Virtual Env Walkthrough - VS CODE

macOS

Install Python Package from Brew

1. Open Terminal.
2. Install Xcode Command Line Tools if you haven't already (may take time):

```
xcode-select --install
```

3. Install Homebrew if you haven't already:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

4. Install Python 3.11 using Homebrew:

```
brew install python@3.11
```

MAC OS VS CODE

5. Install VS Code from the official website: https://code.visualstudio.com/download
6. Launch VS Code.
7. Open the Command Palette by pressing `Cmd + Shift + P`.
8. Type "Python: Select Interpreter" and choose it from the list.
    a. If you do not see this, you'll need to install Python extension for VS Code first!
9. In the Python interpreter selection window, click on "Enter interpreter path...".

a. Select "Find...".

b. Navigate to the directory where you want to create your virtual environment and click "New Folder".

c. Name the folder (e.g., `myenv`) and click "Create".

d. Select the newly created folder and click "Open".

e. In the terminal within VS Code, run the following command to create a virtual environment:

```
python3.11 -m venv .
```

10. VS Code will automatically activate the virtual environment.

    a. Alternatively you can run: `source .venv/bin/activate` in terminal

    b. You should also verify this by checking the Python interpreter path in the bottom-left corner of the VS Code window.

        i. If it does not automatically update, you can change to the correct Python 3.11 environment WITHIN `myenv` (not `base`)

    c. It should display the path to your virtual environment.

11. Open a new terminal in VS Code to ensure the virtual environment is activated.

12. Upgrade pip to the latest version:

```
pip install --upgrade pip
```

13. Check Python Version:

    a. `python --version` or `python3 --version` or `python3.11 --version` whichever one provides that you have python3.11 active in your environment

14. Finally we may have to check about `Jupyter` `Python` and `ipykernel` extensions in VS Code being installed properly for everything to work smoothly (heads-up!)

WINDOWS (VS CODE Venv)

1. Download the Python 3.11 installer from the official website: https://www.python.org/downloads/windows/
2. Run the installer and make sure to check the option "Add Python 3.11 to PATH" during the installation process.
3. Install VS Code from the official website: https://code.visualstudio.com/download
4. Launch VS Code.
5. Open the Command Palette by pressing `Ctrl + Shift + P`.
6. Type "Python: Select Interpreter" and choose it from the list.
    a. If you do not see this, you'll need to install Python extension for VS Code first!
7. In the Python interpreter selection window, click on "Enter interpreter path...".
    a. Select "Find...".
    b. Navigate to the directory where you want to create your virtual environment and click "New Folder".
    c. Name the folder (e.g., "myenv") and click "Create".
    d. Select the newly created folder and click "Select Folder".
    e. In the terminal within VS Code, run the following command to create a virtual environment:

```
python3.11 -m venv .
```

8. ALTERNATIVE to step 7: Create Environment Directly in **Terminal VS Code**
    a. `cd` to your desired workspace folder (for our class materials)
    b. `python3.11 -m venv myenv`
    c. `source myenv/bin/activate`

    d. If you have done it correctly you should see `.venv` instead of `base` at the beginning of your `$` Terminal line

9. VS Code will automatically activate the virtual environment. You can verify this by checking the Python interpreter path in the bottom-left corner of the VS Code window. It should display the path to your virtual environment.
    a. If vs-code does not automatically update, you can change to the correct Python 3.11 environment WITHIN `myenv` (not `base`)
10. Open a new terminal in VS Code to ensure the virtual environment is activated.
11. Upgrade pip as last step to the latest version:

```
python -m pip install --upgrade pip
```

Now you have a Python 3.11 virtual environment set up in VS Code on both macOS and Windows. Your students can use this environment to work on their Python projects, install packages using `pip`, and maintain a consistent Python version across their development environments.

12. To deactivate the virtual environment, simply close the VS Code window or open a new terminal and run:

```
deactivate
```

By following these steps, your students should be able to set up a uniform Python 3.11 environment in VS Code for your Intermediate Python class.

13. Finally we may have to check about `Jupyter` `Python` and `ipykernel` extensions in VS Code being installed properly for everything to work smoothly (heads-up!)

WINDOWS (Backup - Command Prompt)

1. Download the Python 3.11 installer from the official website: https://www.python.org/downloads/windows/
2. Run the installer and make sure to check the option "Add Python 3.11 to PATH" during the installation process.
3. Open Command Prompt.
4. Navigate to the directory where you want to create your virtual environment:

   ```
   cd C:\path\to\your\project
   ```

5. Create a new virtual environment with Python 3.11:

   ```
   python -m venv myenv
   ```

6. Activate the virtual environment:

   ```
   myenv\Scripts\activate
   ```

7. Verify that you are using the correct Python version:

   ```
   python --version
   ```

   It should display "Python 3.11.x".

8. Upgrade pip to the latest version:

   ```
   python -m pip install --upgrade pip
   ```

9. To deactivate the virtual environment when you're done working on your project (after this course or to return to normal environment), simply run:

```
deactivate
```

10. Install VS Code from the official website: https://code.visualstudio.com/download
11. Launch VS Code.
12. Open the Command Palette by pressing `Ctrl + Shift + P`.
13. Type "Python: Select Interpreter" and choose it from the list.
    a. If you do not see this, you'll need to install Python extension for VS Code first!
14. VS Code will automatically activate the virtual environment.
    a. Alternatively you can run: `source .venv/bin/activate` in terminal
    b. You should also verify this by checking the Python interpreter path in the bottom-left corner of the VS Code window.
        i. If it does not automatically update, you can change to the correct Python 3.11 environment WITHIN `myenv` (not `base`)
    c. It should display the path to your virtual environment.

Now you have a Python 3.11 virtual environment set up on Windows. You can install packages using `pip` within this virtual environment without affecting your system-wide Python installation. Wonderful! Let's move on to VS Code Set up: