Python

Qt

Python PyQt6 GUI Project Setup.

This post assumes the following

1. You are on **Windows** OS

2. You have **Python** knowledge

3. You have:
   - **Python** installed and added to PATH
     - (Guide)
   - **VS Code** installed
     - (Guide)
     - (Setup)
   - You have installed the following extensions on VS Code:
     - **Python**
     - **Pylance**

Ready to dive into the world of Python GUI development?

PyQt6 is an excellent choice for creating sleek, modern desktop applications.

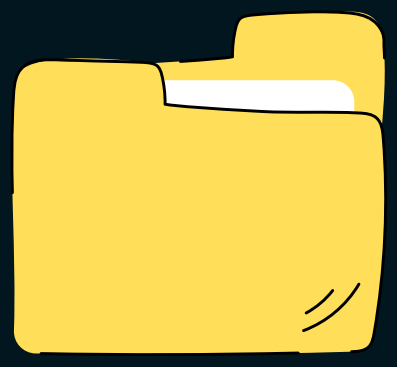
In this post, I'll walk you through setting up your PyQt6 project from scratch.

Environment Setup

It’s best to use a **virtual environment**.

This keeps your project **dependencies isolated** and **manageable**.

Let’s set that up. **Shall we?**



Create a project folder

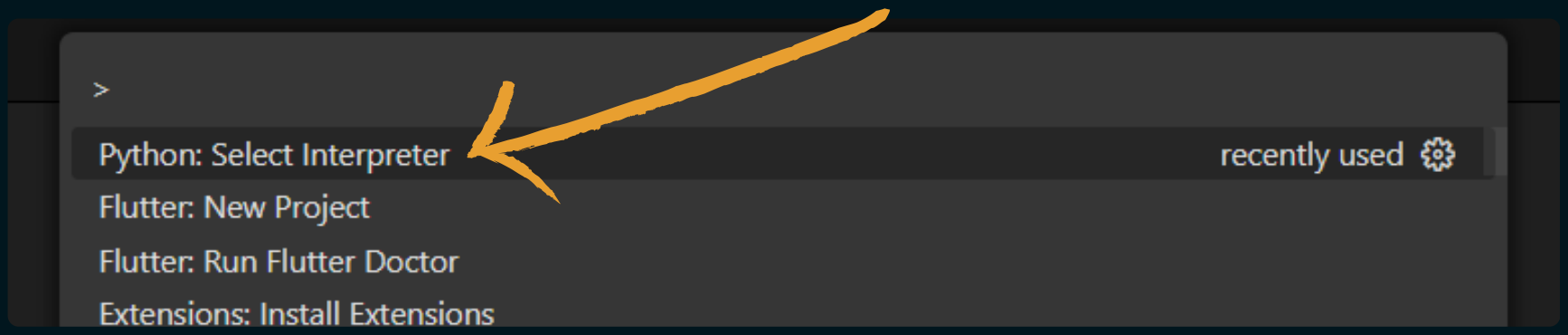
- Open your File Manager and create a folder for your codes.

- Open the folder using VS Code

The “Ctrl+Shift+P” Command

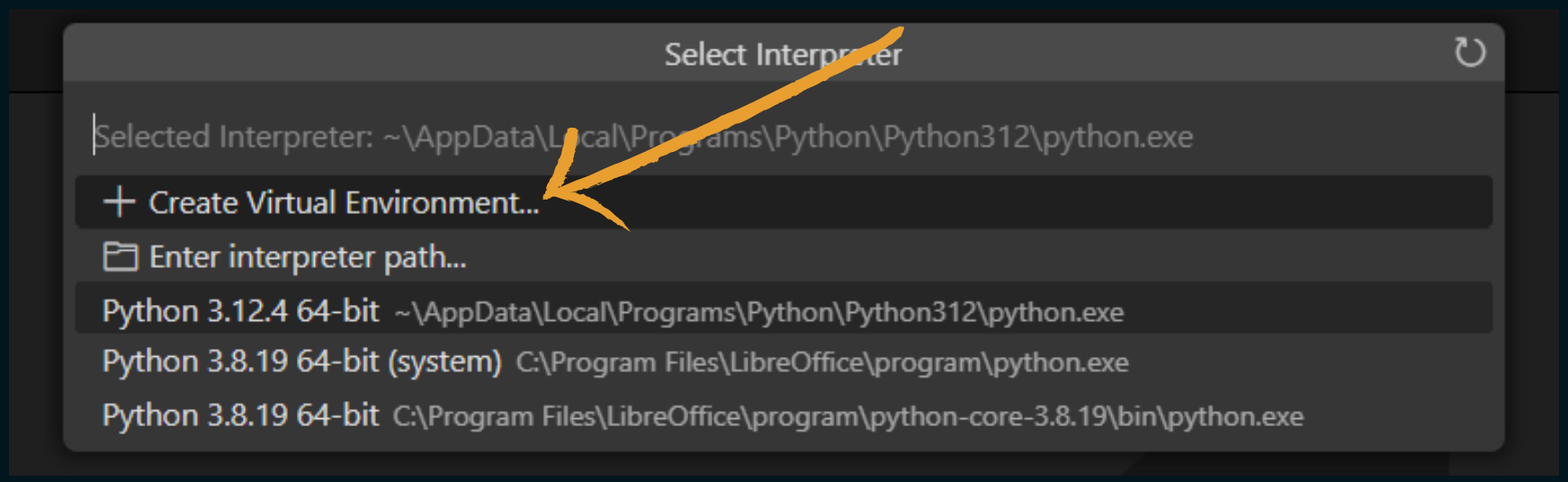
Once VS Code opens, press *ctrl+shift+p*

A popup similar to the one below opens

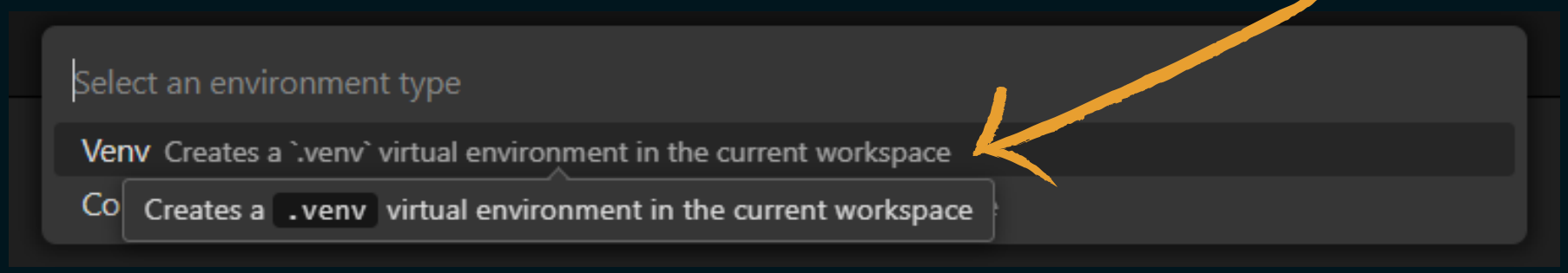


Select the option: “Python: Select Interpreter”

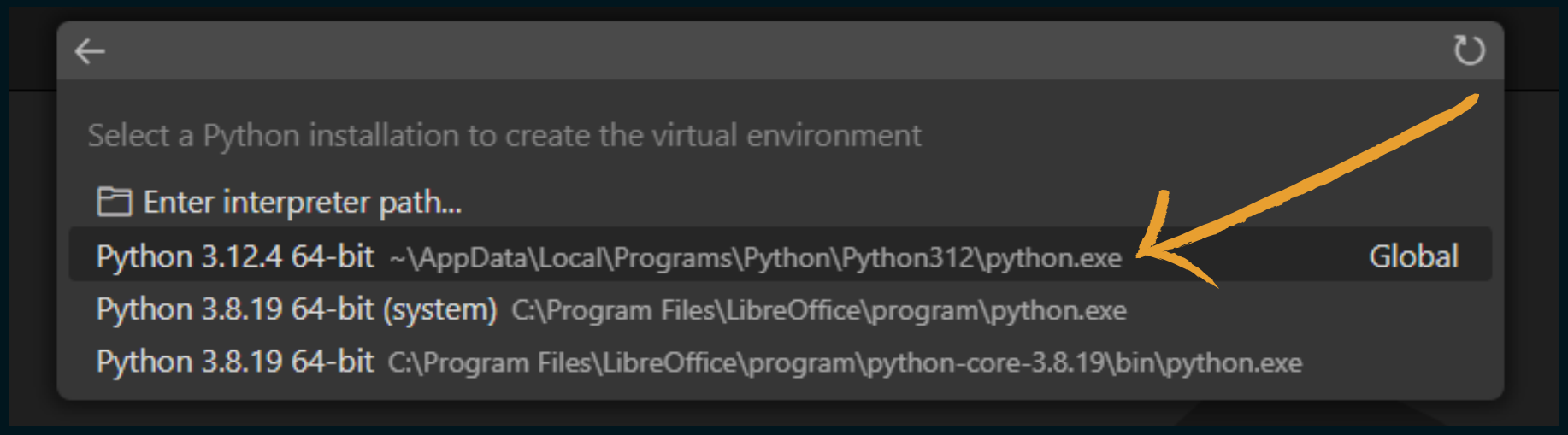
+ CREATE



Select the option: “Create Virtual Environment”



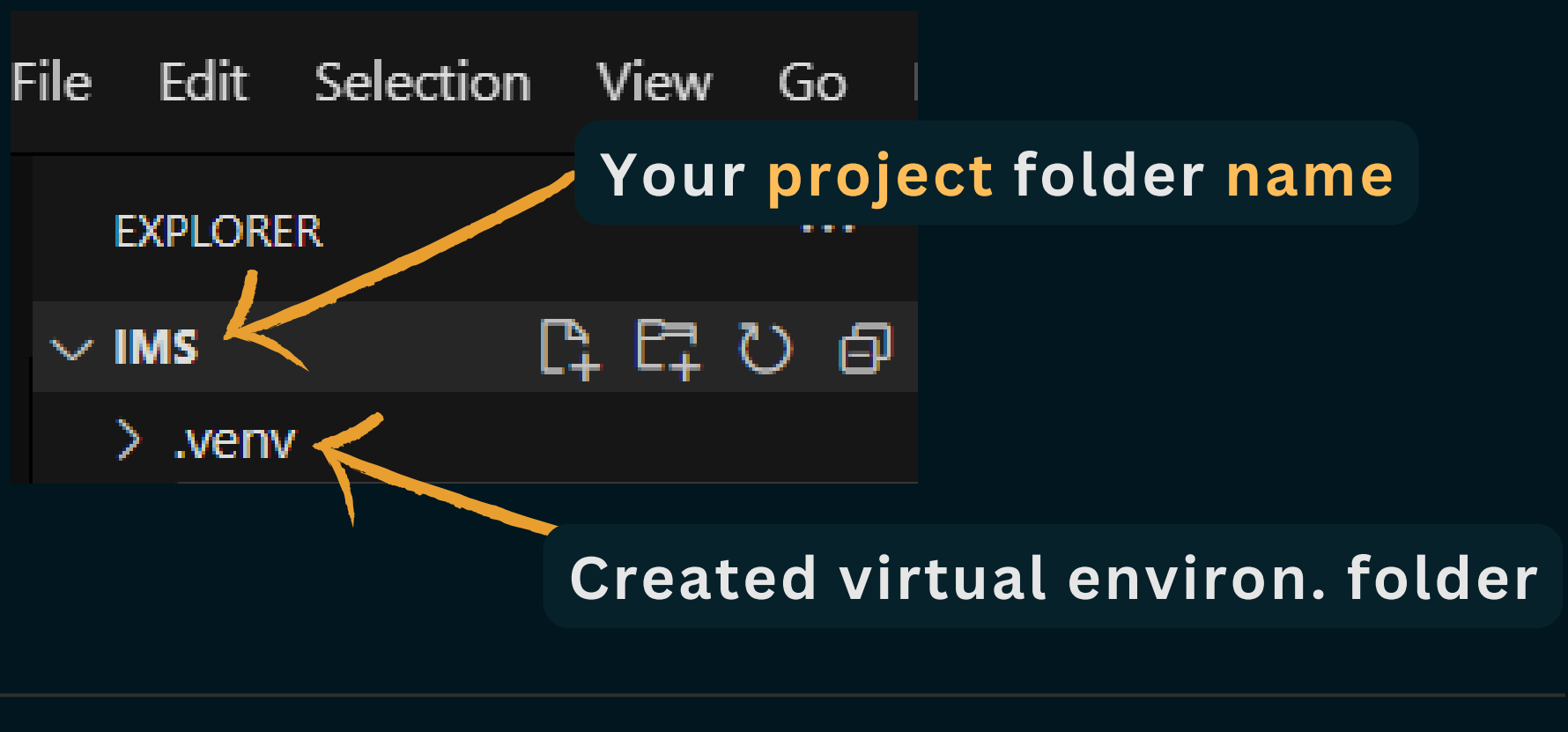
Then select the option: “Venv...”



Select the version of Python, if you have many:

VS Code will take a minute to create the environment. You will see a notification popup on the same.

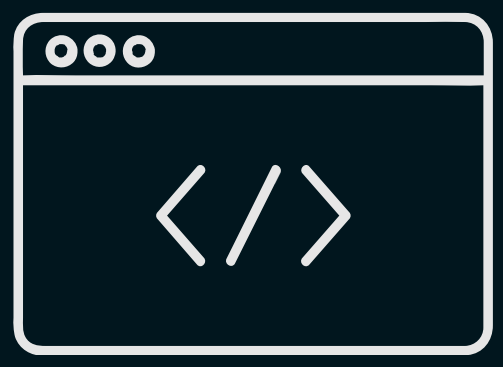
The result, on your left sidebar



VS Code raises errors when I try to open virtual environments in its terminal.

Real quick, let me show you a method I use to open virtual environments.

The Batch script



Create a file named “activate.bat” in the project folder

```
@echo off
REM Define the path to the file you want to execute
set "filePath=path\to\your\project\.venv\Scripts\activate.bat"

REM Run the file
"%filePath%"
```

Add the contents to the file “activate.bat”

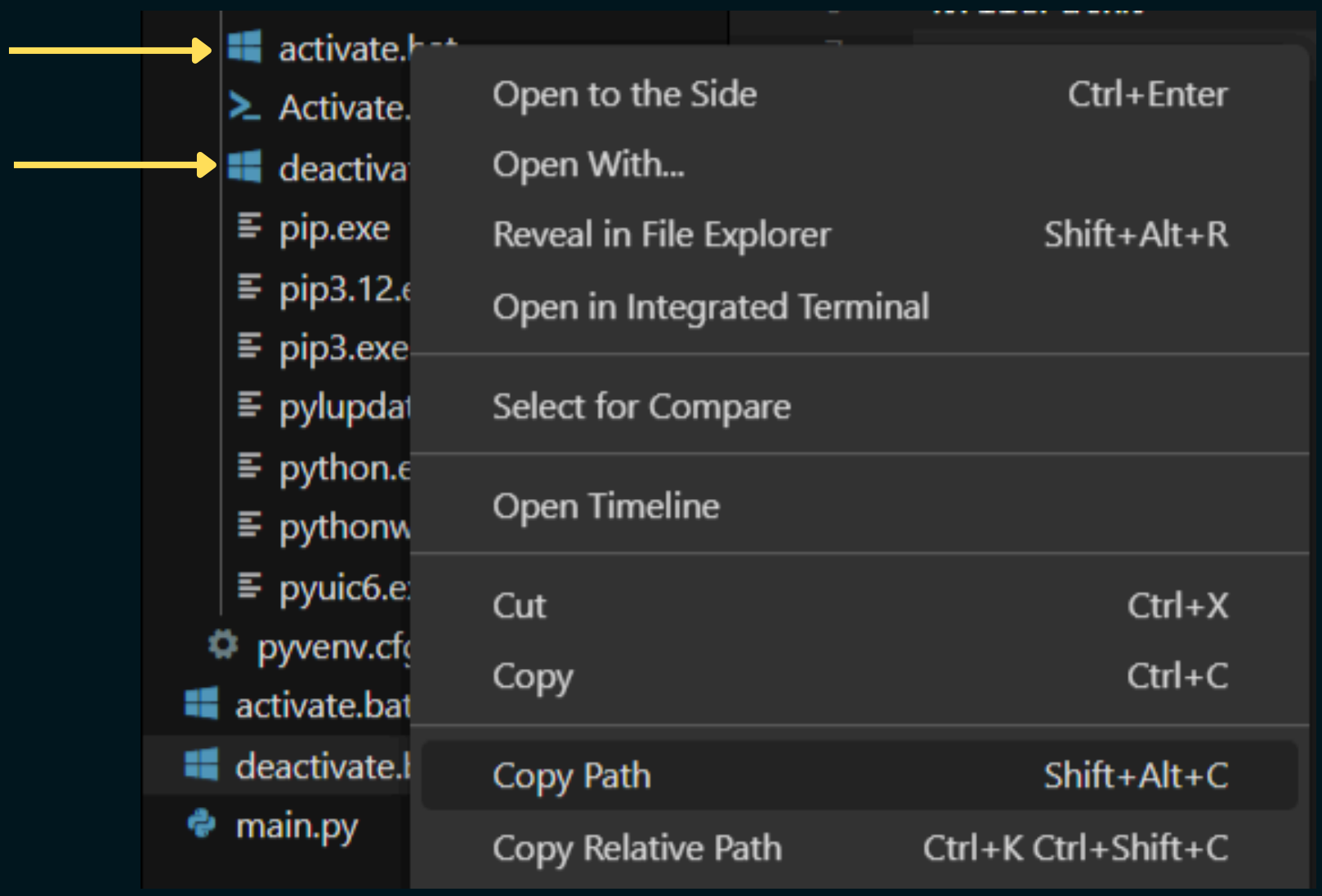
Create another file named “deactivate.bat”

```
@echo off
REM Define the path to the file you want to execute
set "filePath=path\to\your\project\.venv\Scripts\deactivate.bat"

REM Run the file
"%filePath%"
```

Change the path to that of deactivate

Or simply copy and paste the paths by right-clicking them



The paths we're interested in (activate.bat and deactivate.bat) are in the .venv/Scripts folder.

The benefits of creating the two files:

- To reduce the commands involved when (de)activating a virtual environment
- You can place the project and the virtual environment in separate folders and you will be sure that when you call the files in your terminal, the correct virtual environment is (de)activated

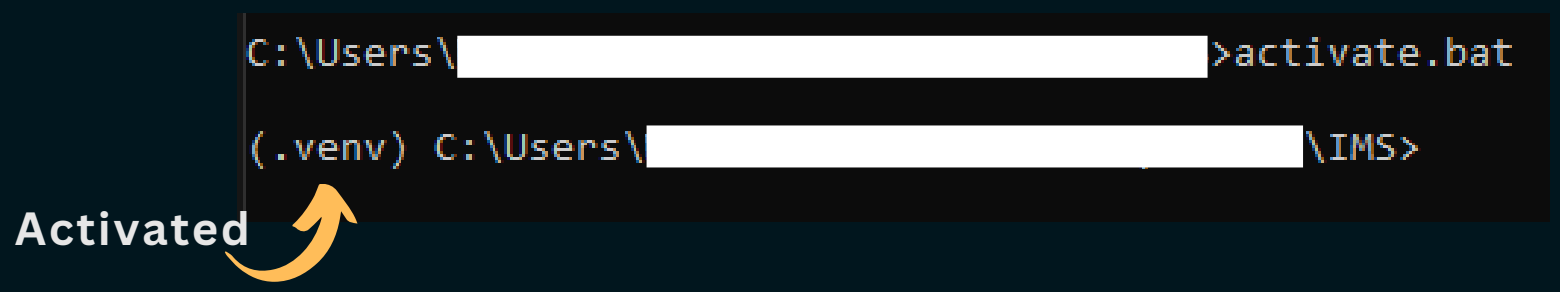
pip install packages (PyQt6)

Open your project folder in the VS Code terminal or the Windows one

(there should be only 1 folder and 2 files)

```
project folder
   |---.venv\
   |---activate.bat
   |---deactivate.bat
```

Type “activate.bat” and press “enter”, no quotes



Next, type the command “pip install PyQt6” and press “enter”

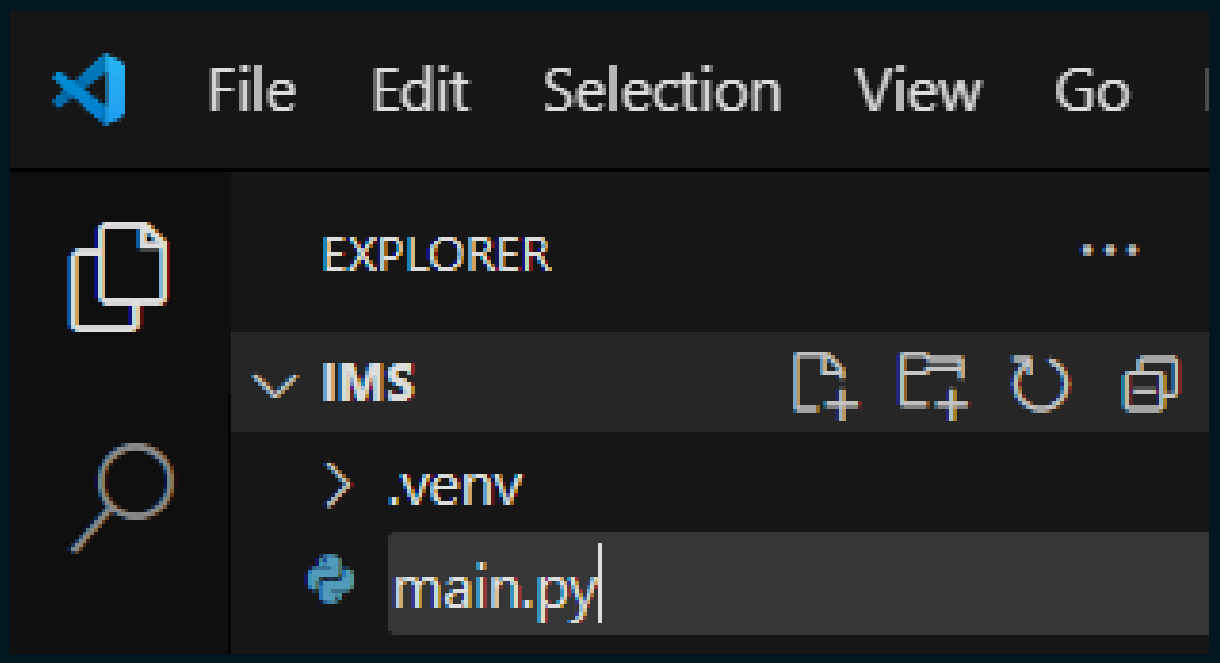
Done with the setup

The packages will download and install. You will see a success message when done.

```
Successfully installed PyQt6-6.7.1 PyQt6-Qt6-6.7.2 PyQt6-sip-13.8.0
```

(The version numbers may be higher, depending on how far into the future you're reading this)

Let’s write our first desktop app



Create a file, in your project folder, named “main.py”

Our first app code

(Write the code below in “main.py”)

```
import sys
from PyQt6.QtWidgets import QApplication, QMainWindow

# Create a class for the main window
class MainWindow(QMainWindow):
    def __init__(self):
        super().__init__()
        # Set the window title
        self.setWindowTitle("Hello PyQt6")
        # Set the window size and position
        self.setGeometry(100, 100, 800, 600)


if __name__ == "__main__":
    # Create the application object
    app = QApplication(sys.argv)
    # Create an instance of the main window
    window = MainWindow()
    window.show() # Show the window
    sys.exit(app.exec()) # Start the application's event loop
```

(Save the code, make sure the environment is activated, and run it)

Challenge:

Create a **Button** (widget) with a text “Change” and add it to the **MainWindow**

Next Topic: Using layout managers to arrange multiple widgets