

**Software Engineering  
Bootcamp**

Hyperiondev

# Setting Up Your Dev Environment

# Lecture – Housekeeping

- ❑ The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. (FBV)
- ❑ No question is daft or silly - **ask them!**
- ❑ There are Q/A sessions midway and at the end of the session, should you wish to ask any follow-up questions.
- ❑ You can also submit questions here:  
<http://hyperiondev.com/sbc4-se-questions>
- ❑ For all non-academic questions, please submit a query:  
[www.hyperiondev.com/support](http://www.hyperiondev.com/support)
- ❑ Report a safeguarding incident:  
<http://hyperiondev.com/safeguardreporting>
- ❑ We would love your feedback on lectures:(FBV)  
<https://hyperiondev.wufoo.com/forms/zsgv4m40ui4i0g/>

# Github Repository – Lecture Examples/Slides

[https://github.com/HyperionDevBootcamps/C4\\_SE\\_lecture\\_examples](https://github.com/HyperionDevBootcamps/C4_SE_lecture_examples)

## Some useful links

Fedora: <https://docs.fedoraproject.org/en-US/quick-docs/dnf/>

Homebrew MacOS: <https://brew.sh>

Chocolatey Windows: <https://chocolatey.org/install>

Pip : <https://pip.pypa.io/en/stable/>

Available pip packages : <https://pypi.org/search>

# Objectives

1. Operating systems
  - a. Different operating systems
  - b. Which package manager to use with which OS
  - c. Installing a package with a package manager
2. Virtual Environments
  - a. What is a virtual environment
  - b. How to create a virtual environment
  - c. Using the virtual environment

# Different operating systems

- MacOS, Windows and Linux based
- Each operating system has its own package managers
- Package managers help us manage dependencies
- It also helps with installing, upgrading and configuring software
- Package managers prevent us from installing everything globally and help keep everything in one place

# Package managers

- MacOS: Homebrew
- Windows: Chocolatey
- Linux: based on operating system, apt-get and dnf

# Virtual Environment

- All of the programs we create will use different packages
- This means we do not want to install packages globally
- This is where the virtual environment comes in

# Virtual Environment

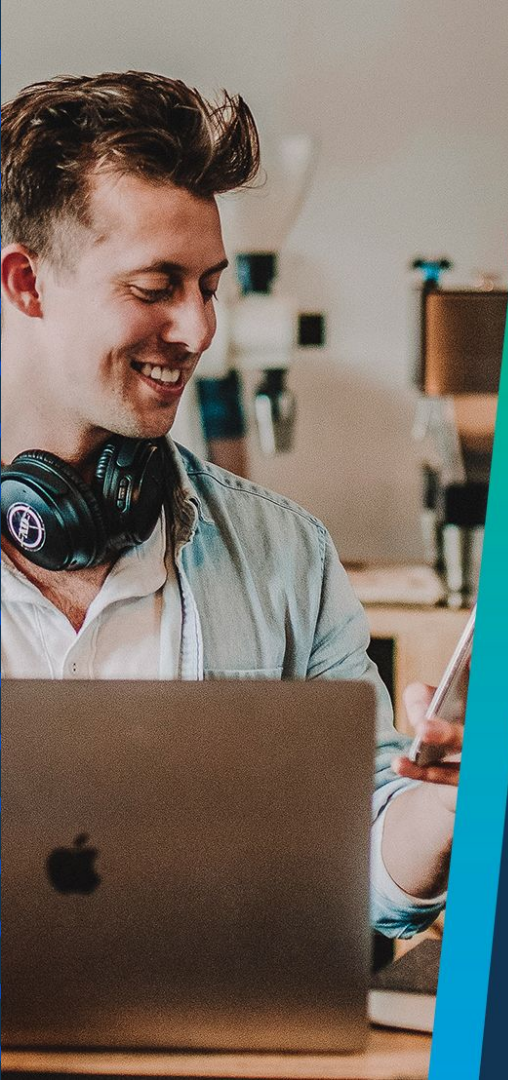
- To create a virtual environment we use the command `python -m venv venv_name`
- To activate the venv we have to navigate to the activate file `venv_name/Scripts/activate`
- To deactivate the venv we simply type `deactivate`



Hyperiondev

# Q & A Section

**Please use this time to ask any questions relating to the topic explained, should you have any**



Hyperiondev

# Thank you for joining us

**Take regular breaks.  
Stay hydrated.  
Avoid prolonged screen time.  
Remember to have fun :)**