

Command Line Interface (CLI)

Your Developer Superpower

What is CLI?

A **text-based interface** where you type commands to interact with your computer

- **CLI:** Like texting - precise, quick, can be saved and repeated
- **GUI:** Like phone calls - immediate but harder to remember exact steps

Why CLI?

GUI (Graphical User Interface) = Restaurant

- Beautiful menus with pictures
- Point and click to order
- Limited to what's on the menu
- Slower for complex orders

CLI (Command Line Interface) = Your Own Kitchen

- Direct access to all ingredients
- Create exactly what you want
- Combine tools in unlimited ways
- Fast once you know the recipes

CLI vs GUI: Created 12 folders

GUI Approach (20+ clicks):

1. Right-click → New Folder → Type name
2. Double-click to enter
3. Right-click → New Folder → Type name
4. Repeat 10 times...

CLI Approach (1 line):

```
mkdir -p project/{src,tests,docs,assets}/{js,css,images}
```

Package Managers: Your Software Assistants

Platform	Tool	Purpose
macOS	brew	Install development tools
Windows	choco	Install development tools
Windows	WSL2	Linux CLI
Linux (Ubuntu)	apt	Install development tools

For Windows, students can use `choco`, but for development purposes, they need to install `WSL2` and use a Linux system on Windows.

Programming	Tool	Purpose
Node.js	npm/npx	Manage JavaScript packages
Python	pip/uv	Manage Python packages

Many applications are deployed as Node.js or Python packages.

Homebrew (macOS)

- Installation
- Make sure the bin directory is in the path.
 - Apple Silicon: /opt/homebrew/bin/
 - Intel: /usr/local/bin

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


Common Commands:

```
# Install software  
brew install node python git
```

```
# Update everything  
brew update && brew upgrade
```

```
# Search for packages  
brew search visual-studio-code
```

```
# Install GUI applications  
brew install --cask visual-studio-code
```

Other Mac CLI tools

Tool	Admin Required?	Best Use Case	Learning Curve
Homebrew	No	General development	Easy
MacPorts	Yes	Scientific/Research	Medium
Fink	Yes	Debian familiarity	Medium
mas	No	App Store apps	Easy
Conda	No	Data science	Medium

In most cases, Homebrew is enough CLI tools for Mac, but Conda may be needed for data science courses.

Chocolatey (Windows)

Installation (PowerShell as Admin):

- Chocolatey adds applications to PATH during installation.
- However, sometimes you need to "refresh (refreshenv)" your environment to see the changes.

```
Set-ExecutionPolicy Bypass -Scope Process -Force  
iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))  
  
refreshenv # refresh
```

Common Commands:

```
# Install software
choco install nodejs python git
# Update everything
choco upgrade all
# Search for packages
choco search vscode
# Install with GUI
choco install vscode googlechrome firefox
```

- Other Windows CLI tools

Tool	Admin Required?	Best Use Case	Learning Curve
Chocolatey	Yes	General software	Medium
Winget	Usually	Official packages	Easy
Scoop	No	Developer tools	Medium
Ninite	Yes	Bulk installs	Very Easy
Conda	No	Data science	Medium

CLI Automation

- The Real Power CLI comes from automation
 - We can automate build, execution, document generation, and tests
- Real-World Analogy: Recipe vs. Cooking Show

Manual Work (Cooking Show)

1. Watch step-by-step
2. Repeat every time
3. Easy to make mistakes
4. Takes the same time each time

Automation (Recipe Script)

```
#!/bin/bash
# setup-project.sh
mkdir $1
cd $1
npm init -y
npm install express
echo "console.log('Hello World!');" > app.js
echo "Project $1 created successfully!"
```


Automation Examples

Daily Developer Tasks:

- setup, deploy, clean, backup, and check the system in one script

```
# Setup new project
./setup-project.sh awesome-app
# Deploy to production
git add . && git commit -m "Update" && git push origin main
# Clean up old files
find . -name "*.log" -mtime +7 -delete
# Backup important files
tar -czf backup-$(date +%Y%m%d).tar.gz src/
# Check system status
ps aux | grep node && df -h && free -m
```

CLI Benefits

- **Speed**
 - Type faster than click
 - One command = many GUI actions
- **Automation**
 - Save and repeat commands
 - Batch operations
 - Scheduled tasks

- **Precision**

- Exact control over parameters
- No misclicks

- **Scalability**

- Manage hundreds of files
- Process large datasets

- **Remote Work**
 - SSH into servers
 - Work on headless systems
- **Version Control**
 - Save command history
 - Share scripts with teams
- **Professional Development**
 - Required for DevOps
 - Essential for system administration