**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

Host a Static Website Locally: Set Up a Local Server Apache and Host a Simple HTML page with your name

Name: Shakthi Lakshmanan V Department : CSE



**Introduction to Hosting a Static Website Locally**

In this **Proof of Concept (POC)**, we'll learn how to **host a static website on our own computer** using an Apache server. Static websites are simple web pages with content like text, images, and links. Since the page doesn't change based on user actions (like login or search), it's called "static."

This POC is perfect for beginners because:

1. **It's simple to set up** – We’ll use basic tools like a text editor and a local server.
2. **You’ll see results instantly** – You can open the site on your browser within a few steps.
3. **It's a great first step into web development and cloud concepts** – Hosting locally helps you understand how websites work before you move to advanced hosting like cloud platforms.

**What’s the Goal of This Project?**

1. **Set Up Apache Server** on your computer (it’s like creating your own personal web server).
2. **Host a Static Website** with Apache, which means your website’s pages will be served directly from your computer.
3. **Build a Simple HTML Page** with your name on it, and **host it locally** (so you can access it without needing an internet connection!).

**Mindmap:** Your Journey from Zero to Hero in Local Hosting

Launch Local Hosting

Host Your Files

Setup Apache Server

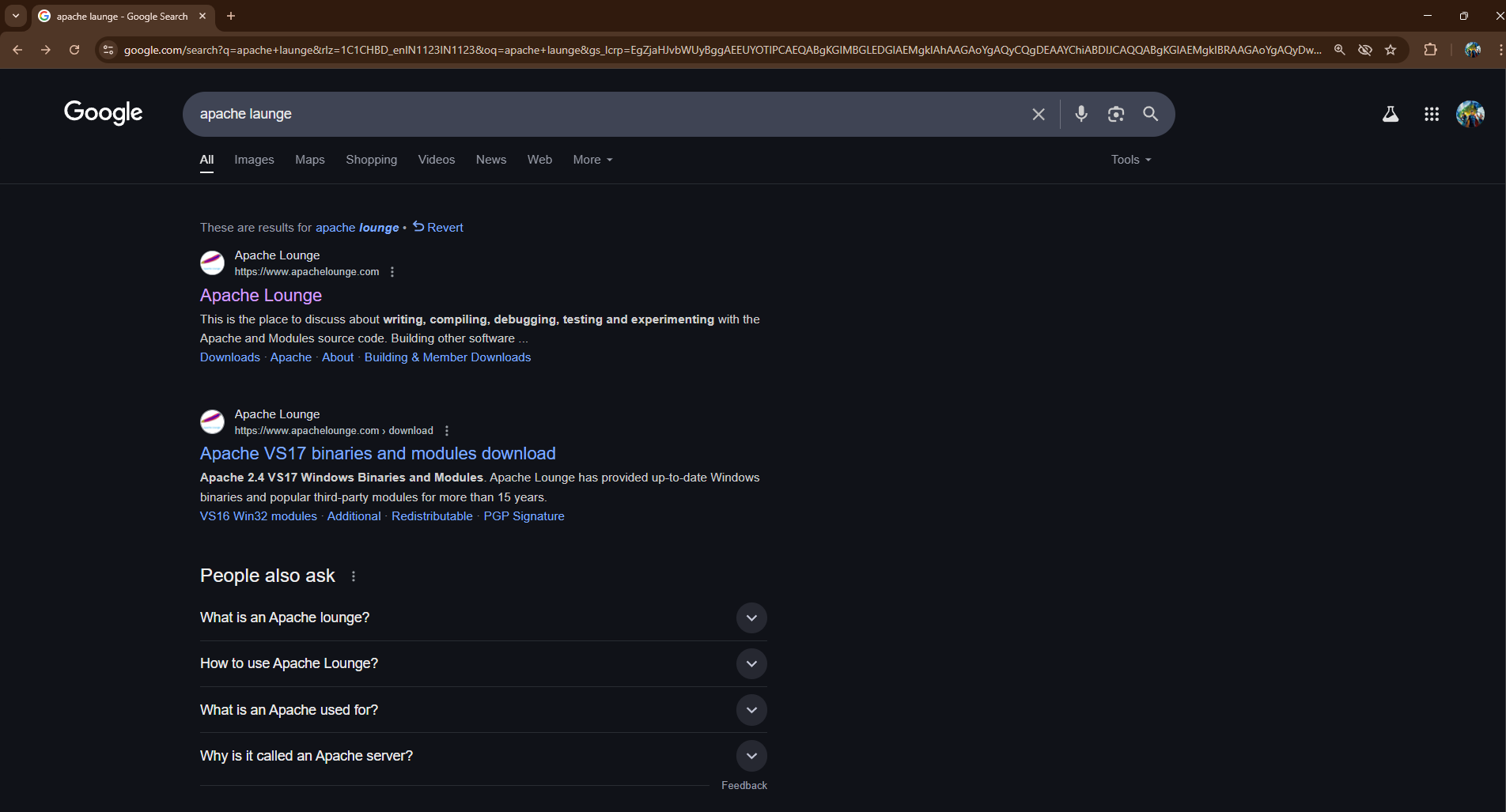
Download Apache Configure Server Create Your HTML Test Locally

Install Apache Edit httpd.conf Display Your Name Open Localhost

**Step-by-Step Overview**

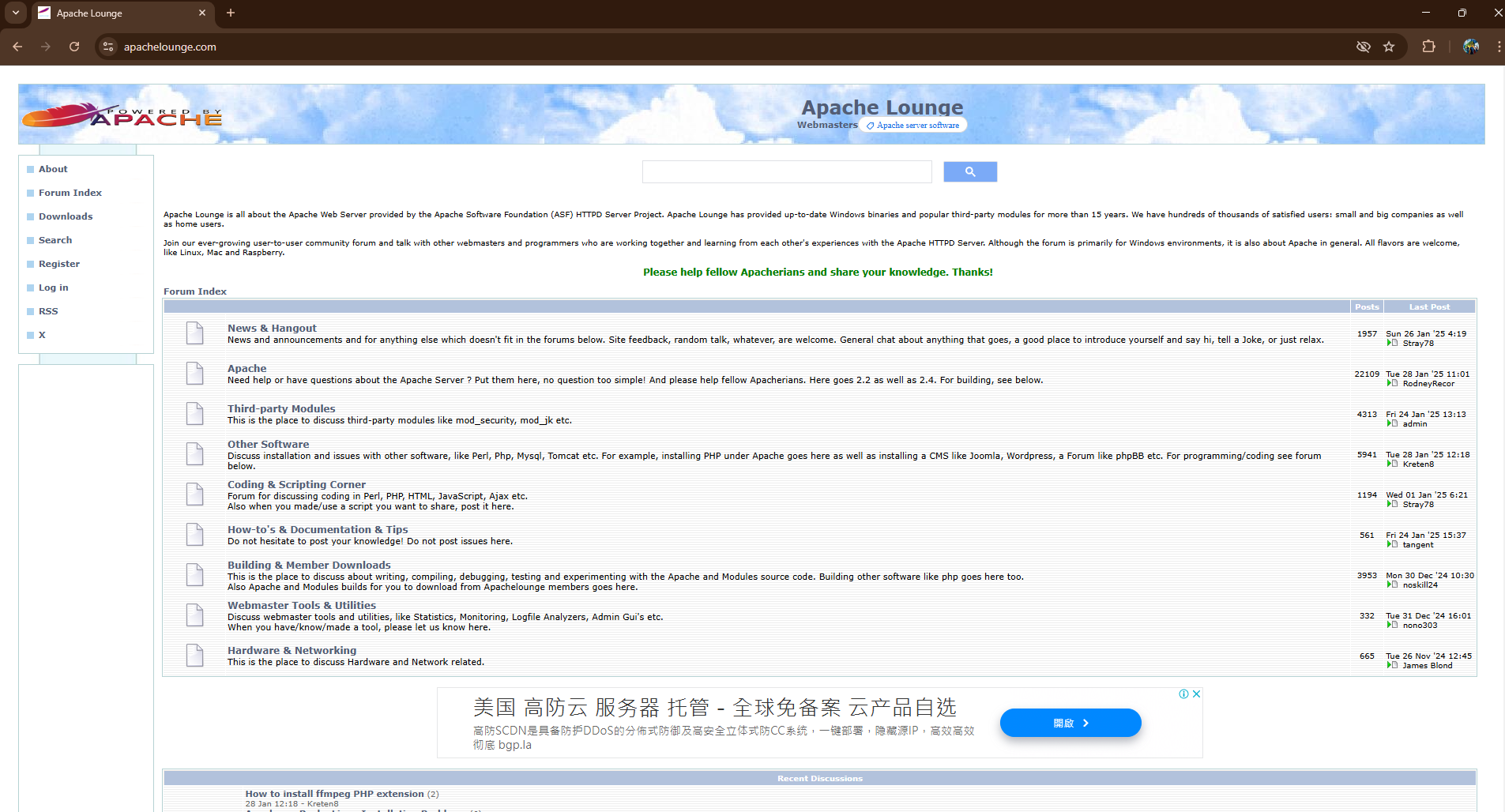
Step1:

Search for "Apache Lounge" on Google and click the first link to access the official website.



Step 2 :

Click on the "Downloads" option located on the left-hand side of the Apache Lounge website.



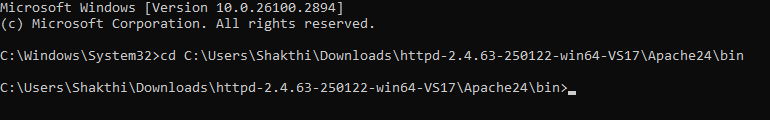
Step 3 :

Click on the link **"Apache 2.4.63-250122 Win64"** (Windows version), download the file, and extract all its contents.



Step 4 :

Open Command Prompt as Administrator (Windows + R, type cmd, right-click and select 'Run as Administrator') and use the command cd C:\path\to\apache\bin to set the path to the Apache bin folder.



Step 5 :

Then Run the installation command :

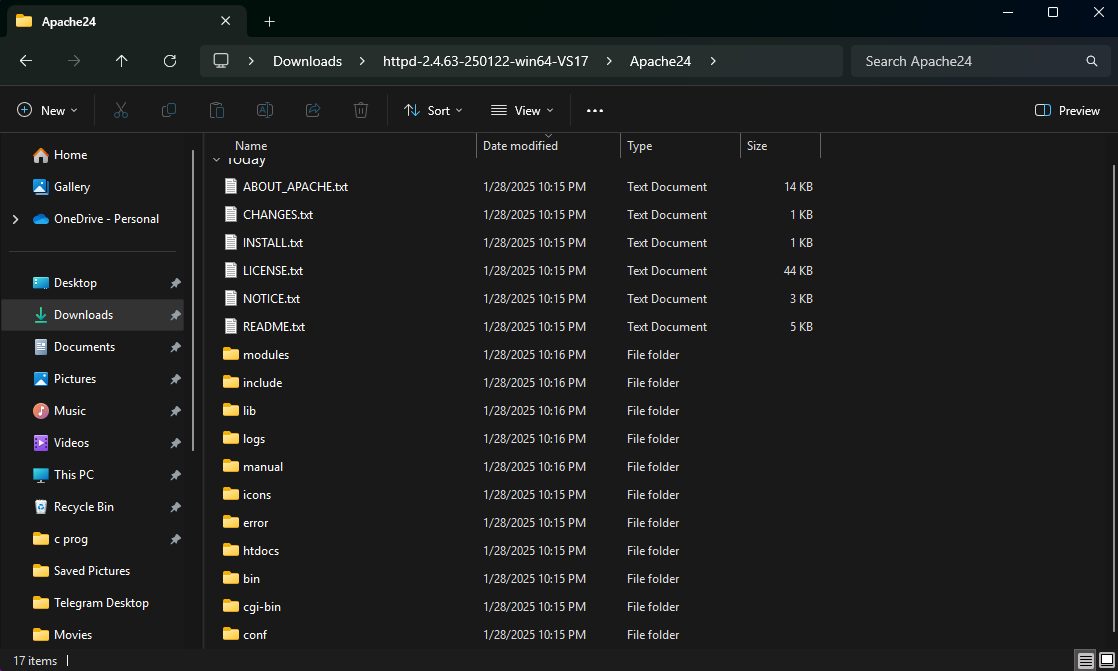
**httpd.exe -k install**

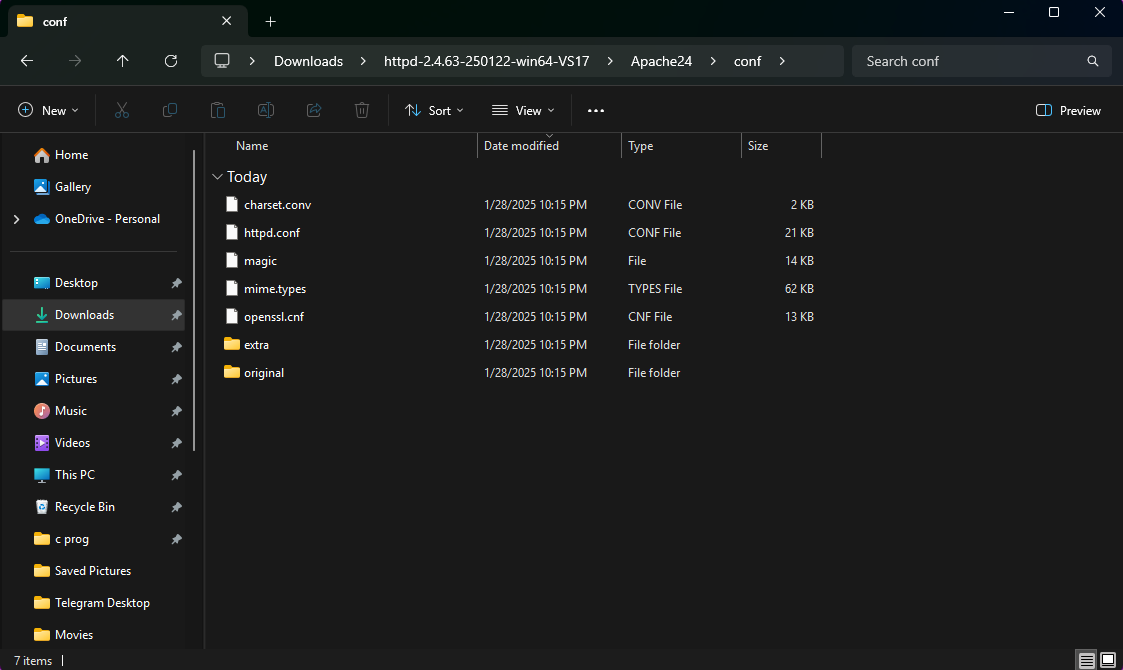


Step 6 :

Navigate to the Apache folder you downloaded, go to the **conf** folder, and right-click on the httpd.conf file; select 'Edit with Notepad'

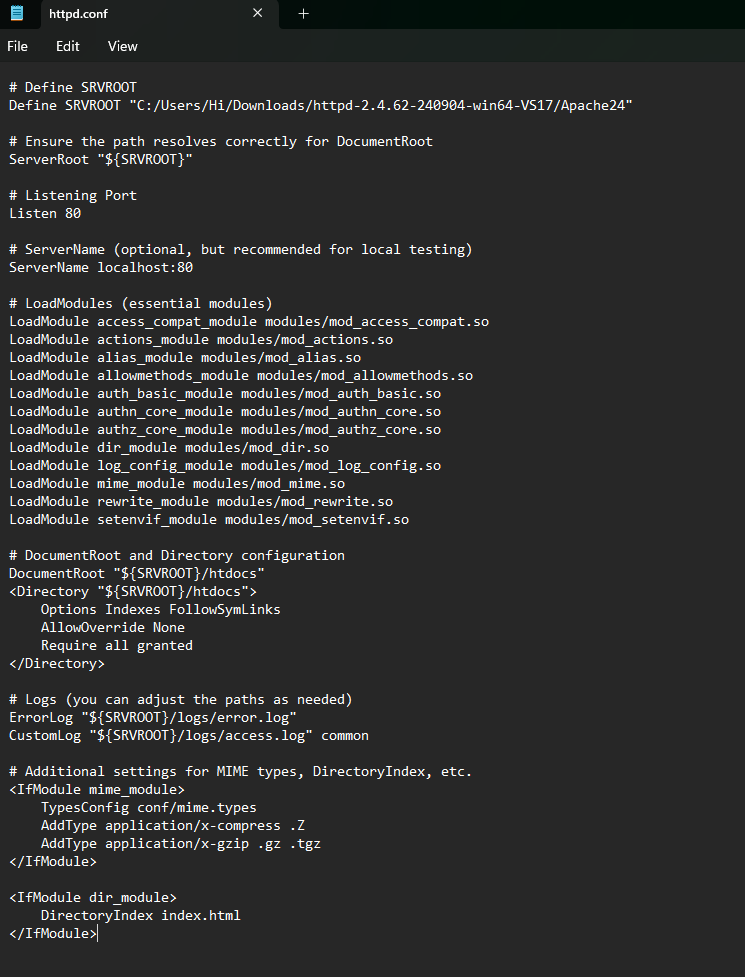
(Apache/conf/httpd.conf)





Step 7 :

Inside the **httpd.conf** file, replace the content with the provided configuration. Ensure you update the SRVROOT directive with your Apache installation path. This configuration defines the server’s root directory, listening port, modules, document root for serving web files, logging paths, and basic permissions, ensuring Apache serves content correctly from the specified htdocs directory.



Step 8 :

Open Command Prompt and type the command **httpd.exe -t** to test the configuration file. If the configuration is correct, you should see **'Syntax OK**'.



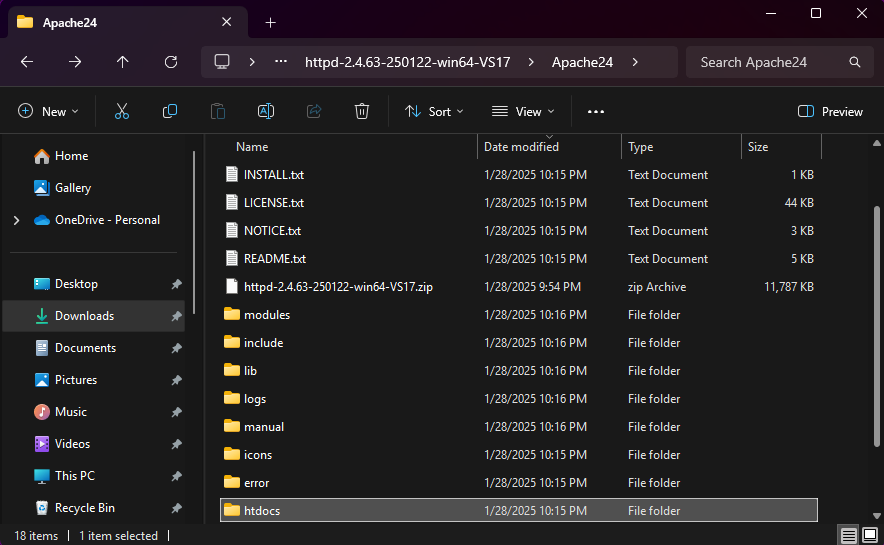
Step 9 :

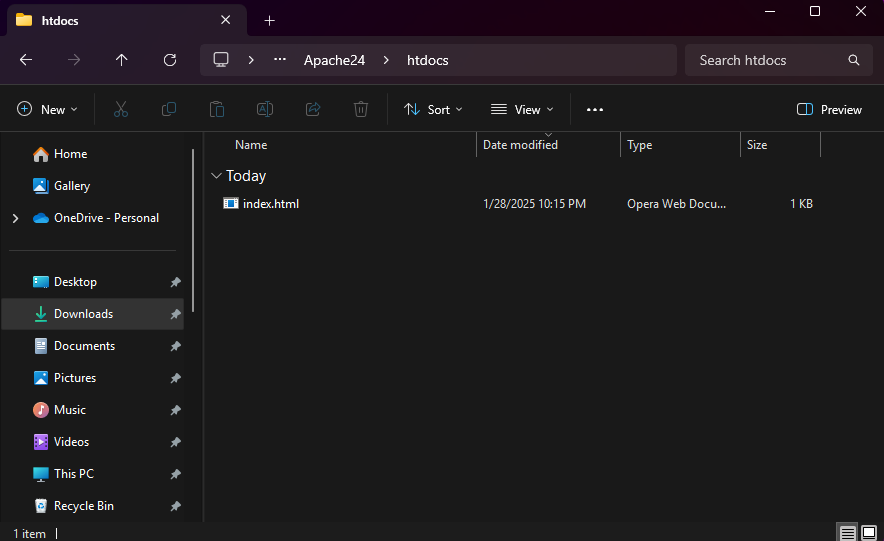
Run the command **httpd.exe -k** start to start the Apache server.



Step10:

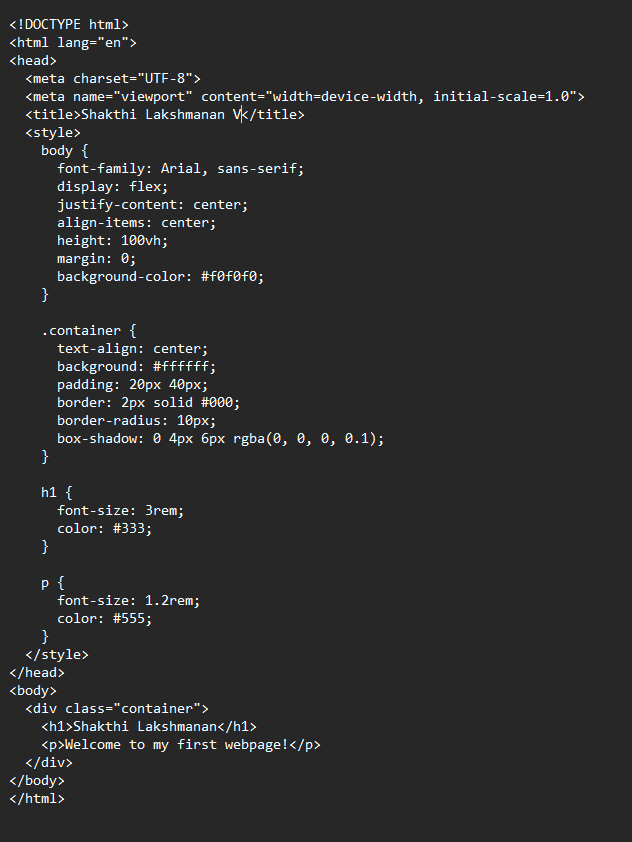
Go to the Apache folder, navigate to the **htdocs** folder, and find the **index.html** file. Right-click on it and select 'Edit with Notepad'.





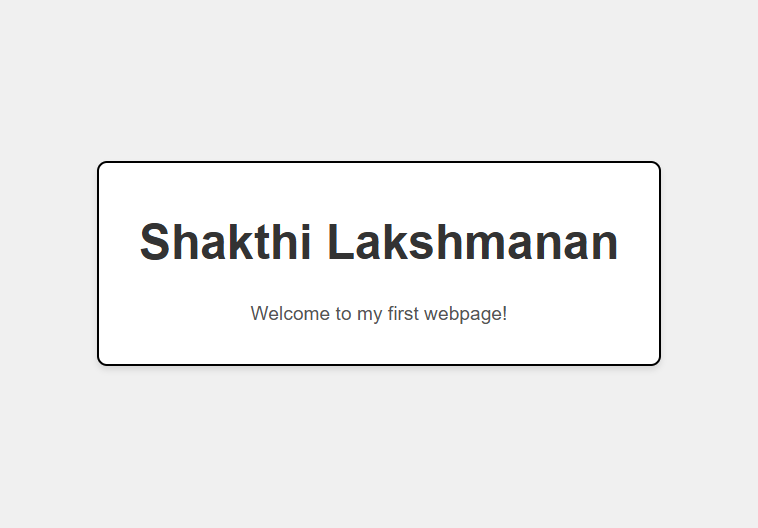
Step 11 :

Create a simple model to display your name in HTML (you may optionally add CSS for styling).



Step 12 :

Open the Chrome browser and type **localhost/index.html** in the address bar. You should be able to see the website hosted successfully.



**Expected Outcome**

By completing this POC, you will:

1. Successfully configure and run an Apache server locally.

2. Host a static HTML website that displays your name.

3. Understand the basics of web server configuration and file hosting.