### **I asked chatgpt o1:**

**Prompt**: I need to build a self paced course to teach linux basics and cli, to univeristy students (age 21 and up) who only used Windows GUI. Suggest chapters and exercises. the total student learning time should be 10 hours. Users have access to ubuntu cli using Windows WSL2.

remove mentions to "man" since they will look in the internet. Add explanations of file permissions Add explanation of "what is a file system". the total duration may be expanded Add chapter on using text processing tools such as awk, sed, tr ...

----------

**Prompt**: 'create a transcript for chapter 1'

And it responded with nice text.

-----------

**Prompt**: " let's say the transcript is ready. Which tool should I use to generate a video with an avatar of me as a talking head"

Here’s a **comparison of D-ID, HeyGen, Synthesia, and DeepBrain AI** based on key factors:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **D-ID** | **HeyGen** | **Synthesia** | **DeepBrain AI** |
| **Realistic AI Avatar** | ✅ Yes | ✅ Yes | ✅ Yes | ✅ Yes |
| **Upload Your Own Photo** | ✅ Yes | ✅ Yes | ❌ No (only pre-built avatars) | ✅ Yes |
| **Voice Cloning** | ✅ Yes (premium) | ✅ Yes (premium) | ❌ No | ✅ Yes |
| **Text-to-Speech** | ✅ Yes | ✅ Yes | ✅ Yes | ✅ Yes |
| **Multilingual Support** | ✅ Yes | ✅ Yes (50+ languages) | ✅ Yes (120+ languages) | ✅ Yes (80+ languages) |
| **Customizable Avatars** | ✅ Yes | ✅ Yes | ❌ No | ✅ Yes |
| **Editing Options** | 🎥 Basic (backgrounds, subtitles) | 🎥 Advanced (scenes, transitions) | 🎥 Moderate (basic editing) | 🎥 Basic |
| **Lip Sync Quality** | ⭐⭐⭐⭐ | ⭐⭐⭐⭐ | ⭐⭐⭐⭐⭐ | ⭐⭐⭐⭐ |
| **AI Voice Quality** | ⭐⭐⭐ | ⭐⭐⭐⭐ | ⭐⭐⭐⭐⭐ | ⭐⭐⭐⭐ |
| **Ease of Use** | ⭐⭐⭐⭐ | ⭐⭐⭐⭐ | ⭐⭐⭐⭐⭐ | ⭐⭐⭐ |
| **Export Formats** | 🎥 MP4 | 🎥 MP4, GIF | 🎥 MP4 | 🎥 MP4 |
| **Free Trial** | ✅ Yes (limited) | ✅ Yes (limited) | ❌ No | ✅ Yes |
| **Pricing** | 💰 Starts ~$5/mo | 💰 Starts ~$30/mo | 💰 Starts ~$22/mo | 💰 Starts ~$30/mo |

### **Best For Different Needs**

* **Best for Personal Talking Head (Use Your Own Photo):** **D-ID, HeyGen, DeepBrain AI**
* **Best for High-Quality AI Voice & Lip Sync:** **Synthesia**
* **Best for Video Editing Features:** **HeyGen**
* **Best for Budget Users:** **D-ID (cheapest option)**
* **Best for Multilingual Support:** **Synthesia** (120+ languages)

### **Final Recommendation**

* **If you want to use your own photo & get a talking avatar** → **D-ID or HeyGen**
* **If you need high-quality avatars and voices for professional courses** → **Synthesia**
* **If you want more control over editing (transitions, scenes)** → **HeyGen**
* **If you want a simple, easy tool with voice cloning** → **DeepBrain AI**

Would you like help with choosing one based on your **specific needs and budget**?



### **Expanded Self-Paced Linux Basics & CLI Course**

**Target Audience:** University students (21+) who have only used Windows GUI

**Environment:** Ubuntu CLI on Windows WSL2

**Total Learning Time:** ~12–14 hours

## **Course Structure**

Each chapter includes core topics and hands-on exercises.

Box[ [why more people dont use linux](https://www.youtube.com/watch?v=_H4E6eTRHAo&ab_channel=ThePrimeTime)]

### **Chapter 1: Introduction to Linux & WSL2 (~1 Hour)**

* What is Linux? History and philosophy
* Linux vs. Windows: Key differences
* Overview of Ubuntu and WSL2
* Installing and configuring WSL2 with Ubuntu
* **What is a File System?**
  + How files are stored
  + Common Linux file systems (ext4, XFS, etc.)
  + Differences from Windows (NTFS vs. ext4)

**Exercises:**

1. Verify WSL2 installation and list available distributions (wsl --list --verbose)
2. Launch Ubuntu in WSL2 and check its version (lsb\_release -a)
3. Explore the root file system (ls /) and compare it to Windows' C:\

### **Chapter 2: Navigating the Linux File System (~1 Hour)**

* Understanding the Linux directory structure (/, /home, /etc, /var, etc.)
* Navigating directories (cd, pwd, ls)
* Absolute vs. relative paths
* Hidden files and directories (ls -a)
* File and directory operations (mkdir, touch, rm, rmdir)

**Exercises:**

1. Navigate to different directories and list their contents
2. Create a new directory (mkdir myfolder) and move into it
3. Create an empty file (touch myfile.txt) and remove it (rm myfile.txt)

### **Chapter 3: Working with Files and Directories (~1 Hour)**

* Viewing file contents (cat, less, head, tail)
* Copying, moving, and renaming files (cp, mv)
* Finding files (find, locate)
* Understanding file types (file command)

**Exercises:**

1. Create a file, write text into it, and display its contents (echo "Hello Linux" > hello.txt && cat hello.txt)
2. Copy a file and rename it (cp hello.txt hello\_copy.txt)
3. Use find to locate all .txt files in the home directory

### **Chapter 4: File Permissions and Ownership (~1.5 Hours)**

* **Understanding File Permissions**
  + The ls -l output (rwxr-xr--)
  + User, group, and others
  + Permission types: Read (r), Write (w), Execute (x)
* **Modifying Permissions**
  + Changing permissions with chmod (symbolic & numeric modes)
  + Changing ownership with chown
  + Changing group ownership (chgrp)
* Special permissions (setuid, setgid, sticky bit)

**Exercises:**

1. Check the permissions of a file (ls -l)
2. Change a file to be readable and writable only by the owner (chmod 600 myfile.txt)
3. Give execute permissions to a script (chmod +x script.sh)
4. Change the owner of a file (sudo chown user:group myfile.txt)

### **Chapter 5: Basic Linux Commands (~1 Hour)**

* System information commands (whoami, hostname, date, uptime, uname -a)
* Process management (ps, kill, jobs, bg, fg)
* Keyboard shortcuts (Ctrl+C, Ctrl+Z, tab completion)

**Exercises:**

1. Find out system uptime (uptime)
2. List running processes (ps aux | grep bash)
3. Start a process in the background and bring it back (sleep 100 &, fg)

### **Chapter 6: Text Processing & Editing (~1.5 Hours)**

* Viewing text (cat, less, tail, head)
* Searching within files (grep, egrep)
* Counting words, lines, and characters (wc)
* Sorting and filtering (sort, uniq)
* Editing with nano

**Exercises:**

1. Search for a word in a text file (grep "Linux" myfile.txt)
2. Count the number of lines in a file (wc -l myfile.txt)
3. Sort a list of words and remove duplicates (sort words.txt | uniq)

### **Chapter 7: Advanced Text Processing Tools (~1.5 Hours)**

* **sed (Stream Editor)**
  + Find and replace text (sed 's/old/new/g' file.txt)
  + Delete lines matching a pattern (sed '/pattern/d' file.txt)
* **awk (Pattern Scanning and Processing Language)**
  + Print specific columns (awk '{print $2}' file.txt)
  + Perform calculations (awk '{sum+=$3} END {print sum}' data.txt)
* **tr (Translate and Delete Characters)**
  + Convert lowercase to uppercase (tr 'a-z' 'A-Z' < file.txt)

**Exercises:**

1. Replace "foo" with "bar" in a file (sed 's/foo/bar/g' myfile.txt')
2. Extract column 2 from a CSV file (awk -F, '{print $2}' data.csv)
3. Convert lowercase letters to uppercase in a file (cat input.txt | tr 'a-z' 'A-Z')

### **Chapter 8: Users, Groups & Permissions (~1 Hour)**

* User accounts (whoami, id, passwd)
* Listing users and groups (cat /etc/passwd, cat /etc/group)
* Managing permissions (recap)

**Exercises:**

1. Change your password (passwd)
2. Create a new group and add a user (sudo usermod -aG mygroup user)

### **Chapter 9: Package Management (~1 Hour)**

* Installing, updating, and removing software (apt, dpkg)
* Searching for software (apt-cache search)
* Managing services (systemctl)

**Exercises:**

1. Install htop (sudo apt install htop)
2. Remove a package (sudo apt remove htop)

### **Chapter 10: Networking & Remote Access (~1 Hour)**

* Checking network settings (ip a, ifconfig, ping)
* Downloading files (wget, curl)
* SSH basics (ssh user@host)

**Exercises:**

1. Check your IP address (ip a)
2. Connect to a remote server via SSH

### **Chapter 11: Shell Scripting (~2 Hours)**

* Writing simple scripts (#!/bin/bash)
* Variables and user input (read, echo)
* Conditionals (if, else)
* Loops (for, while)
* Running a script (chmod +x script.sh && ./script.sh)

**Exercises:**

1. Write a script that prints "Hello, Linux!"
2. Create a script that takes a user’s name as input and greets them
3. Write a script to list all .txt files in a directory

### **Final Project (~1.5 Hours)**

* Students will complete a practical challenge:
  + Organizing files into directories based on extensions
  + Writing a script to automate a repetitive task

### **Additional Notes**

* Provide students with a Linux command cheat sheet
* Encourage independent research when stuck
* Suggest online resources for deeper learning

---------------------------------- ------------------ ----------------- ------------------

Topics to add:

Using rsync