

# ENDG 233: Setup Instructions

```
object to mirror  
mirror_mod.mirror_object  
operation == "MIRROR_X":  
    mirror_mod.use_x = True  
    mirror_mod.use_y = False  
    mirror_mod.use_z = False  
operation == "MIRROR_Y":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = True  
    mirror_mod.use_z = False  
operation == "MIRROR_Z":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = False  
    mirror_mod.use_z = True
```

```
selection at the end -add  
mirror_ob.select= 1  
mirror_ob.select=1  
context.scene.objects.active  
["Selected"] + str(modifier)  
mirror_ob.select = 0  
= bpy.context.selected_objects  
data.objects[one.name].select  
print("please select exactly
```

--- OPERATOR CLASSES ---

```
bpy.types.Operator):  
    "X mirror to the selected  
    object.mirror_mirror_x"  
    "mirror X"
```

# Installation Steps

## Setting up development environment:

- Step 0: Chromebook users pre-setup
- Step 1: Download and install VS Code
- Step 2: Install Python Extension
- Step 3: Install Python
- Step 4: Set Default Terminal to Command Prompt or Bash
- Step 5: Select Python Interpreter
- Step 6: Setup Working Directory
- Step 7: Create Virtual Environment and activate virtual environment

# Step 0: Chromebook users only

Enabling Linux Development environment:

1. Open **settings**
2. Locate **Linux Beta** option (you can use the search settings field in the top and type in Linux to find it)
3. **Turn on** Linux Beta and follow the installation steps

# Step 0: Chromebook users only

Know your device hardware:

1. Open Chrome browser
2. Type this in the navigation bar  
**Chrome://system**
3. Scroll down to **cpuinfo** and press  
**expand**
4. Locate the **model name** info
  - Look up your processor model online to know if it is an ARM processor or not

This will come in handy in Step 1

# Step 1: Download VS Code and install

Go to <https://code.visualstudio.com/download>

Linux and Chromebook users download the 64 bit unless:

- Processor is ARM then download ARM for 32 bit systems or ARM 64 or 64 bit systems



↓ Windows

Windows 7, 8, 10

User Installer	64 bit	32 bit	ARM
System Installer	64 bit	32 bit	ARM
.zip	64 bit	32 bit	ARM



↓ .deb

Debian, Ubuntu

↓ .rpm

Red Hat, Fedora, SUSE

.deb	64 bit	ARM	ARM 64
.rpm	64 bit	ARM	ARM 64
.tar.gz	64 bit	ARM	ARM 64

Snap Store



↓ Mac

macOS 10.11+

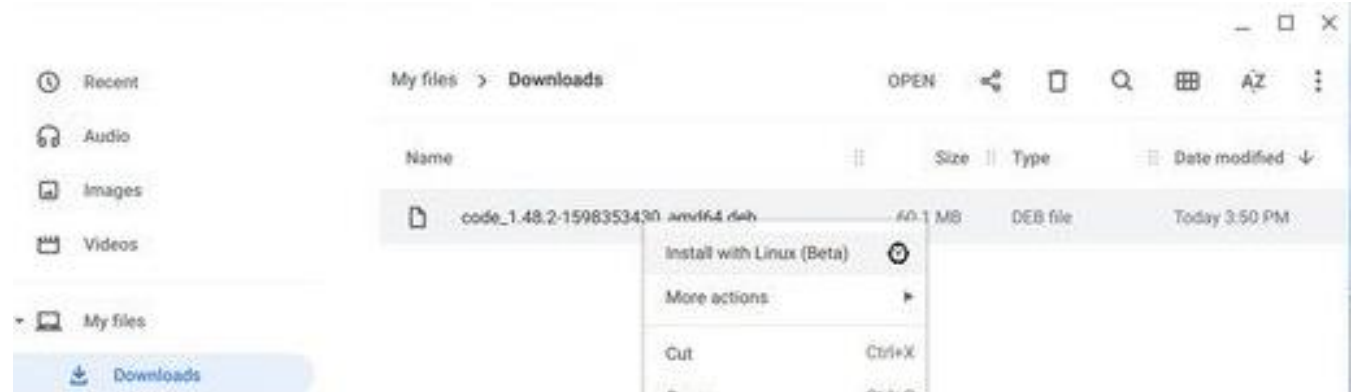
.zip Universal Intel Chip Apple Silicon

Download 64 bit option unless:

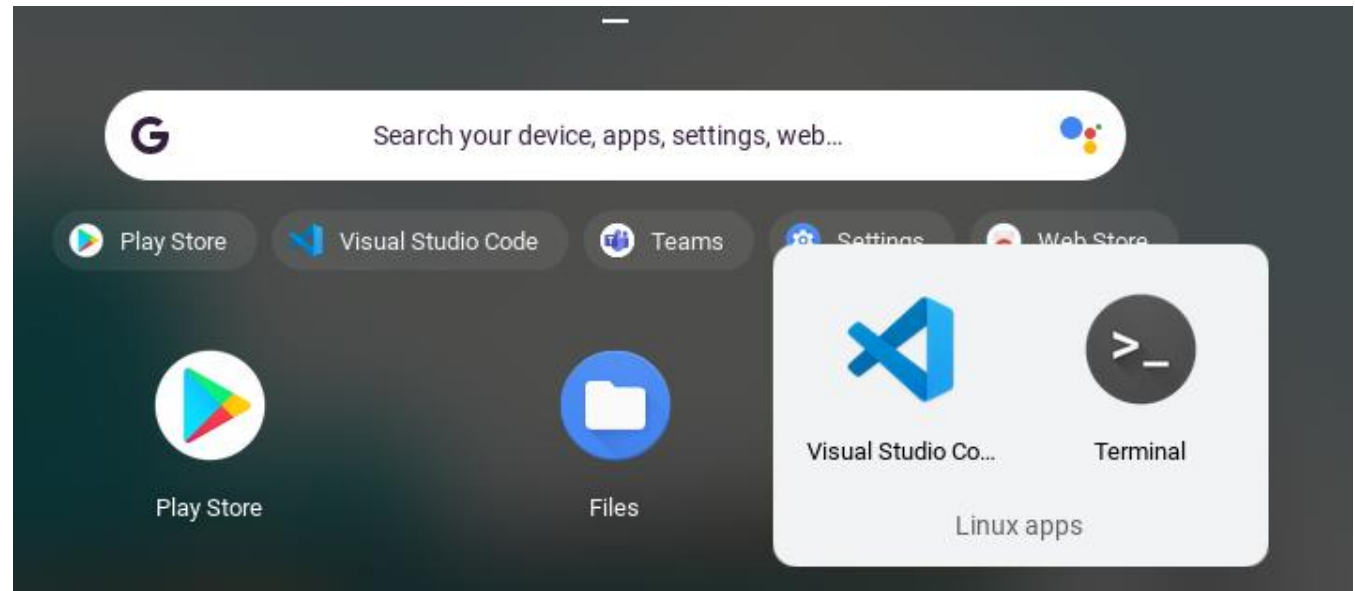
- Windows version on your device is 32 bit
- Surface X users (ARM processors) download ARM option

# Step 1: Install VS code special instructions for Chromebooks

1. Right-click the downloaded installer and select install with Linux Beta

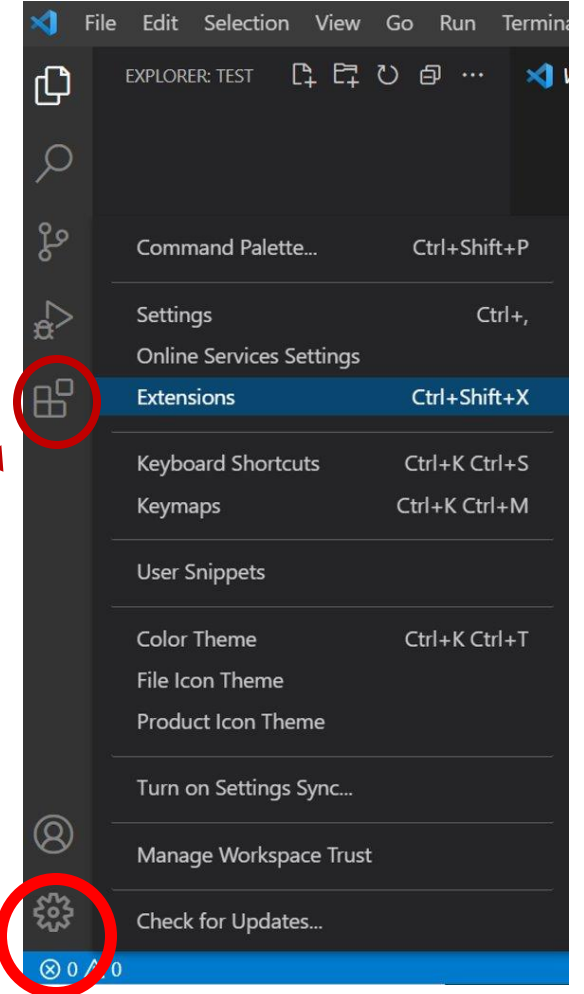


2. To launch VS Code, you can find it in your app folder inside the Linux apps



## Step 2: Install Python Extensions

- Open Visual Studio code
- Go to **Settings** (gear symbol) - > **Extensions** (Ctrl+Shift+X)
- or, use the quick access icon





## Step 2 (continued)

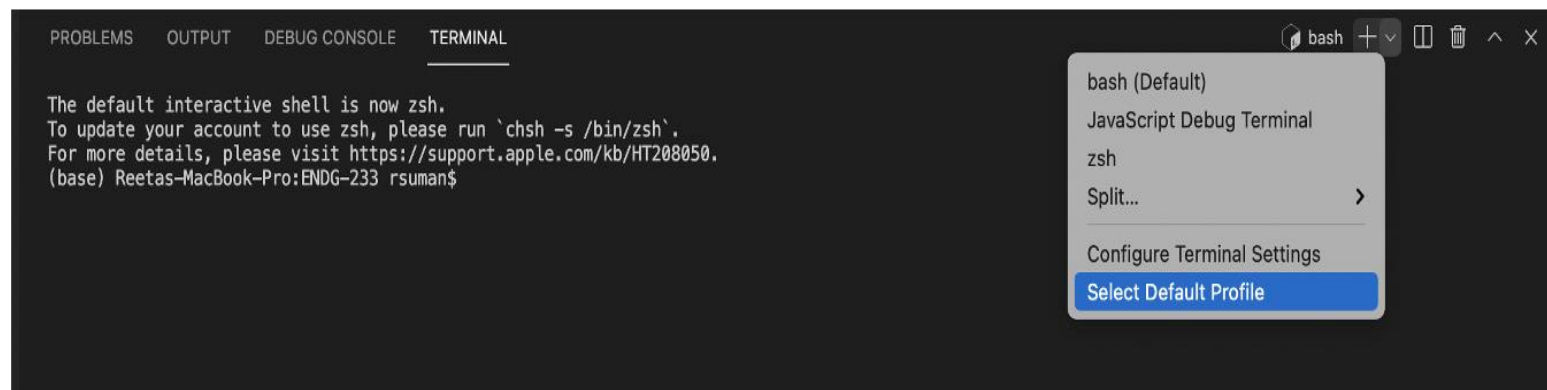
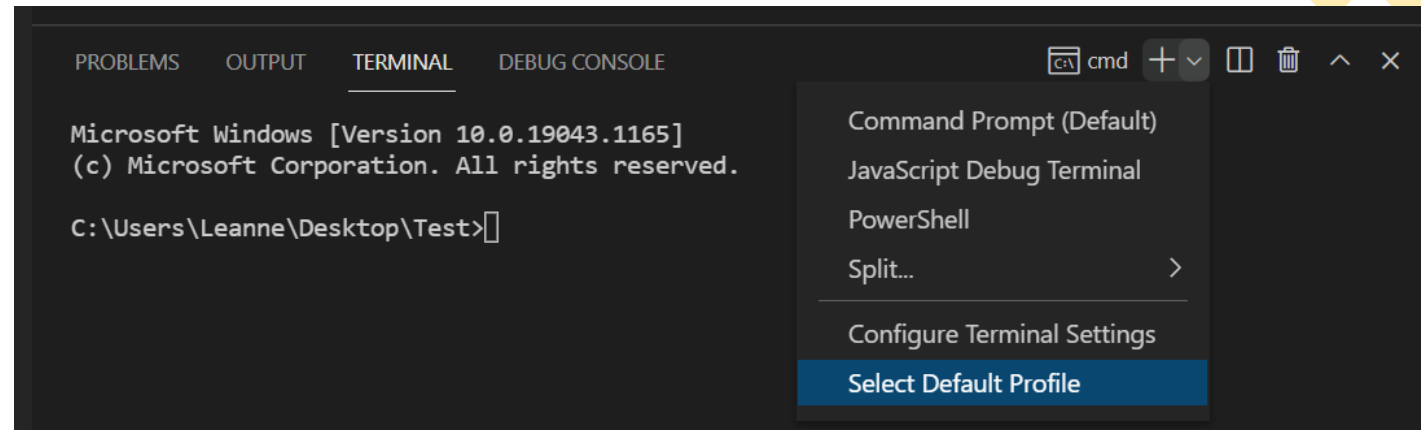
- Search **python** -> select one from the Microsoft publisher -> click **Install**





# Step 3: Set Default Terminal to Command Prompt

- Go to Terminal menu -> New Terminal
- [Windows]- Click on the down arrow next to the plus symbol -> click on Select Default Profile -> pick the Command Prompt option from the drop-down menu.
- [Mac/Linux] - Click on the down arrow next to the plus symbol -> click on Select Default Profile -> pick the Bash option from the drop-down menu.



## Step 3 (continued)

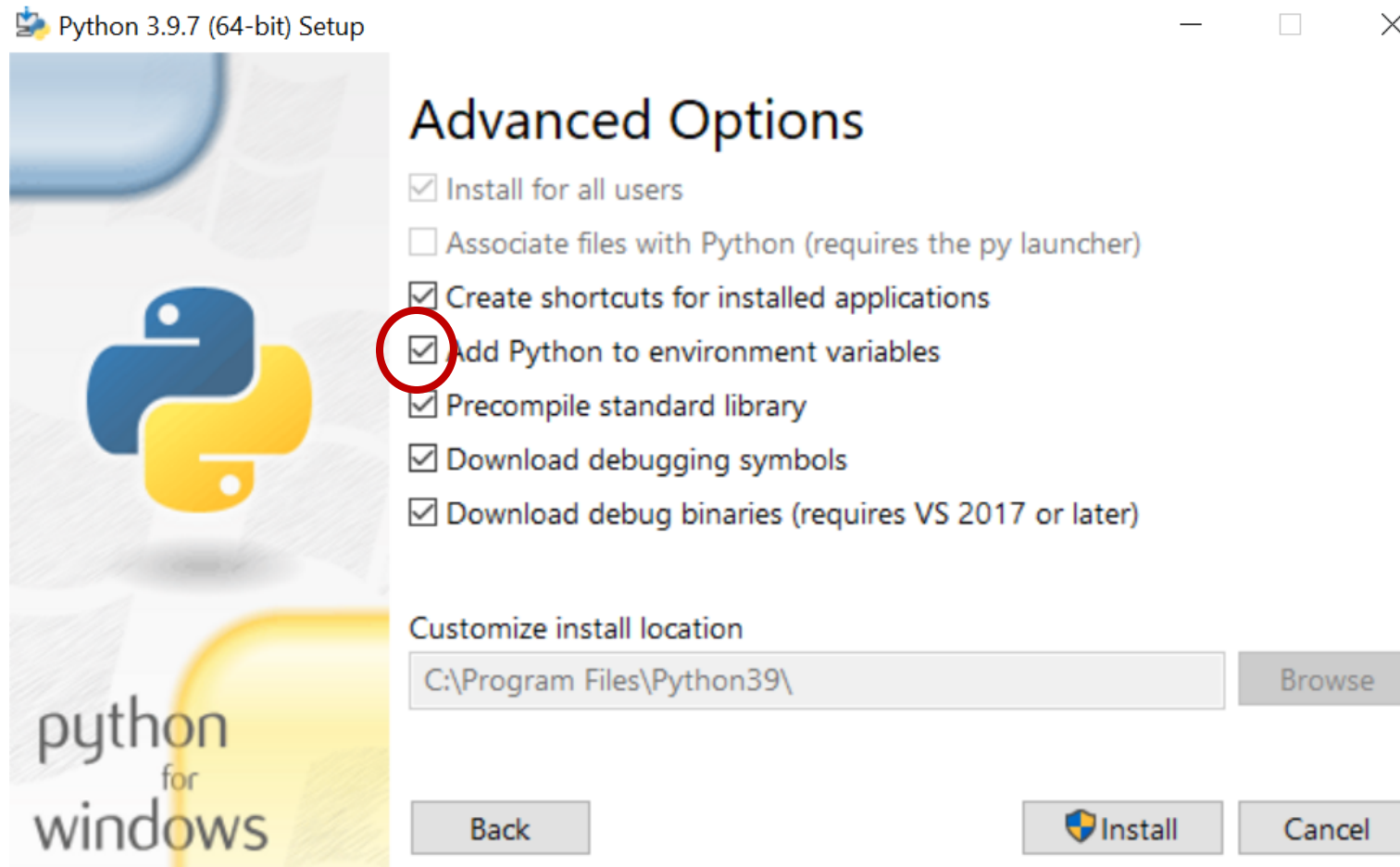
- Double-check when you open VS Code that the correct terminal is used
- If a different terminal pops up, click the trash can button to kill the terminal, then use the Terminal menu to open a new terminal

## Step 4: Install Python

- Go to the Microsoft store (will open if you type python in terminal)
- Or Go to <https://www.python.org/>
- You will need version 3.9 or higher
- Do not use default Python installation included with Mac computers

# NOTE

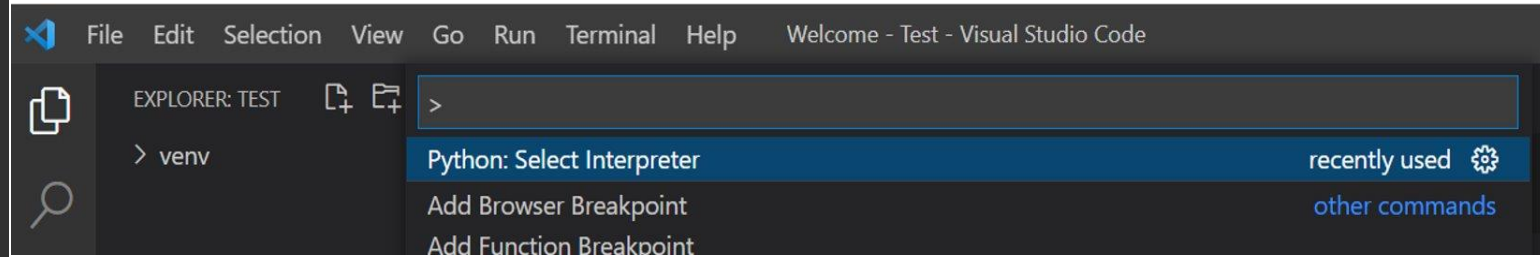
- Make sure to check the Add to environment variables (PATH) option checkbox in the advanced options window, **and restart your device after installation is complete**




# Step 4: Install Python (Linux and Chromebooks)

1. Launch terminal
  - For Chromebook users, it should be on your app list under linux apps
2. Update your Linux package installer
  - `sudo apt-get update`
  - `sudo apt-get install -y gnome-keyring`
  - `sudo apt-get install build-essential`
3. Install python and supporting packages that we will need
  - `sudo apt-get install python3`
  - `sudo apt-get install python3-pip`
  - `sudo apt-get install python3-venv`

# Step 5: Select Python Interpreter



- Access the **Command Palette** (Ctrl+Shift+p) or (Command+Shift+p) for Mac users or
- Click on settings  (**gear symbol**) and select Command Palette
- Search for **Python: Select Interpreter**
- Select the Python version you installed (or enter in the file path for the python.exe file)
- Interpreter should show up at the bottom-left corner of the VS Code program (purple/blue bar)
- Use command **python --version** or **py --version** to check the python version (**python3 --version** for Mac users)

## Step 6: Setup Working Directory

- Create a course project folder anywhere on your computer
  - Ex. **ENDG233**
- Create a test folder
  - Ex. **Week 1**
- Note: it is good practice to have a separate folder for each project



# Step 7: Create Virtual Environment

- Navigate to test folder (ex. Week 1) using **File** menu -> **Open Folder**
- Open a new terminal
- Create virtual environment folder named venv using terminal command:

**python -m venv venv** or

**py -m venv venv** or

**python3 -m venv venv** (Mac/Linux/Chromebook)

- Activate virtual environment:

**venv\Scripts\activate**

Or **source venv/bin/activate** for (Mac/Linux/Chromebook)

- Double-check: (venv) should be added to the front of the text line or  
(venv) for window users
- When finished, you can deactivate the virtual environment with  
command **deactivate**

```
(base) C:\Users\Leanne\Desktop\Test>venv\Scripts\activate.bat
```

```
(venv) (base) C:\Users\Leanne\Desktop\Test>
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

```
The default interactive shell is now zsh.  
To update your account to use zsh, please run `chsh -s /bin/zsh`.  
For more details, please visit https://support.apple.com/kb/HT208050.  
(base) Reetas-MacBook-Pro:ENDG-233 rsuman$ source venv/bin/activate  
(venv) (base) Reetas-MacBook-Pro:ENDG-233 rsuman$
```

## Step 7 (continued)

- You will need a virtual environment for each project (creation instruction is only done once for a project, activation is whenever we want to use the virtual environment)
- Your code files will **NOT** be saved inside the venv folder, they will be inside your project folder, the venv folder should not be tampered with directly by user
- Using a virtual environment protects the rest of your files from being impacted from changes to your code
- You can save all your projects in the same virtual environment, but in industry, typically a separate venv for each project is used so that issues in one project do not impact another