Installing Essential Packages

AiCE Software Engineering Foundations — Setting Up Your Developer Environment

With Ubuntu installed and running through WSL, this lesson covers pinning it to your desktop for quick access, and installing the four packages every lesson in this course assumes you already have: vim, git, python3, and python3-pip.

1. Pinning Ubuntu to Your Desktop

Rather than searching for Ubuntu in the Windows search bar every time you want to open it, pin it to your desktop for one-click access:

1. Find Ubuntu in your Windows search results, right-click it, and choose Open file location.

2. In the folder that opens, right-click the Ubuntu shortcut again, choose Show more options, hover over Send to, and select Desktop (create shortcut).

3. Ubuntu now appears as an icon on your desktop, ready to open anytime.

2. Updating Ubuntu Before Installing Anything

Open Ubuntu from your new desktop shortcut. Before installing any package, always update Ubuntu's list of available software first:

```
sudo apt update
```

You'll be prompted for your Ubuntu password — type it and press Enter. As before, nothing will appear on screen as you type; this is expected. Wait for the update to finish, and make sure your laptop stays connected to the internet throughout.

> **Update Every Time You Reopen the Terminal**
> *Whenever you close Ubuntu and reopen it in a new session, run sudo apt update again before installing anything new. It only takes a moment, and it ensures you're always installing the latest available version of a package.*

3. Installing vim

To install any package on Ubuntu, the command follows the same pattern: sudo apt install, followed by the package's name. You don't need to know exactly what each package does yet — you'll get to know all of them well as this course progresses. Start with vim, a terminal-based text editor:

```
sudo apt install vim
```

Press Enter, then type Y and press Enter again to confirm. Leave it to download and install, keeping your internet connection active.

Once it finishes, confirm the installation succeeded by checking its version. To check any package's version, type its name, a space, two dashes, then “version”:

```
vim --version
```

If this runs without an error and shows a version number, vim installed successfully. Clear your screen before continuing to the next package:

```
clear
```

4. Installing git

git is the version control tool you'll use constantly throughout this course. Install it the same way:

```
sudo apt install git
```

Once it finishes, confirm with:

```
git --version
```

A version number confirms success.

5. Installing python3

If you've closed and reopened your terminal since the last package installation, run sudo apt update once more before continuing — it's a habit worth building now. Then install Python 3, the programming language this entire course is built around:

```
sudo apt update
sudo apt install python3
```

Confirm with:

```
python3 --version
```

6. Installing python3-pip

pip is Python's package installer — it lets you install additional Python tools and libraries later in this course. Install it with:

```
sudo apt install python3-pip
```

Type Y and press Enter to confirm when prompted. Once finished, confirm with:

```
pip3 --version
```

A version number confirms pip3 is ready.

7. Full Package Reference

Package	Install Command	Verify Command
vim	sudo apt install vim	vim --version
git	sudo apt install git	git --version
python3	sudo apt install python3	python3 --version
python3-pip	sudo apt install python3-pip	pip3 --version

Key Takeaways

Every package installs the same way — sudo apt install package-name — and every install is worth confirming immediately with package-name --version. Updating with sudo apt update before installing anything, especially after reopening the terminal in a new session, is a habit that prevents outdated or failed installs. With vim, git, python3, and pip3 installed and verified, your terminal now has everything this course's lessons will assume is already there. The next lesson covers making the terminal itself look and feel the way you want it to.