Streamlining Invoice Processing: Document Understanding Transformer for Efficient Business Operations in Python

Pre-Workshop Guide

January 2024

https://algorit.ma

Contents

Introduction 2

Installation 3

- Installing Python using *Miniconda*
- Installing Visual Studio Code
- Installing Python & Jupyter Extension in VS Code

System Verification 4

- Verify Miniconda Installation
- Verify VSCode Installation
- Verify Python & Jupyter Extension in VS Code Installation

Introduction

This guide is a resource for students at Algoritma to use in setting up their laptop or environment prior to the scheduled workshops. In this guide, students can find a list of prerequisites that will be consistently used throughout the entire course. These prerequisites are required to be **completed before** the start of the workshop.

For new students, we will run through the installation process to ensure that the necessary programming languages and tools - such as Python - are installed. The next section will then talk about methods on how to verify whether the installs were completed successfully.

For recurring students, we recommend repeating the System Verification section oncemore to confirm past completed installations.

Installation

Installing Python using *Miniconda*

To install Python, we will use **Miniconda,** a minimal version of Anaconda that includesonly *conda*, Python, and the essential packages they rely on. Miniconda provides not only Python but also the required packages (such as numpy and pandas) used in our workshops. Please follow the link below and **select Python version 3 for installation.**

Use this link:
https://docs.conda.io/en/latest/miniconda.html#latest-miniconda-installer-links for Miniconda

More info on *Miniconda*:
https://docs.conda.io/en/latest/miniconda.html

Installing Visual Studio Code

To set up a code editor, we'll be using Microsoft's Visual Studio Code, often referred to as **VS Code.** This robust and versatile code editor supports multiple programming languages and offers a wide range of extensions to enhance its functionality. Follow the steps below to install VS Code:

1. Open your web browser and **go to the official Visual Studio Code downloadpage** using this link: https://code.visualstudio.com/download
2. On the download page, you'll find options for Windows, macOS, and Linux. **Selectthe appropriate download link that matches your operating system.**
3. **Once the installation is complete, launch VS Code.** You'll be greeted with awelcome screen like the picture below.

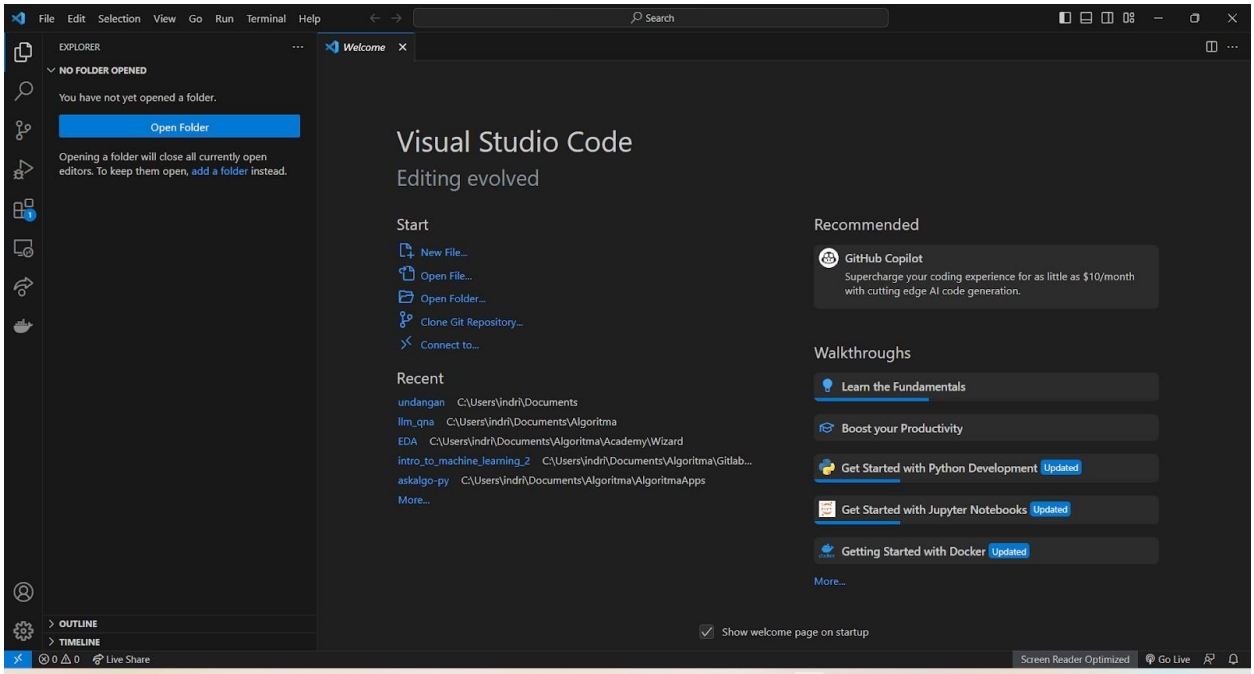


Figure 1: Visual Studio Code's Welcome Page

Installing Python & Jupyter Extension in VS Code

To set up VS Code with Miniconda, we'll require the 'Python' and 'Jupyter' extensions in VS Code. The 'Python' extension is necessary for managing Python environments and providing code assistance. Moreover, the 'Jupyter' extension is essential for editing, running, and interacting with Jupyter notebooks within VS Code. Follow the steps belowto install VS Code:

1. Open VSCode and **navigate to the 'Extension' menu on the sidebar** (*highlighted in green boxes in the image below*)**.** Alternatively, you can also open this menu using the shortcut **Ctrl+Shift+X** on Windows or **Cmd+Option+X** on macOS.

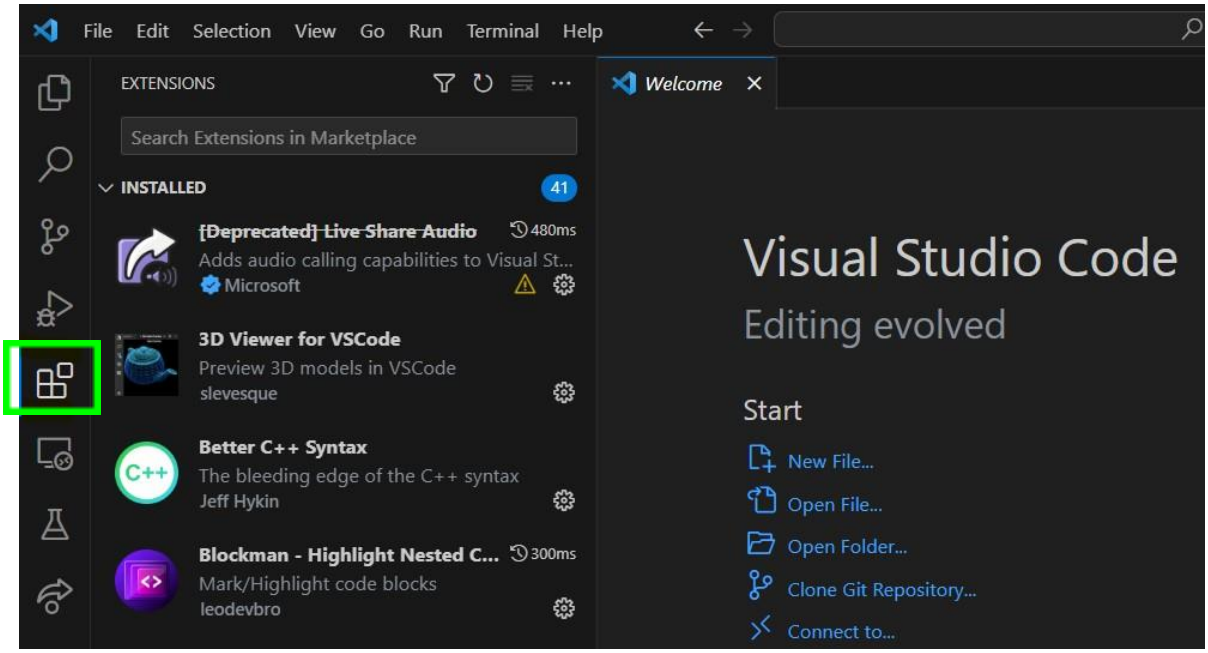


Figure 2: VS Code Extensions on the Activity Bar

Python Extension Installation

2. **Search 'Python' on Extension** search bar. Then, press **Install**

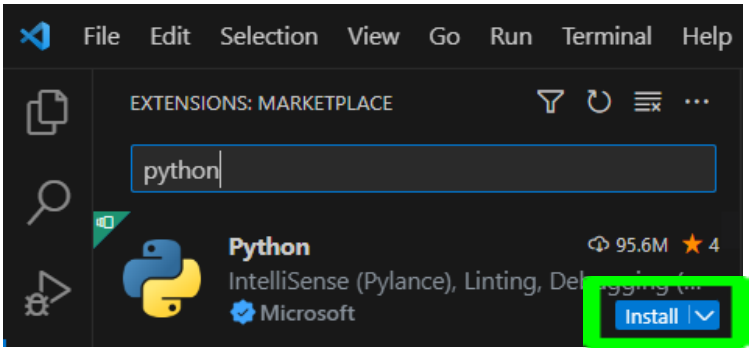


Figure 3: Searching for PythonExtension in VS Code Extensions

3. **Once the 'Python' extension installation is complete,** the page view is like picture below

Figure 4: PythonExtension Installed in VS Code Extensions

Jupyter Extension Installation

4. **Search 'Jupyter' on Extension** search bar. Then, click **Install**

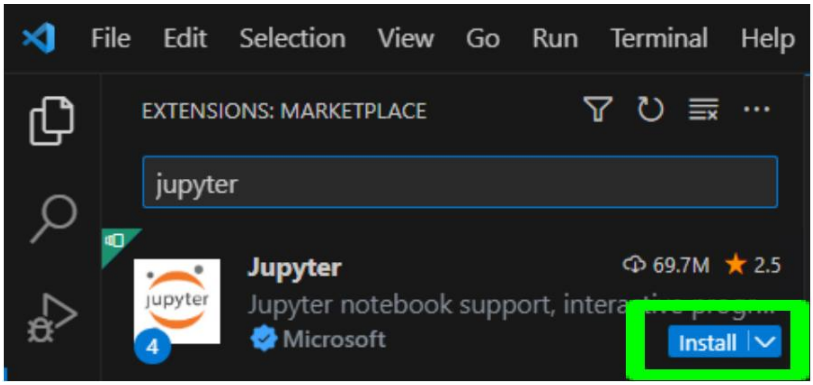


Figure 5: Searching for Jupyter Extension in VS Code Extensions

5. **Once the 'Jupyter' extension installation is complete,** the page view is like picture below

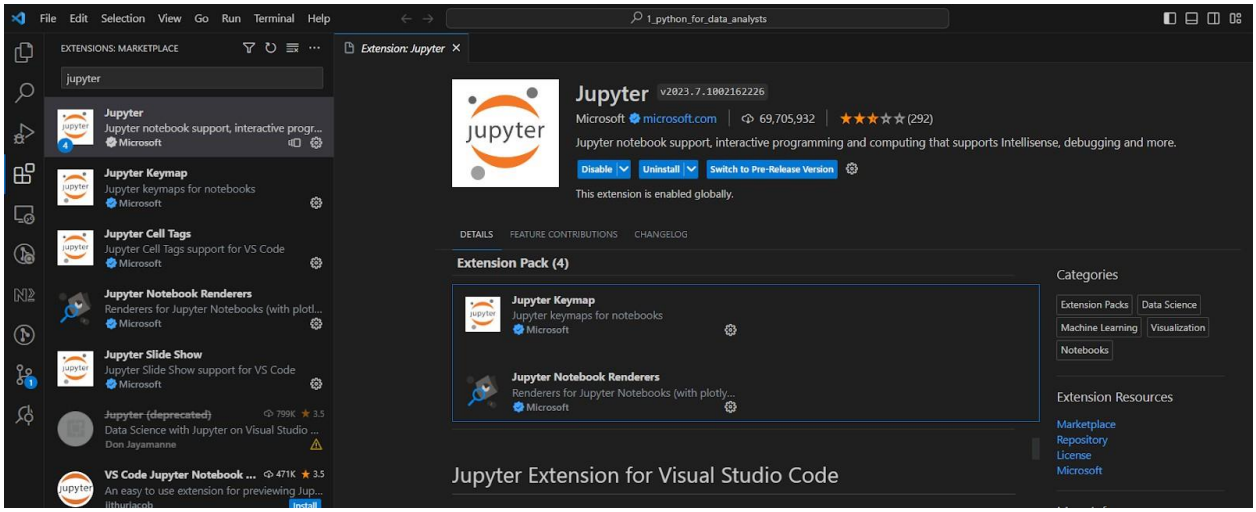


Figure 6: Jupyter Extension Installed in VS Code Extensions

System Verification

- **For Mac OS X and Linux-based OS:** Open "*Terminal*"
- **For Windows:** Open "*Anaconda Prompt (miniconda3)*"

Verify Anaconda Installation

1. Type the command `conda list` in your *"Terminal"* or *"Anaconda Prompt(miniconda3)"*.
2. If the installation was completed successfully, your terminal will give a response oflist of packages like the example below.
3. If your terminal does not give any response, please check the installation section'sWarning, if the problem still persists, kindly reach out for further assistance via email at mentor@algorit.ma.

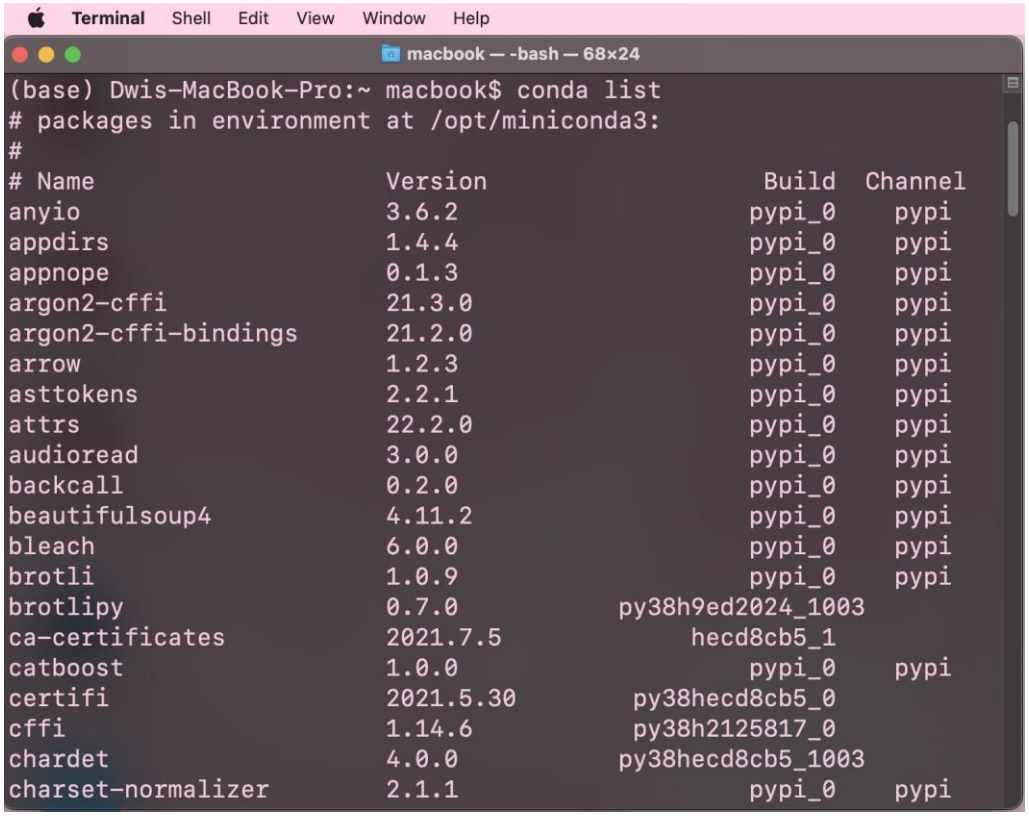


Figure 7: `conda list` Response on Mac OS X Terminal

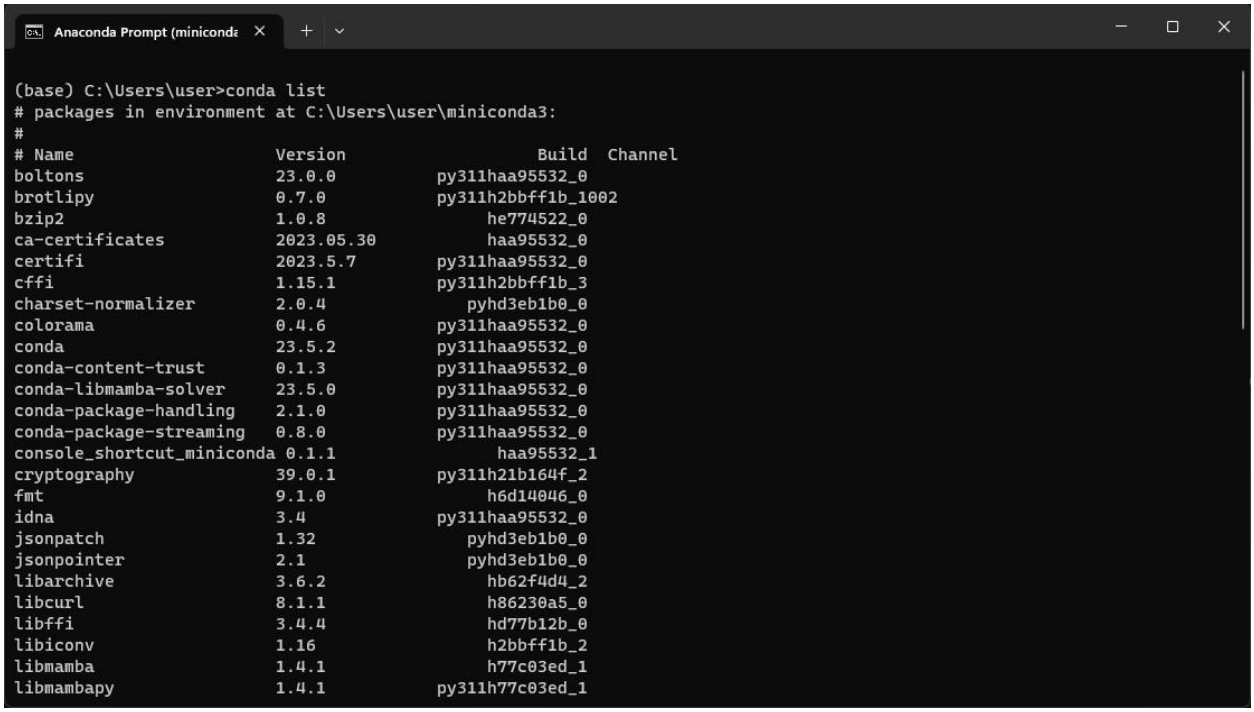


Figure 8: `conda list` Response on Anconda Prompt (miniconda3)

Verify 'Python' and 'Jupyter' Extension on VS Code Installation

1. Open VS Code and press **Ctrl+Shift+P** on Windows or **Cmd+Option+P** on MacOSuntil a modal appears like the one pictured below. This modal is called the Command Palette, and from here, we can access all the functionality of VS Code.

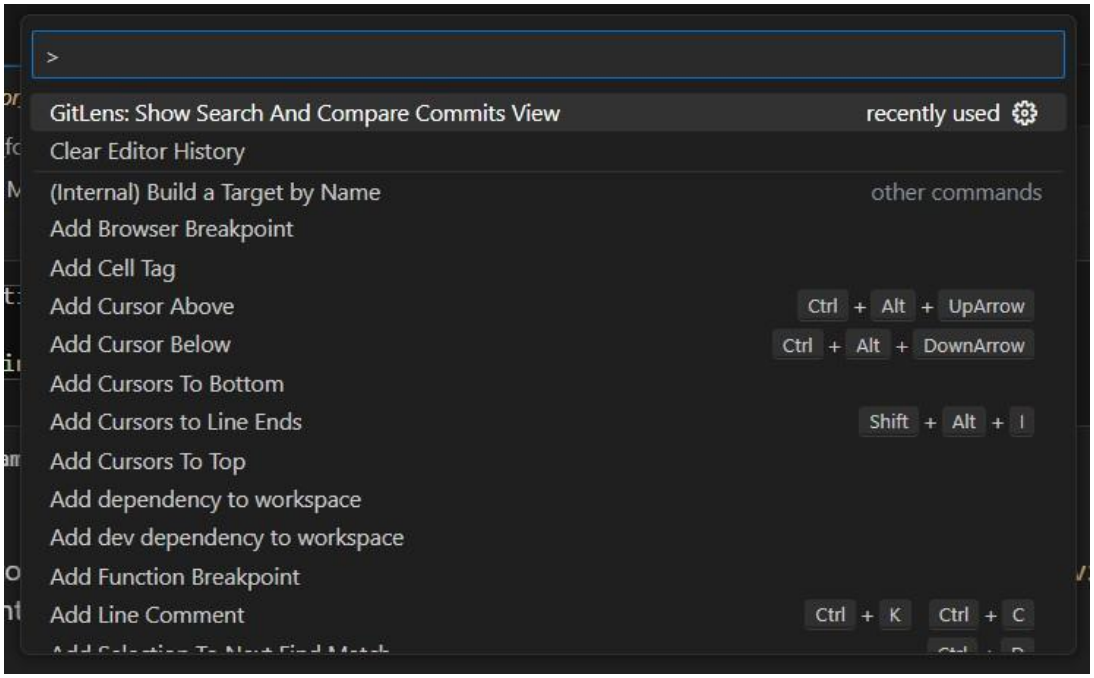


Figure 9: Opening the Command Palette in VS Code

2. Type 'interpreter' to the search bar

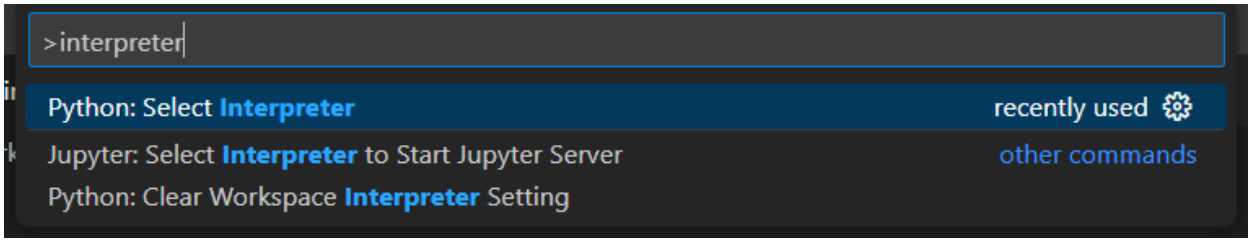


Figure 10: Typing 'Interpreter' to Find the 'Python: Select Interpreter' Menu

3. Click on "Python: Select Interpreter" then you can select the **base** environment (~\miniconda3\python.exe).

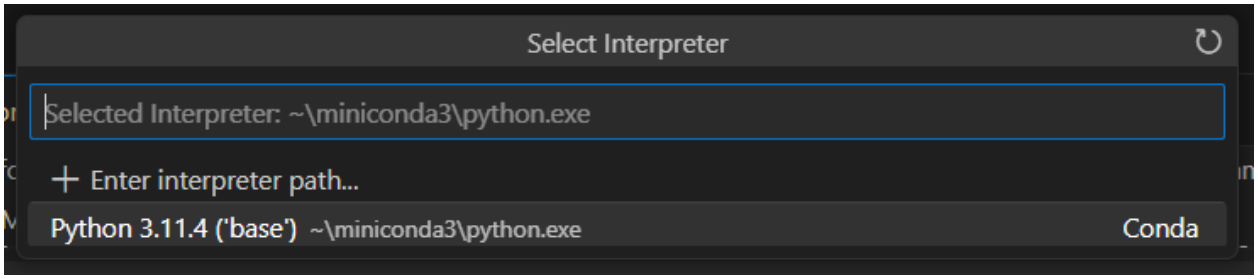
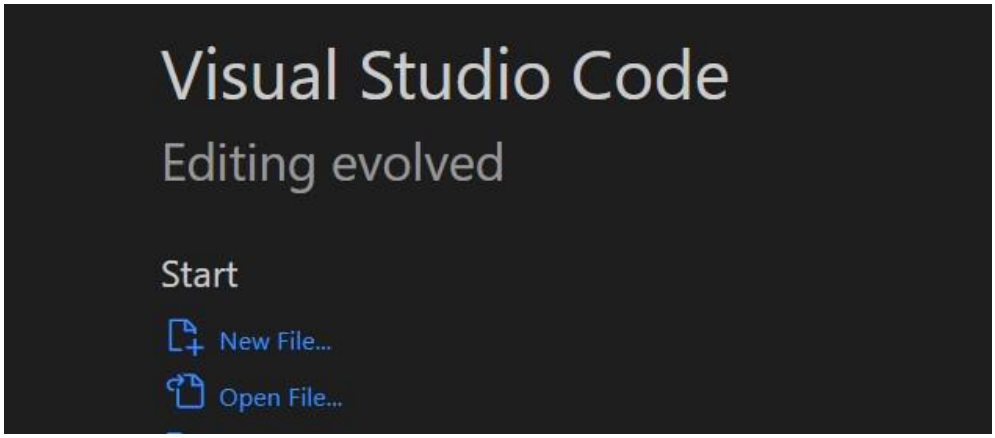


Figure 11: Selecting 'Python 3 ('base') ~\miniconda\python.exe'

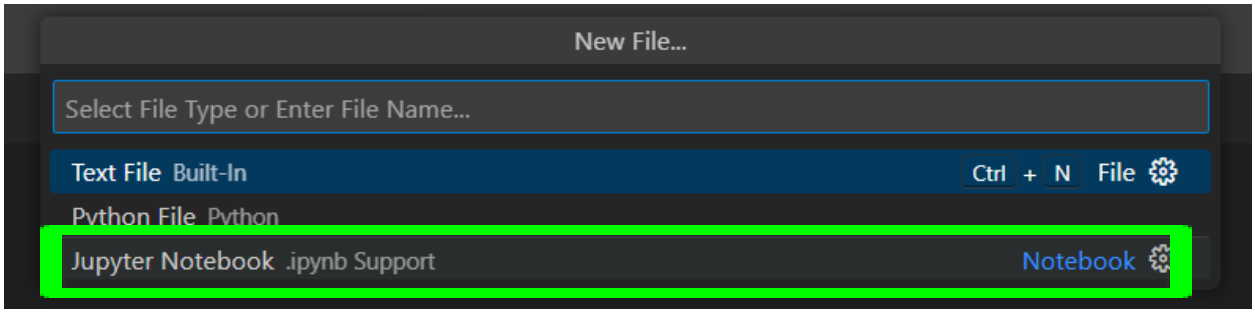
Alternative Kernel Selection

Instead of manually using Ctrl+Shift+P or Cmd+Shift+P in VSCode, you can select the kernel by clicking on "Select Kernel" in the upper right-hand corner of your notebook

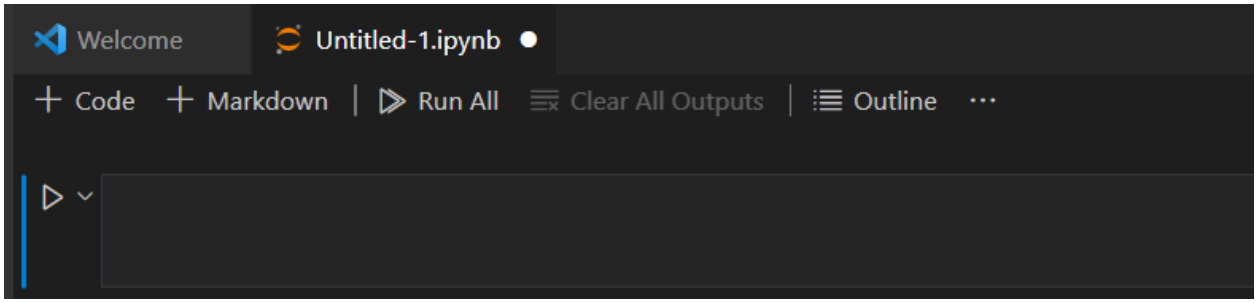
1. Create **a new notebook** by clicking the **'New File'** button.

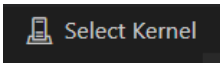


2. A modal will appear, prompting you to choose a file type. **Select the 'Jupyter Notebook .ipynb Support' file.**

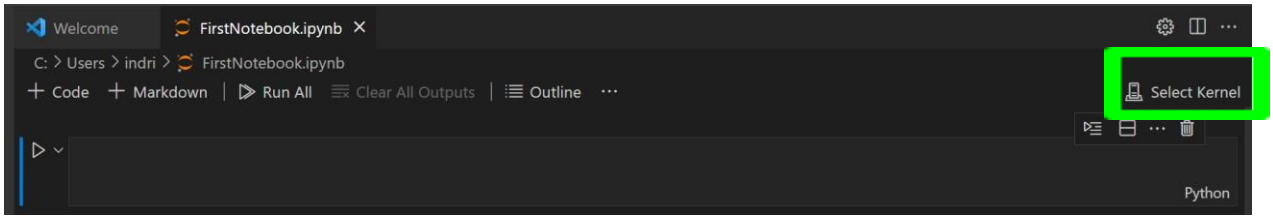


3. Afterward, an 'Untitled.ipynb' file will appear as shown below. Save it as **'FirstNotebook.ipynb.'"**

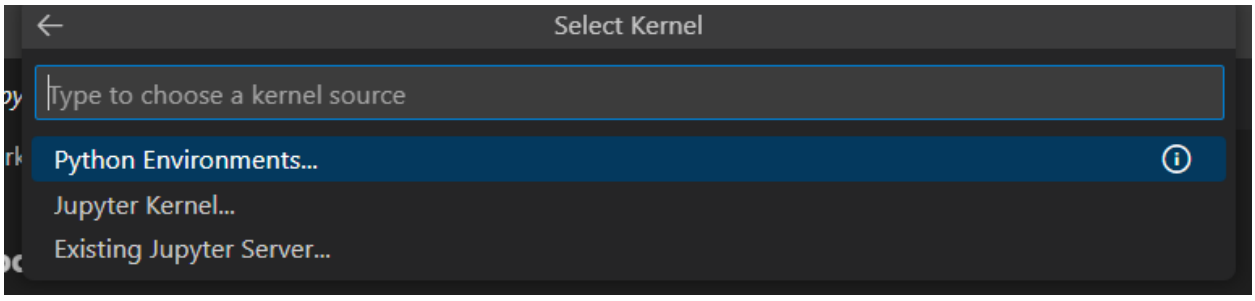


4. Look for the 

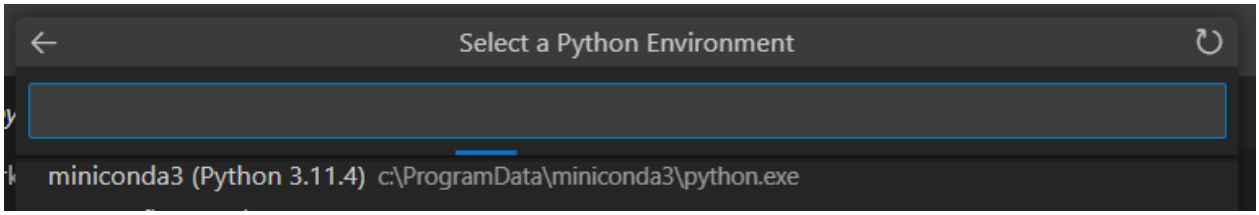
option on the upper right-hand corner of your notebook interface. It's typically a dropdown menu or a button with a kernel icon.



5. Click on it to see a list of available kernels. Kernels are associated with specific programming languages or environments, so choose the one that matches yourrequirements.
    a. Select "**Python Environments**"



b. After installing Miniconda for the first time, you can select the **'miniconda3'** kernel, which is the default Python 3.X.X kernel provided by Miniconda



6. Once you've selected the desired kernel, your notebook will start using that kernel for code execution.

This makes it easy to switch between different programming languages or environments within a Jupyter Notebook without having to remember keyboard shortcuts.

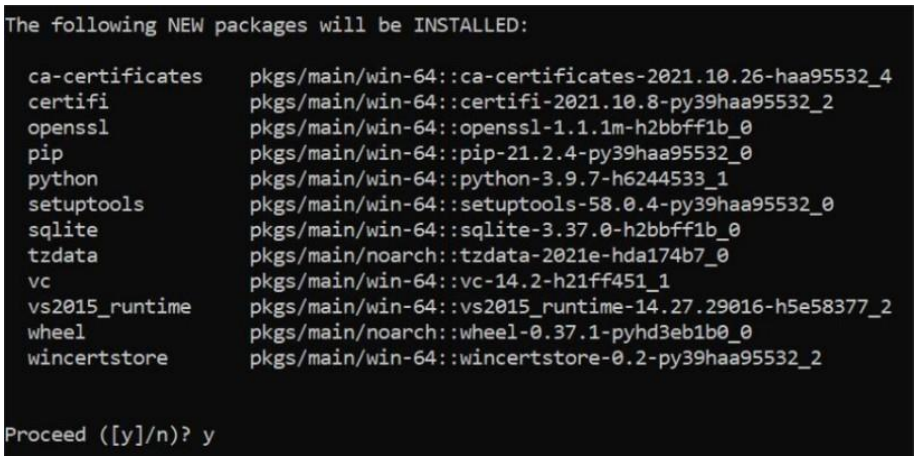
Virtual Environment and Package Preparation *

1. Search and open **Anaconda Prompt (miniconda3)** application for windows, or **Terminal** for Mac

2. Create a new virtual environment named **dss_donut** using Python version **3.10**

   ```
   conda create -n dss_donut python=3.10
   ```

3. Proceed the installation by typing **y**



4. Activate the newly created virtual environment, namely **dss_donut**

   ```
   conda activate dss_donut
   ```

5. Change your terminal directory to the path where the **requirements.txt** is located.For example, if your txt file is in the Downloads directory, use: cd Downloads

6. For standardization packages and libraries installation, please install using the **requirements.txt** shared to you.

   ```
   pip install -r requirements.txt
   ```

   Wait until the installation is complete, and after that, you are good to go.

Notes: If you choose not to install the package via the shared requirements.txt, make sure you have the following package version for standardization:

1. git+https://github.com/huggingface/transformers.git
2. torchvision==0.16.1
3. gradio==4.7.1
4. sentencepiece==0.1.99
5. protobuf==4.23.4
6. pandas==2.1.4

* : will be discussed later in day 1 class