CONESTOGA
Connect Life and Learning

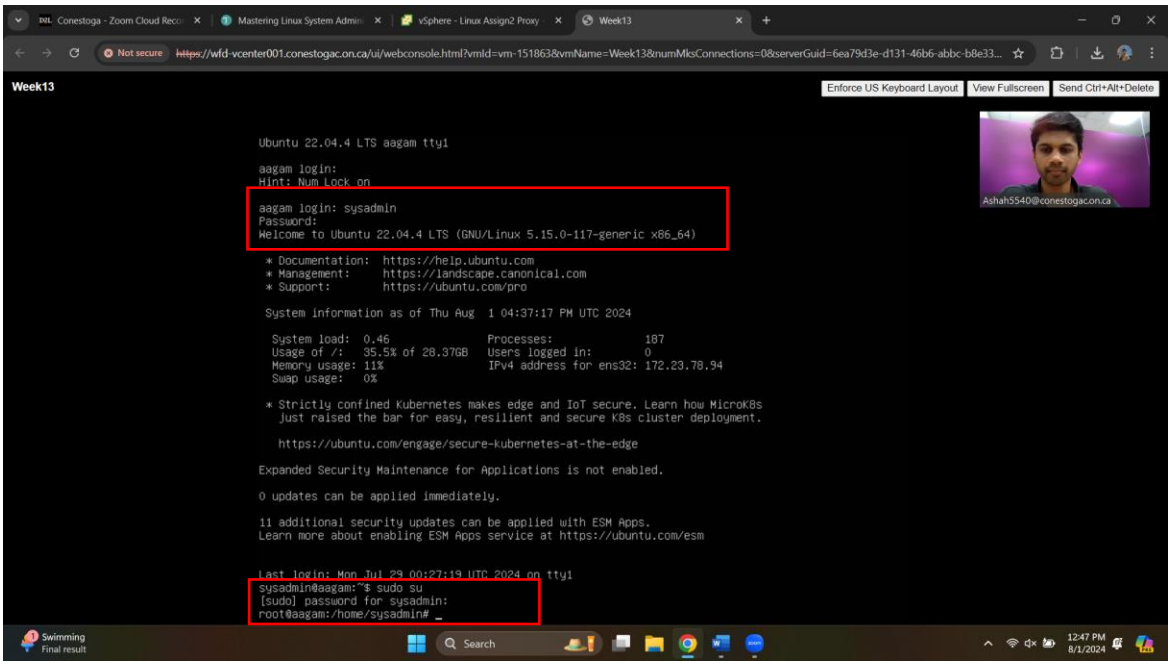
Student Name:	Aagam Sanjay Shah
Deliverable:	In-Class Tasks Week 13 Assignment
Course Name:	NTWK8141-24S-Sec3-Linux Server

Date Assigned:	31/07/2024
Date Due:	01/08/2024
Rules:	• Individual. • Cheating is not allowed. • Plagiarism counts as cheating! • That FAILURE to submit work in the course can result in a grade of 'F' or 'I' for failure to complete the course!

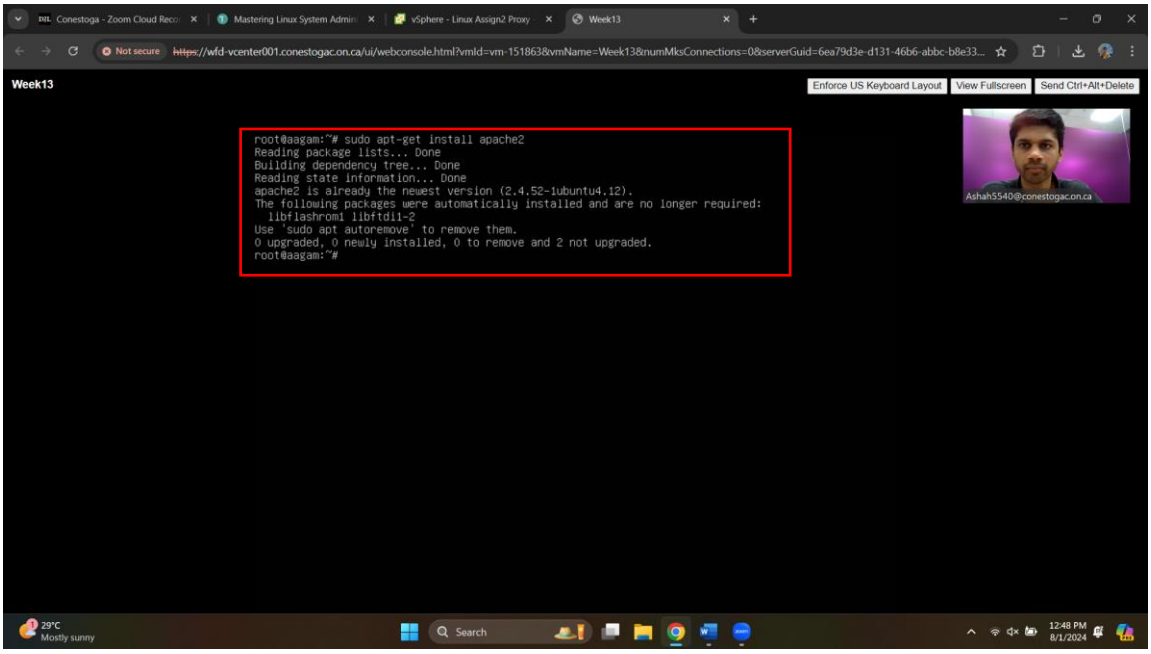
Week 13 Slide 21

Complete the Real World Scenario: Testing Your Web Server from the Command Line in Ch 20 TESTING YOUR WEB SERVER FROM THE COMMAND LINE

1. Log in as root or acquire root privileges by using su or sudo with each of the following commands.

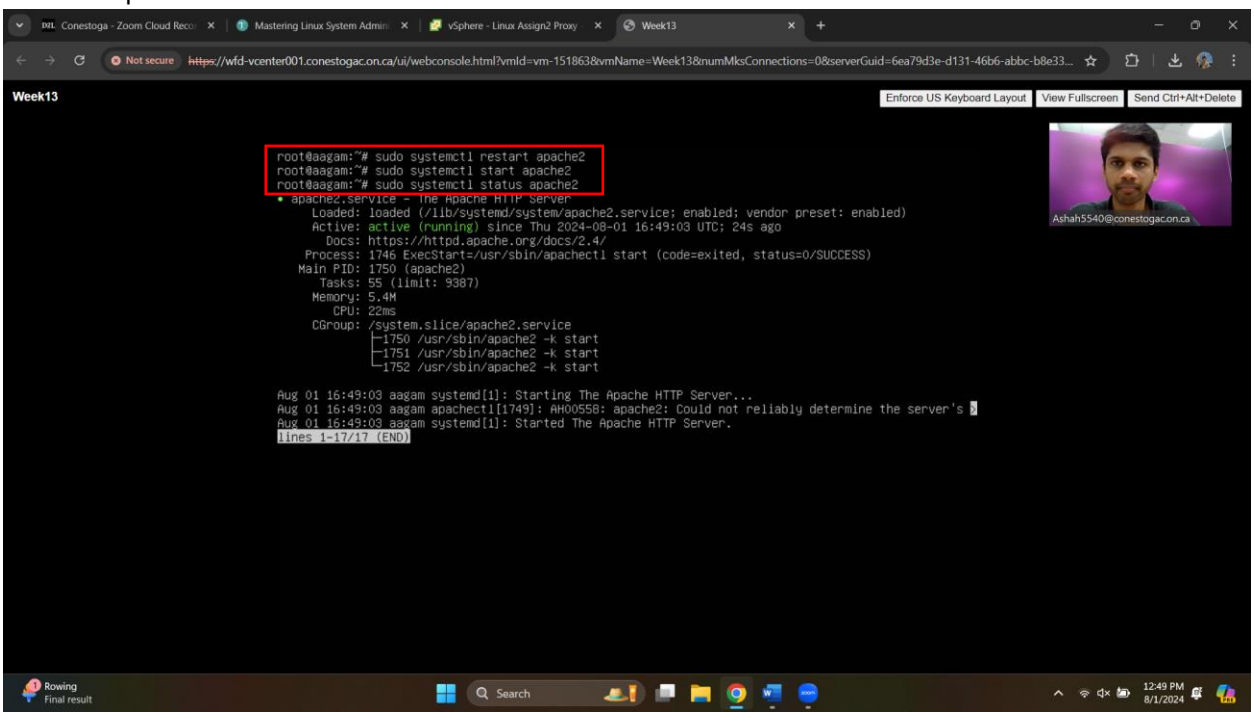


2. Ensure that you have the Apache web server installed. Open a command prompt and then type **sudo apt-get install apache2** to install the Apache web server package on your Ubuntu server, or **sudo dnf install httpd** to install Apache on your Red Hat server.

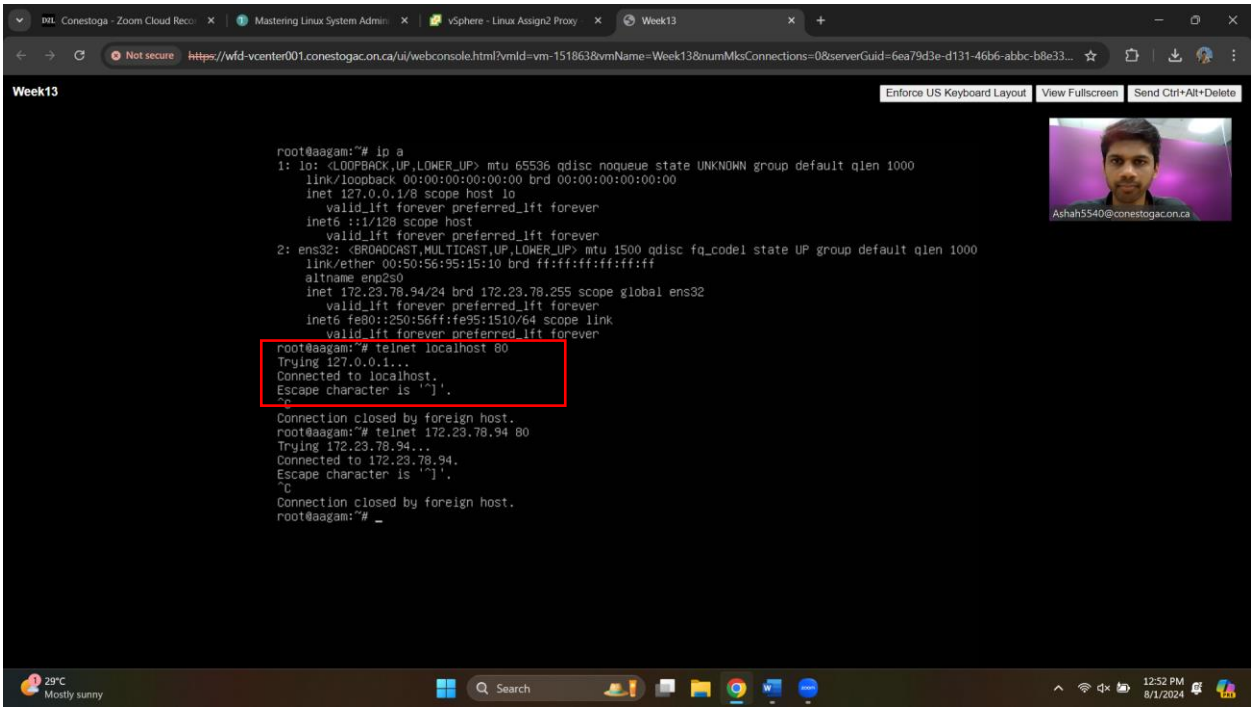


3. For Red Hat servers, start the Apache web server by typing **sudo systemctl start httpd**.

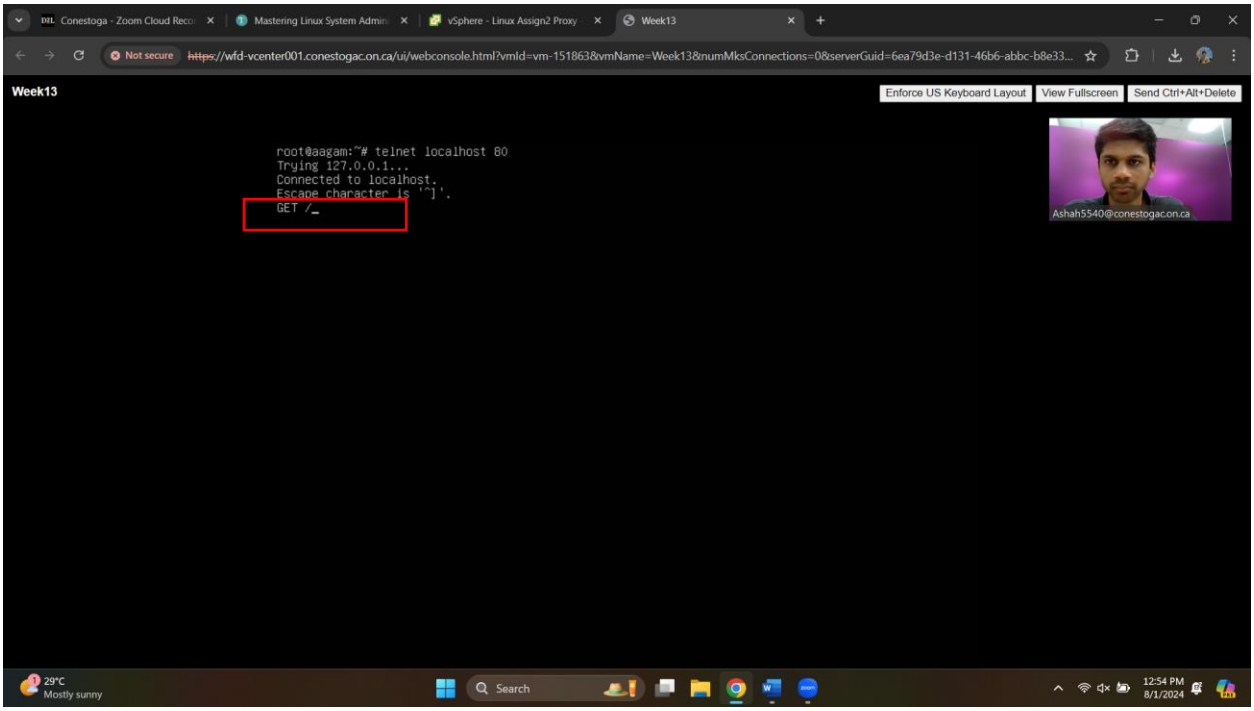
Ubuntu: Apache2 status



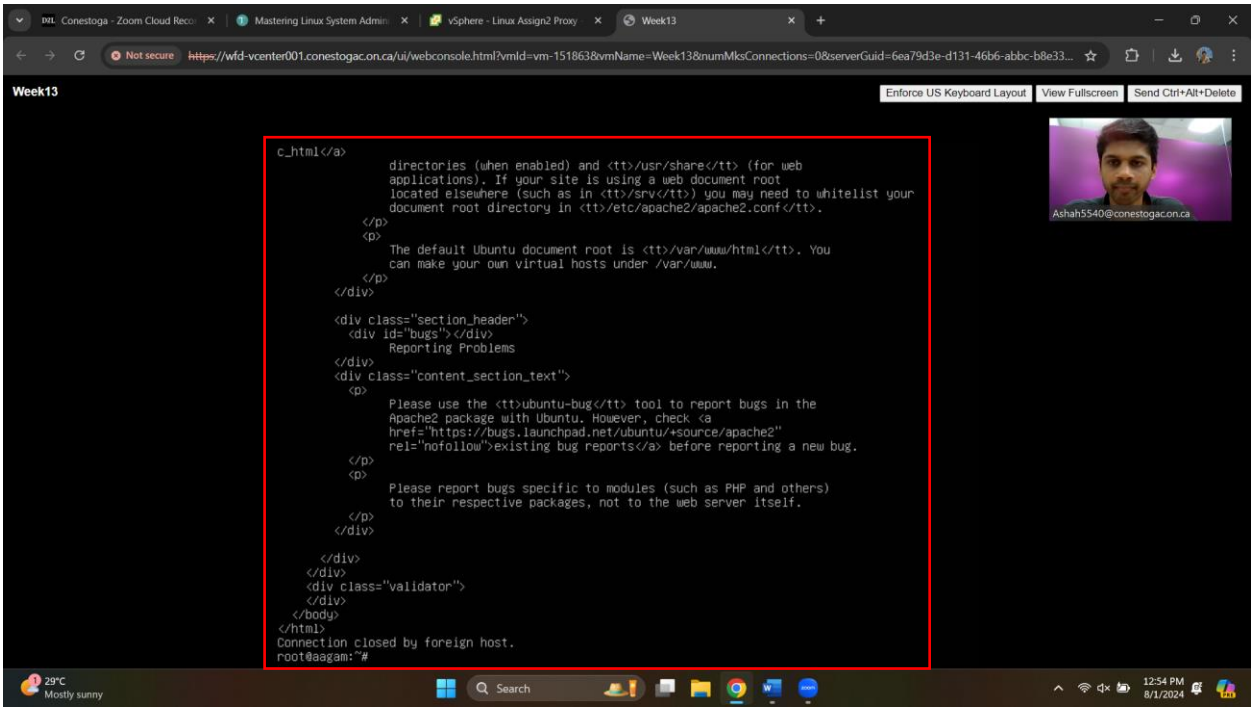
4. Test the Apache web server by typing **telnet localhost 80** at a command line (you may have to also install the Telnet package on your system). If your system has the Apache web server running, you should be greeted by the Telnet banner, but nothing from the Apache web server as it's waiting for your request.



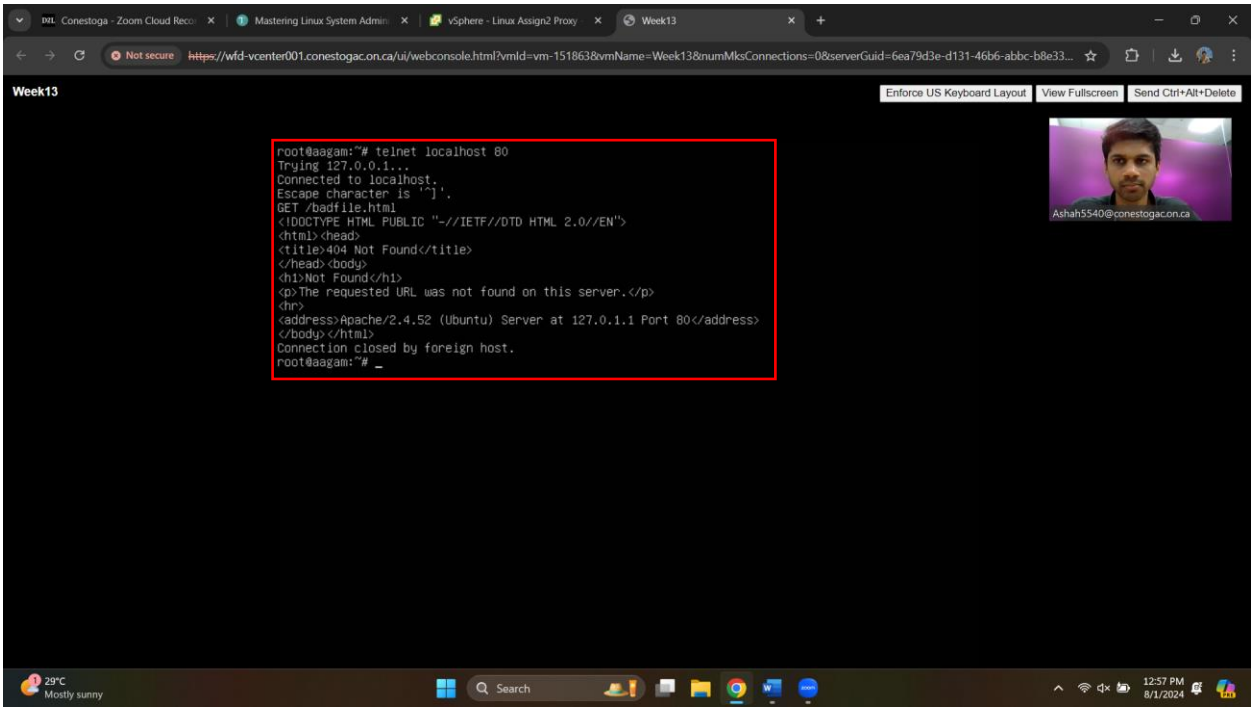
5. Request the default web page from the server by typing **GET /**, and then hit the Enter key to submit the request.



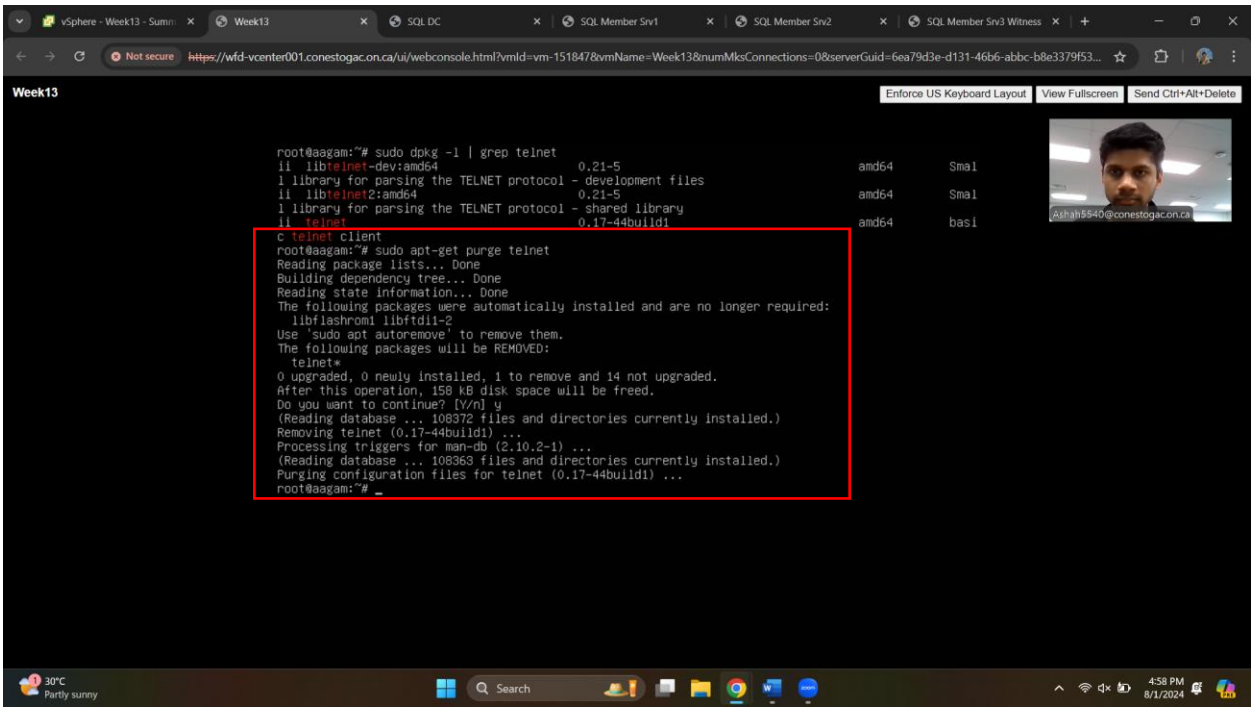
6. The Apache web server should return the HTML code contained in the index.html file from the /var/www/html directory.



7. Now test attempting to retrieve an invalid web page. Connect to the Apache web server by typing **telnet localhost 80** at a command line. At the prompt, type **GET /badfile.html** and then hit the Enter key. You should see some HTML code for a generic error message web page that the browser would display to inform you that it could not find the file on the system.



8. Remove the Telnet package from your server, as it can be a security risk if left enabled.

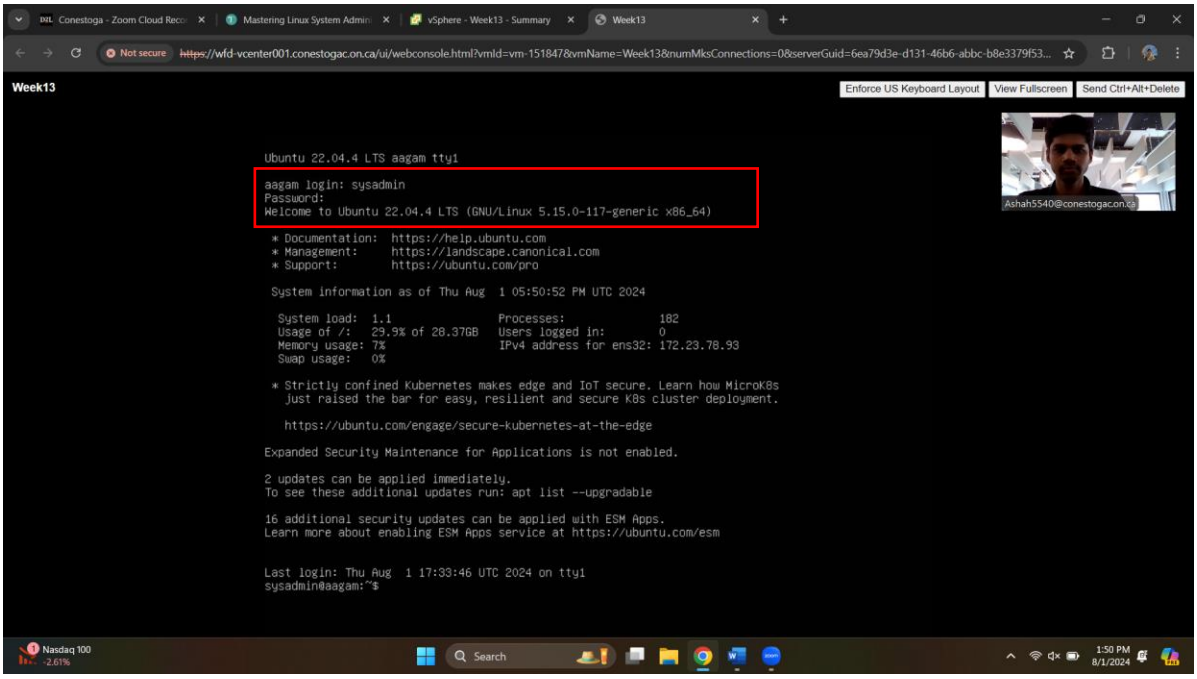


Week 13 Slide 34

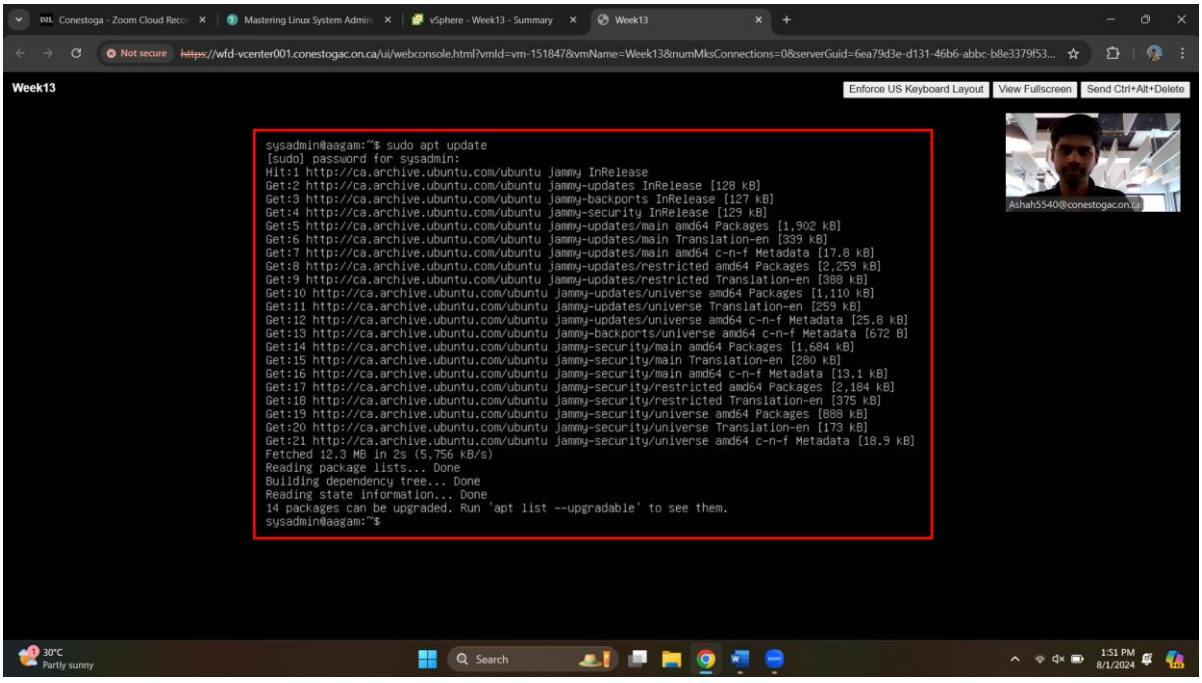
Complete the Real World Scenario: Installing the MariaDB Server on Your Ubuntu System in Ch 21

INSTALLING THE MARIADB SERVER ON YOUR UBUNTU SYSTEM

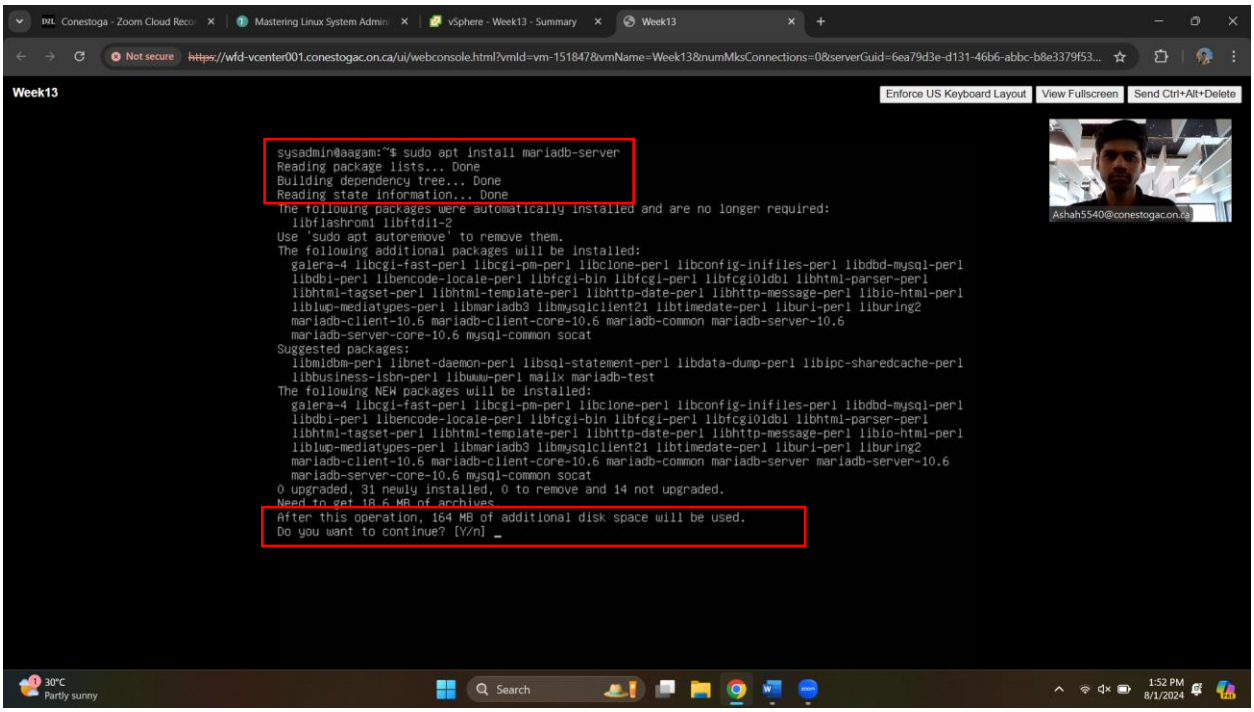
1. Log into your Ubuntu Linux system, using the sysadmin account and the password you created for it.



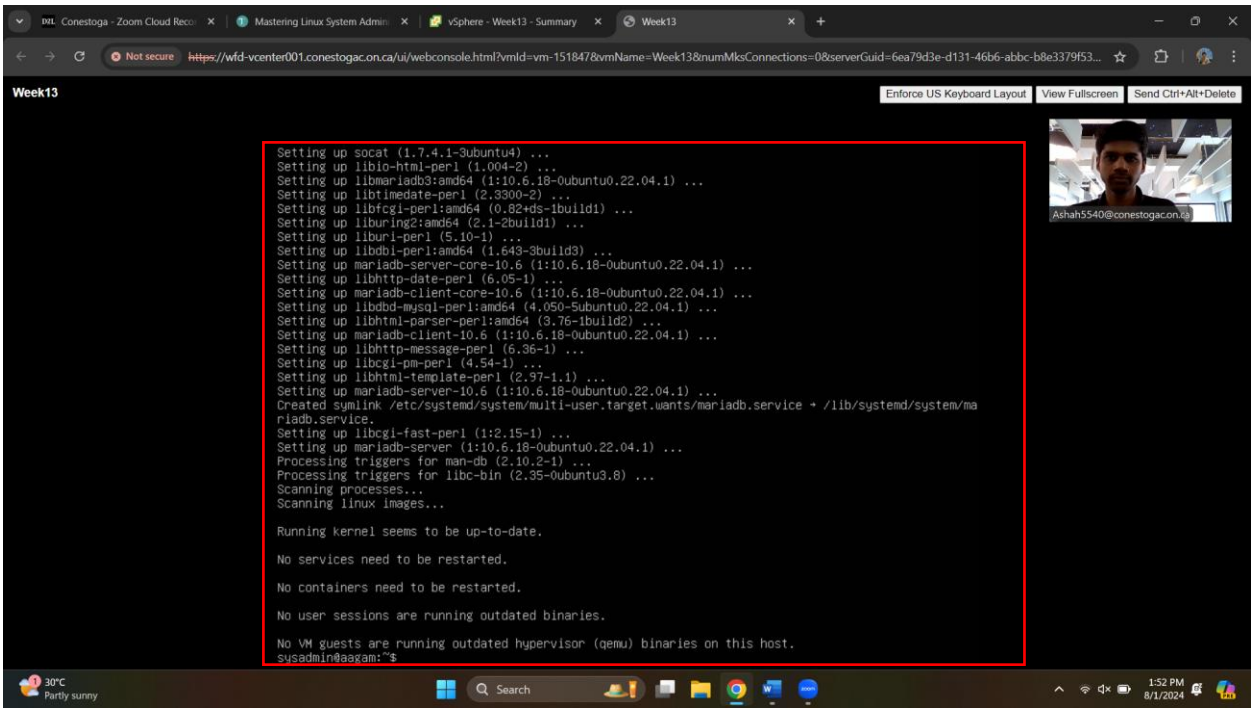
2. Update the package information on your Ubuntu system by typing **sudo apt update** and pressing Enter. Enter your account's password, if the system asks for one.



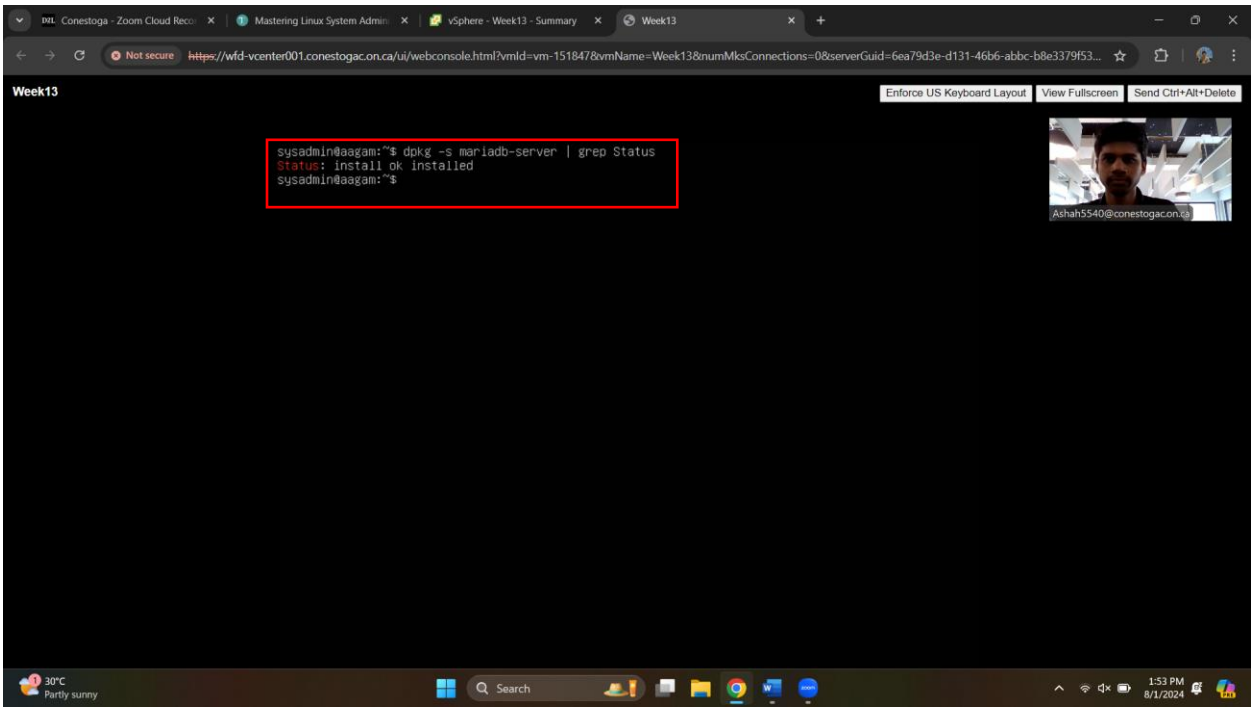
3. After you receive the command-line prompt back, install the MariaDB package by typing **sudo apt install mariadb-server** and pressing Enter. Enter your account's password, if the system asks for one.



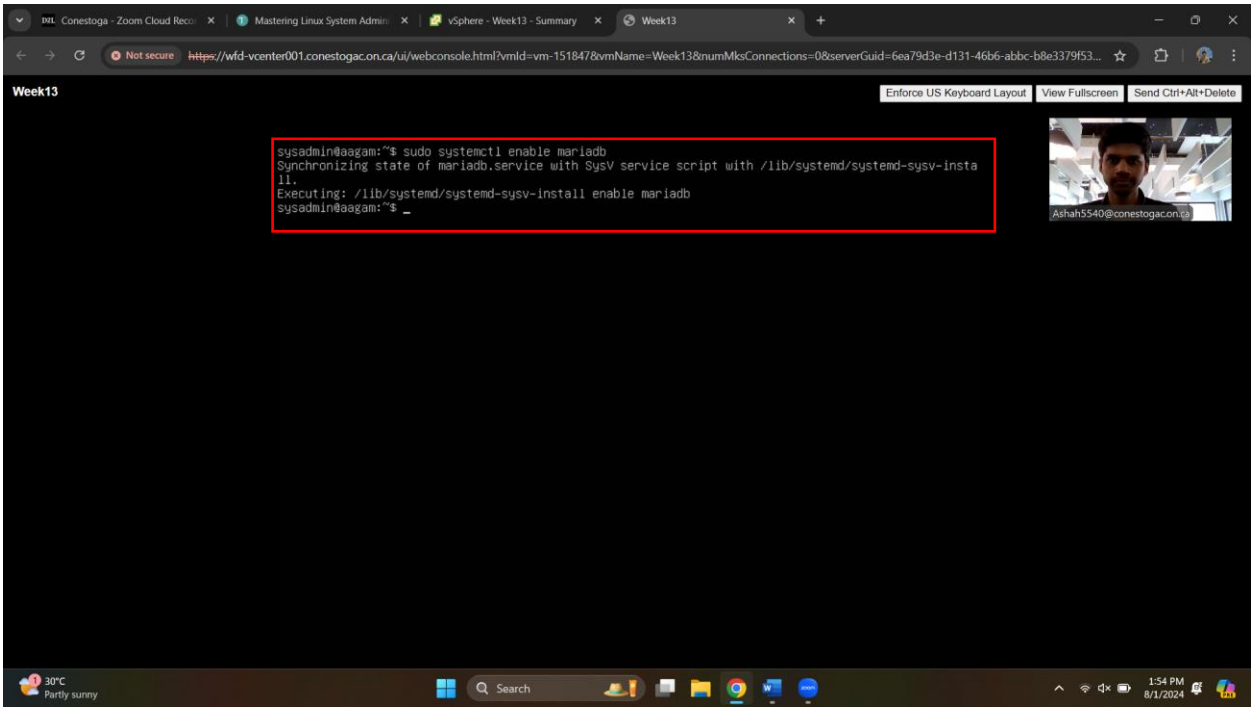
4. During the installation process, when you receive the Do you want to continue? [Y/n] prompt, type **Y** and press Enter.



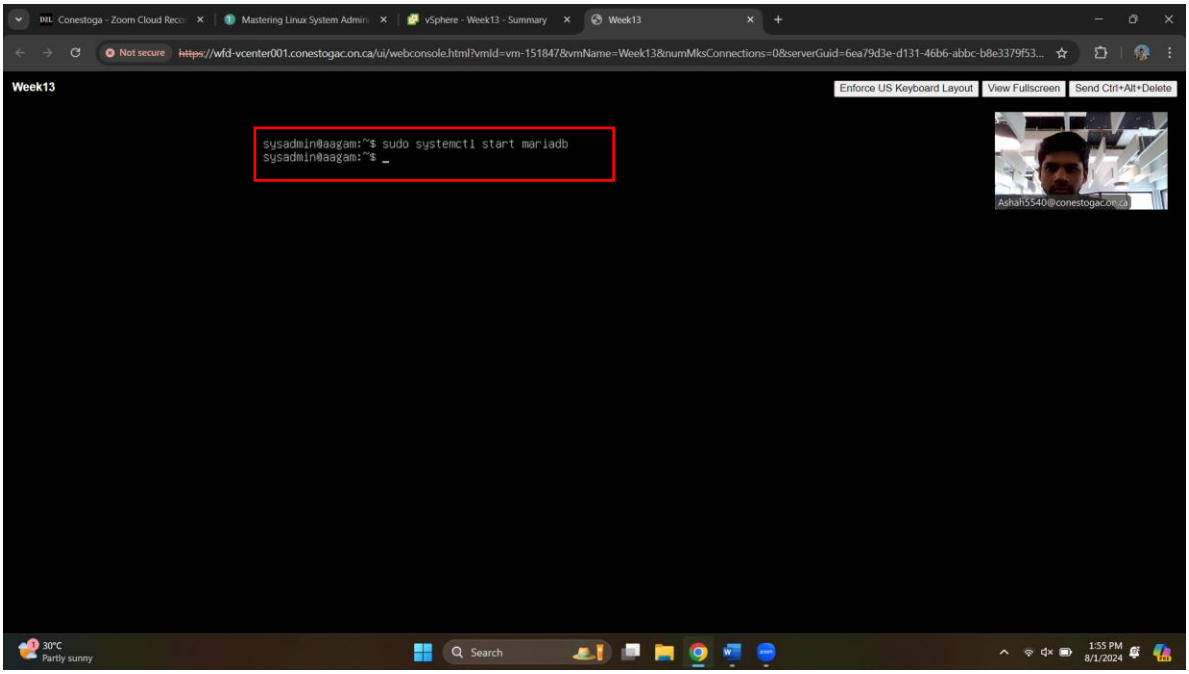
5. When you receive the command-line prompt back, check to see if the installation was successful by typing **dpkg -s mariadb-server | grep Status** and pressing Enter. If you receive a message similar to Status: install ok installed, continue on to the next step. If you do not receive this message, troubleshoot the installation. You can get additional help from Chapter 3.



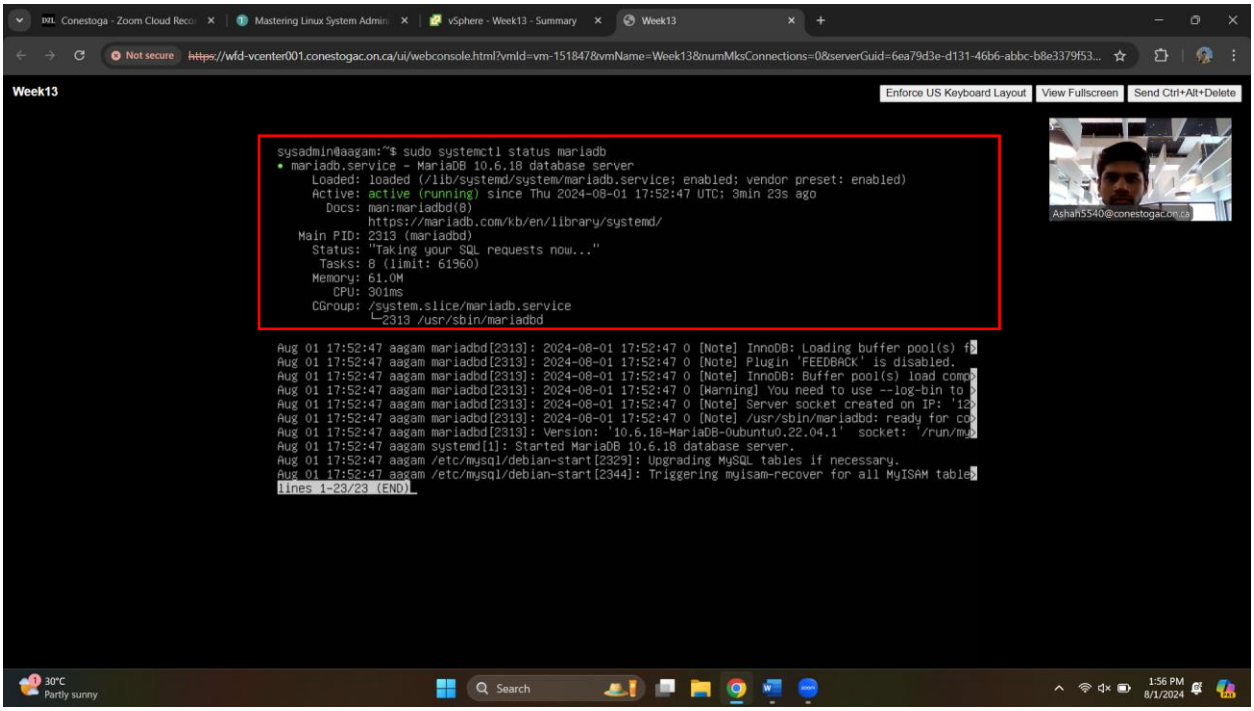
6. Set the MariaDB server to start when the system is booted, by typing **sudo systemctl enable mariadb** and pressing Enter. Enter your account's password, if the system asks for one.



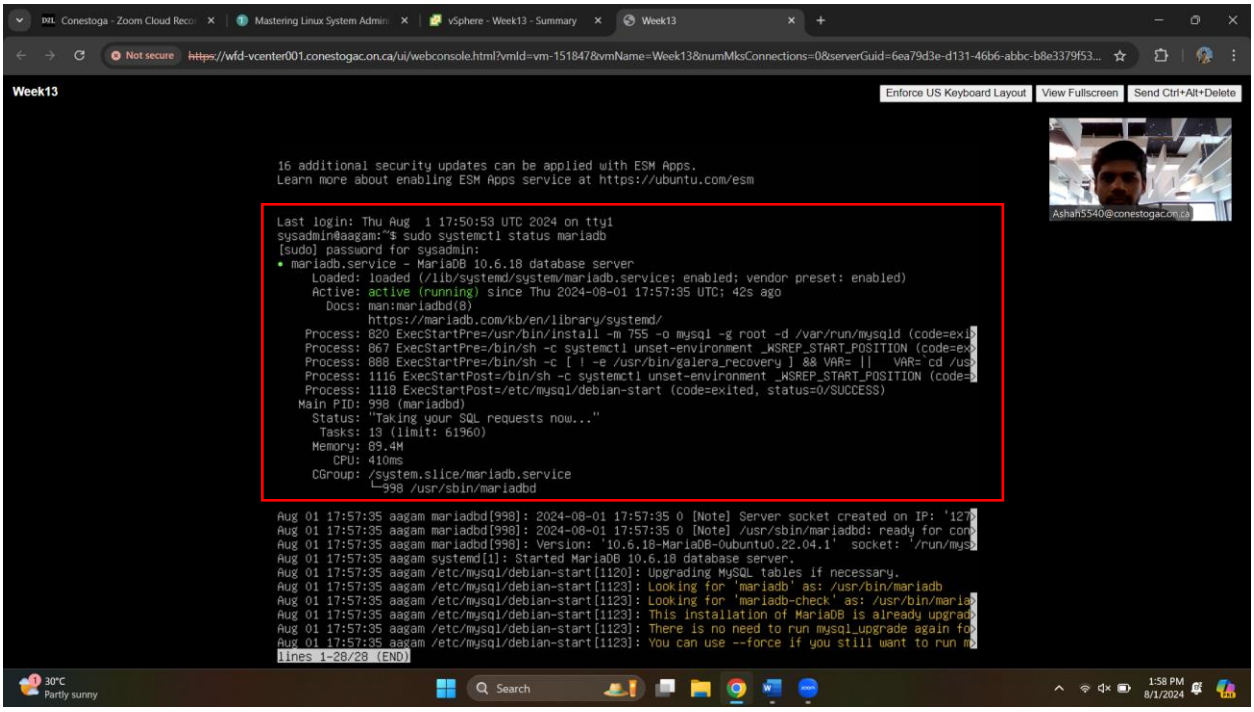
7. Start the MariaDB server by typing **sudo systemctl start mariadb** and pressing Enter. Enter your account's password, if the system asks for one.



8. Check to ensure that the MariaDB service is enabled (will start at boot) and active (is currently running), by typing **systemctl status mariadb** and pressing Enter. Look through the command's output for the words enabled and active. If you do not see these words, go back to step 6 and reissue the commands in steps 6 and 7, noting any error messages. If you do see these words, congratulations! Your installation of MariaDB on Ubuntu was successful, and the RDBMS is ready to manage your data.



After restarting ubuntu server (it its active)

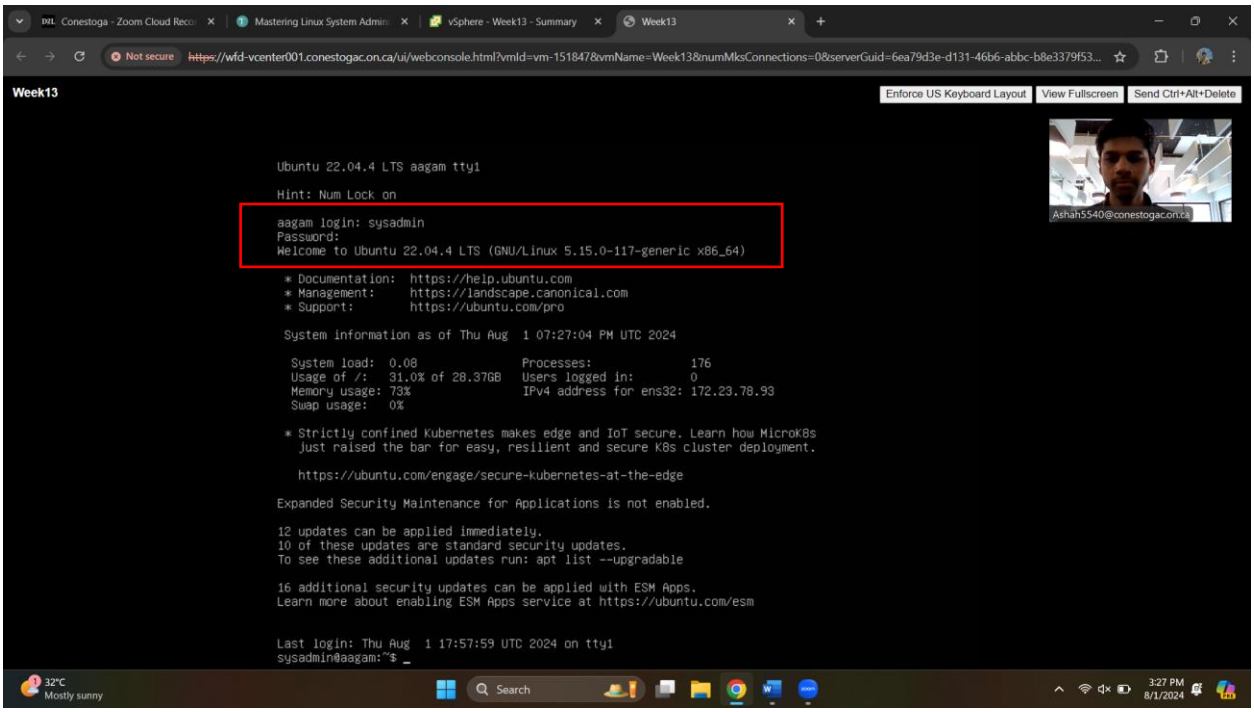


Week 13 Slide 37

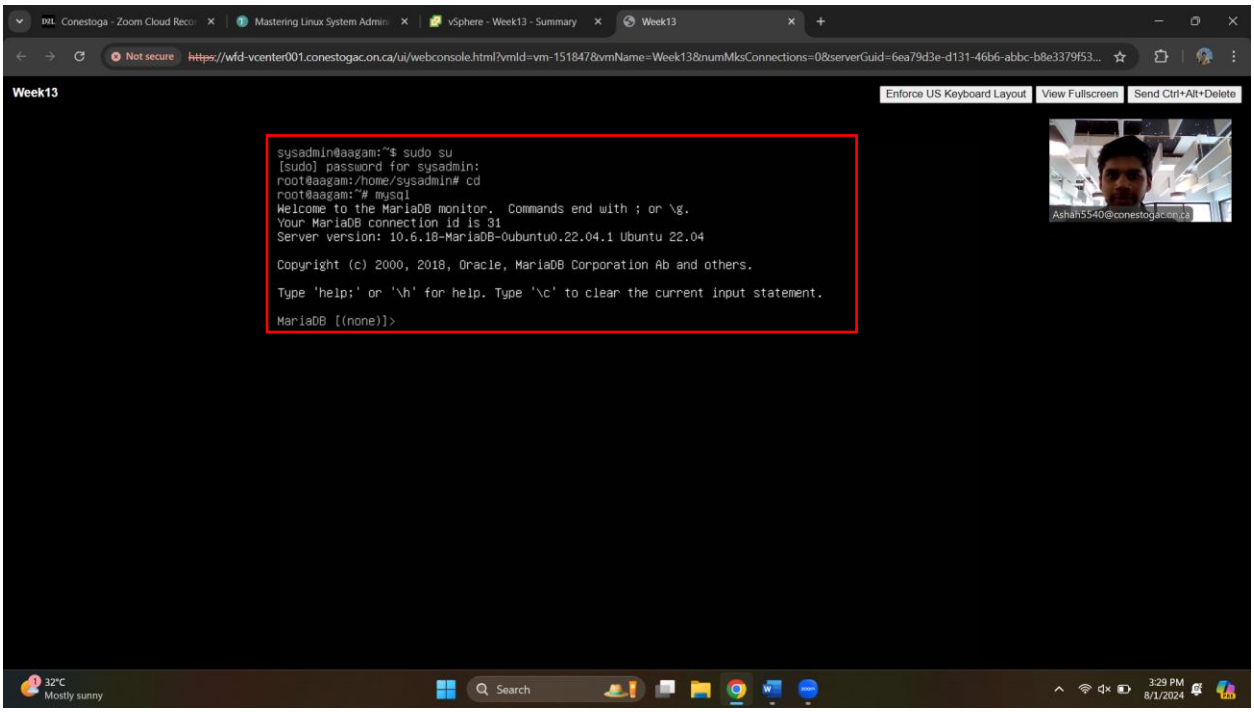
Complete the Real World Scenario: Creating a Database with MariaDB in Ch 21

CREATING A DATABASE WITH MARIADB

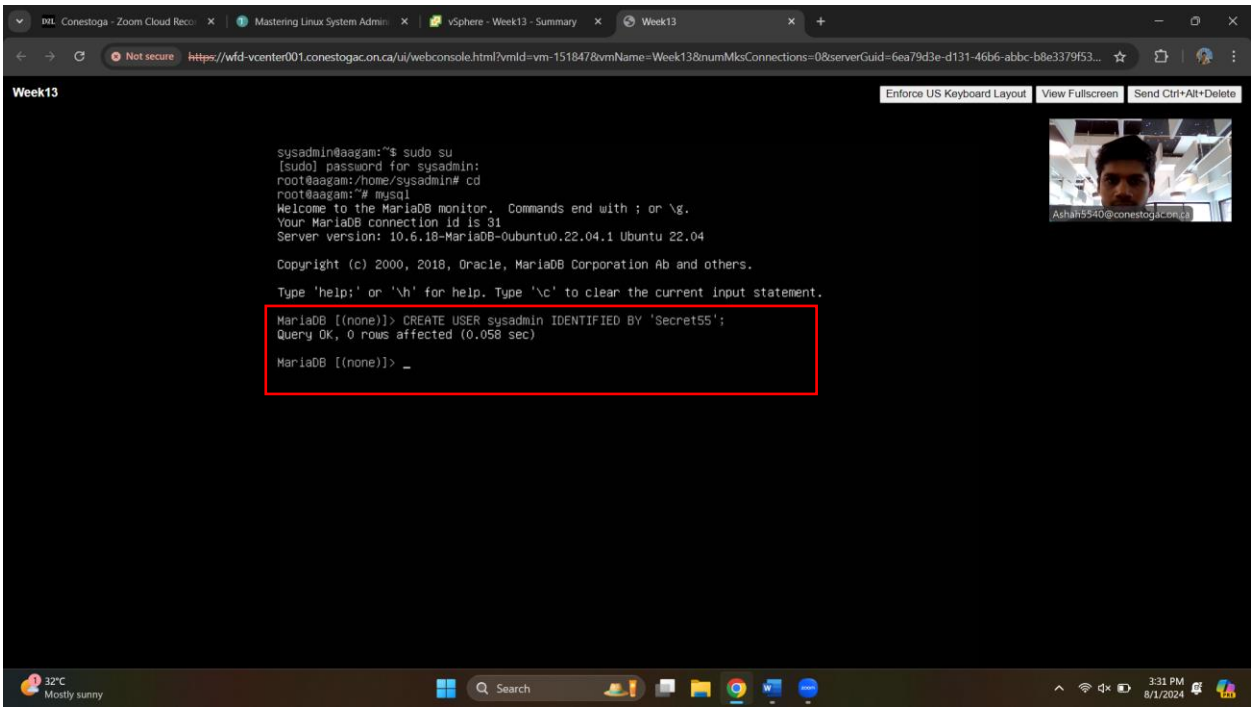
1. Log into your Linux system using the sysadmin account and the password you created for it.



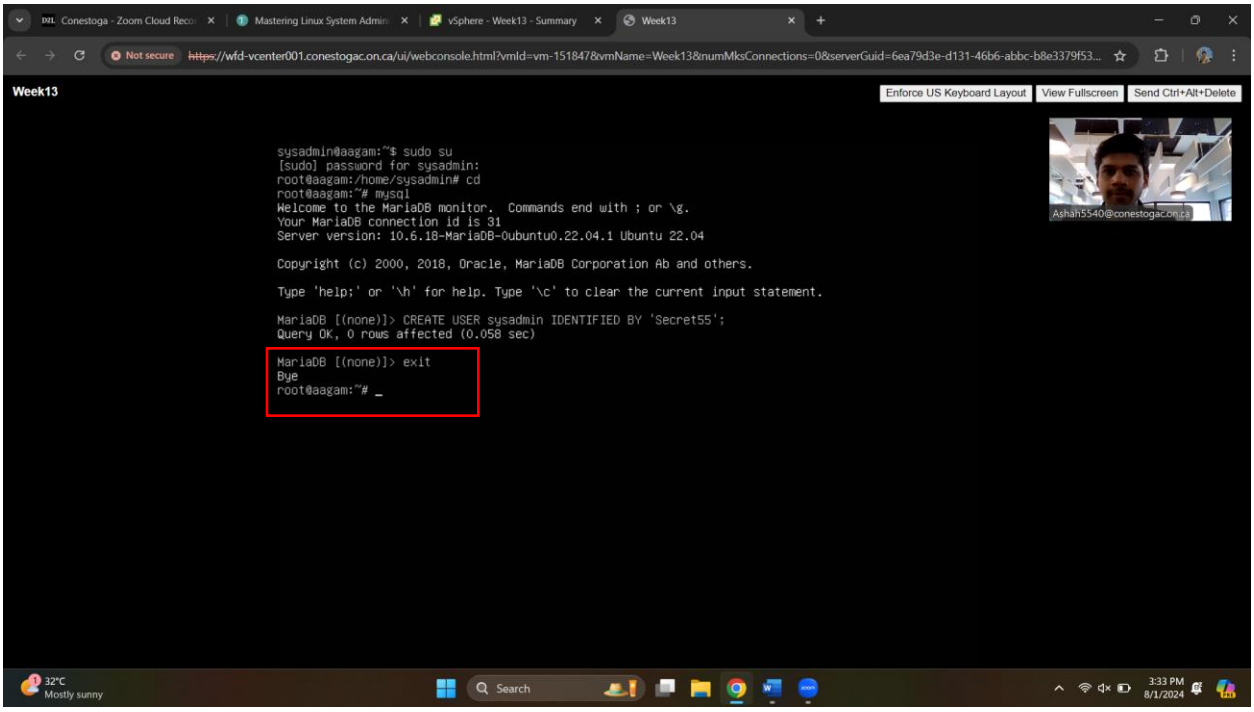
2. Using the appropriate privilege escalation method for your Linux distribution, use super user privileges (such as entering **sudo** or logging into the root account), type **mysql**, and press Enter.



3. At the MariaDB SQL interface prompt, type **CREATE USER sysadmin IDENTIFIED BY ' *password* ';**, replacing *password* with the password of your choice, and press Enter.



4. Leave the SQL interface by typing **exit** at the prompt and pressing Enter.



5. Pick one of your favorite hobbies to create a database for. Using word processing software or even just a few pieces of paper, list all the data concerning this hobby that you want to track.

➔ Hobbies as (reading)Library, books, singing.

6. Normalize your hobby data by determining the different categories the data should go into.

➔ Different books and variety of songs

7. For your categorized data, determine what tables should be created, what they should be named, and what data fields they will contain. Record this information in your document.

➔ Columns like ID, genre, date, publisher, author.

8. Determine what data field will be the primary key or create one, such as an identification number. This field will exist in every table. Record this information in your document.

➔ ID column is used as primary key

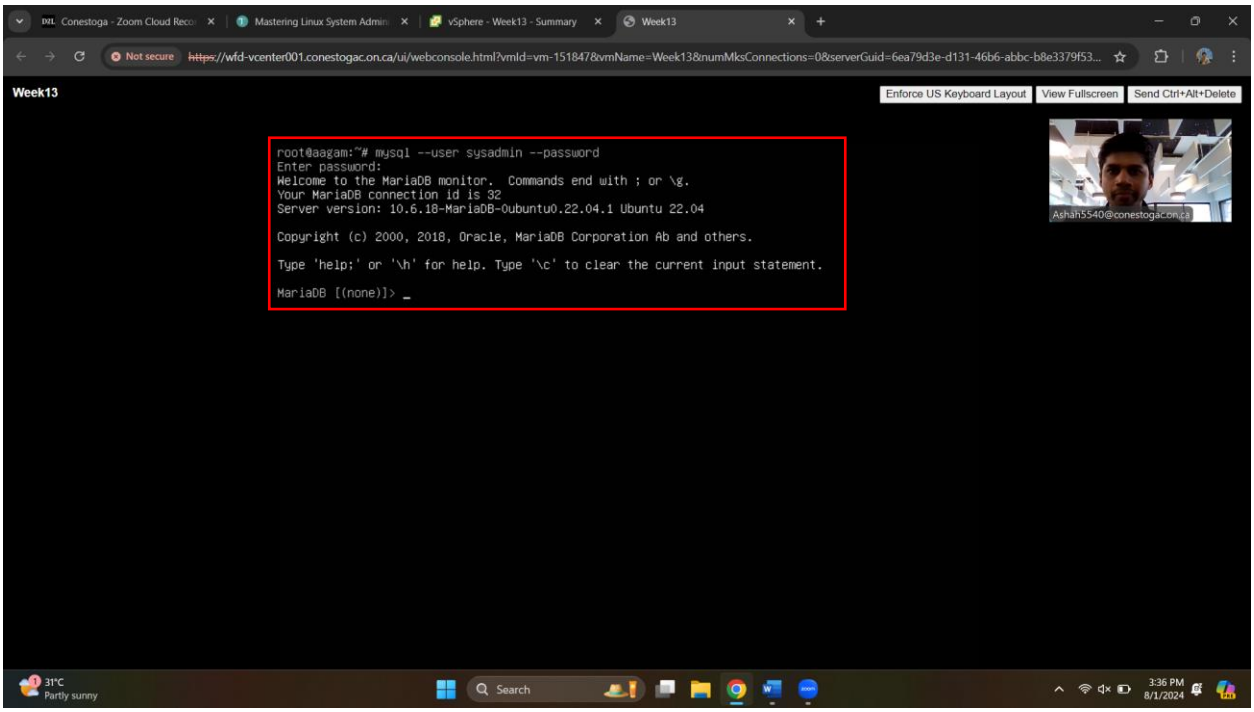
9. Decide the data type for every data field, and record this in your document.

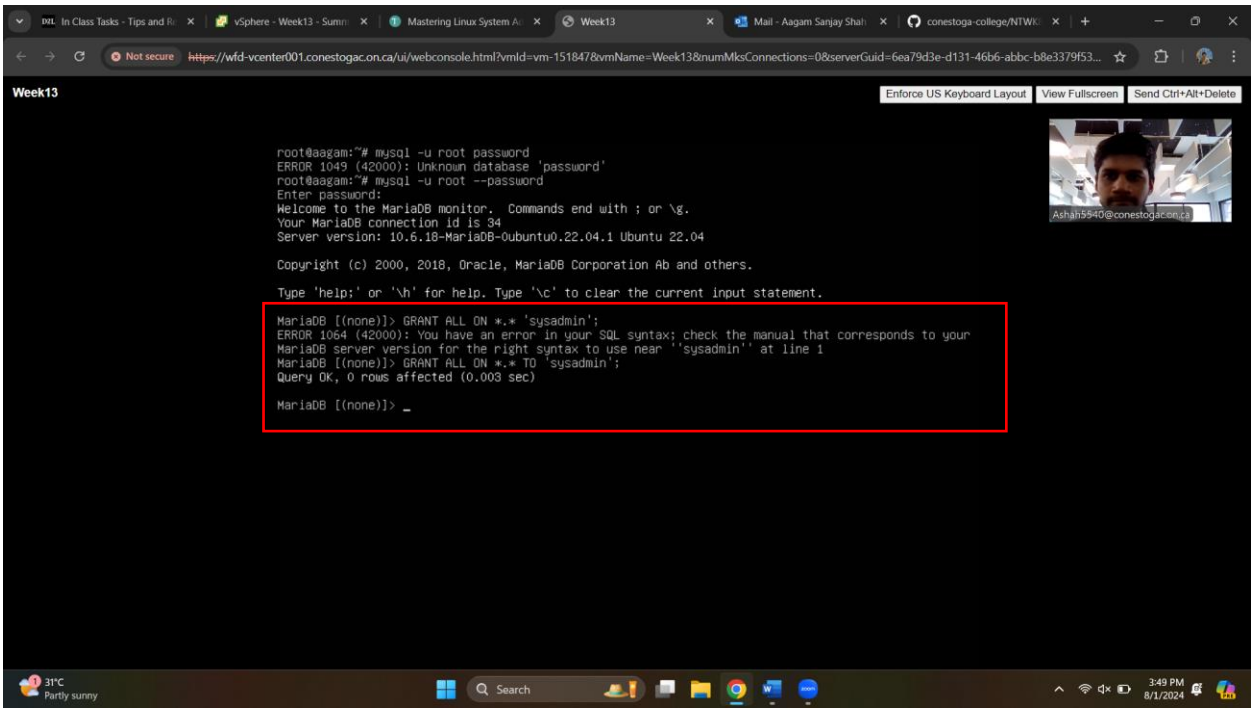
➔ Each column have different data types like text, int and date

10. Choose a name for your hobby database.

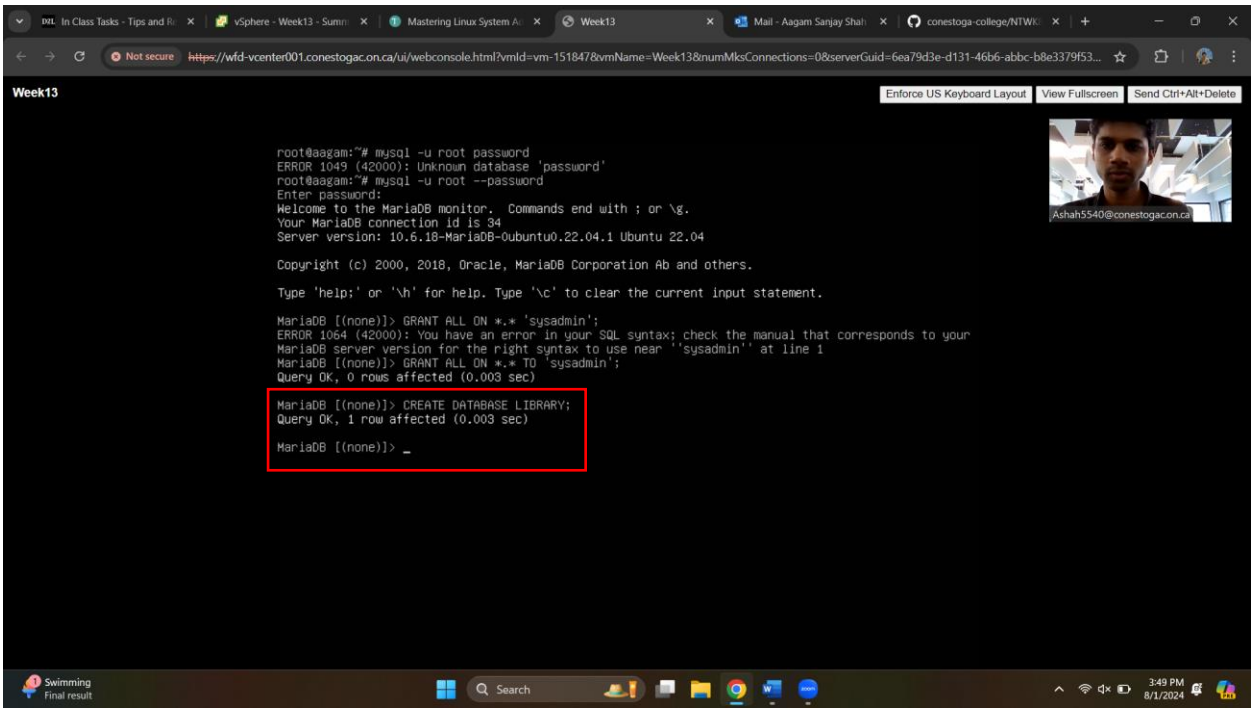
➔ Hobby database named as Library

11. Access the MariaDB server's SQL interface by typing **mysql --user sysadmin --password**, and press Enter. When asked for a password, enter the one you created in step 3 for this interface.

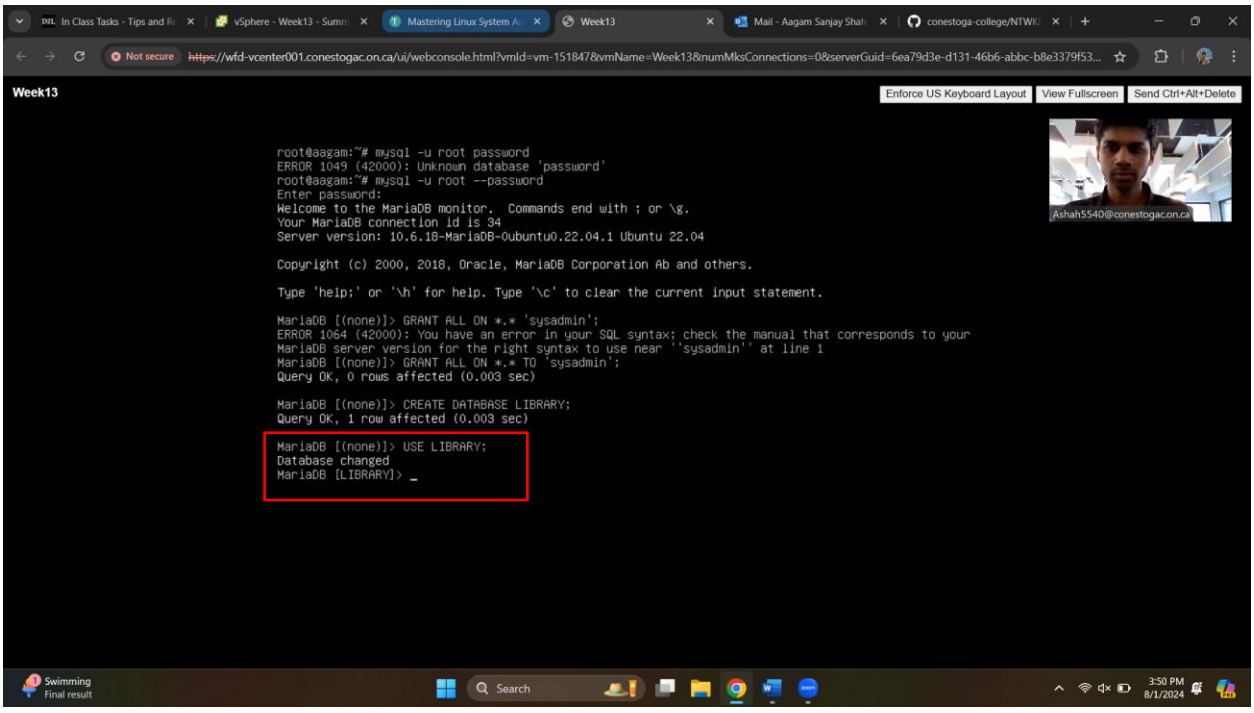




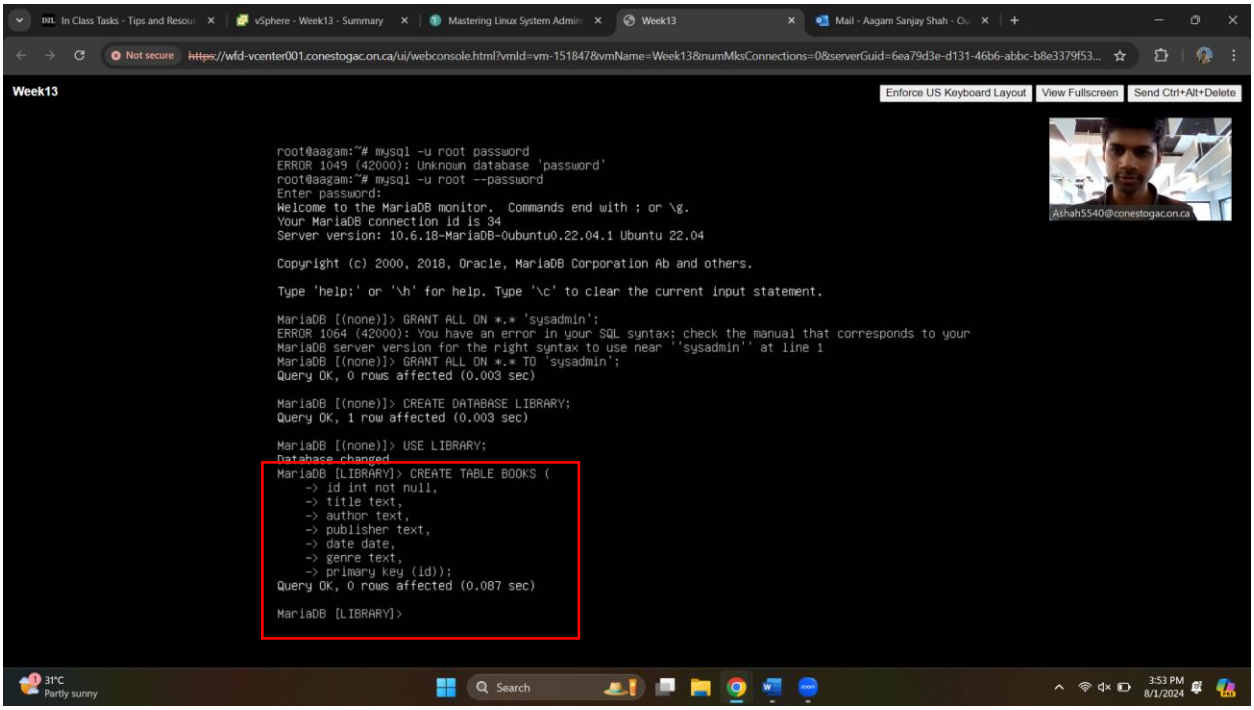
12. Create your hobby database by typing **CREATE DATABASE *database-name ;***, replacing *database-name* with the name of your hobby database, and press Enter.



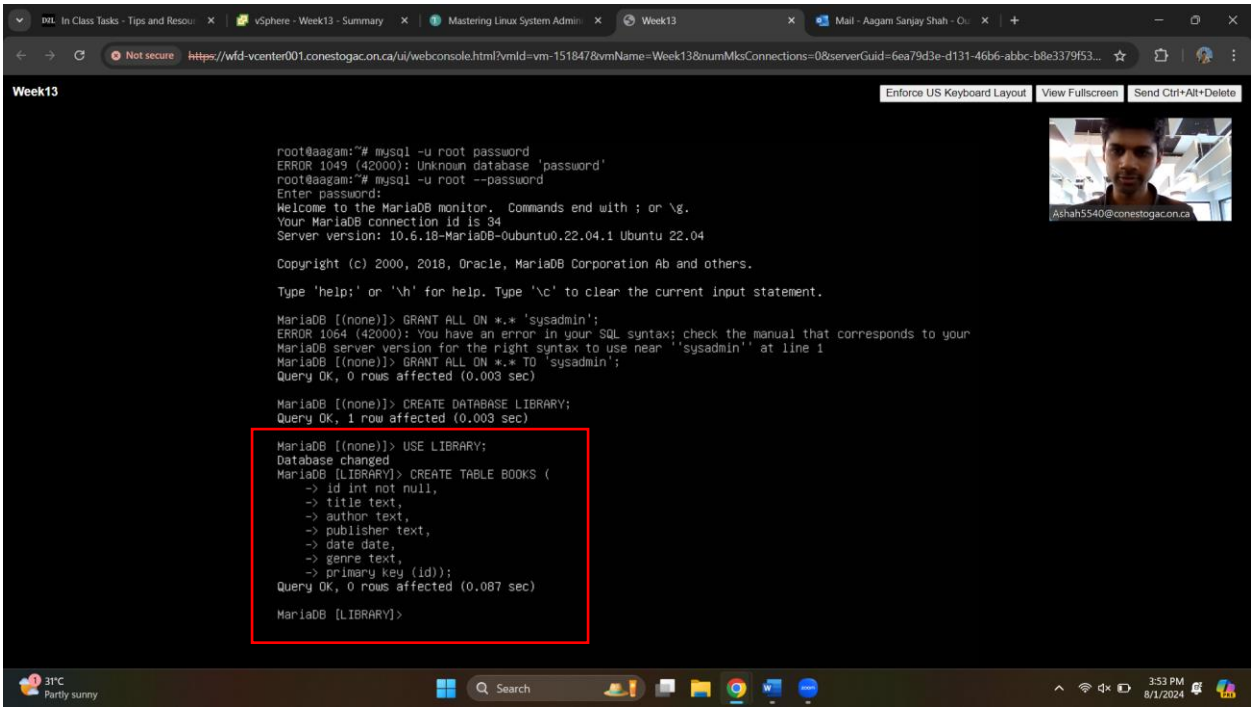
13. Select your hobby database by typing **USE *database-name ;***, replacing *database-name* with the name of your hobby database, and press Enter.



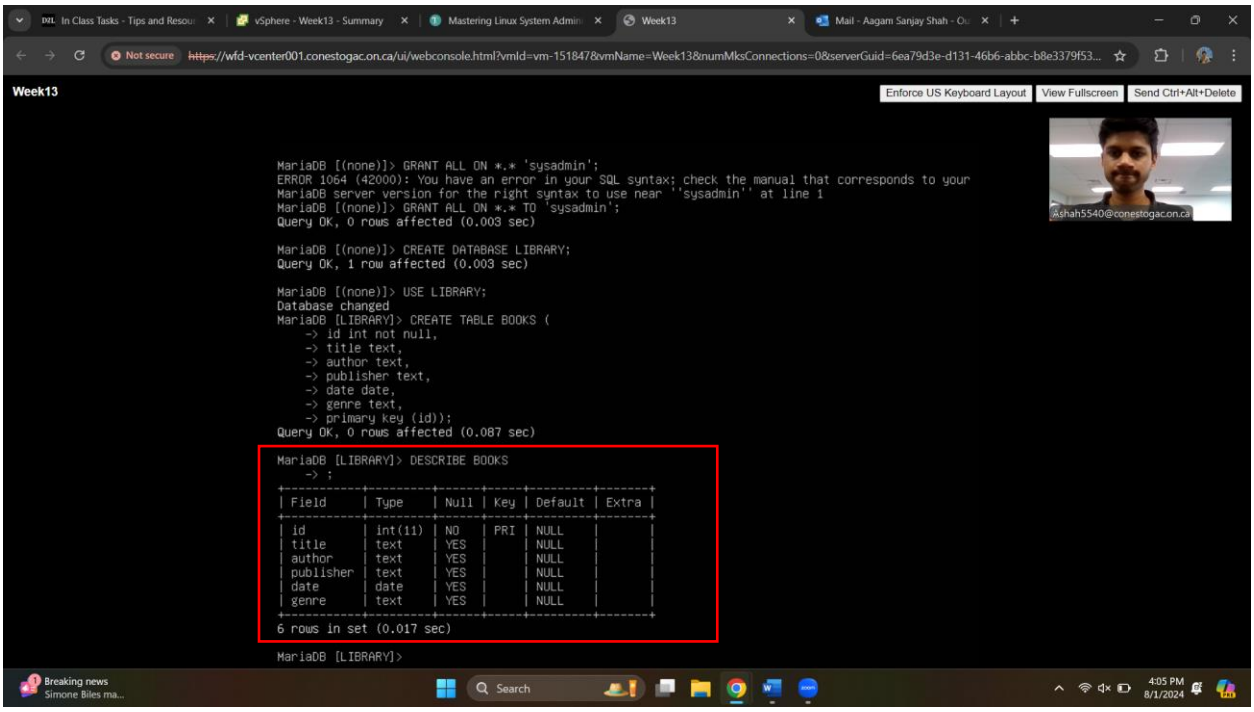
14. Create your first table in your database. Gather your table's field information from your document, and begin the process by typing **CREATE TABLE *table-name* (**, replacing *table-name* with the name of your first table, and press Enter.



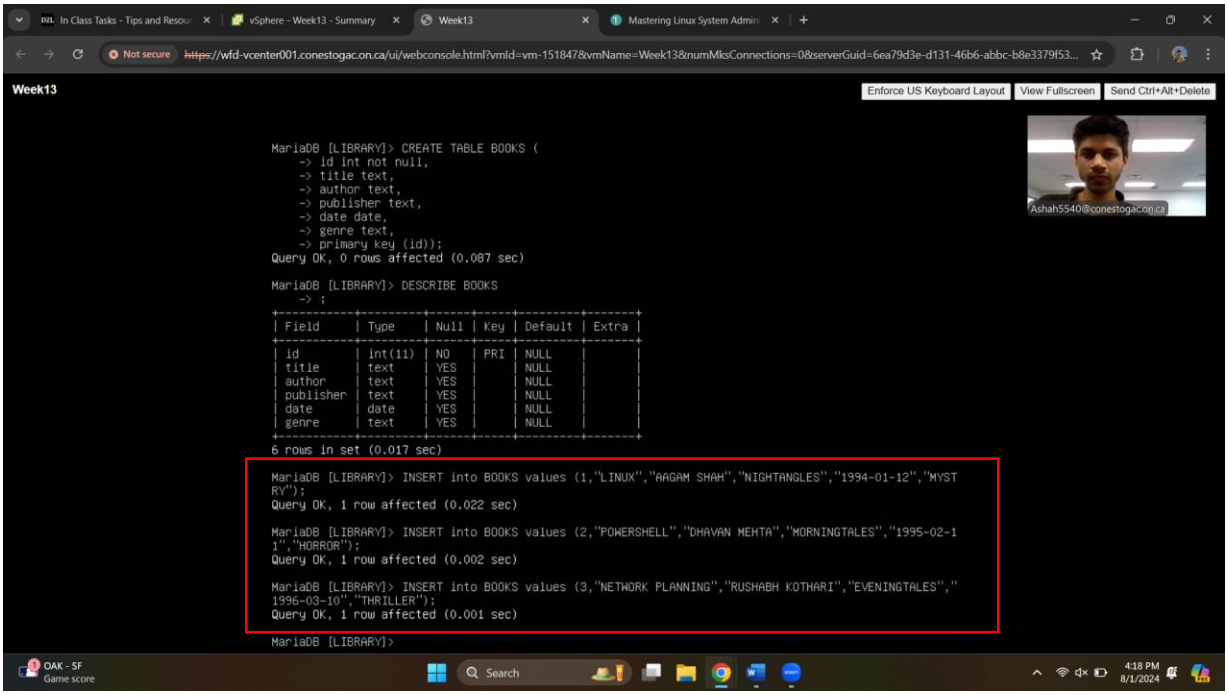
15. Using your documented table field information and guidance from this section, continue to create the table by putting in each table field and its data type. Don't forget to include the primary key information.



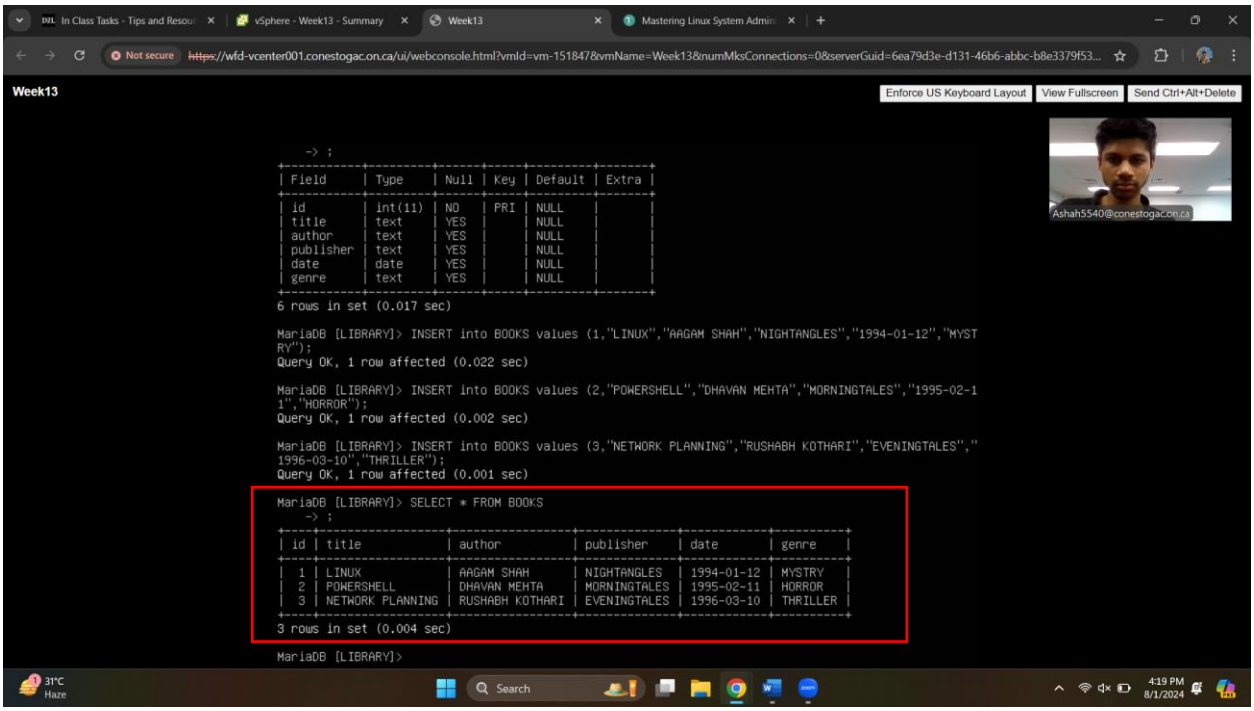
16. After you have completed creating your first table, review its structure by typing **DESCRIBE *table-name* ;**, replacing *table-name* with the name of your first table, and press Enter. If you find any problems, you can use the **DROP** command to delete the table and re-create it properly.



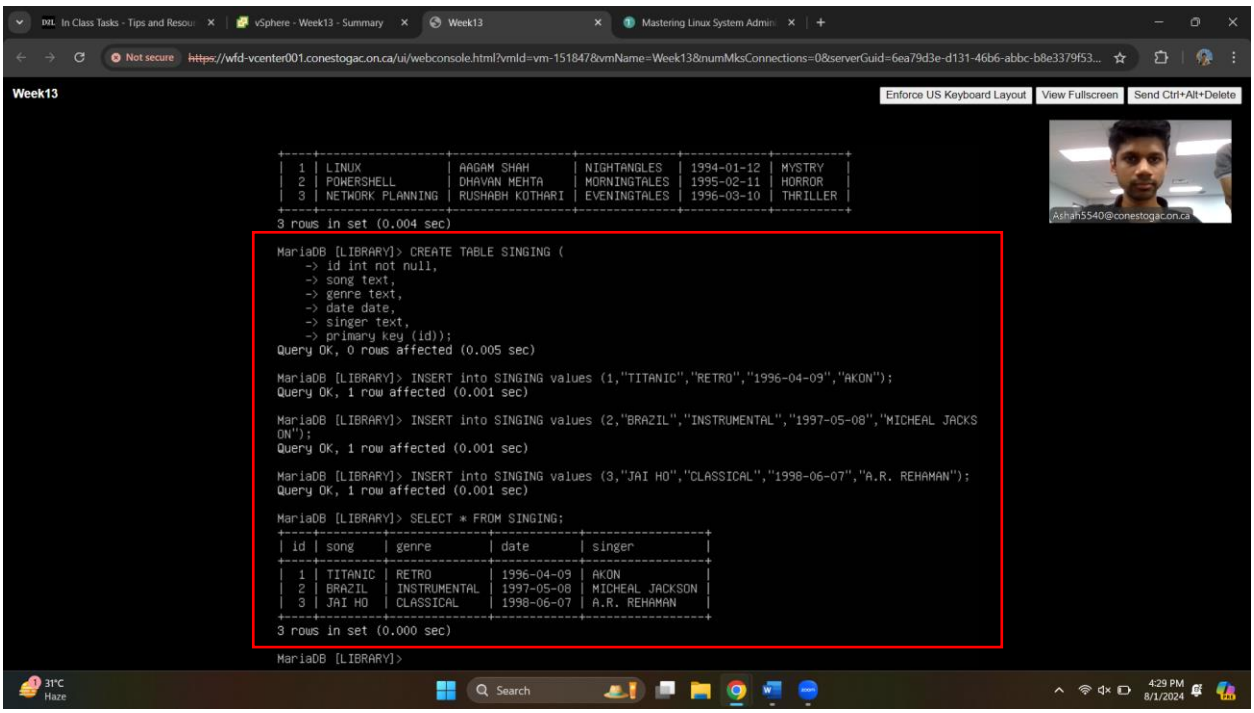
17. When you are satisfied with your first table's structure, begin populating the table with your hobby data using the appropriate INSERT INTO commands along with guidance from this section.



18. After you have completed your data entry into your first table, review the data by typing **SELECT * FROM *table-name* ;**, replacing *table-name* with the name of your first table, and press Enter. If you find any incorrect data, modify it using the **UPDATE** command along with guidance from this section.



19. When your first table is completed to your satisfaction, continue creating and populating the tables needed for your hobby database.



20. When you are done, type **exit** and press Enter to leave the SQL interface for MariaDB. Congratulations! We hope your hobby database will assist in your enjoyment of your favorite hobby.