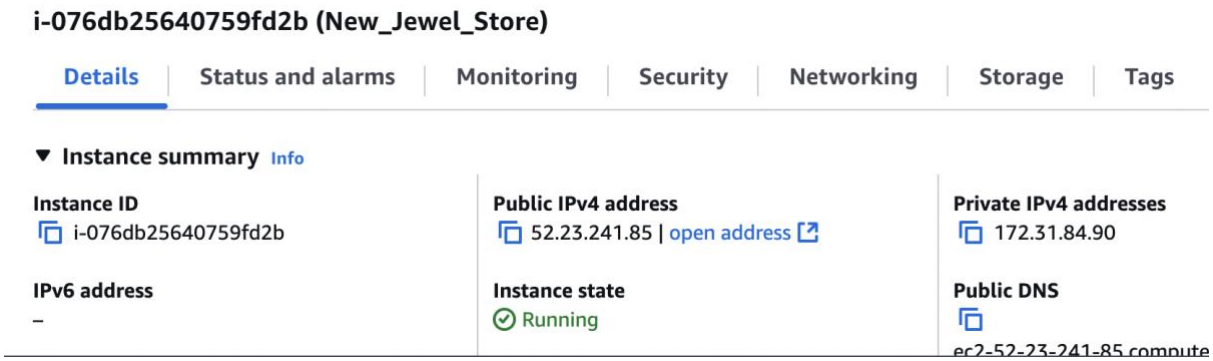
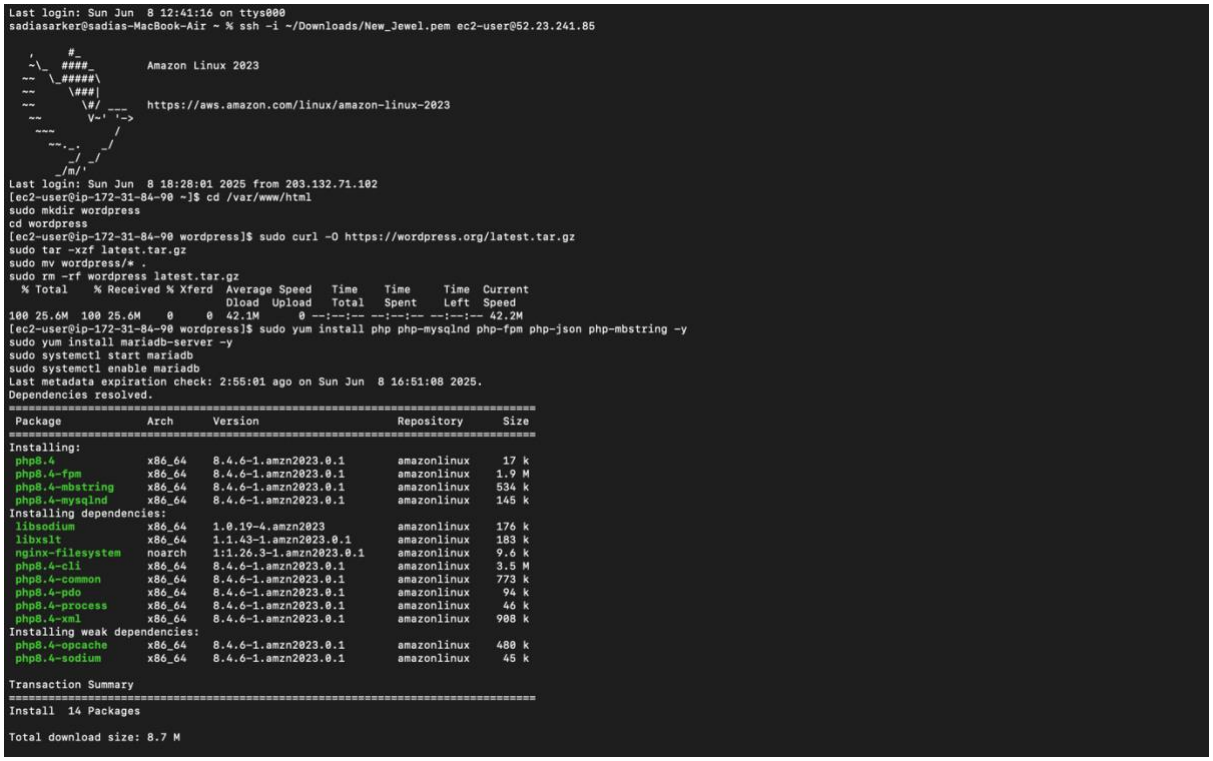
Custom INDEX, STYLE, SCRIPT Website

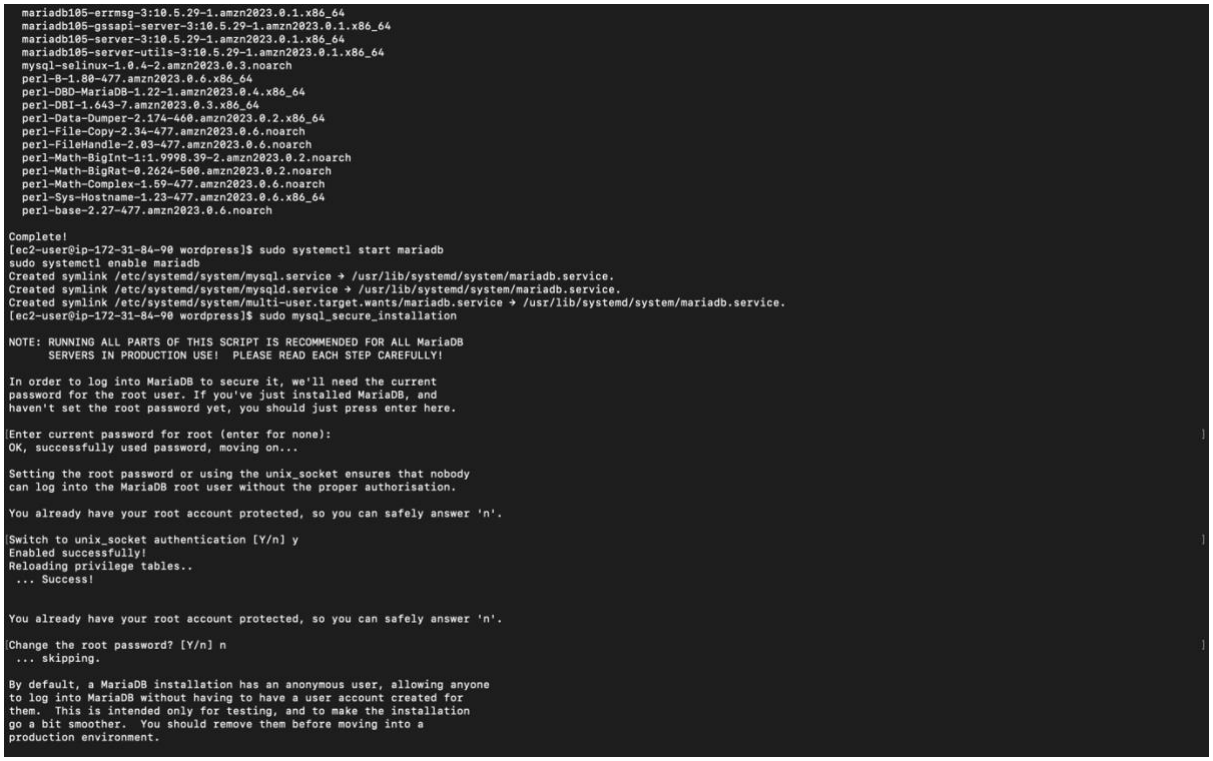


Screenshot 2: PHP Installation



Installing PHP, MariaDB, and required dependencies on Amazon Linux 2023 via SSH connection

Screenshot 3: MySQL Mariadb



Securing and starting the MariaDB database server using MySQL Secure Installation

Manual WordPress Site Setup

For my second website, I wanted to challenge myself by manually deploying WordPress on my AWS EC2 instance instead of using a prebuilt solution. This process involved several technical steps, including setting up

the environment, configuring the database, and troubleshooting errors until the site finally went live.

Launching the EC2 Server

I used Amazon EC2 to launch a Linux instance and connected to it via SSH using my PEM key. I verified that the connection was successful by checking the welcome message and Linux version in the terminal.

Preparing the Environment

Inside the EC2 terminal, I installed Apache as the web server, and then installed PHP and its dependencies like php-mysqli, php-json, and php-fpm. I also installed MariaDB for database management.

Installing and Securing MariaDB

After installing MariaDB, I started and enabled the mariadb service. I ran the secure installation script to remove test users, disable remote root login, and clean up the default database settings. This was important to make sure the database was secure before connecting it to WordPress.

Creating the WordPress Database

Using the MySQL terminal, I created a new database called wordpress, and added a user named wpuser with the password wppassword. I granted full privileges to this user on the WordPress database and then flushed the privileges to apply the changes.

Downloading and Configuring WordPress

I downloaded the latest WordPress files using the terminal and extracted them into the /var/www/html/wordpress directory. I copied the sample configuration file and renamed it wp-config.php, then manually updated the database name, username, and password to match the ones I had set in MySQL.

Fixing Permissions

To make sure WordPress could access its files properly, I changed ownership of the WordPress directory to the Apache user (apache:apache) using the chown command.

Going Live

After restarting Apache, I opened the browser and navigated to http://52.23.241.85/wordpress. The WordPress setup screen loaded successfully, which confirmed the database connection was working. I completed the installation by setting up the site title, username, and password through the browser.

I didn’t edit this website as much as it was raw html setup and honestly it was a bit hard to navigate through it. I still added my store name added sparkles but no images. I connected WordPress to this site and had a new website layout and added images and made another store there as it was easier for me to navigate through.