Learning Unix in Google Colab - Video Tutorial Script

Hello everyone! Welcome to this exciting tutorial on learning basic Unix commands using Google Colab. My name is [Your Name], and today we're going to explore how you can use Unix commands in a super easy and fun way right here within your Jupyter Notebooks. So, grab your coffee, relax, and let's dive in!

# Section 1: Setting the Scene

Alright, let’s start by talking about why learning Unix is so important. Unix is a foundational operating system used in almost all bioinformatics and computational biology work. If you’re handling large datasets, organizing files, or running complex pipelines, knowing a few key Unix commands will save you tons of time. And guess what? You don’t need a special system to start learning Unix—you can do it right in Google Colab! That’s what this notebook is all about!

# Section 2: Checking Your Current Directory

Now, every time you're working with files, it’s crucial to know where exactly you are in your computer’s directory. Think of it as checking your location on a map. In Unix, we can check where we are by using the `pwd` command. Simple, right? `pwd` literally stands for ‘print working directory,’ and it will tell us the folder we're currently in. Let’s give that a try!

\*Pause for demonstration, run the command.\*

See? That’s our current location in the system!

# Section 3: Listing the Files in Your Directory

Once we know where we are, the next step is to see what’s around us. To do this, we use the `ls` command, which lists all the files and directories in the current folder. It’s like peeking inside your backpack to see what tools you have before starting your work. Let’s take a look.

\*Pause for demonstration, run the `ls` command.\*

Here’s everything in our directory right now. You’ll probably see some default files that Google Colab creates, and if you’ve uploaded anything, it’ll show up here too!

# Section 4: Creating a New Directory

Great! Now, let’s say you want to get organized and create a new folder to keep all your important files. That’s super easy with the `mkdir` command, which stands for ‘make directory.’ Imagine you’re building a new folder, a virtual drawer, to store your files. Let's create one now!

\*Pause for demonstration, run the `mkdir` command.\*

And just like that, we’ve made a new directory! You can create as many as you need to stay organized.

# Section 5: Moving Files Around

Next, let’s say you want to move files from one folder to another, or even rename them. The `mv` command is your go-to tool for this. Think of it as packing up a file and moving it to another room—or giving it a new label. It’s simple, powerful, and helps keep things neat.

\*Pause for demonstration, run the `mv` command.\*

Perfect! Our file has been moved, or if we chose, renamed. This is super handy when organizing large datasets.

# Section 6: Cutting Out Parts of a File with the `cut` Command

Now, let's talk about something really useful for bioinformatics—extracting parts of a file. You might have data in columns or separated by certain characters. Unix lets you extract specific sections using the `cut` command. For example, if you have a file full of data and you only need the first column or specific fields, `cut` can do that for you. Let me show you how it works!

\*Pause for demonstration, show the `cut` command.\*

In this example, we’re using `cut` to grab everything before a specific character—pretty cool, right? This is super useful when you’re cleaning up data or preparing it for analysis.

# Wrap-Up

Congratulations! You’ve just taken your first steps into the world of Unix commands, and you did it all in Google Colab! These basic commands are incredibly powerful, and you’ll find yourself using them constantly as you dive deeper into computational biology, bioinformatics, or any data-heavy research. Just remember, learning Unix is like unlocking a superpower for handling your data efficiently and effectively. So keep practicing, keep exploring, and soon you’ll be a Unix master!

Thank you so much for watching this tutorial! If you found this helpful, don’t forget to share it with your fellow classmates and colleagues. Until next time—happy coding, and happy exploring!