don't have to be a developer to take the course!

Mike Boucher



training description

This presentation will provide a recipe for turning ChatGPT **unknown unknowns** into **known unknowns** into **knowns** by leveraging the free 1-hour online course from DeepLearning.AI called, ChatGPT Prompt Engineering for Developers.

After this presentation, you'll have the knowledge and a roadmap to begin your ChatGPT learning journey.

Everything you need to know to take the free course, ChatGPT Prompt Engineering for developers

outline

01 Introduction

02 Accounts Setup

03 (Optional) Computer Setup

04 (Optional) Using Jupyter Notebook

Presentation in a Nutshell

- Intro to ChatGPT with a brief overview of the course
- Create a ChatGPT account
- Sign up for the course at Deeplearning.AI

The rest is optional

- Install Anaconda
- Install 3 libraries in the Anaconda Python 3.10 instance
  - python-dotenv
  - openai
  - Redlines
- Get and install an OpenAI key on your PC
- Doing simple operations in Jupyter Notebook

lesson 01

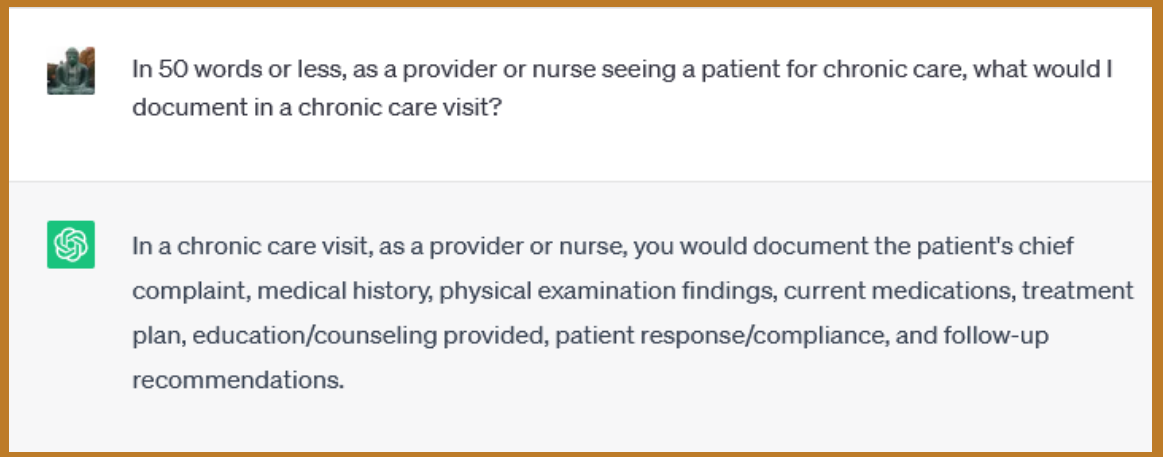
Introduction to ChatGPT and the course, ChatGPT Prompt Engineering for Developers

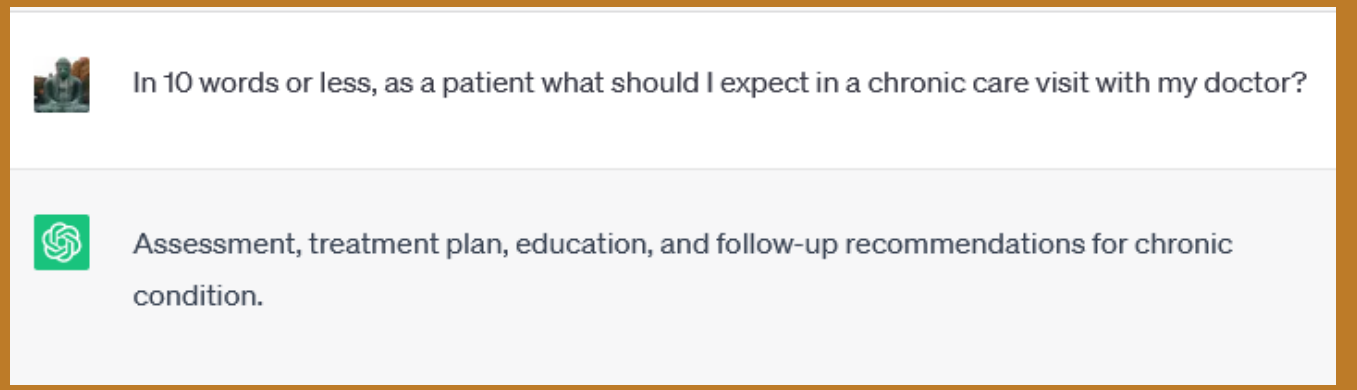
What is ChatGPT?

- ChatGPT is a large language model (LLM) that can engage in human-like conversations
- ChatGPT is trained on a massive text dataset, including books, articles, and websites
- There are 2 ways to interact with ChatGPT
  - Web Interface (with a browser like doing a Google search)
  - API Interface (with a software program)

Prompts and Responses

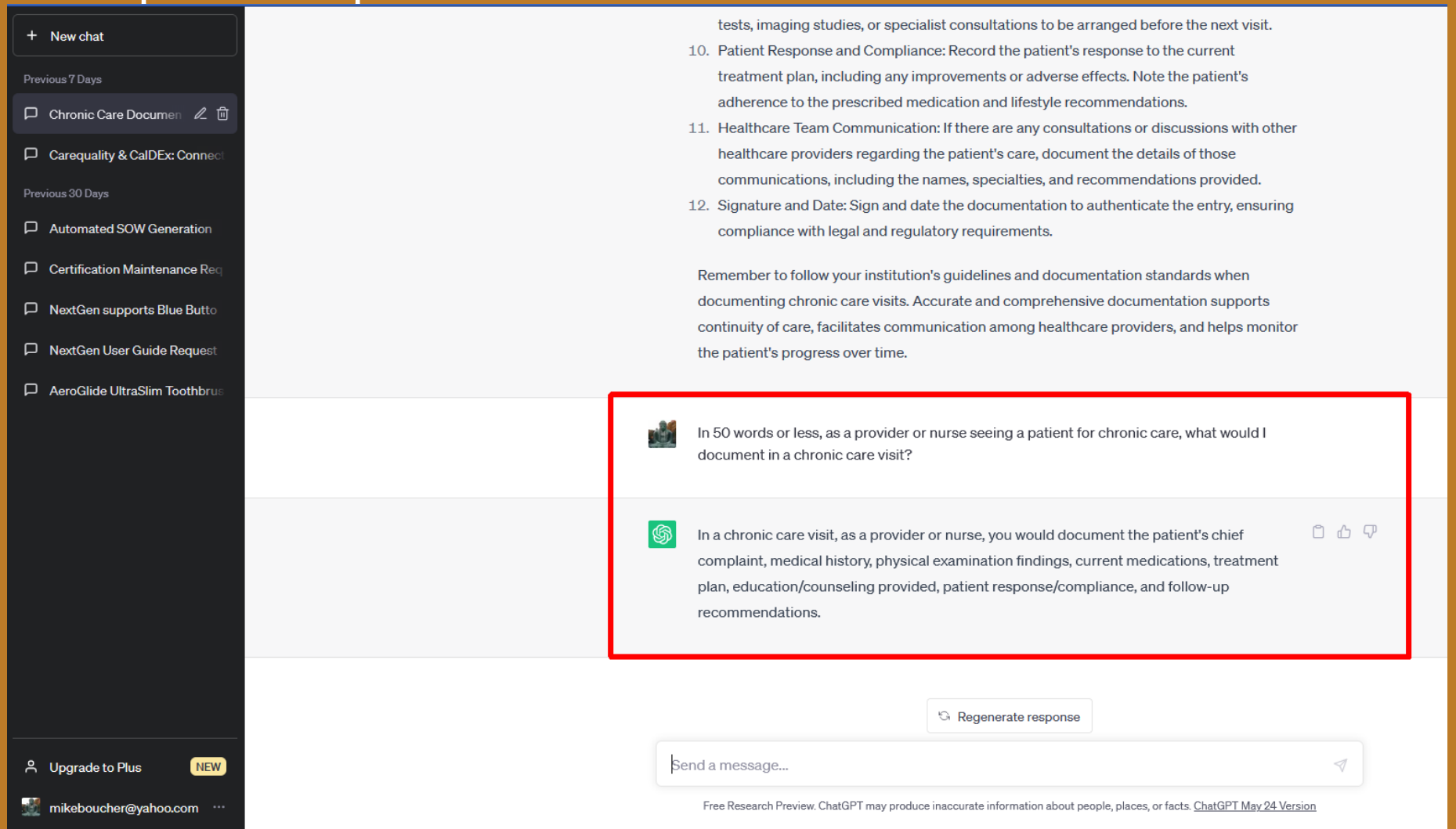
- Prompt – the query that is input to ChatGPT
- Response – the reply from ChatGPT
- Prompt engineering is the process of creating and reviewing high-quality prompts to guide language models.





ChatGPT Web Interface

https://chat.openai.com/



ChatGPT API Interface

jupyter 5 Transforming Last Checkpoint: 05/16/2023 (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

In [1]:
```
import openai
import os

from dotenv import load_dotenv, find_dotenv
_ = load_dotenv(find_dotenv()) # read local .env file

openai.api_key  = os.getenv('OPENAI_API_KEY')
```

In [2]:
```
def get_completion(prompt, model="gpt-3.5-turbo", temperature=0):
    messages = [{"role": "user", "content": prompt}]
    response = openai.ChatCompletion.create(
        model=model,
        messages=messages,
        temperature=temperature,
    )
    return response.choices[0].message["content"]
```

In [3]:
```
prompt = f"""
Translate the following English text to Spanish: \
```Hi, I would like to order a blender```
"""
response = get_completion(prompt)
print(response)
```

Prompt

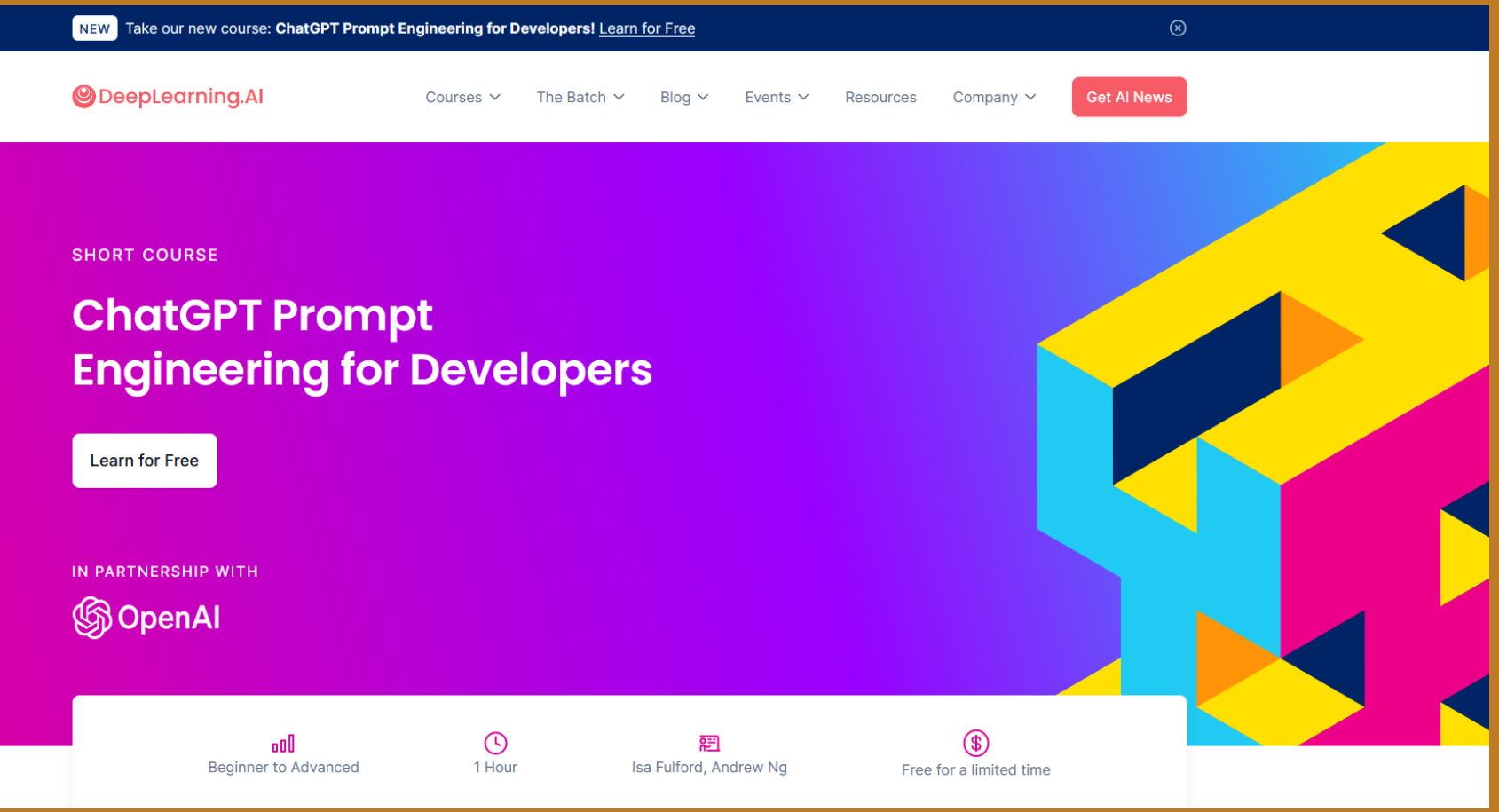
```
Hola, me gustaría ordenar una licuadora.
```

Response


```

ChatGPT Prompt Engineering for Developers

https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/

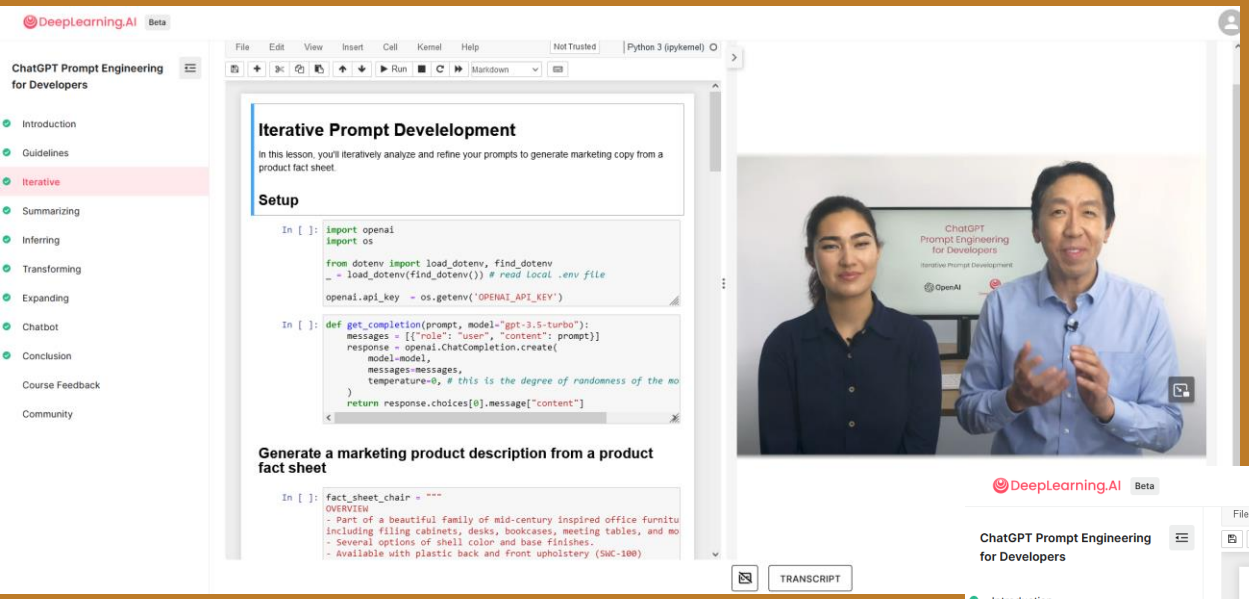


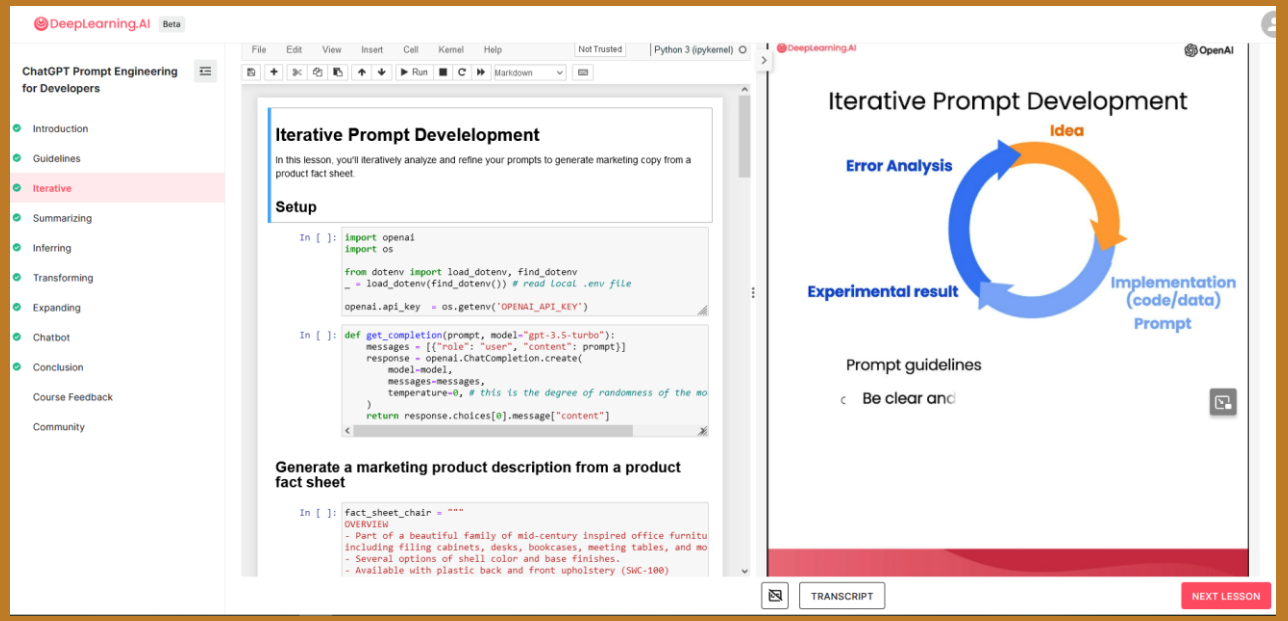
1-hour course

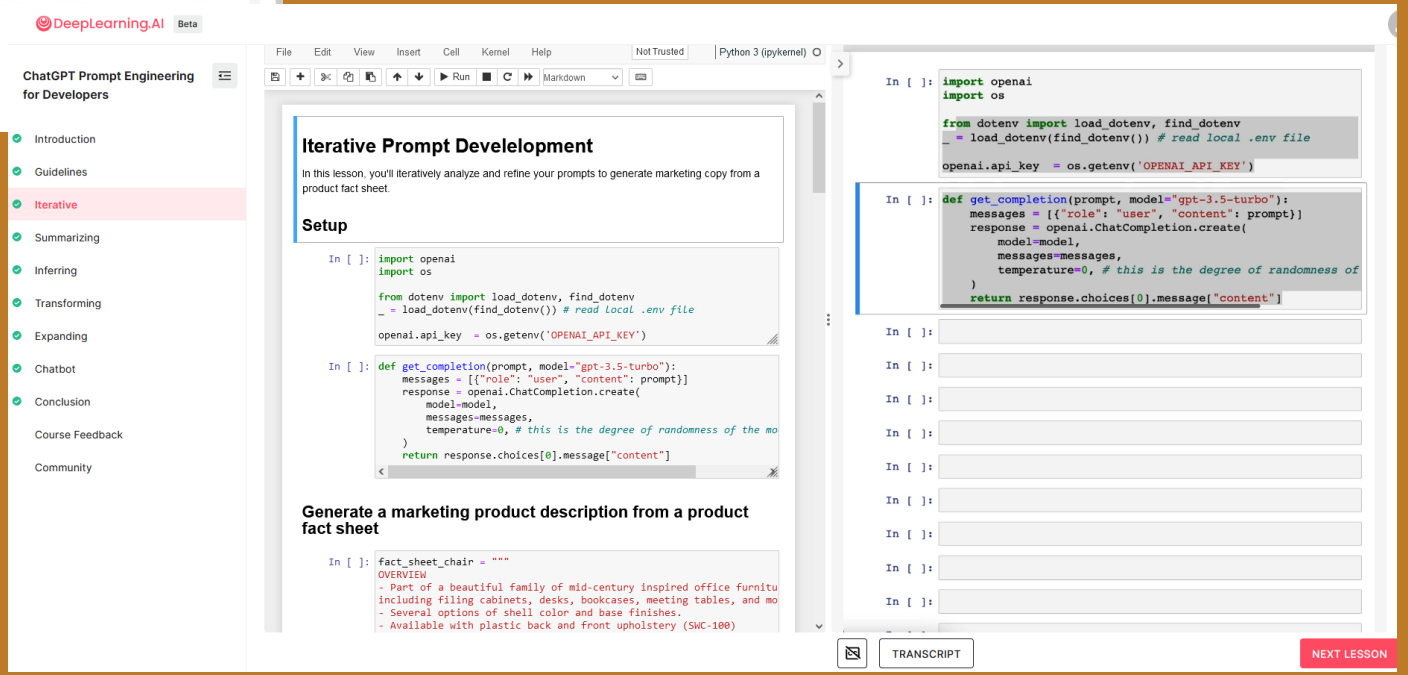
- Explains what ChatGPT is and different ways to use it
- Demonstrates the concepts with simple code
- Provides code examples so you can follow along and experiment on your own

Everything you need to know to take the free course, ChatGPT Prompt Engineering for developers

Course Format







How to Use the Course

- Listen to Isa Fulford and Andrew Ng
- Watch
- Do

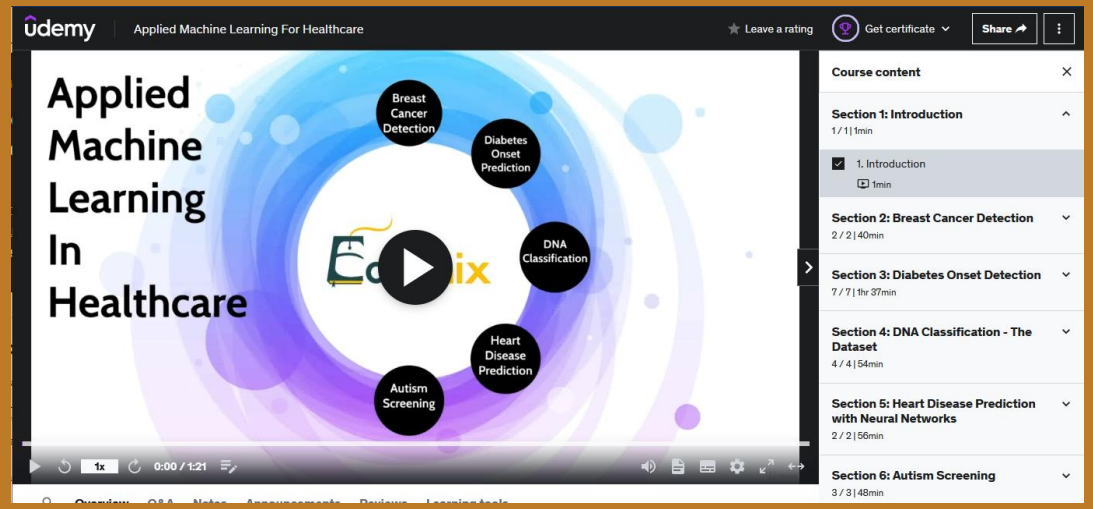
Everything you need to know to take the free course, ChatGPT Prompt Engineering for developers

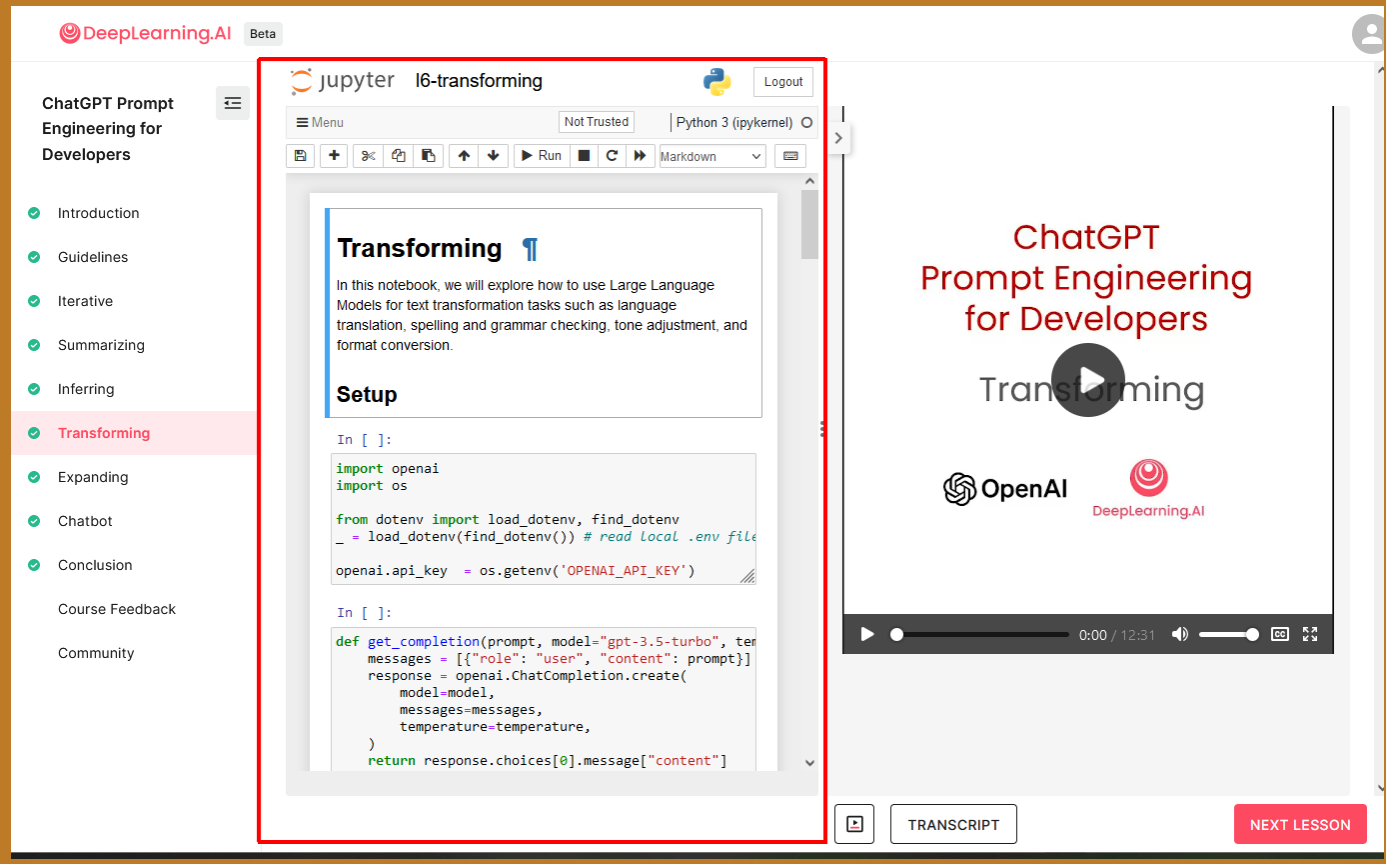
ChatGPT Concepts in the Course

- **Iterating your prompts** – how to refine your prompts to get what you want
- **Summarizing** – using ChatGPT to summarize information
- **Inferring** – using ChatGPT to infer the emotion of a written product review
- **Transforming** – using ChatGPT to translate text to a different language or to a certain writing style or to correct grammar
- **Expanding** – Using ChatGPT to take a short piece of text and generate a longer piece of text such as an email response or an essay
- **Building an AI Chatbot** – building a chatbot to take restaurant orders

Running Examples in the Course

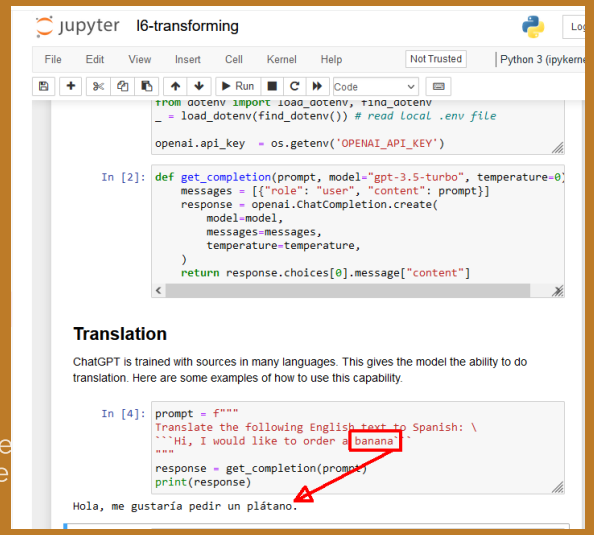
- The course embeds a tool called Jupyter Notebook which allows you to view, edit, and run the (Python) code examples from the course
- Jupyter Notebook is used by many data scientists to develop machine learning models using the Python programming language with specialized machine learning libraries and tools
  - E.g., Applied Machine Learning in Healthcare

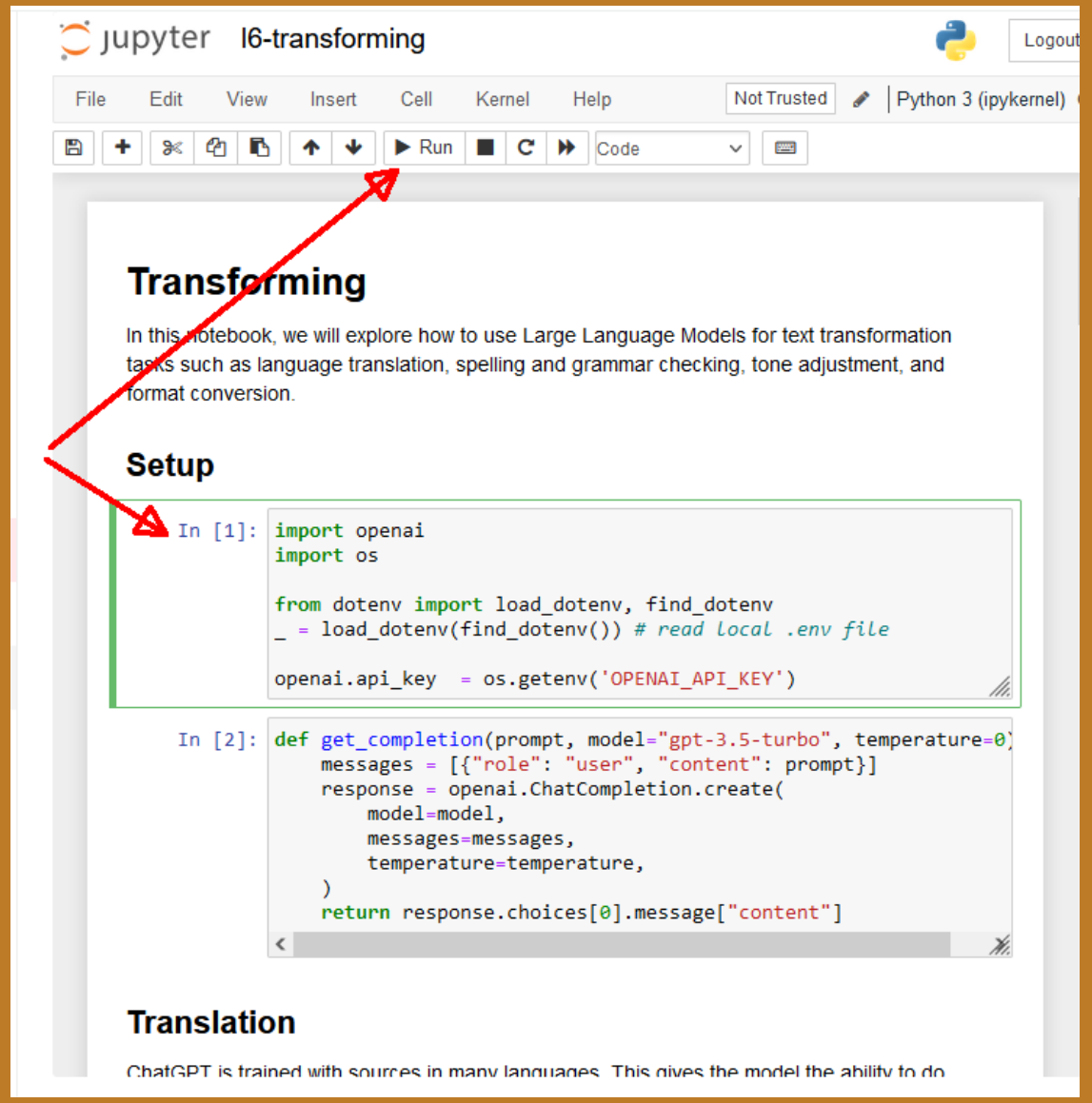




Tips for Using Jupyter Notebook

- Always start in the top (first) cell
- Use the ▶ Run control to execute code in a cell
- If the cell calls to ChatGPT, allow a few seconds for a response to show up
- If you don't get a response after a minute or 2, try rerunning the cell or start from the top again
- If the Jupyter notebook window is empty, refresh your browser
- Use the left button of your mouse to select or go back to an earlier cell
- Edit the code in a cell and try running that





lesson 1 summary

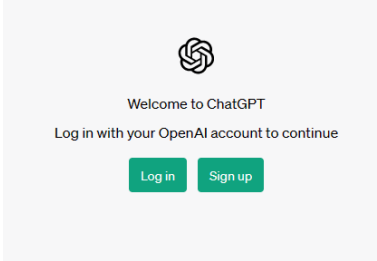
- ChatGPT is a large language model (LLM) that can engage in human-like conversations using a Prompt & Response mechanism
- You can interact with ChatGPT via a web browser (like a Google search) or via an API with a software program
- *ChatGPT Prompt Engineering for Developers* is a 1-hour course from a company called DeepLearning.AI that uses simple code examples, running in Jupyter Notebook, that utilize the ChatGPT API to teach the concepts for using ChatGPT
  - Jupyter Notebook allows you to run and modify the code from the course

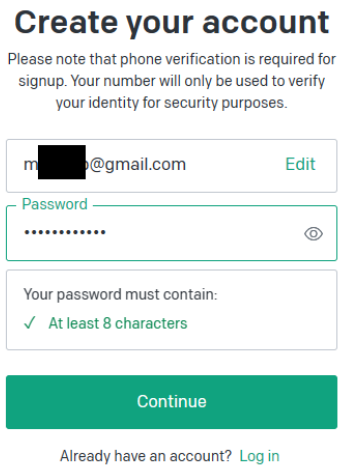
lesson 02

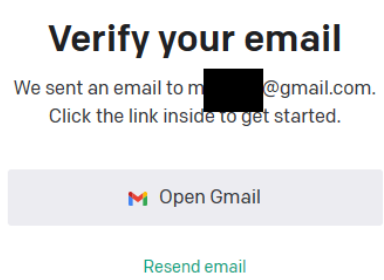
Creating Your Accounts with OpenAI and DeepLearning.AI

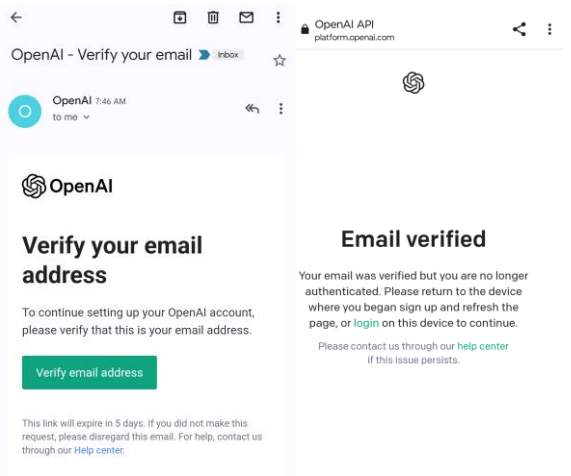
Creating your OpenAI Account

1. Go to https://chat.OpenAi.com
2. Sign up



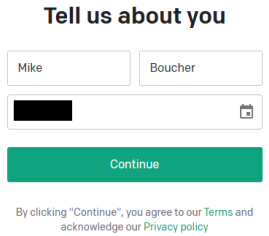


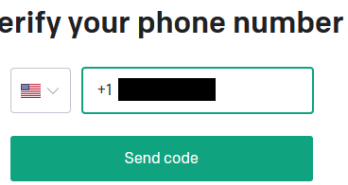


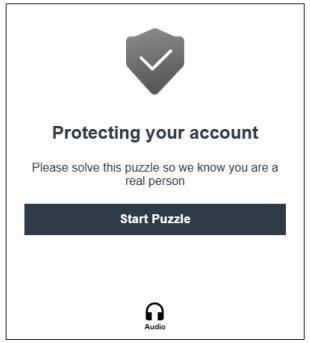


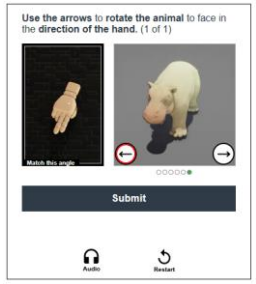
I had problems creating my account. I kept getting the error, “Signup is currently unavailable, please try again later.”

The way I got it to work was to complete the signup on my phone, with Firefox in incognito mode. After that I was able to login normally with Firefox on my PC.







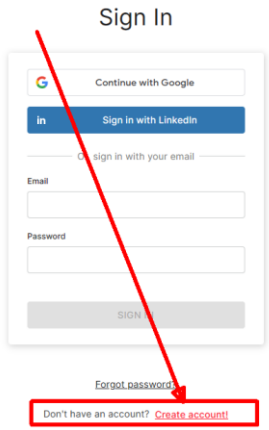


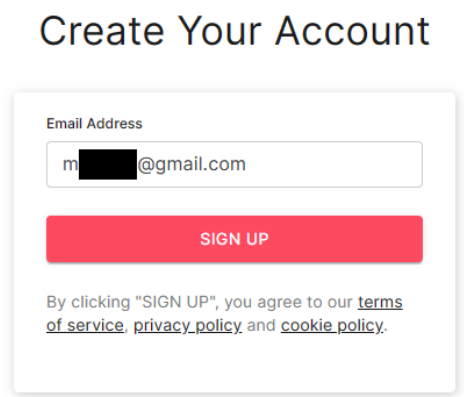
SMS Verification

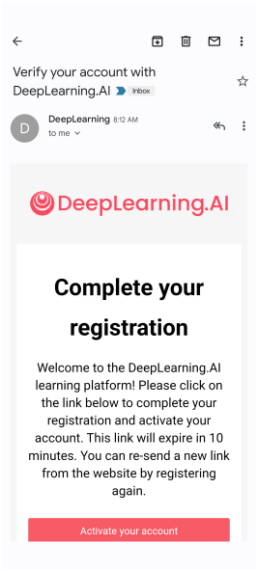
Course Signup

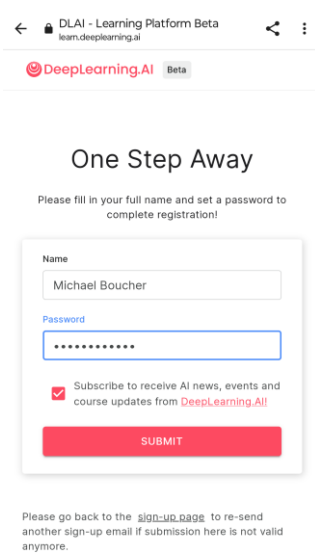
1. Go to ChatGPT Prompt Engineering for Developers - DeepLearning.AI
2. Select 'Learn for Free' and follow the prompts

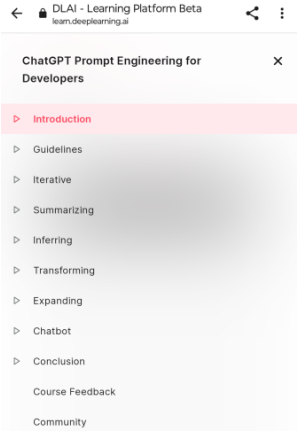
https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/











Everything you need to know to take the free course, ChatGPT Prompt Engineering for developers

lesson 2 summary

- At this point you've successfully created a ChatGPT account & have signed up for the course, *ChatGPT Prompt Engineering for Developers*
- You are now ready to take the course and play with the course's code examples using the Jupyter Notebook that is running inside the course.
- OPTIONAL: If you want to run and experiment with the code samples outside the course, then you'll need to install a few things on your computer

- *In the next lesson we're going to get your computer set up so that you can run all the examples and follow along with the with course using Jupyter Notebook separate from the course. This will enable you to save and rerun the course code on your own at any time without being logged into the course.*

lesson 03

Setting up Your computer to follow along with the course

Lesson 3 topics

Install Anaconda/Jupyter Notebook

Install 3 libraries
- python-dotenv
- openai
- Redlines

Setup your OpenAI key

Why do I have to install stuff on my computer?

- The course runs Python code examples using a tool called Jupyter Notebook
  - Jupyter Notebook is a handy way to write programs in small steps
  - Jupyter Notebook (and Python) are used a lot by Data Scientists
  - Anaconda is the tool you install to get Jupyter Notebook
- The course uses a few Python libraries that don't come with a stock Python install
  - We're going to install the libraries ahead of time, so you don't get errors when you are follow along in the course
- To connect your program to ChatGPT you need an API key that's unique to you, so you'll get the key from OpenAPI and install it on your computer

topic 1 of 3

Installing Anaconda / Jupyter Notebook

Steps

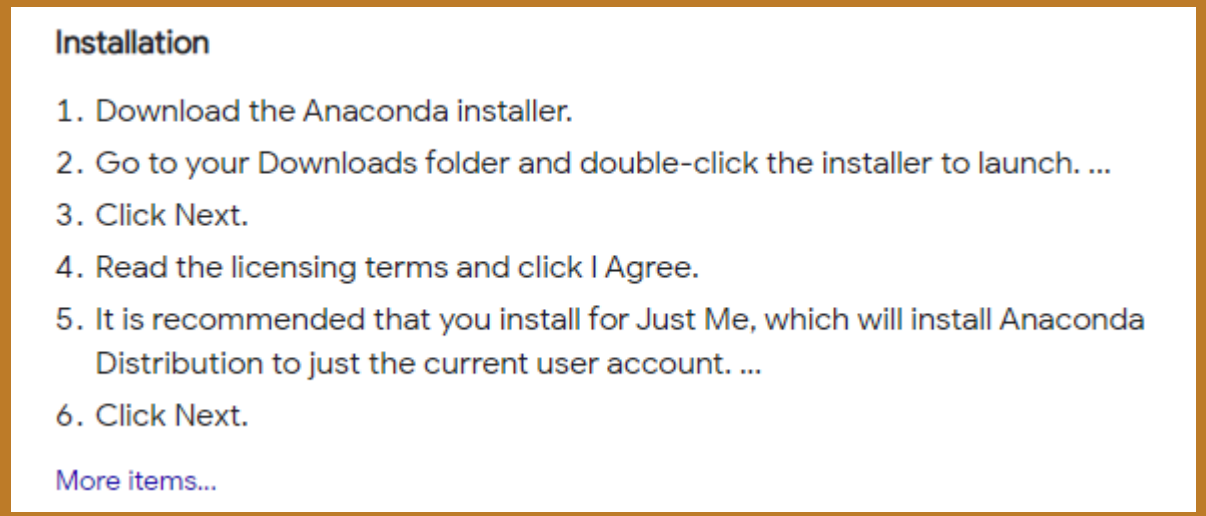
1. Install Anaconda
2. Launch Anaconda
3. Create a working project folder
4. Launch Jupyter Notebook
5. Create your 1st Jupyter Notebook

Progress

1. **Install Anaconda / Jupyter Notebook**
2. Install 3 Python libraries
   - python-dotenv
   - openai
   - Redlines
3. Setup your OpenAI key

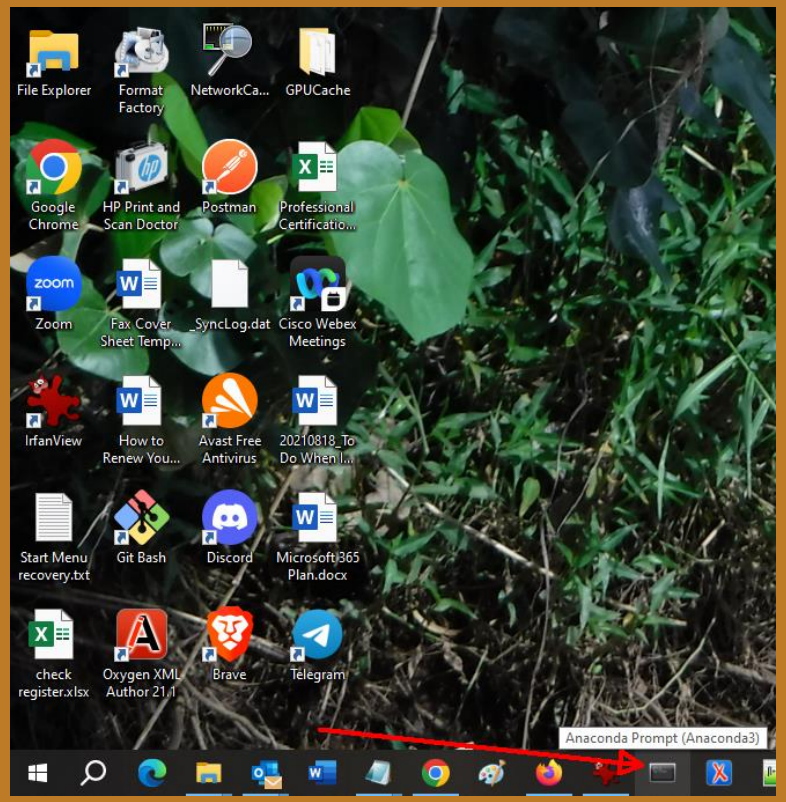
Install Anaconda / Jupyter Notebook

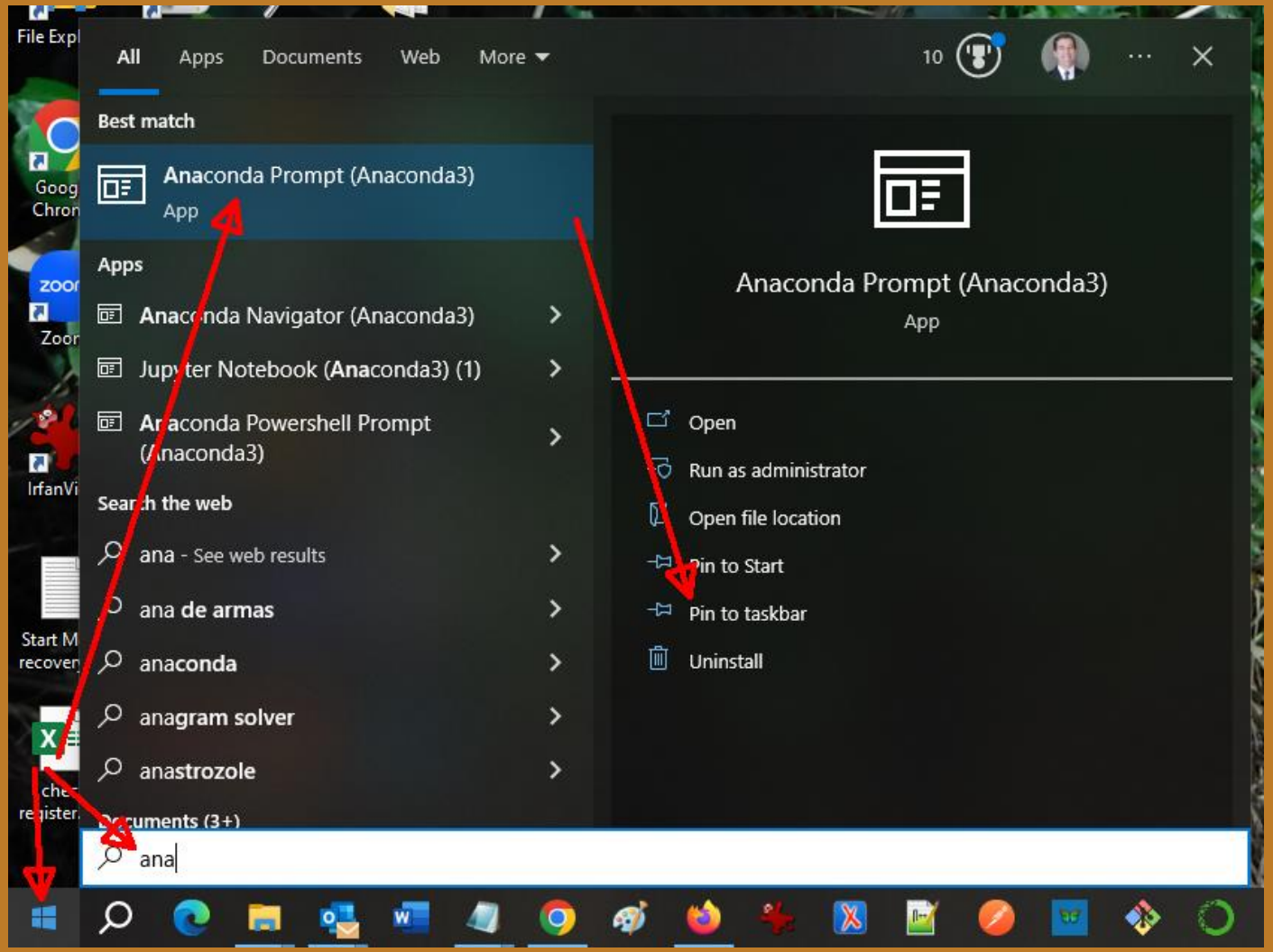
- Follow the instructions from here: https://docs.anaconda.com/free/anaconda/install/windows/
- Accept the defaults
  - Install for **Just Me** (not for all users)
  - Do not add Anaconda to your PATH environment variable
- When done, create a shortcut



Pin Anaconda to the Task Bar

1. Windows Start
2. “ana”
3. Pin to taskbar

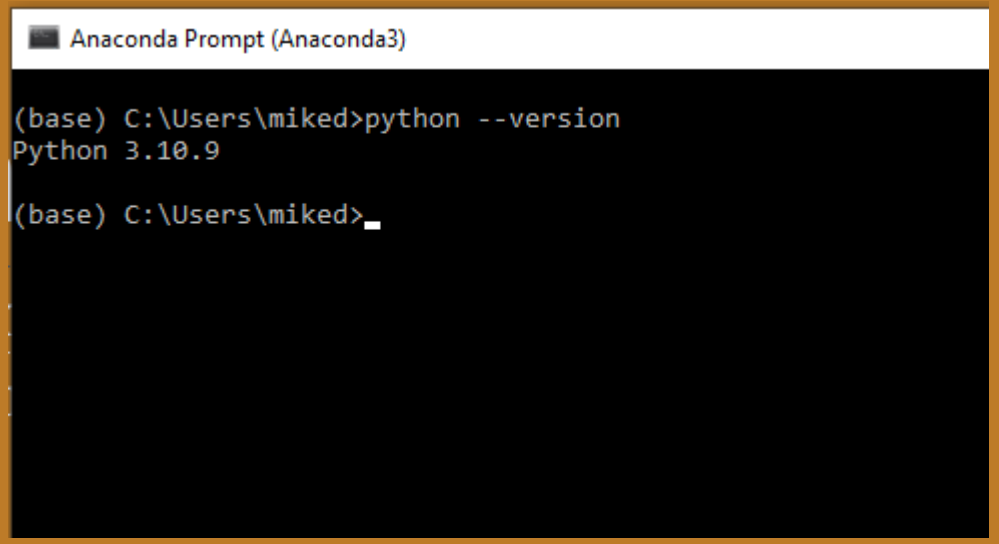




Verify Python Version

Open an Anaconda prompt from the task bar and then run the following command to verify that you're running Python 3.10 (or later)

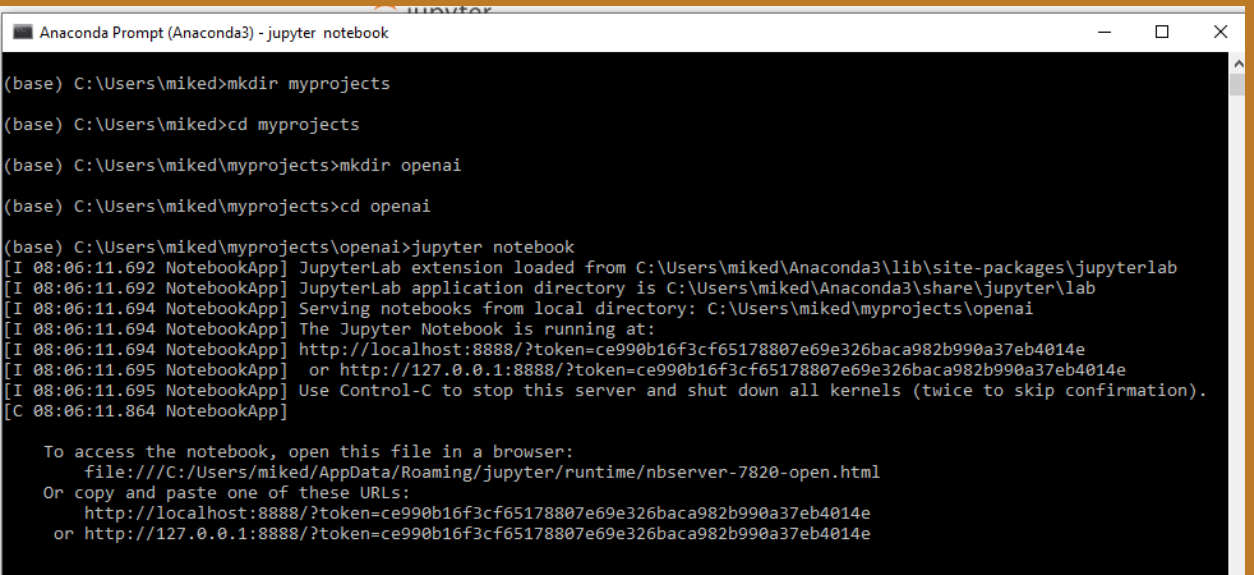
python --version

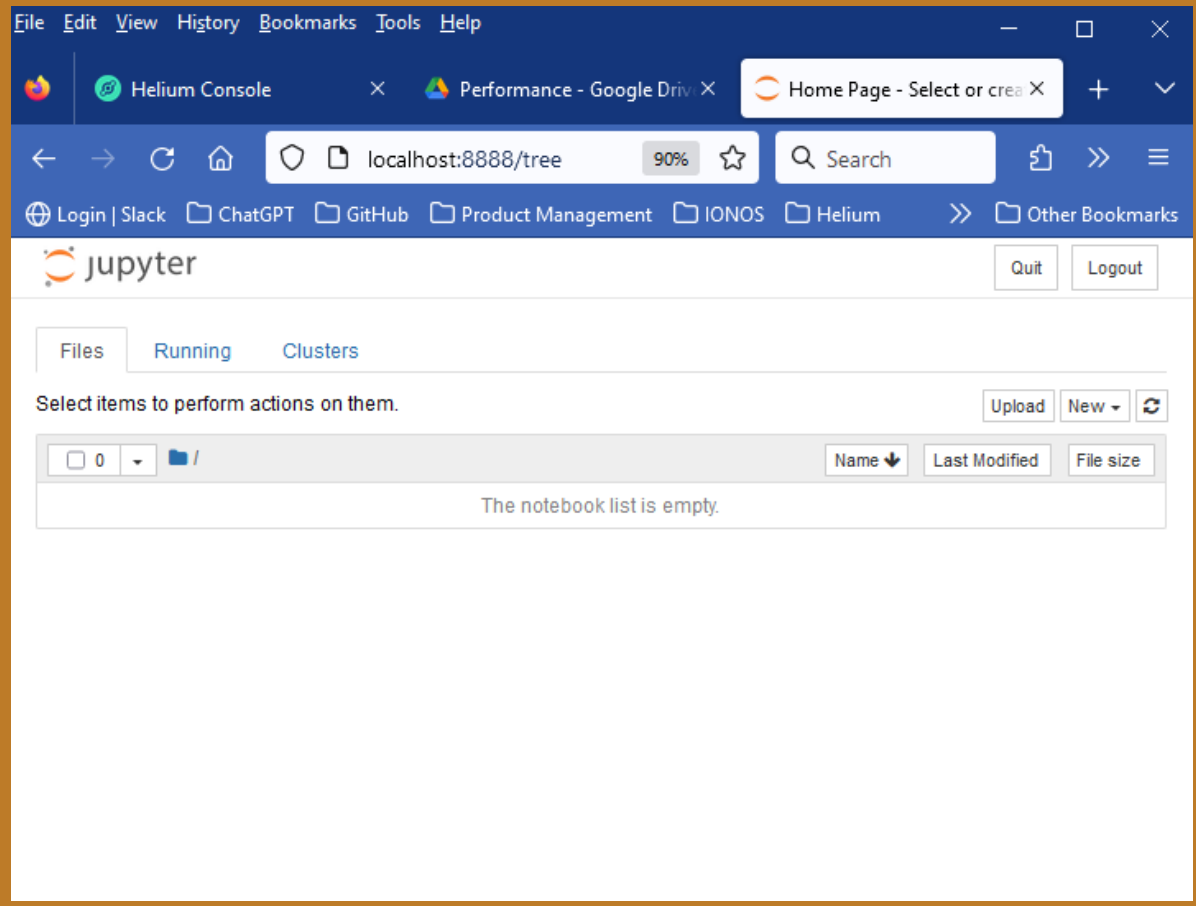


Create a Project folder, then open Jupyter Notebook

Open an Anaconda prompt from the task bar and then run the following commands:

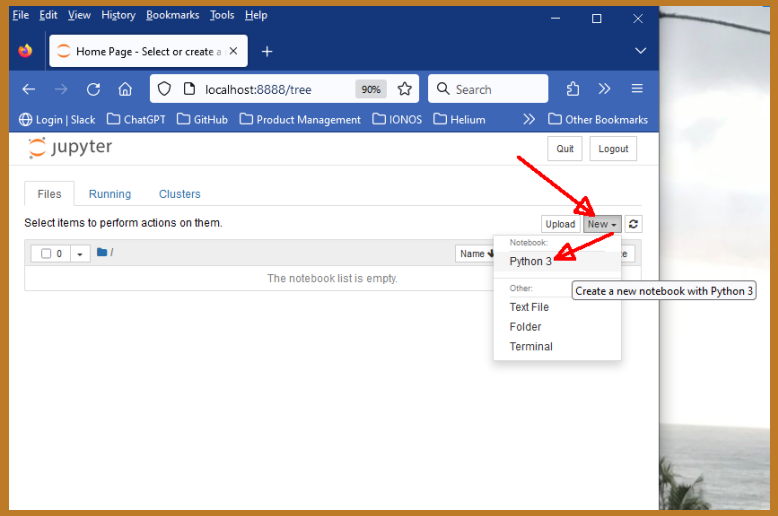
1. *dir*
2. *mkdir myprojects*
3. *cd myprojects*
4. *mkdir openai*
5. *cd openai*
6. *jupyter notebook*

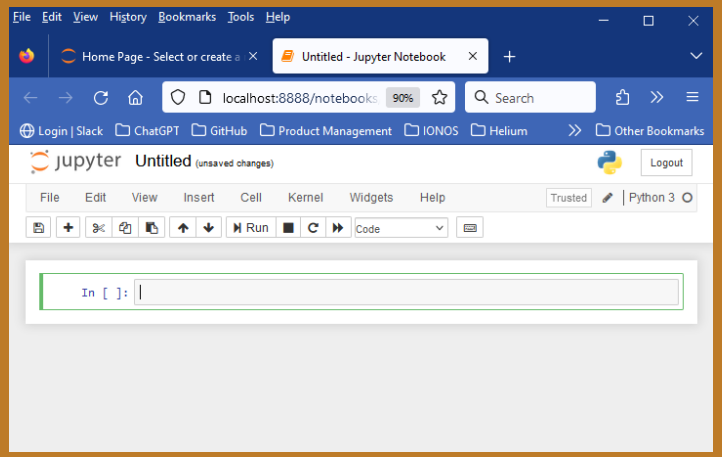


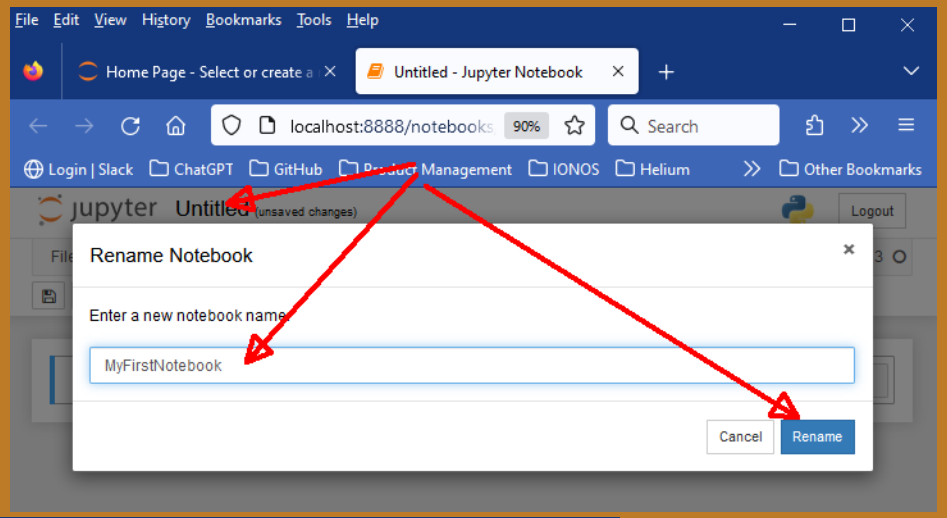


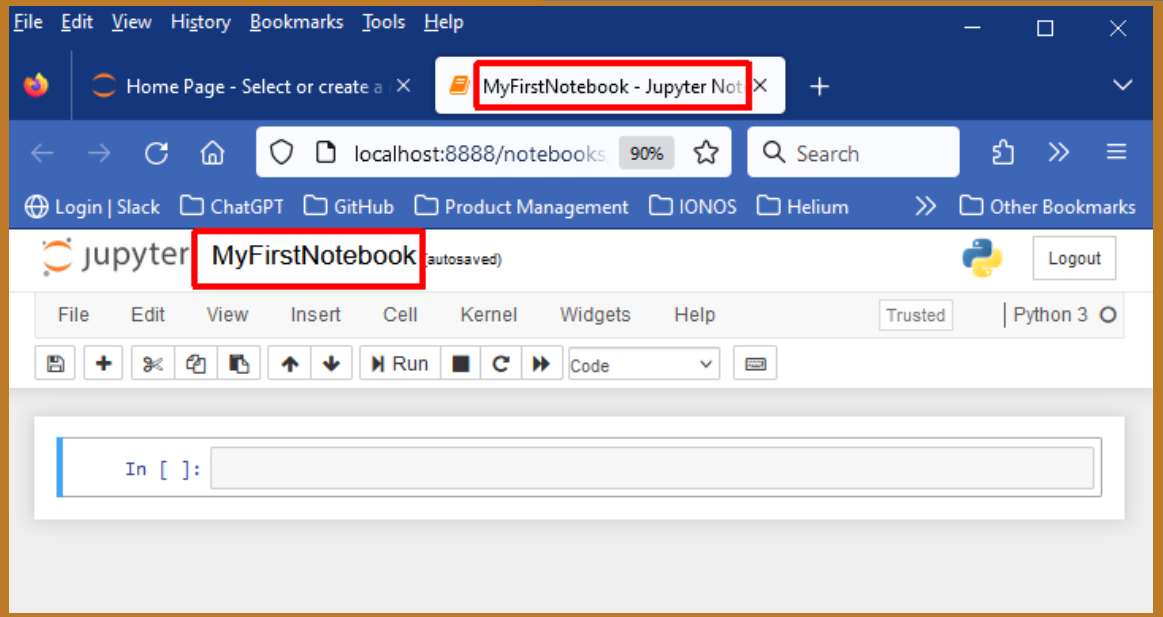
Create a new Jupyter Notebook

New > Python3 to create the notebook, then rename the notebook









Congrats

You’re now running Anaconda and Jupyter Notebook on your PC and have created your first jupyter notebook!

Everything you need to know to take the free course, ChatGPT Prompt Engineering for developers

Progress

1. ~~Install Anaconda / Jupyter Notebook~~
2. **Install 3 Python libraries**
   - python-dotenv
   - openai
   - Redlines
3. Setup your OpenAI key

topic 3 of 3

Installing 3 Python libraries

Steps

1. Install the Python library, python-dotenv
2. Install the Python library, openai
3. Install the Python library, Redlines

Installing python-dotenv

Trying to run the first bit of code from the course, I got a dotenv error.

```
In [1]: import openai
        import os

        from dotenv import load_dotenv, find_dotenv
        _ = load_dotenv(find_dotenv())

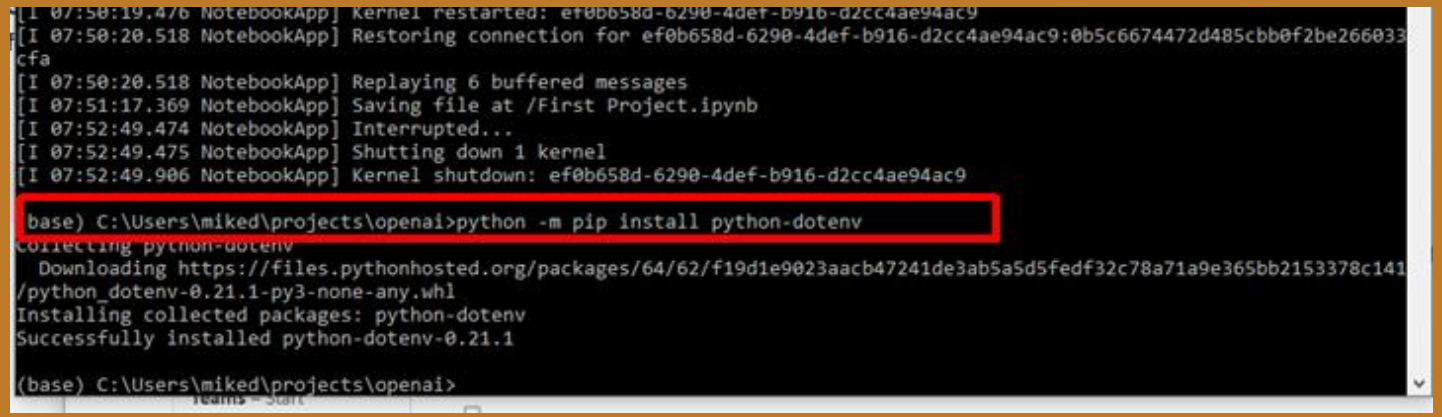
        openai.api_key  = os.getenv('OPENAI_API_KEY')
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
<ipython-input-1-0863633b1d8c> in <module>
      2 import os
      3
----> 4 from dotenv import load_dotenv, find_dotenv
      5 _ = load_dotenv(find_dotenv())
      6

ModuleNotFoundError: No module named 'dotenv'
```

Solution: from the Anaconda prompt install the library, python-dotenv with the following command:

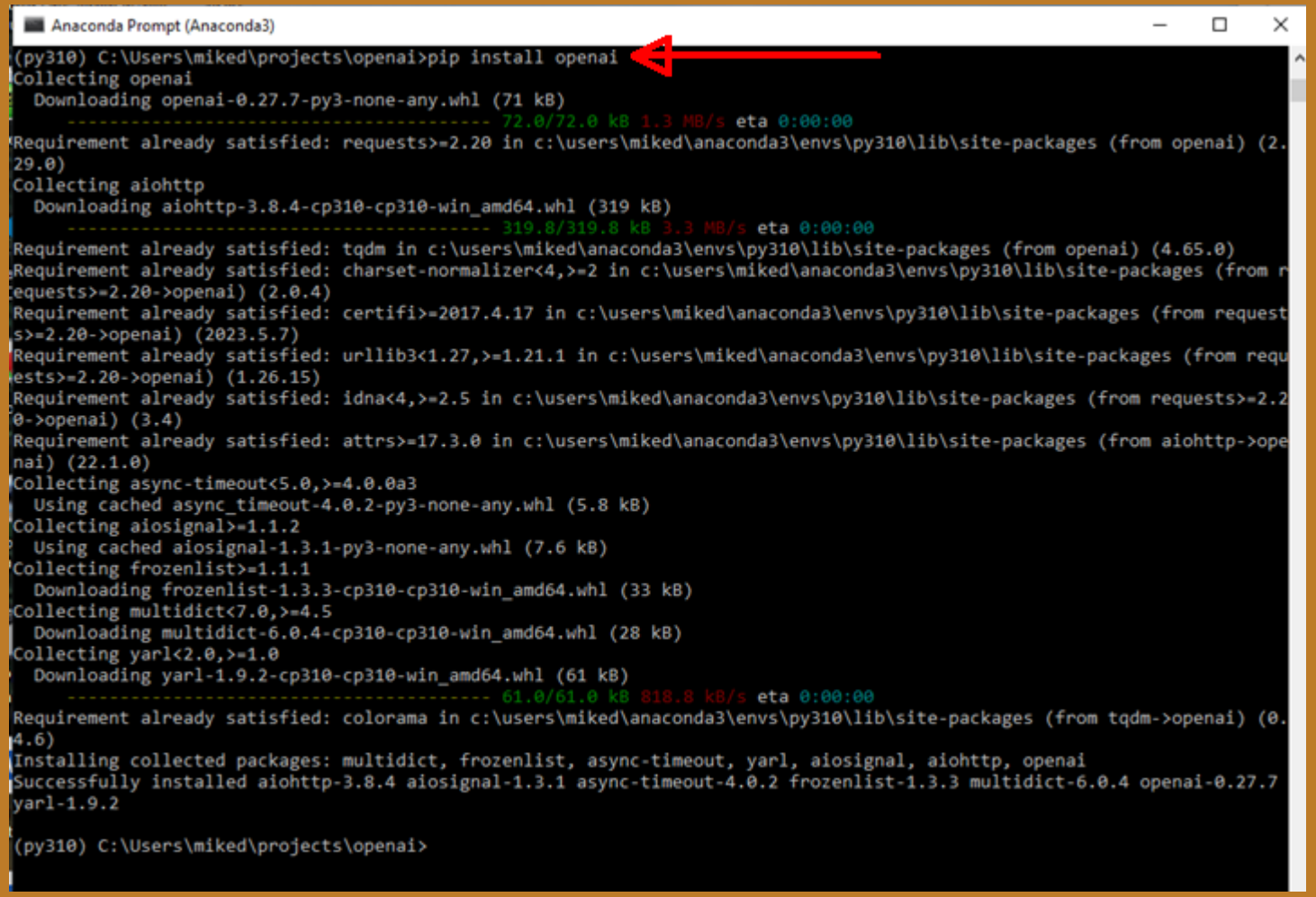
python -m pip install python-dotenv



Installing openai

From the Anaconda prompt install the library, openai with the following command:

pip install openai



Installing Redlines

The course chapter on Transforming requires the Redlines python package. Here's the error you get when you try to use Redlines without having it installed.

```
In [27]: from redlines import Redlines

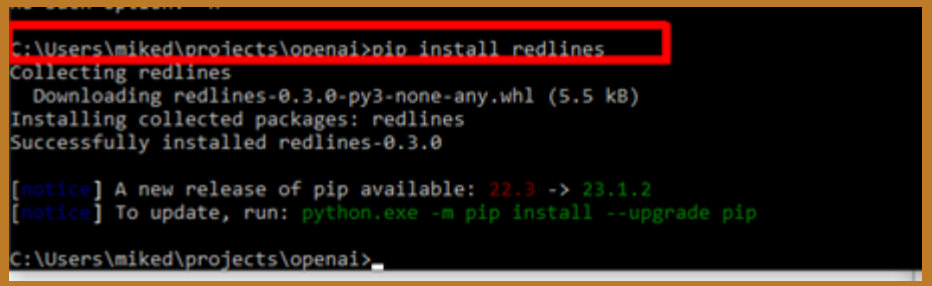
         diff = Redlines(text,response)
         display(Markdown(diff.output_markdown))

---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
<ipython-input-27-4c4c6ae221b6> in <module>
----> 1 from redlines import Redlines
      2
      3 diff = Redlines(text,response)
      4 display(Markdown(diff.output_markdown))

ModuleNotFoundError: No module named 'redlines'
```

Solution: from the Anaconda prompt install the library, Redlines, with the following command:

pip install redlines



Progress

1. ~~Install Anaconda / Jupyter Notebook~~
2. ~~Install 3 Python libraries~~
    - python-dotenv
    - openai
    - Redlines
3. Setup your OpenAI key

topic 3 of 3

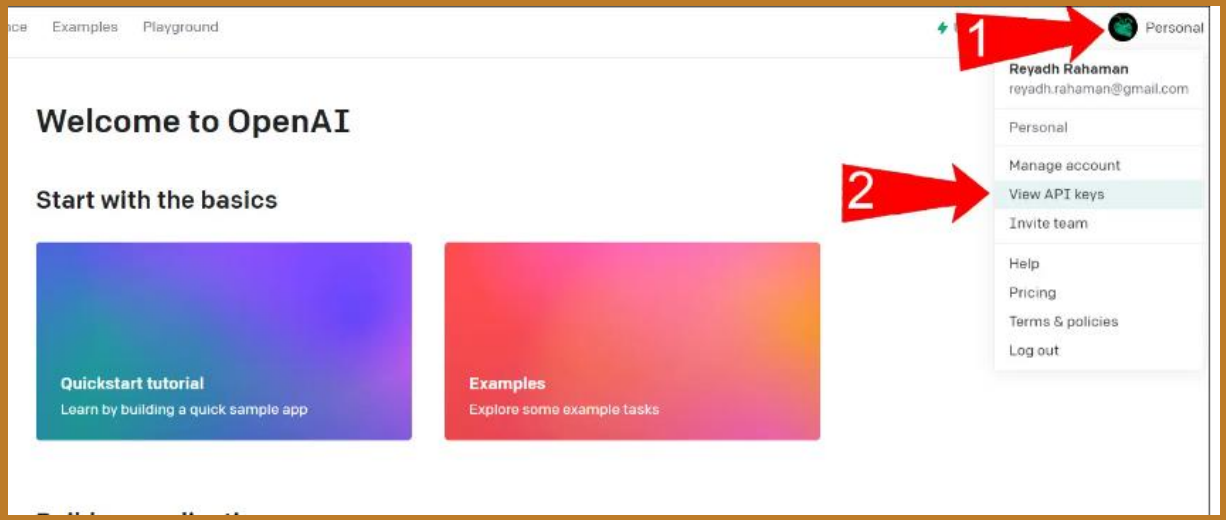
Getting and Installing your OpenAI Key

Steps

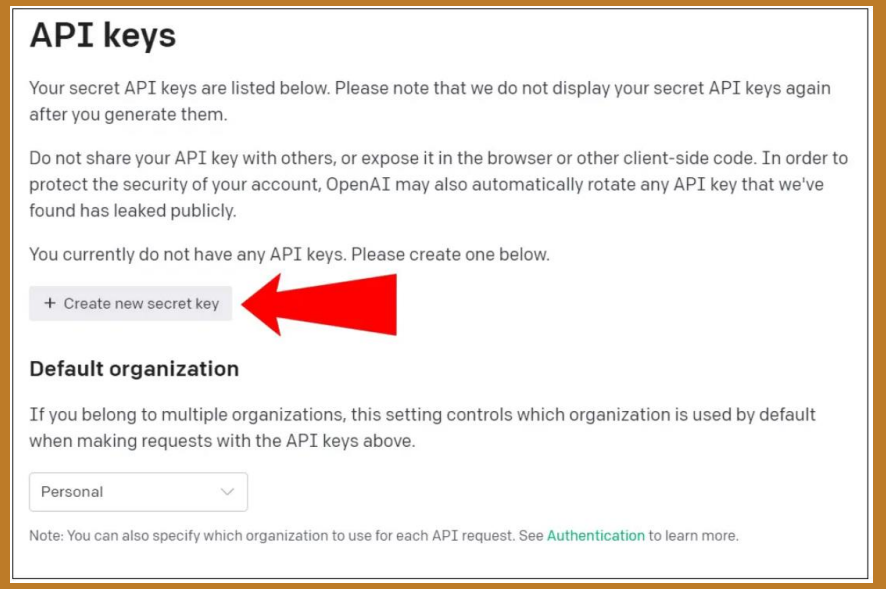
1. Create a Key in your OpenAI account
2. Put the Key into an environment variable on your PC

Getting Your OpenAI Key

Based on instructions from here:
https://www.howtogeek.com/885918/how-to-get-an-openai-api-key/



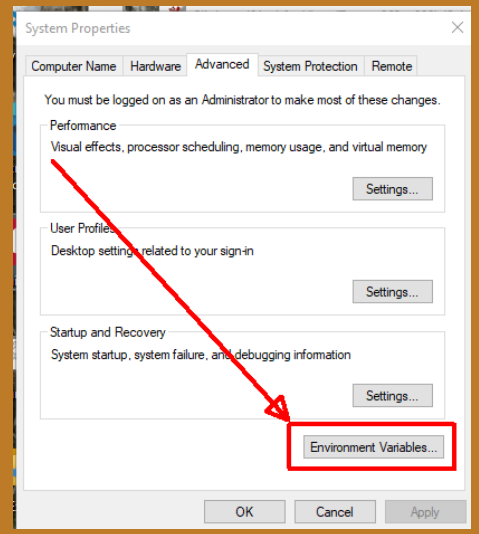
1. Go to OpenAI's Platform website at https://platform.openai.com and sign in with your OpenAI account.
2. Click your profile icon at the top-right corner of the page and select "View API Keys"
3. Click "Create New Secret Key" to generate a new API key
4. Name the key “My First openai Key”
5. Copy the key to Notepad or any other handy text editor

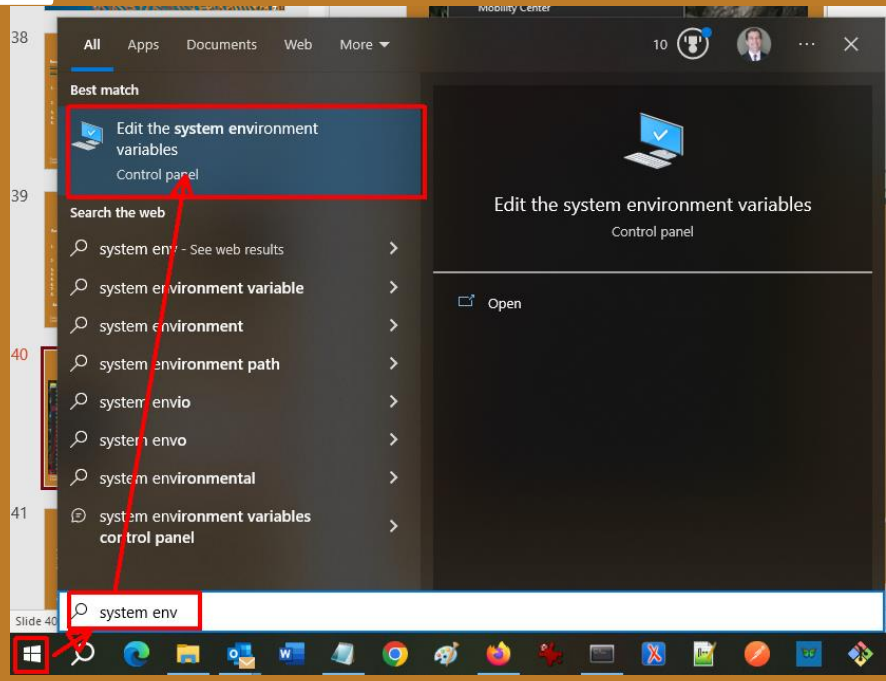


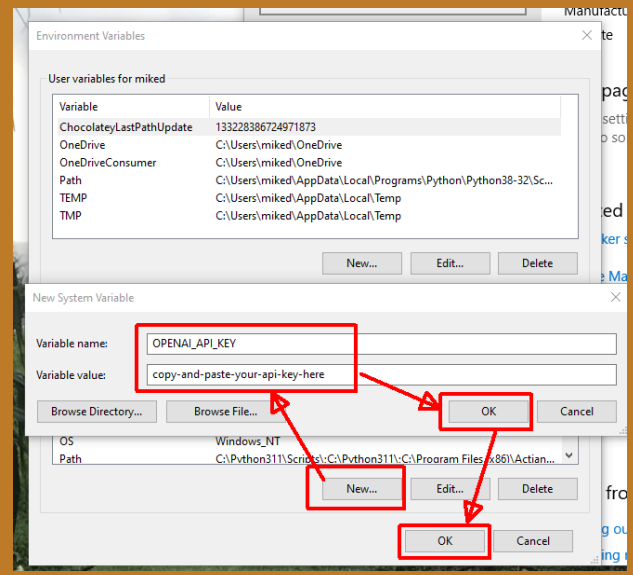
Creating an OPENAI_API_KEY environment variable

Based on instructions from here:

1. On the Windows taskbar, right-click the Windows icon type ‘System env’ then select ‘Edit the system environment variables’
    1. If you don’t see the option to edit environment variables, search for
2. On the Advanced tab, click **Environment Variables**. ...
3. Under System Variables, click **New** to create a new environment variable.
4. Set **Variable Name** to OPENAI_API_KEY
5. Set Variable Value to the key you got from your OpenAI account
6. Select **OK**
7. Select **OK**

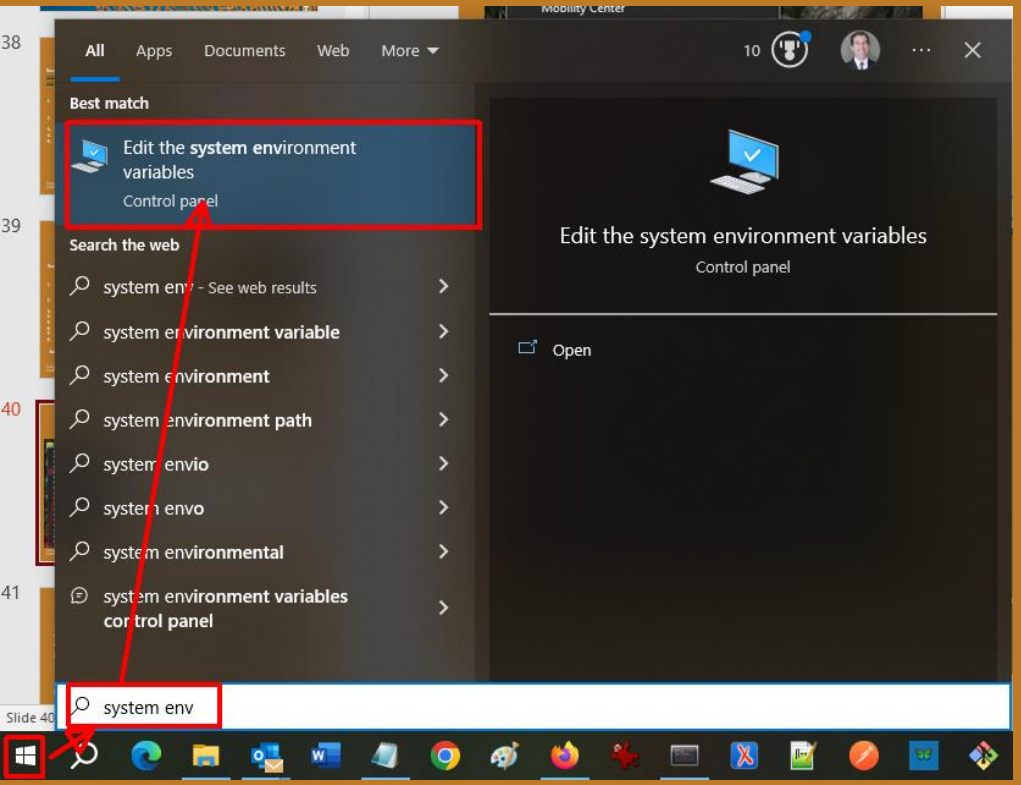


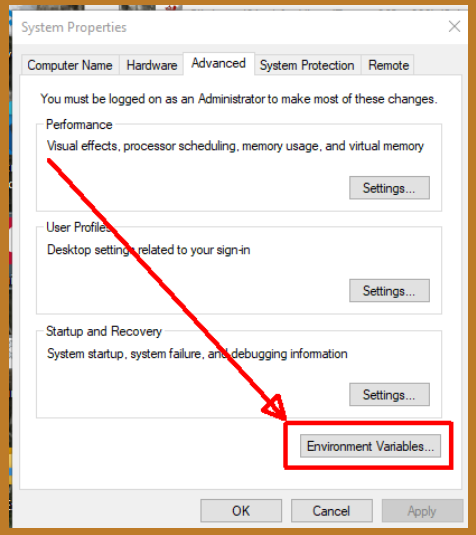


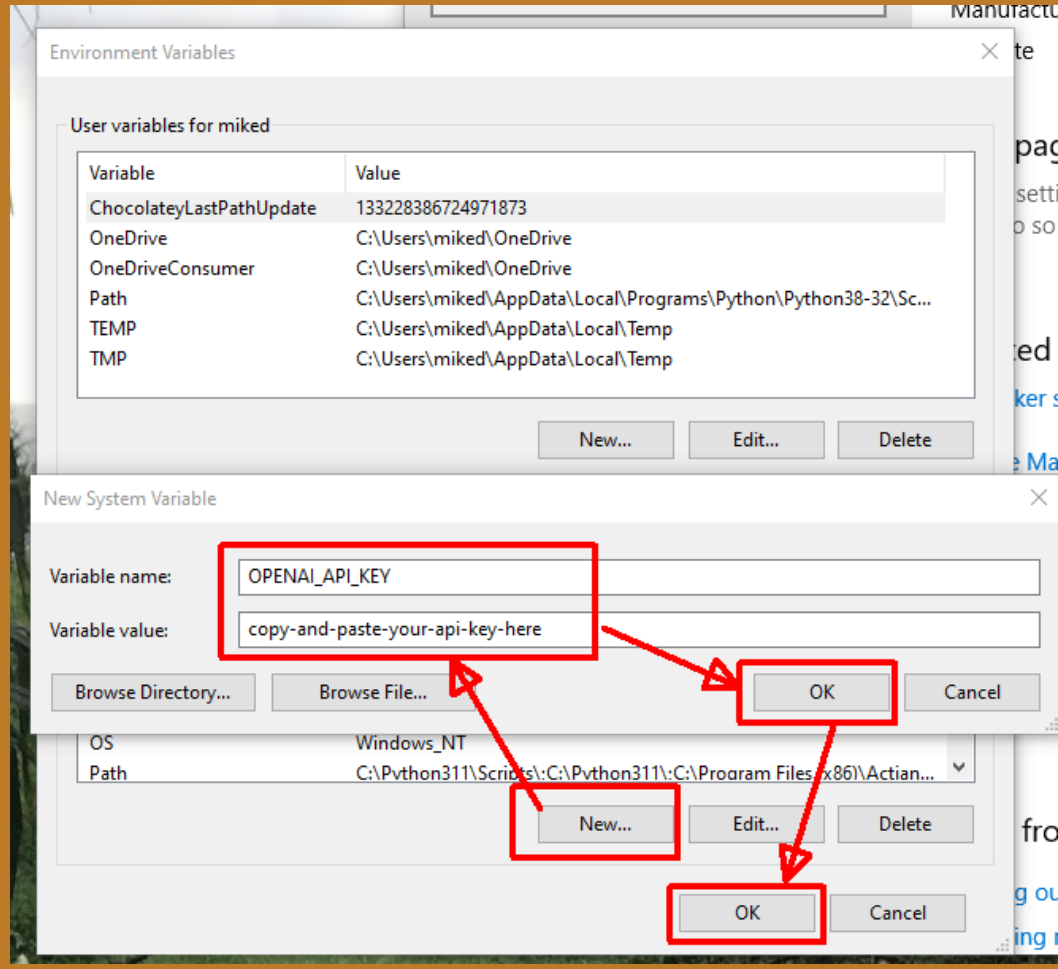


See bigger screenshots on the next slide

Bigger Screenshots







Progress

1. ~~Install Anaconda / Jupyter Notebook~~
2. ~~Install Python 3.10 in Anaconda~~
3. ~~Install 3 Python libraries~~
    - python-dotenv
    - openai
    - Redlines
4. ~~Setup your OpenAI key~~

lesson 3 summary

- You've now set up the following tools on your computer
  - Installed Anaconda /Jupyter Notebook
  - Installed some extra Python libraries the course uses
  - Installed an OpenAI key so you programs can make calls to ChatGPT
- Your computer should be set up to run all the course examples

- *In the next lesson we're going to run through the basics of using Jupyter Notebook so you'll be comfortable following along with the course*

Congrats

You've installed the extra Python libraries needed to run all the course examples

AND

You've installed an OpenAI Key on your PC so your programs can call ChatGPT

Everything you need to know to take the free course, ChatGPT Prompt Engineering for developers

lesson 04

Using Jupyter Notebook

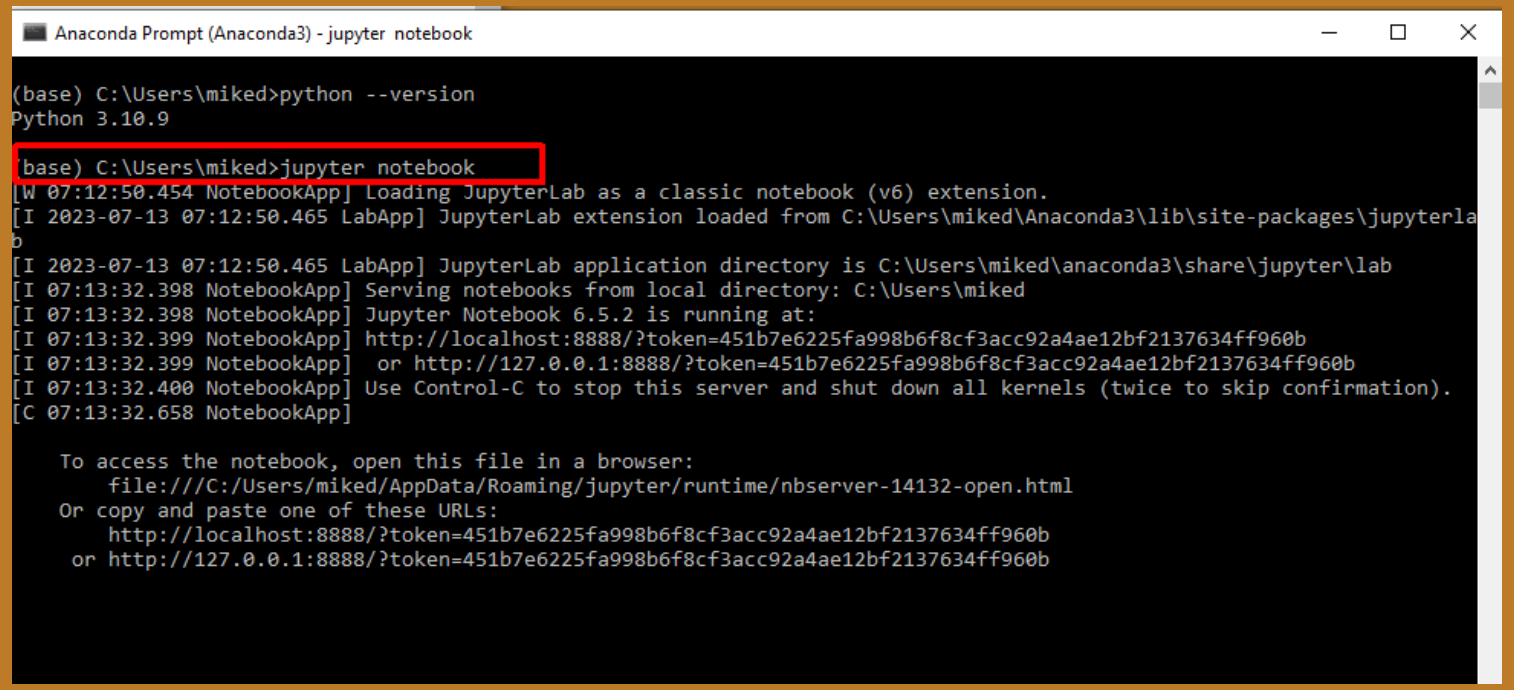
Using Jupyter Notebook

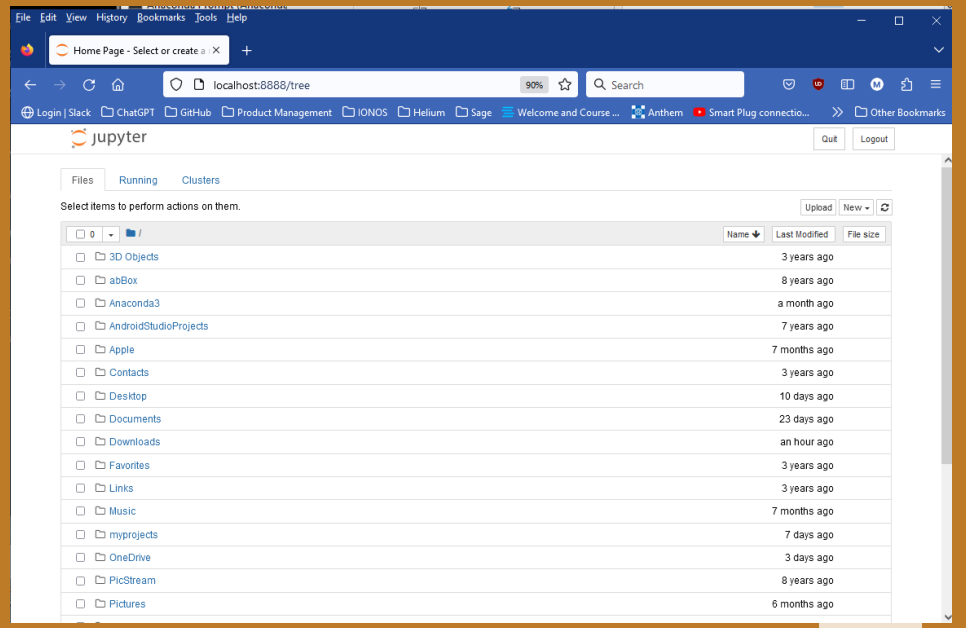
1. Starting Jupyter Notebook from an Anaconda prompt
2. Creating a new notebook
3. Renaming a notebook
4. Saving a notebook
5. Closing a notebook
6. Opening an existing notebook
7. Adding code to a notebook
8. Deleting a notebook cell
9. Running code in a notebook

Starting Jupyter Notebook

Open an Anaconda prompt, then run the following command:

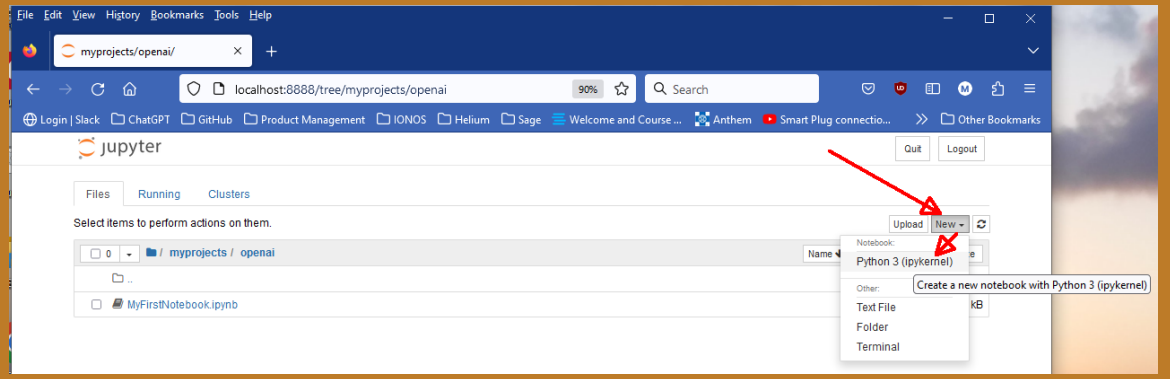
jupyter notebook

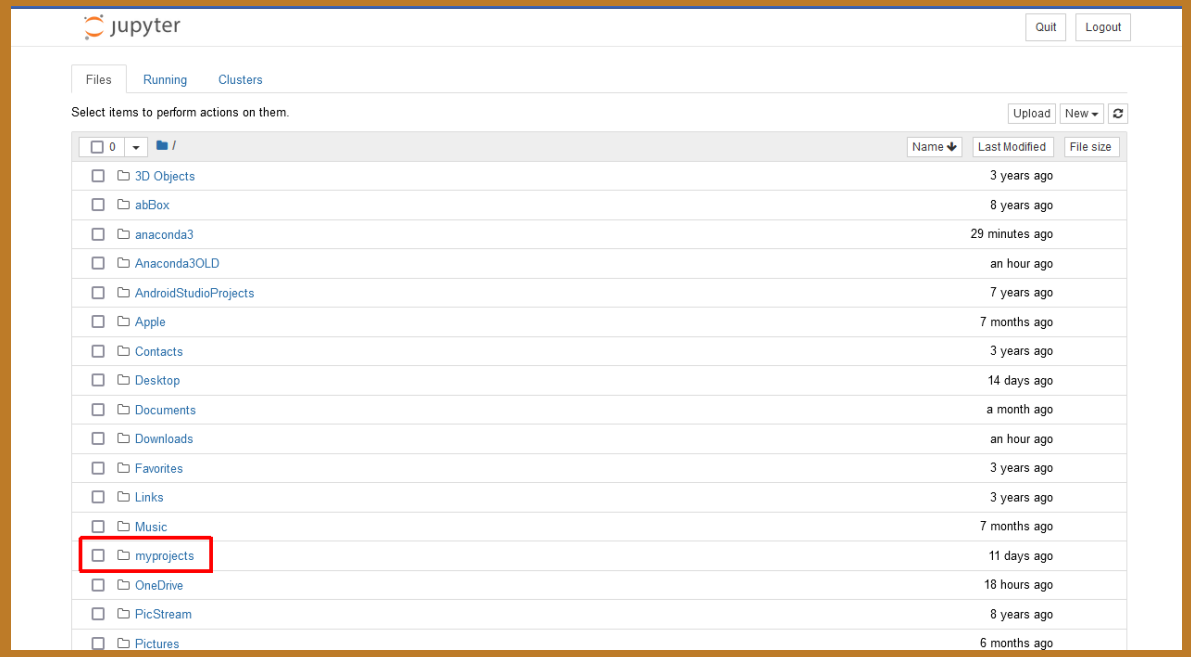


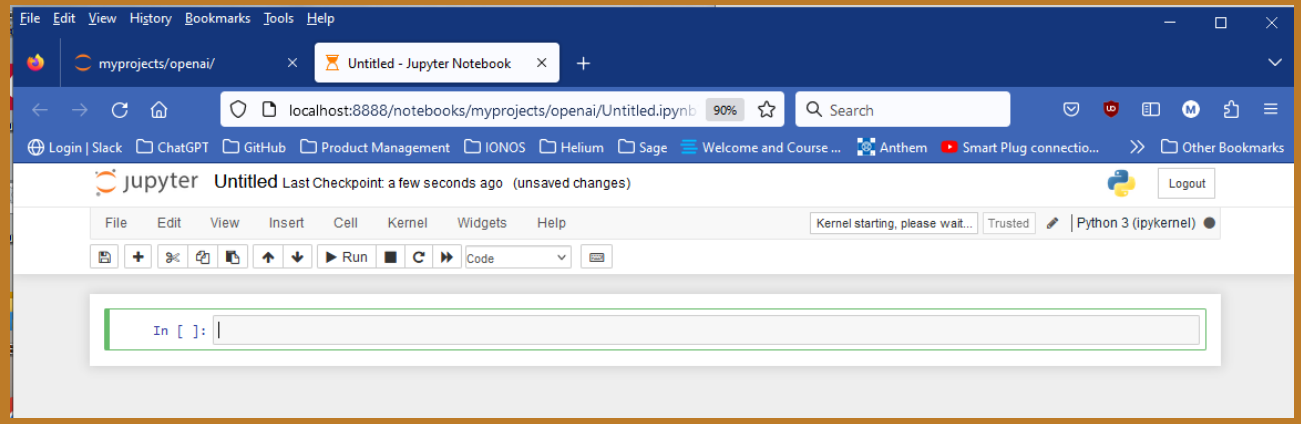


Creating a new Notebook

1. Open Jupyter Notebook
2. Navigate to the folder where you want to create the notebook
   - Myprojects > openai
3. New > Python 3

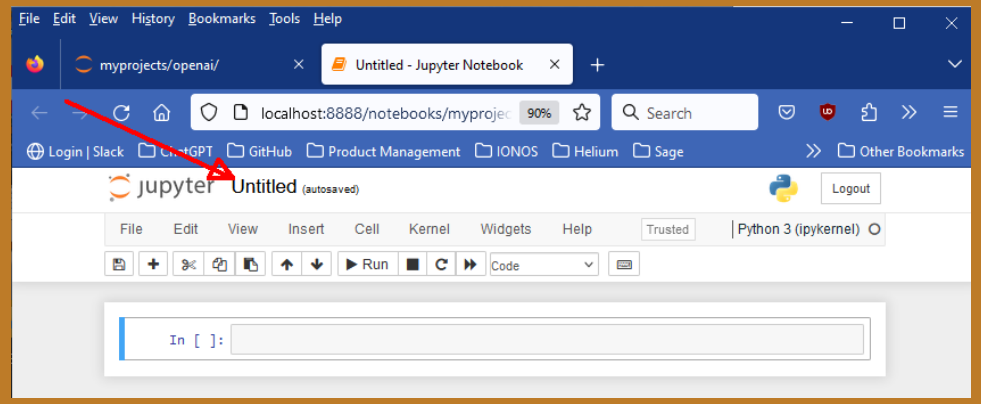


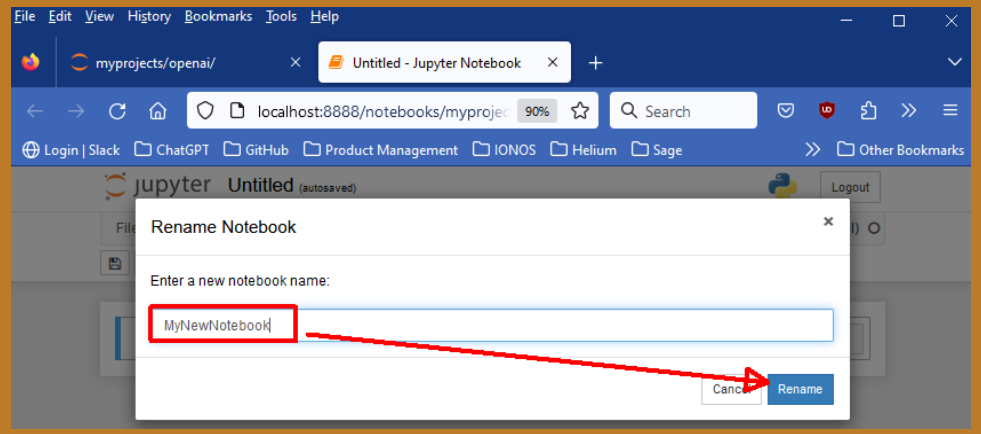


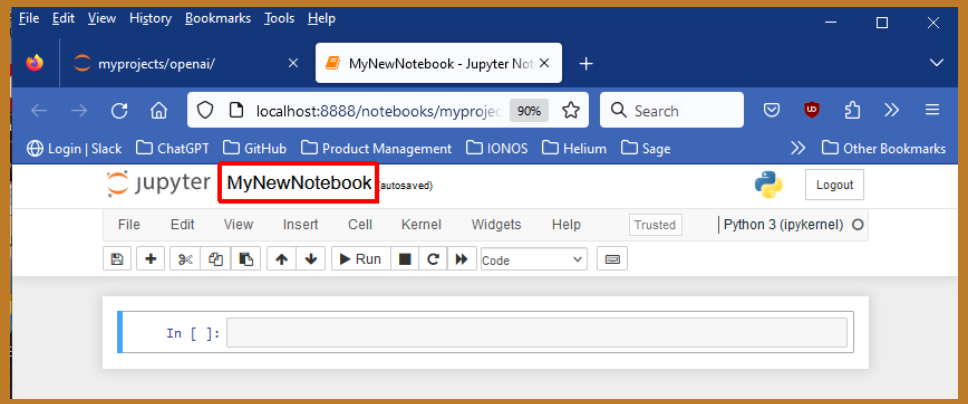


Renaming a Notebook

1. Click on the title to bring up the Rename Notebook dialog box
2. Enter a new notebook name, *1 Guidelines for Prompting*, then select **Rename**





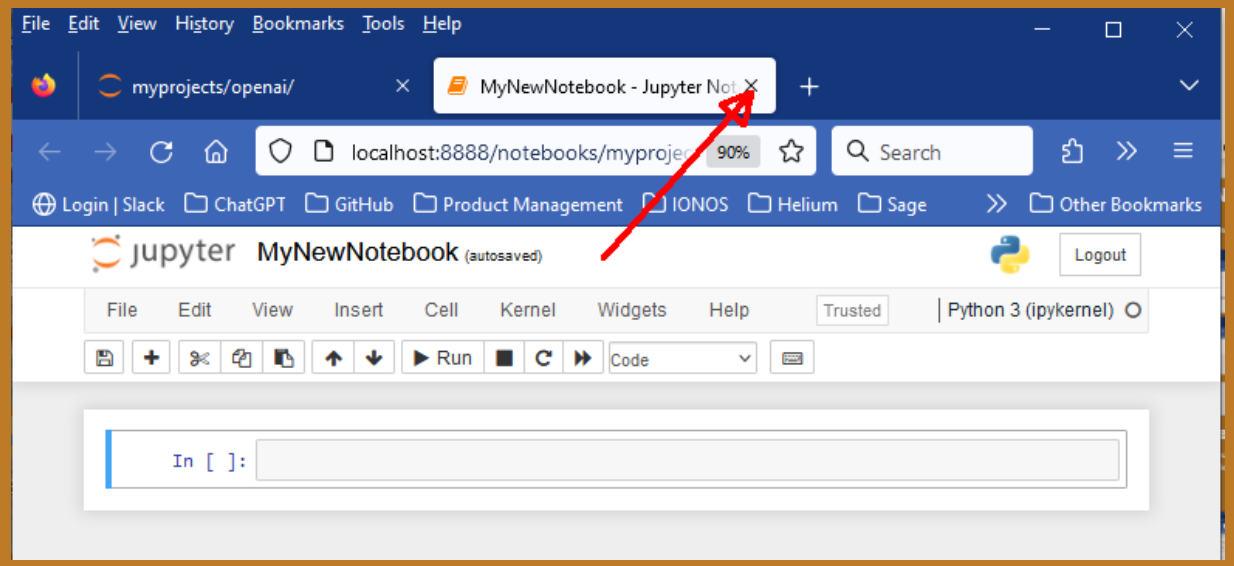


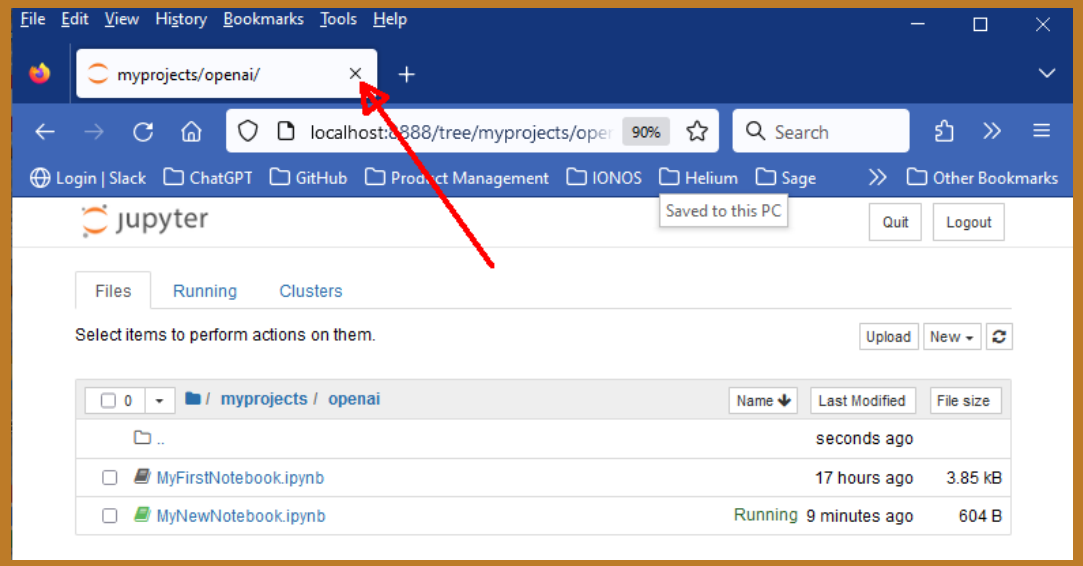
Saving a Notebook

Use the  icon to save your changes (Jupyter Notebook periodically auto-saves too)

Closing a Notebook

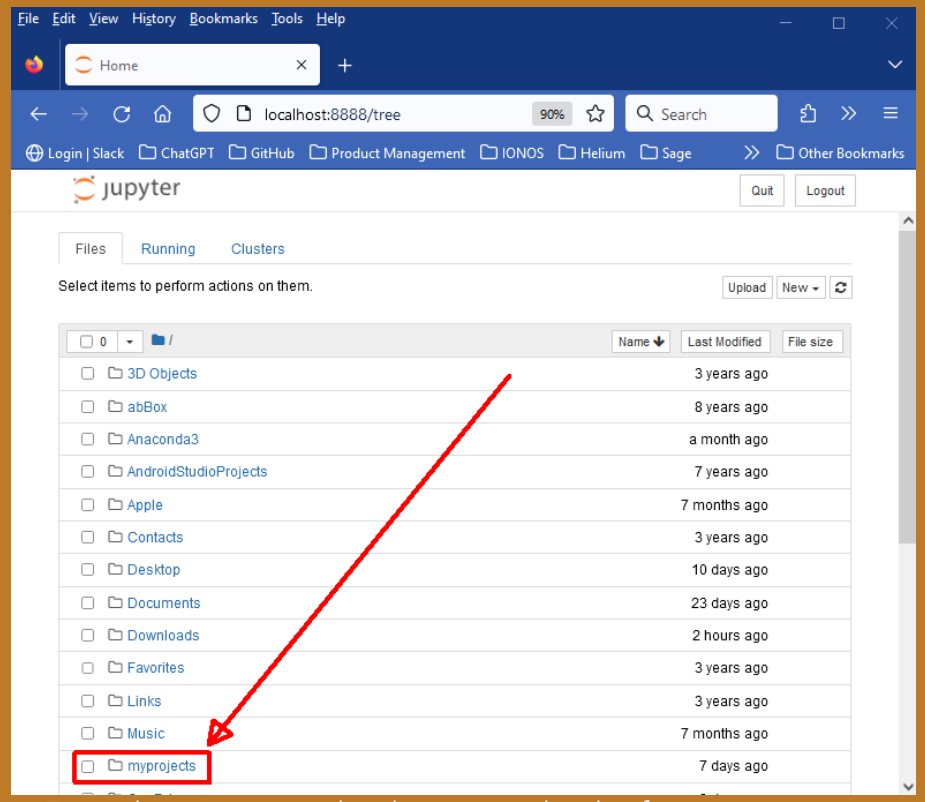
1. Close the browser window running the Notebook
2. (optional) Close the browser window showing the notebook listing
3. (optional) Close the Anaconda prompt window

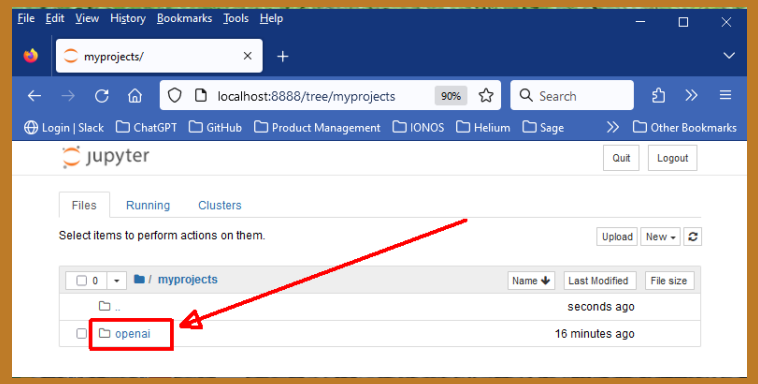


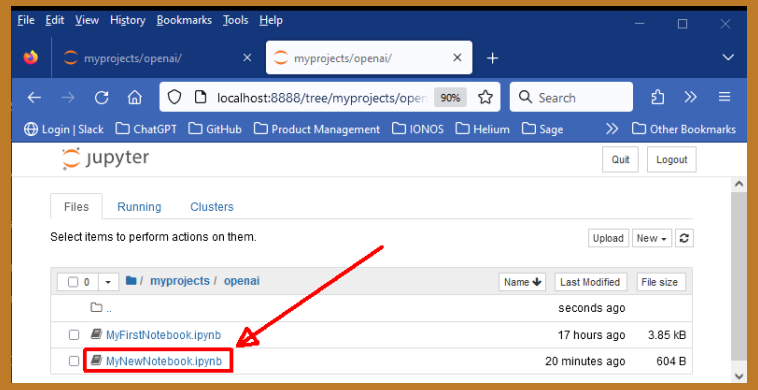


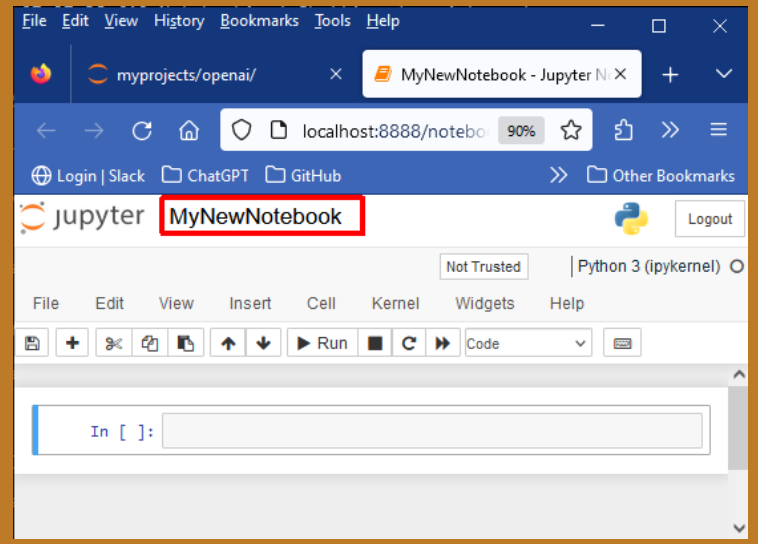
Opening an existing Notebook

1. Open Jupyter Notebook (start Anaconda, then run *jupyter notebook*)
2. Navigate to the folder containing the notebook
3. Double-click the notebook



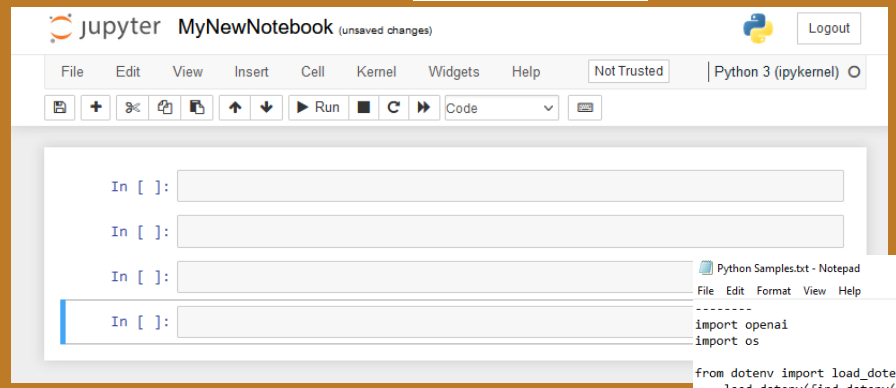


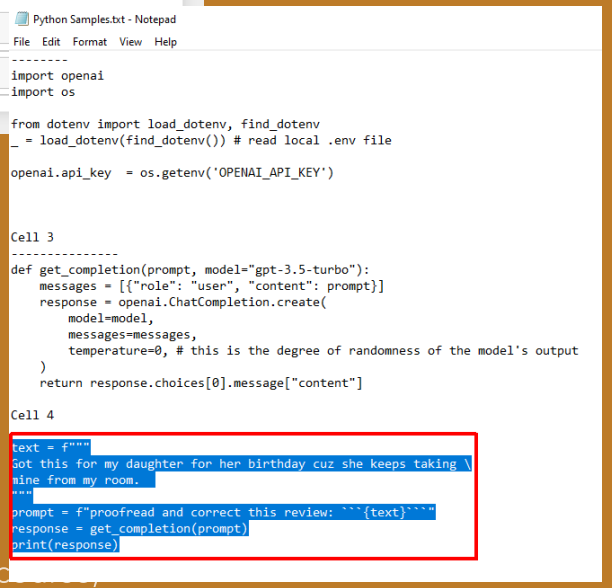


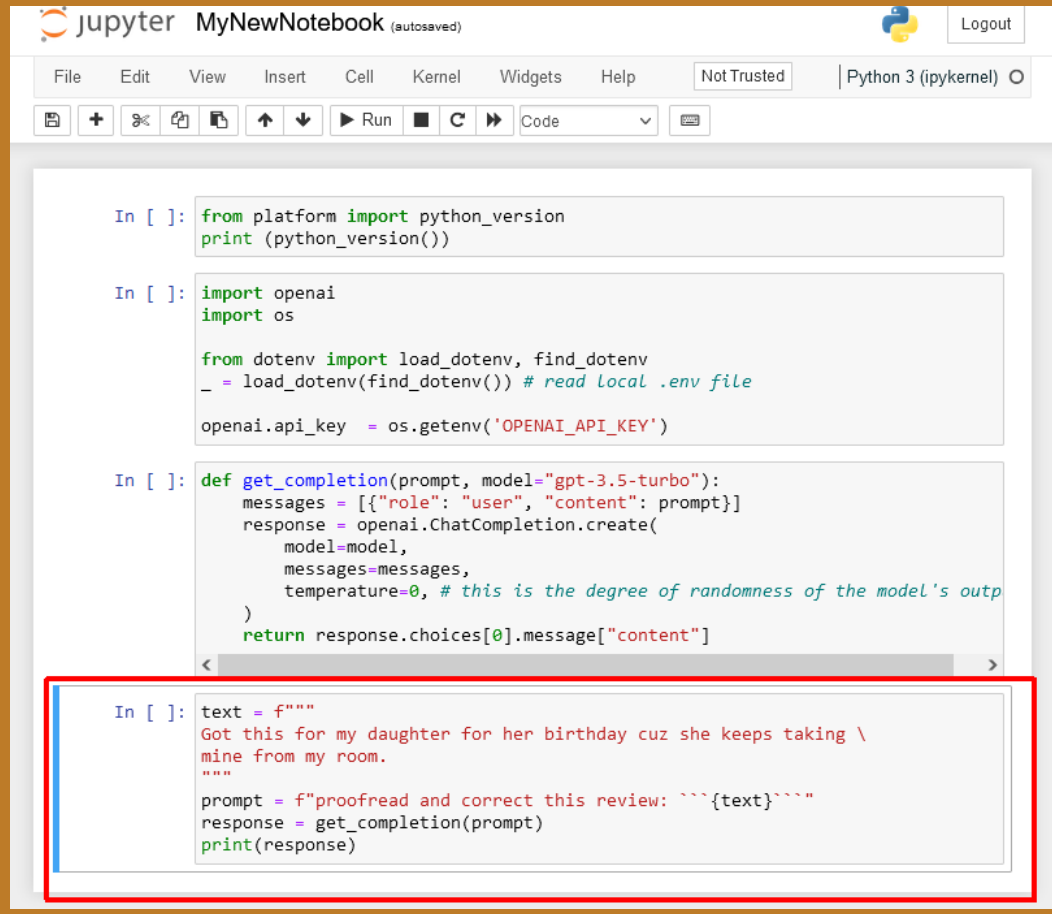


Adding Code to a Notebook

1. Use the + icon to add additional cells
2. Write or paste some code into the cells
3. Use the ▶ Run icon to run/step thru 1 cell at a time (start from the 1st cell)



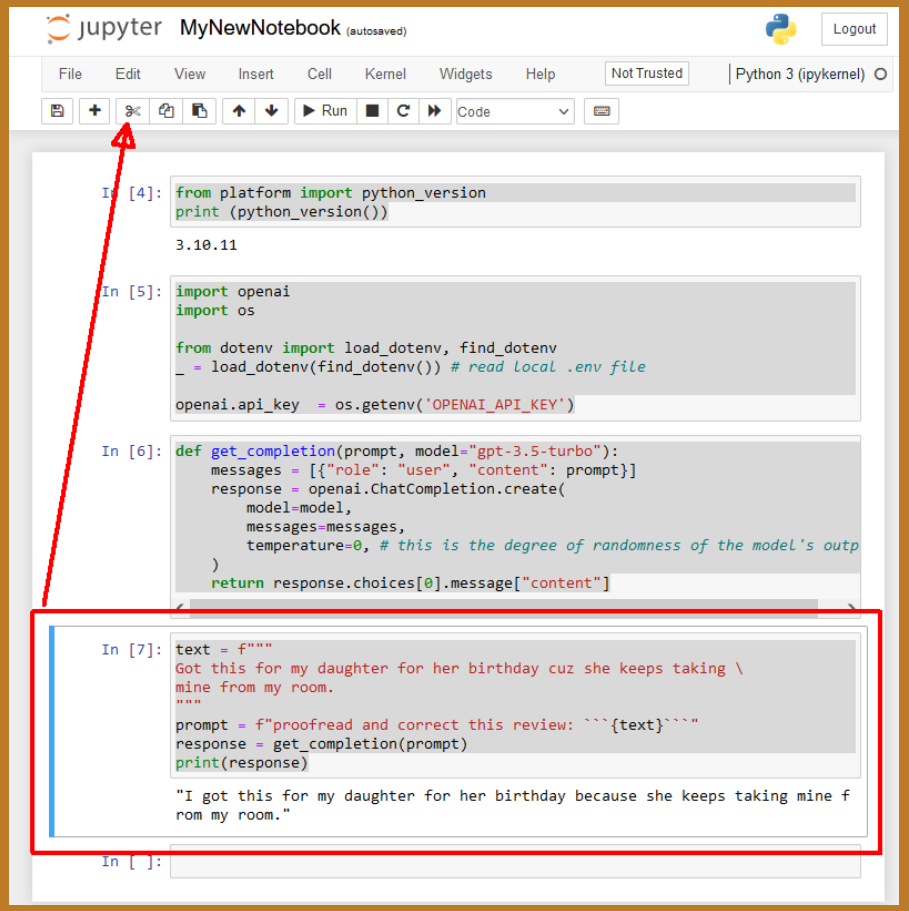


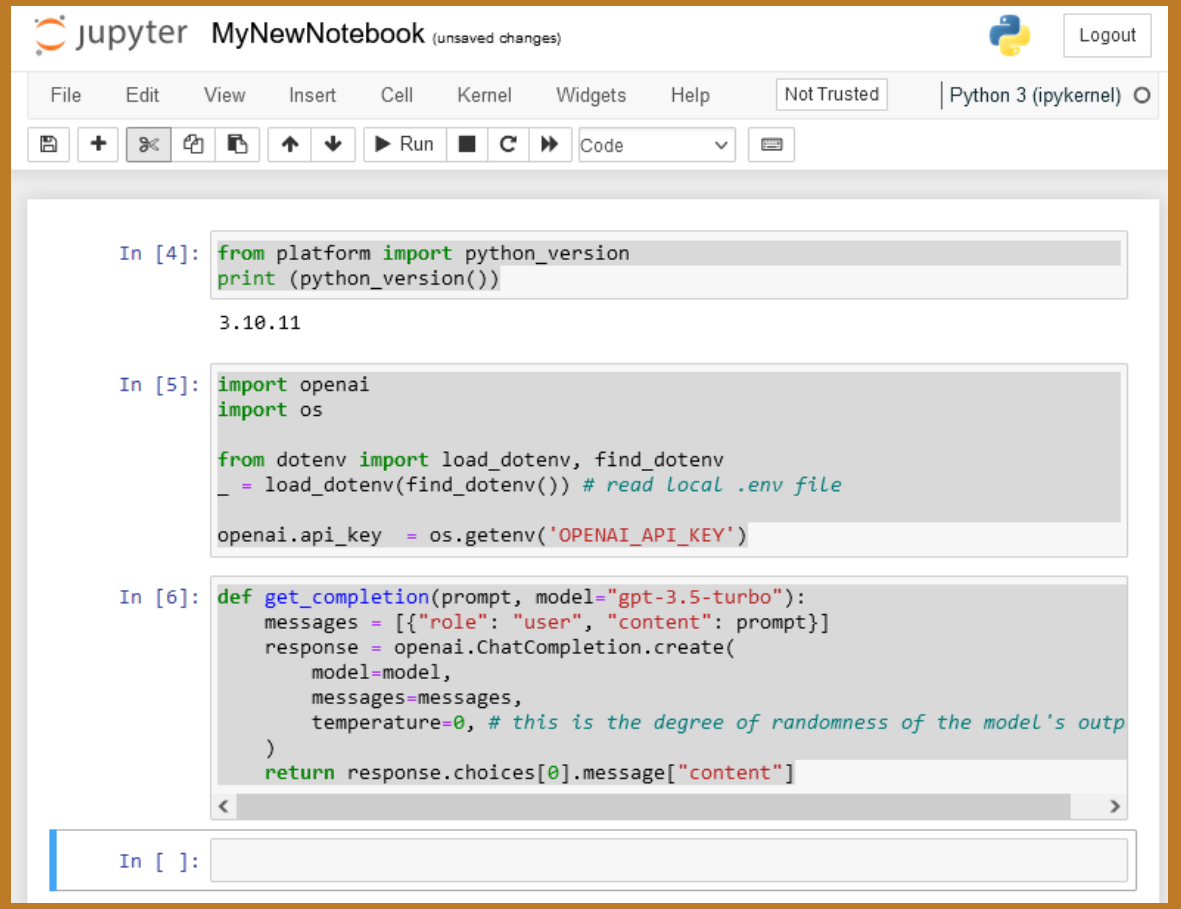


Everything you need to know to take the free c
ChatGPT Prompt Engineering for developers

Deleting a Notebook Cell

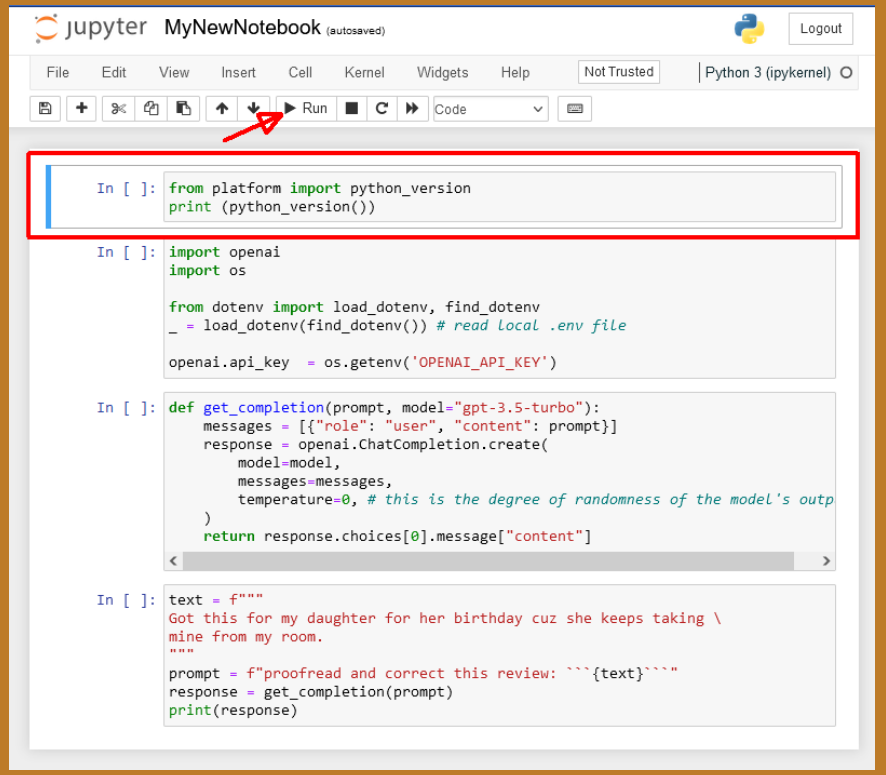
Select the cell to be deleted, then use the Use the ✂ icon

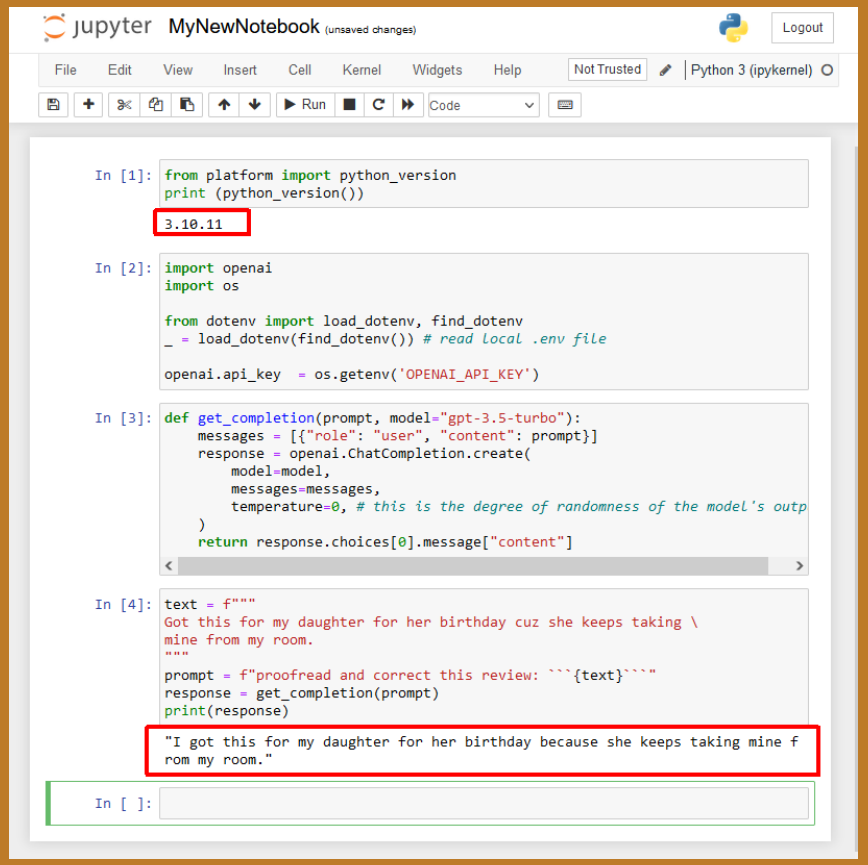




Running Code in a Notebook

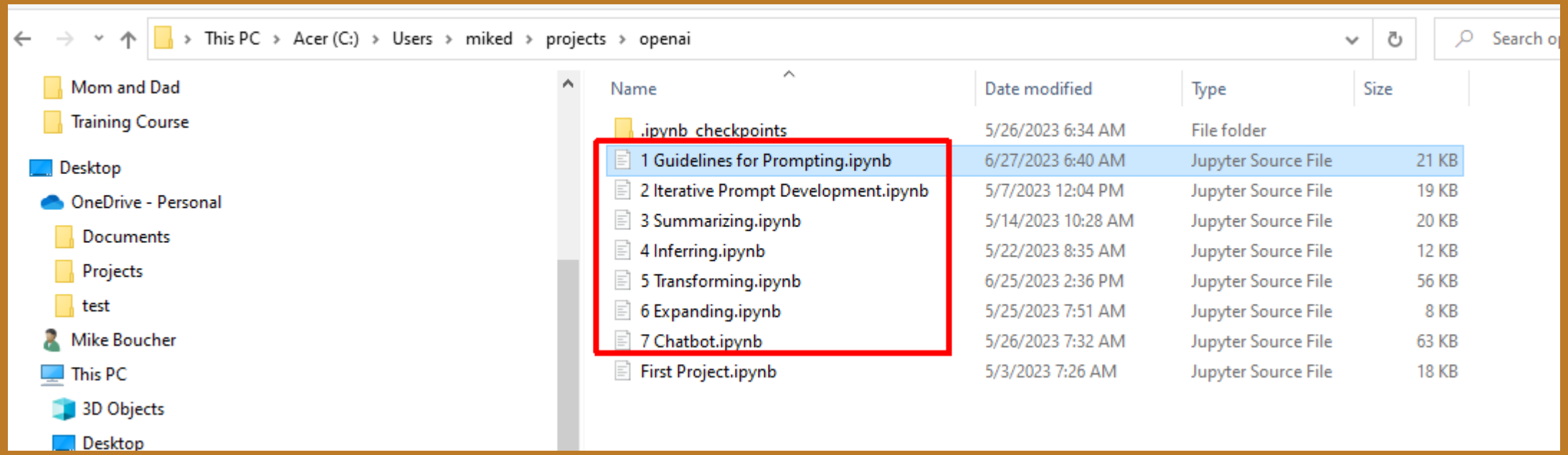
1. Select the 1st cell
2. Use the ▶ Run icon to run/step thru 1 cell at a time
3. Cells that have output will show it under the cell





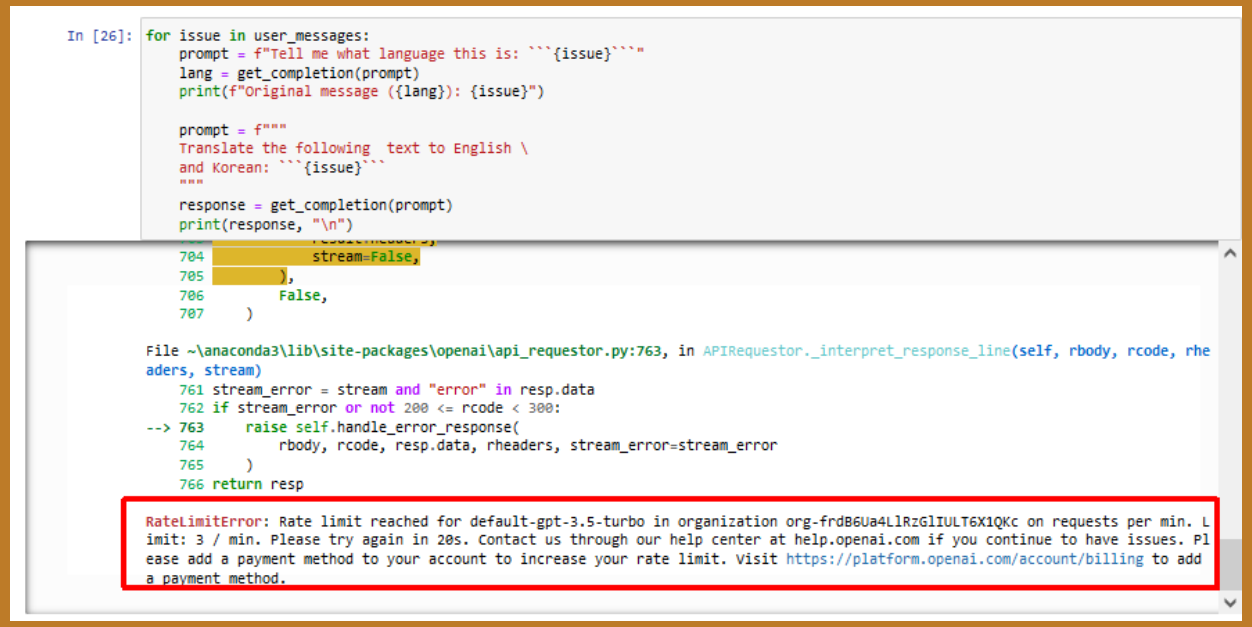
Suggestion – Create Notebooks for the Class

Using the knowledge gained in this section, you may want to create a Notebook for each chapter of the class.



Occasional Timeout Error

- When you're running an example from the course that makes multiple calls to the openai API, sometimes you'll get a timeout error.
- This is because the free account allows 3 API calls per minute. If you try more than that, you'll get an error.
- When this occurs, just move on (or sign up for a paying account) or introduce a time delay into the code.
- If you're running the Jupyter Notebook from within the course you will not get this error.



```
In [26]: for issue in user_messages:
             prompt = f"Tell me what language this is: ```{issue}```"
             lang = get_completion(prompt)
             print(f"Original message ({lang}): {issue}")

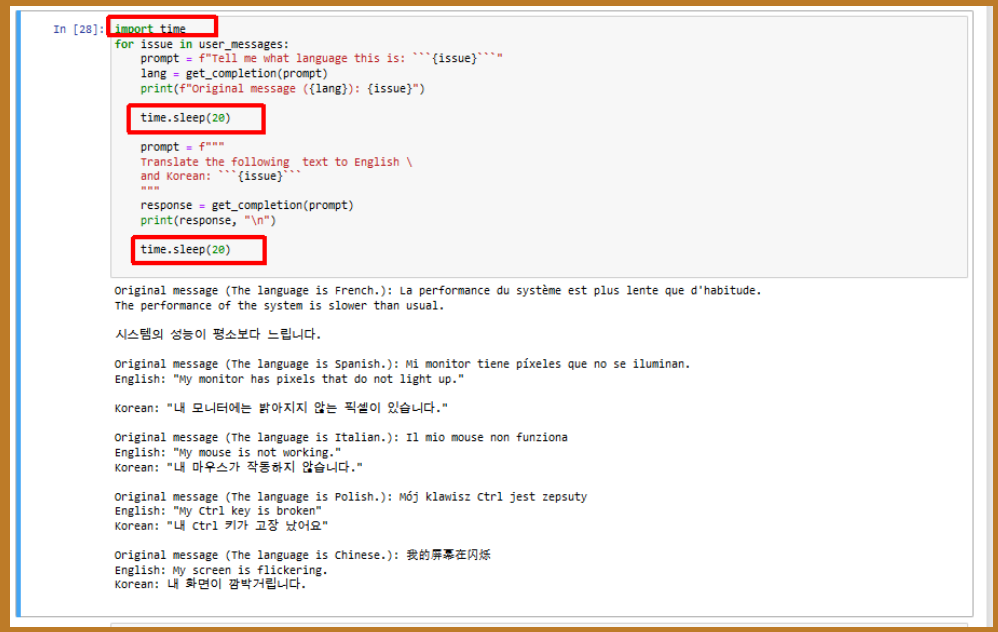
             prompt = f"""
             Translate the following  text to English \
             and Korean: ```{issue}```
             """
             response = get_completion(prompt)
             print(response, "\n")
```

```
    704             stream=False,
    705         ),
    706         False,
    707     )

File ~\anaconda3\lib\site-packages\openai\api_requestor.py:763, in APIRequestor._interpret_response_line(self, rbody, rcode, rheaders, stream)
    761 stream_error = stream and "error" in resp.data
    762 if stream_error or not 200 <= rcode < 300:
--> 763     raise self.handle_error_response(
    764         rbody, rcode, resp.data, rheaders, stream_error=stream_error
    765     )
    766 return resp

RateLimitError: Rate limit reached for default-gpt-3.5-turbo in organization org-frdB6Ua4LlRzGlIULT6X1QKc on requests per min. Limit: 3 / min. Please try again in 20s. Contact us through our help center at help.openai.com if you continue to have issues. Please add a payment method to your account to increase your rate limit. Visit https://platform.openai.com/account/billing to add a payment method.
```

(above) Hitting the Rate Limit Error



```
In [28]: import time
         for issue in user_messages:
             prompt = f"Tell me what language this is: ```{issue}```"
             lang = get_completion(prompt)
             print(f"Original message ({lang}): {issue}")

             time.sleep(20)

             prompt = f"""
             Translate the following  text to English \
             and Korean: ```{issue}```
             """
             response = get_completion(prompt)
             print(response, "\n")

             time.sleep(20)
```

```
Original message (The language is French.): La performance du système est plus lente que d'habitude.
The performance of the system is slower than usual.

시스템의 성능이 평소보다 느립니다.

Original message (The language is Spanish.): Mi monitor tiene píxeles que no se iluminan.
English: "My monitor has pixels that do not light up."

Korean: "내 모니터에는 밝아지지 않는 픽셀이 있습니다."

Original message (The language is Italian.): Il mio mouse non funziona
English: "My mouse is not working."
Korean: "내 마우스가 작동하지 않습니다."

Original message (The language is Polish.): Mój klawisz Ctrl jest zepsuty
English: "My Ctrl key is broken"
Korean: "내 Ctrl 키가 고장 났어요"

Original message (The language is Chinese.): 我的屏幕在闪烁
English: My screen is flickering.
Korean: 내 화면이 깜박거립니다.
```

(above) Using 20 second time delays to avoid the Rate Limit Error

Congrats

Your PC is set up and you know everything you need to know to successfully follow along with every example from the course, ChatGPT Prompt Engineering for developers

Everything you need to know to take the free course, ChatGPT Prompt Engineering for developers

Recap

Set up OpenAI and DeepLearning.AI accounts

Installed Anaconda/Jupyter Notebook and supporting Python libraries

Set up an OpenAI key on your PC

Learned how to use Jupyter Notebook

Next Steps

1. Create a new Jupyter Notebook called, *1 Guidelines for Prompting*
2. Start the course
3. Use your newly created notebook, *1 Guidelines for Prompting*, to run the examples from the first chapter of the course

references

Reference	Link
ChatGPT signup/login	https://chat.openai.com/auth/login
Course - ChatGPT Prompt Engineering for Developers	https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/
Installing Anaconda on Windows	https://docs.anaconda.com/free/anaconda/install/windows/
Installing Latest Version of Python in Anaconda	https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-python.html
How to Generate an OpenAI key and install it on your PC	https://www.howtogeek.com/885918/how-to-get-an-openai-api-key/

The End

Contact: mikeboucher@yahoo.com

Appendix Slide - Installing Python 3.10 in Anaconda

If you're are running an older version of Anaconda on your PC already, you may not have Python 3.10 installed. In that case, you'll need to install Python 3.10 in Anaconda. Alternatively, uninstall Anaconda, then reinstall the latest version (that's what I did).

The following slides explain how to do that.

Steps

1. Install Python 3.10 in Anaconda
2. Run Python 3.10 in Anaconda
3. From a Jupyter Notebook, write and run your 1st program to check the python version

Python 3.10

- Anaconda installs Python version 3.7 by default
- Anaconda allows you to run different versions of Python
- One of the examples in the course uses a Python library that requires Python version 3.10 or higher
- So let's install Python version 3.10 in Anaconda

Installing a different version of Python

To install a different version of Python without overwriting the current version, create a new environment and install the second Python version into it:

1. Create the new environment:
   - To create the new environment for Python 3.9, in your terminal window or an Anaconda Prompt, run:

```
conda create -n py39 python=3.9 anaconda
```

Note

Replace `py39` with the name of the environment you want to create. `anaconda` is the metapackage that includes all of the Python packages comprising the Anaconda distribution. `python=3.9` is the package and version you want to install in this new environment. This could be any package, such as `numpy=1.19`, or multiple packages.

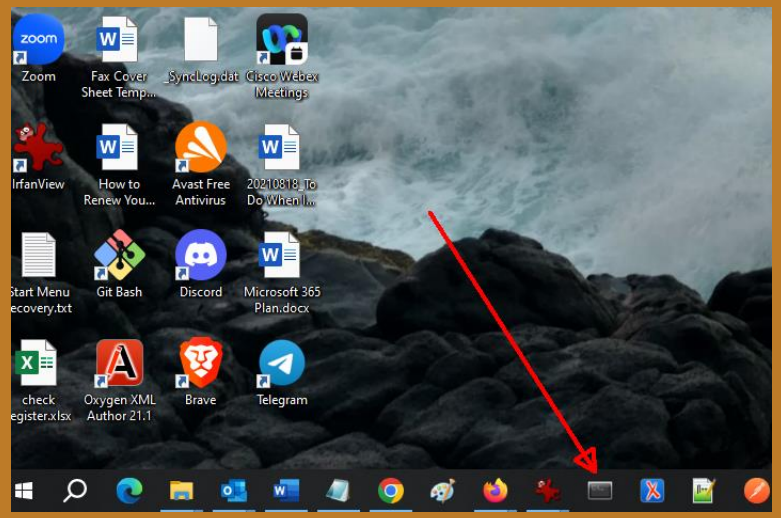
2. Activate the new environment.
3. Verify that the new environment is your current environment.
4. To verify that the current environment uses the new Python version, in your terminal window or an Anaconda Prompt, run:

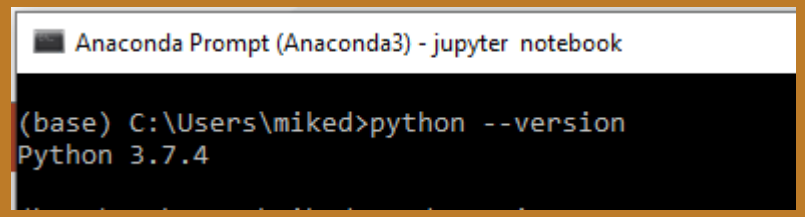
```
python --version
```

We'll follow the instructions from here: https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-python.html

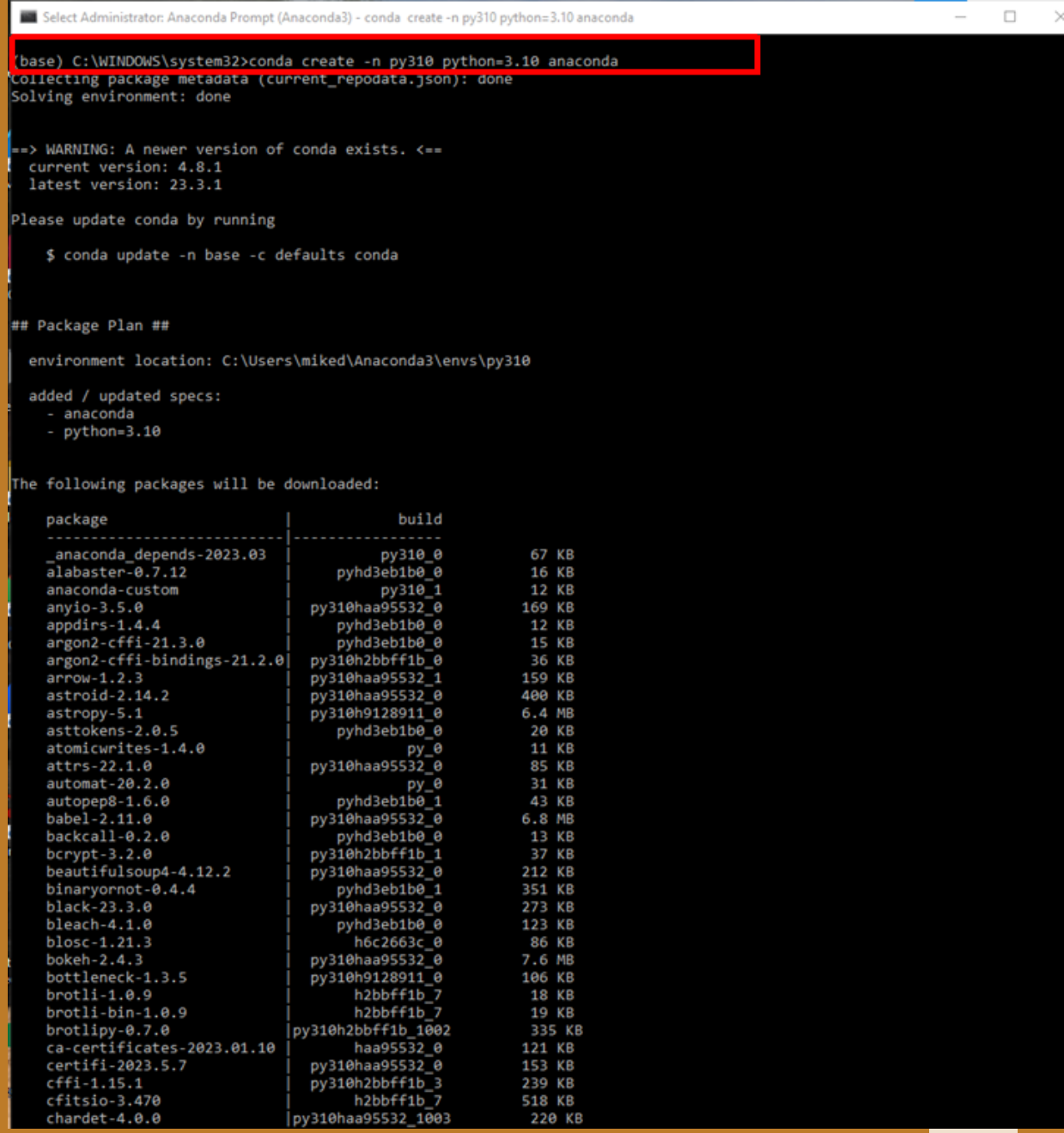
- Open an anaconda prompt
- Check your Python version with: *python --version*
- Run the command: *conda create -n py310 python=3.10 anaconda*
- Then when that's done, activate py310 with the following command: *conda activate py310*
- Verify you're running Python 3.10 with: *python --version*

Installing Python 3.10

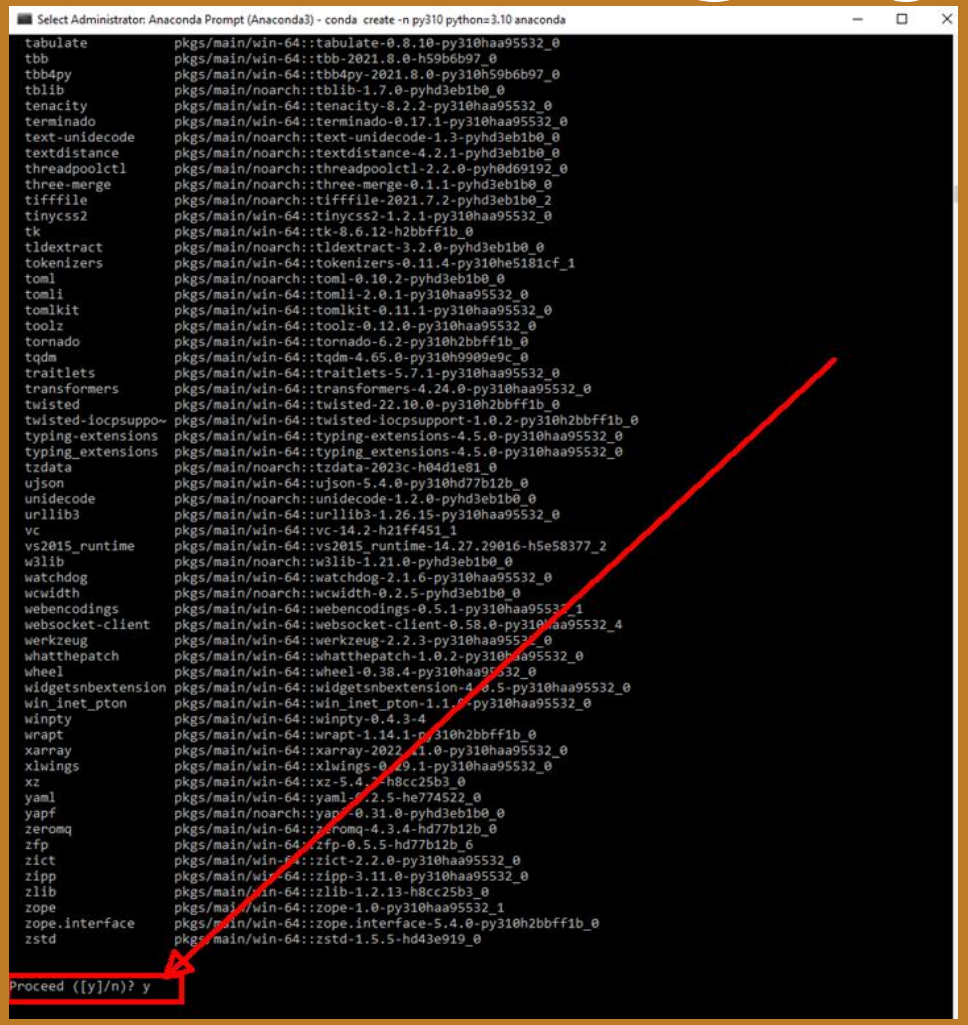




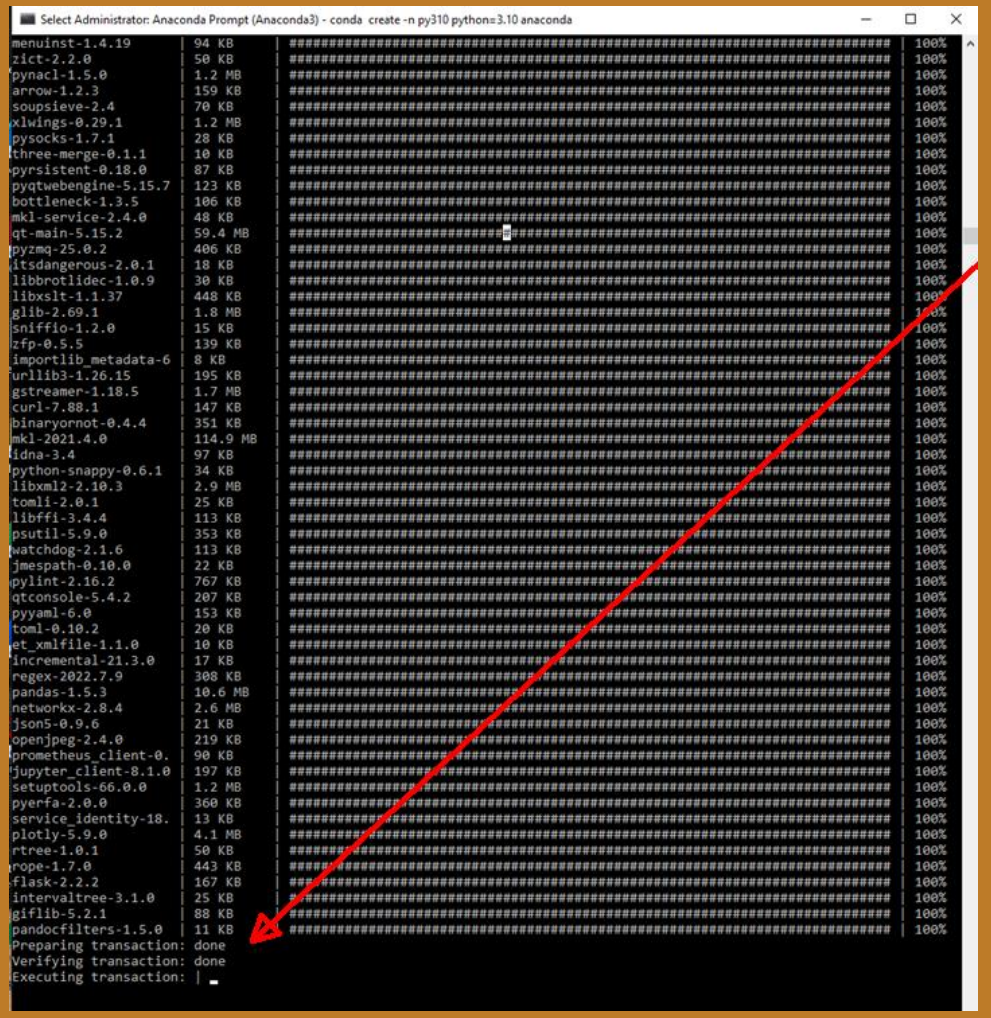
1. Open an anaconda prompt
2. Check your Python version with the command, *python –version*
3. Run the command, *conda create -n py310 python=3.10 anaconda*



Installing Python 3.10

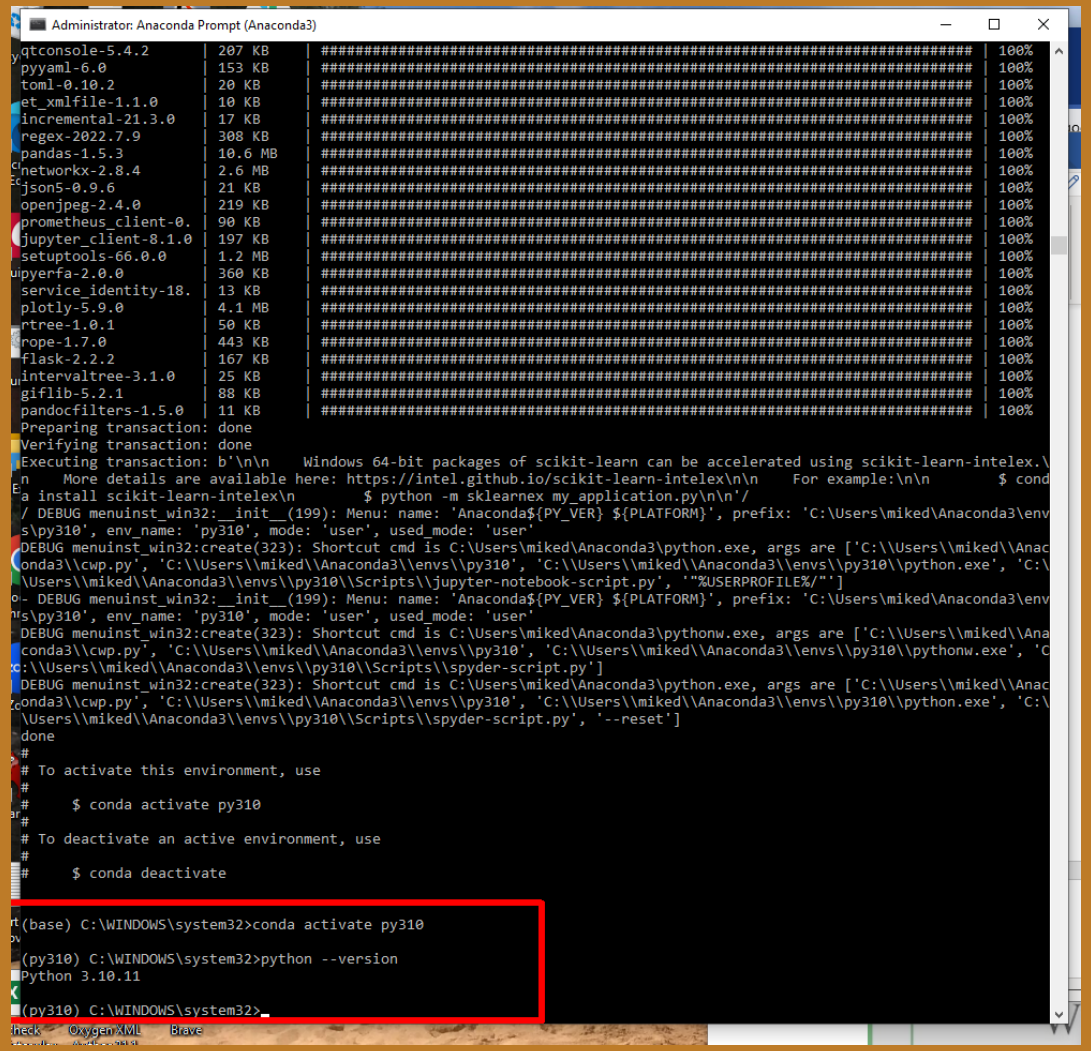


Enter ‘y’ to proceed with the install



If it gets stuck at **Executing transaction, do a** CTRL-C

Installing Python 3.10 – Last Step



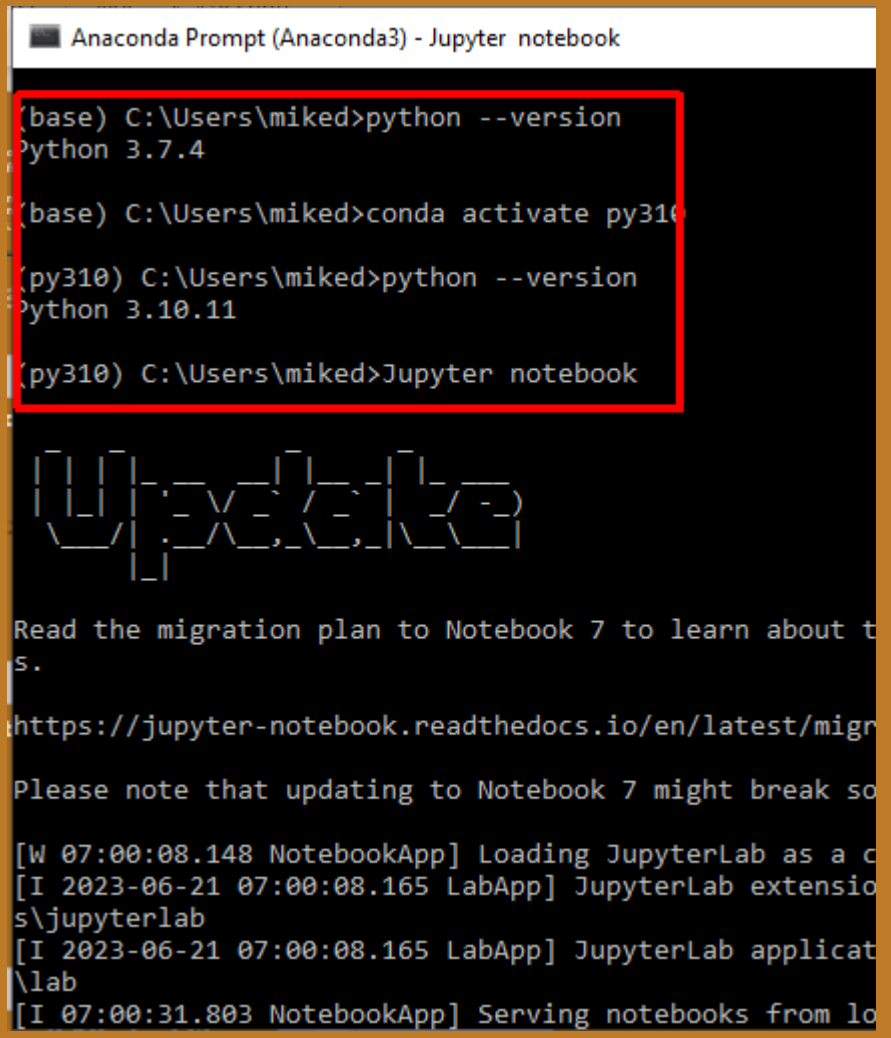
- After the install completes, activate py310 with the following command: *conda activate py310*
- Verify you're running Python 3.10 with: *python --version*

Launching Jupyter Notebook on Python 3.10

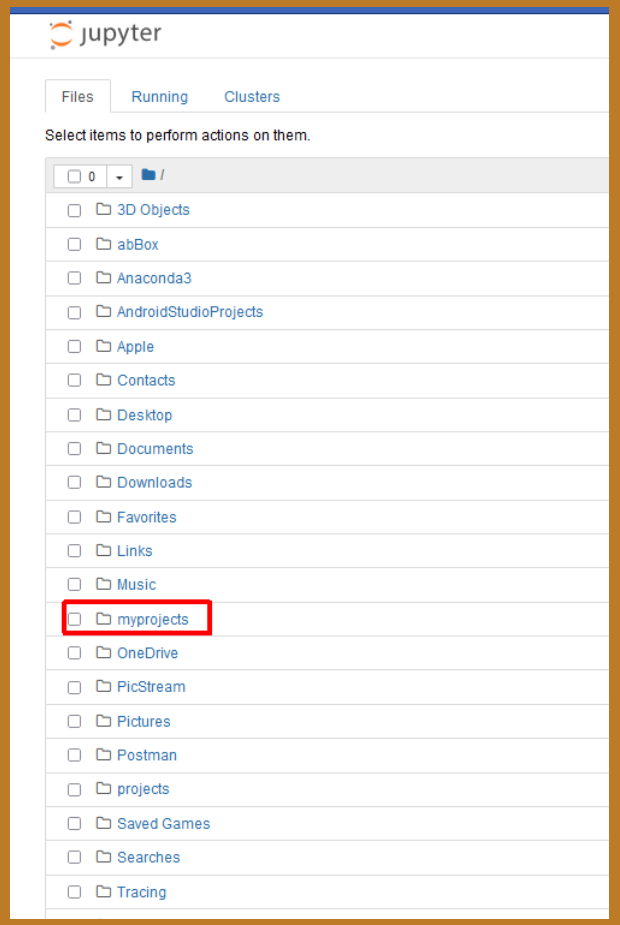
From the Anaconda prompt:

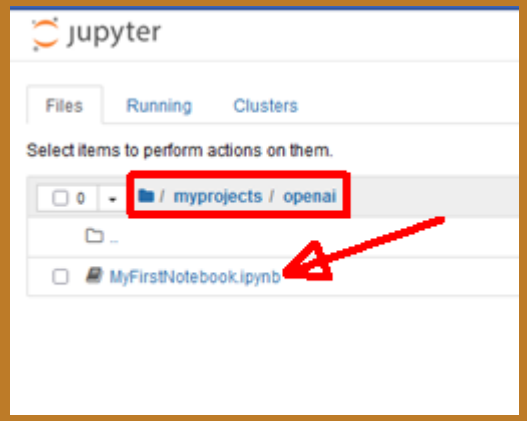
1. *python --version // Check the Python version that is running*
2. *conda activate py310 // Activating Python 3.10*
3. *python –version // Verify that you're running Python 3.10*
4. *Jupyter notebook // open Jupyter Notebook*

Wait for a minute Jupyter Notebook to open in your default browser



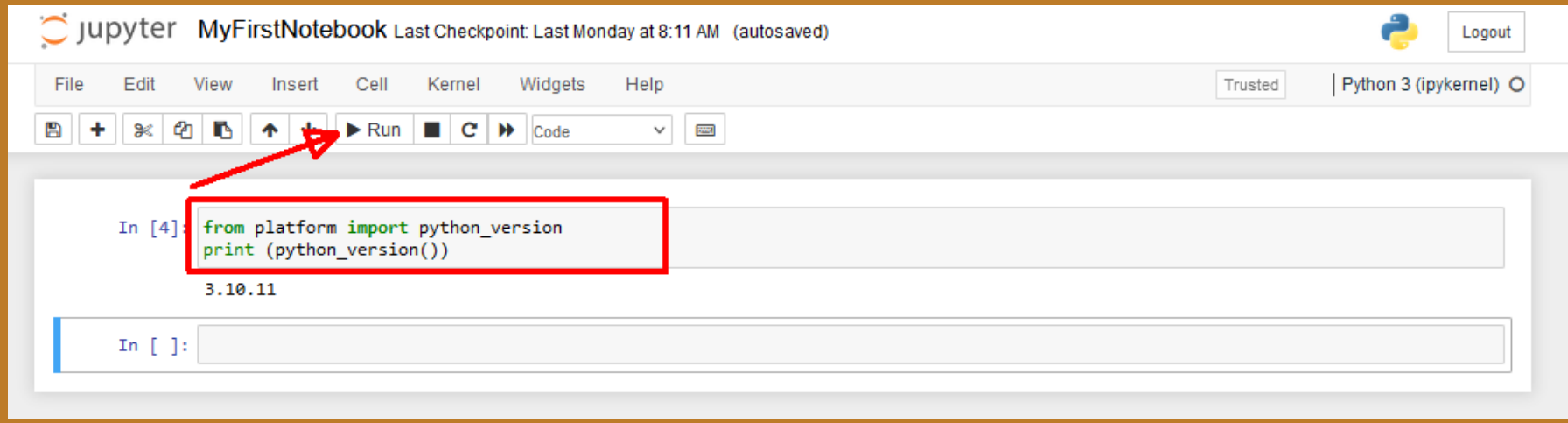
Checking Python Version in Jupyter Notebook





1. In jupyter, navigate to myprojects > openai
2. Open MyFirstNotebook
3. In the notebook, enter and then run the following commands

   from platform import python_version
   print (python_version())



Congrats

You’re now running Python 3.10 and have written and run your first python program in Jupyter Notebook!

Feel free to add Python and Jupyter Notebook as skills on your LinkedIn profile

Everything you need to know to take the free course, ChatGPT Prompt Engineering for developers