

## **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: Jayadasan S

Department : CSE

# Introduction and Overview

In this POC, we will learn how to host a static website locally using the Apache HTTP Server. This process involves setting up a local web server, configuring it correctly, and hosting a simple HTML page. By following these steps, you'll get hands-on experience with configuring and running a local Apache server, which is a foundational skill for web hosting and server management.

## Objective

The goal of this project is to:

1. Set up a local web server using Apache.
2. Configure the server to host static files.
3. Create and host a simple HTML page displaying your name.

## Importance of Local Hosting

Local hosting is an essential skill for developers, as it allows them to test and experiment with web applications in a controlled environment. It offers several advantages, such as:

**Hands-On Learning:** Gain practical experience with server setup and configuration.

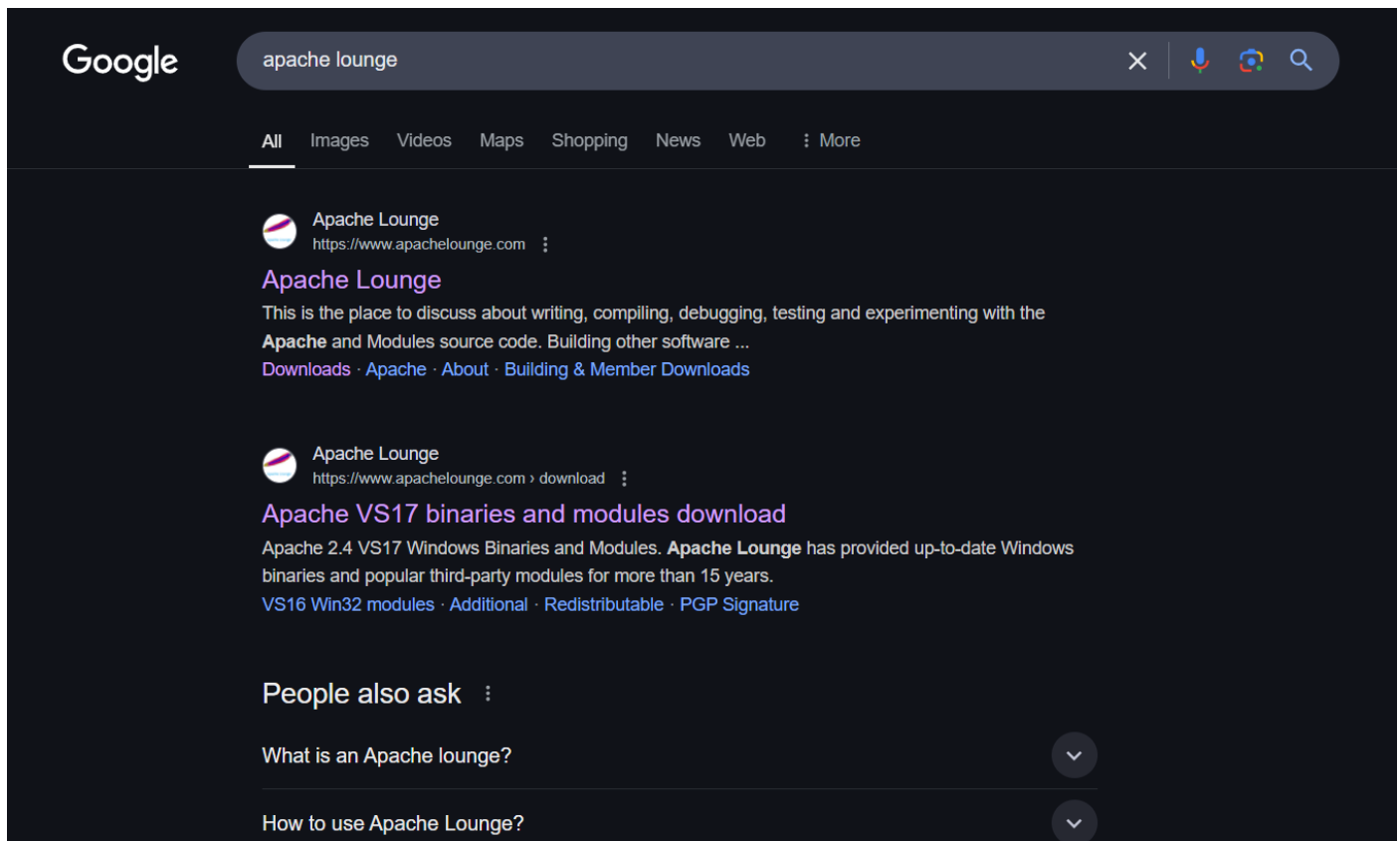
**Testing Ground:** Safely test and debug websites before deploying them to a live server.

**Offline Development:** Work on web projects without requiring an active internet connection.

# 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.

Click on the link "**Apache 2.4.62-240904 Win64**" (Windows version), download the file, and extract all its contents.

Apache Lounge has provided up-to-date Windows binaries and popular third-party modules for more than 15 years. We have hundreds of thousands of satisfied users: small and big home users. Always build with up to date dependencies and latest compilers, and tested thorough. The binaries are referenced by the ASF, Microsoft, PHP etc. and more and more software with our binaries and modules.

The binaries, are build with the sources from ASF at <http://apache.org>, contains the latest patches and latest dependencies like zlib, openssl etc. which makes the downloads here more reliable than downloads from other places. The binaries **do not run** on XP and 2003. Runs on: 7 SP1, Vista SP2, 8/8.1, 10, 11 Server 2008 SP2 / R2 SP1, Server 2012 / R2, Server 2016/2019/2022

Build with the latest Windows Visual Studio C++ 2022 aka VS17. Has improvements, fixes and optimizations over VS16 in areas like Performance, MemoryManagement, New standard features, Code generation and Stability. For example code quality tuning and improvements done across different code generation areas for "speed". And makes more use of latest supported Windows editions (win7 and up) internal features.

**VS17 is backward compatible,** That means, a VS16/15/14 module can be used inside the VS17 binary.

**Be sure** you installed latest 14.42.34433.0 Visual C++ Redistributable Visual Studio 2015-2022 : [vc\\_redist\\_x64](#) or [vc\\_redist\\_x86](#) see [Redistributable](#)

### Apache 2.4.63-250122 Win64

[Info & Changelog](#)

[httpd-2.4.63-250122-win64-VS17.zip](#) 22 Jan '25 11.787k  
[PGP Signature \(Public PGP key\)](#), [SHA1-SHA512 Checksums](#)

### Apache 2.4.63-250122 Win32

 [httpd-2.4.63-250122-win32-vs17.zip](#) 22 Jan '25 10.589k  
[PGP Signature \(Public PGP key\)](#), [SHA1-SHA512 Checksums](#)

To be sure that a download is intact and has not been tampered with, use PGP, see [PGP Signature](#)

打开 >

HK高防雲, 美國高防雲, 雲服務, 彈性雲, 600G HK香港本地防禦, 全球 3TB 防禦, 100倍故障賠償 [bgp.la](http://bgp.la)

## Apache 2.4 modules VS17

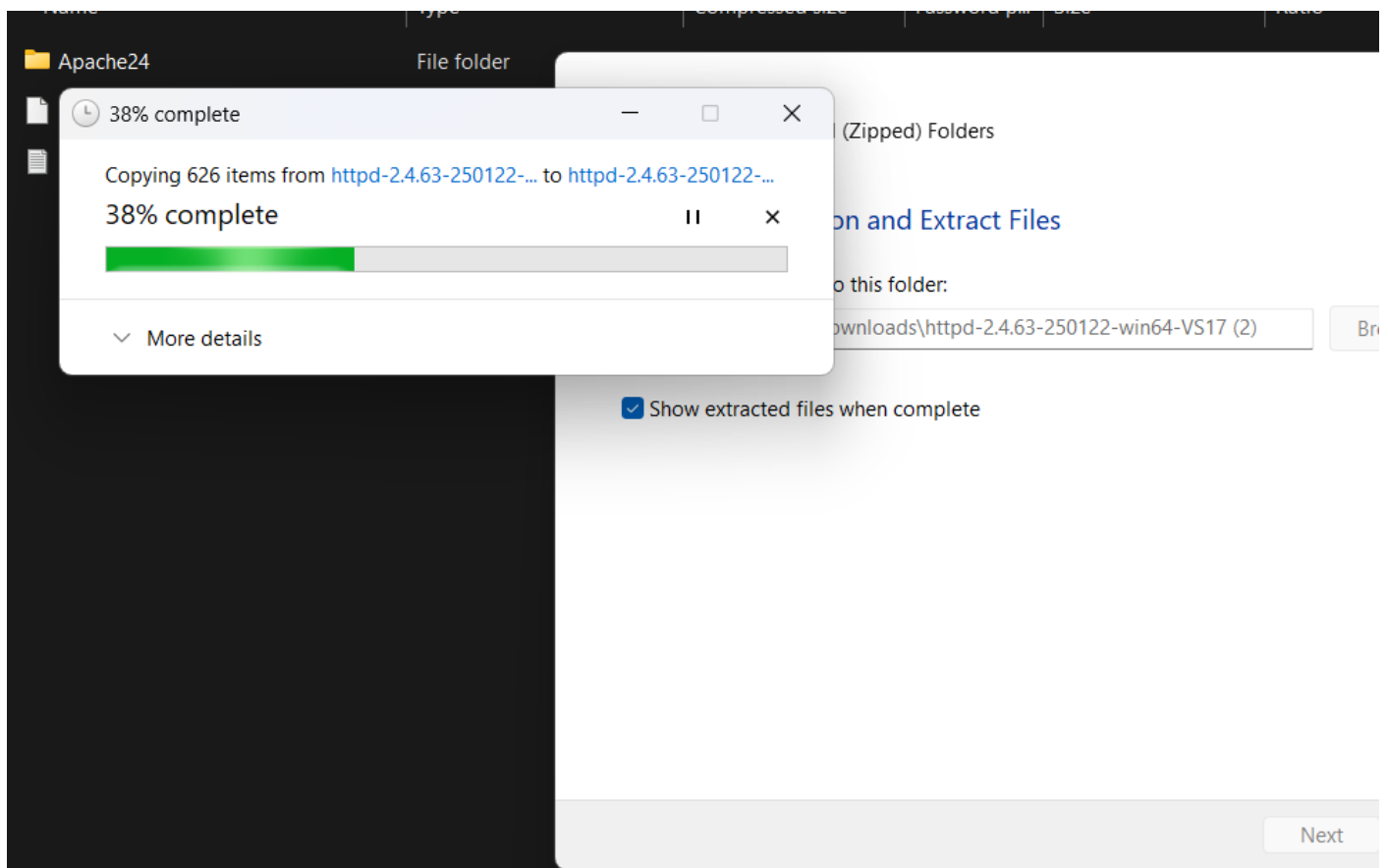
Mail for the PGP signatures and/or SHA checksums to verify the contents of a file.

download/VS17/binaries/httpd-2.4.63-250122-win64-VS17.z...

## Step 4:

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

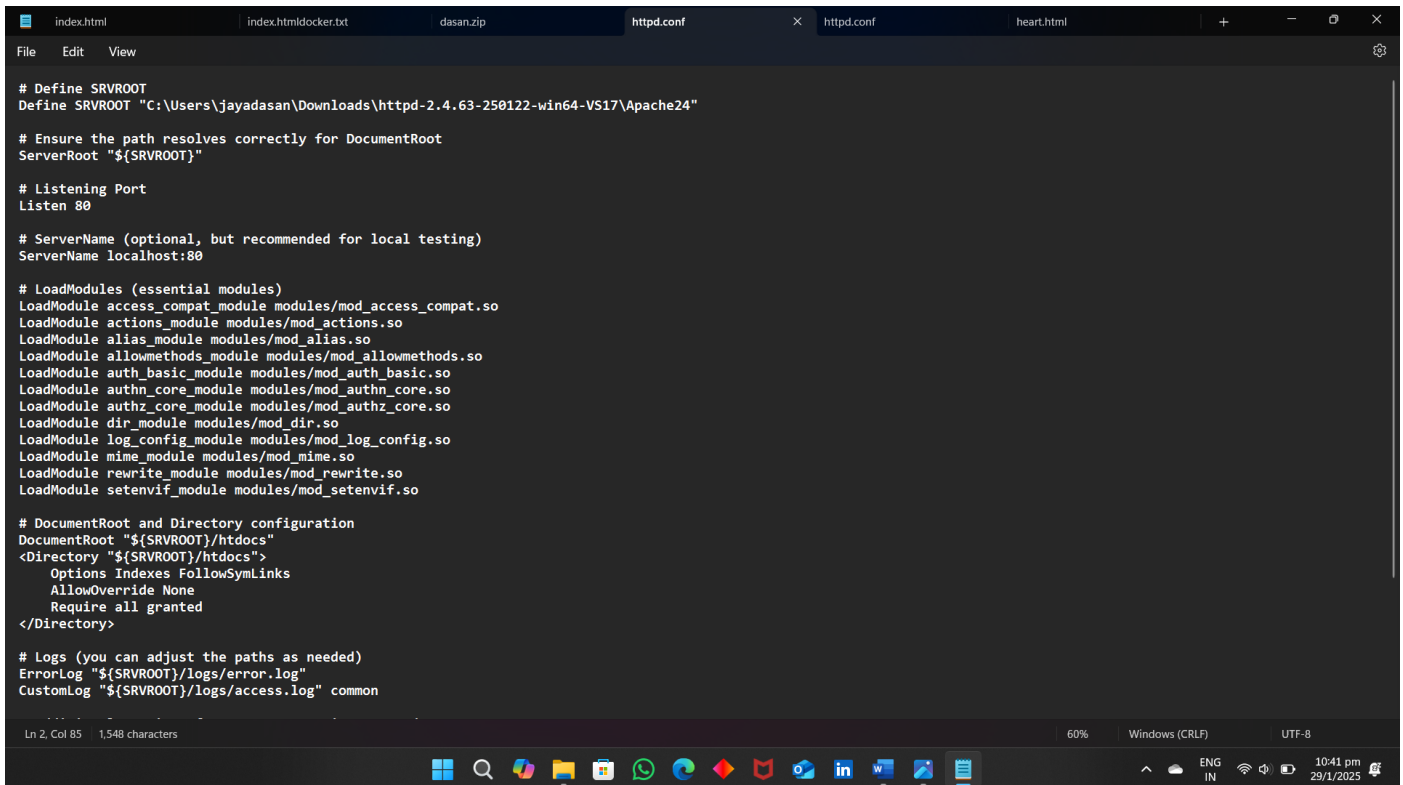
(Apache/conf/httpd.conf)



## Step 5 :

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.

A screenshot of a code editor window with multiple tabs. The active tab is 'httpd.conf'. The editor shows the configuration for the Apache HTTP server. The configuration includes defining the server root, setting the document root to the htdocs directory, loading various modules, and setting up logging. The status bar at the bottom indicates the cursor is at line 2, column 85, and the file contains 1,548 characters.

```
# Define SRVROOT
Define SRVROOT "C:\Users\jayadasan\Downloads\httpd-2.4.63-250122-win64-VS17\Apache24"

# Ensure the path resolves correctly for DocumentRoot
ServerRoot "${SRVROOT}"

# Listening Port
Listen 80

# ServerName (optional, but recommended for local testing)
ServerName localhost:80

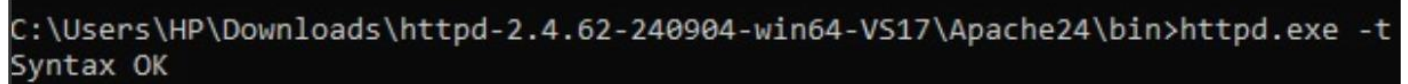
# LoadModules (essential modules)
LoadModule access_compat_module modules/mod_access_compat.so
LoadModule actions_module modules/mod_actions.so
LoadModule alias_module modules/mod_alias.so
LoadModule allowmethods_module modules/mod_allowmethods.so
LoadModule auth_basic_module modules/mod_auth_basic.so
LoadModule authn_core_module modules/mod_authn_core.so
LoadModule authz_core_module modules/mod_authz_core.so
LoadModule dir_module modules/mod_dir.so
LoadModule log_config_module modules/mod_log_config.so
LoadModule mime_module modules/mod_mime.so
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule setenvif_module modules/mod_setenvif.so

# DocumentRoot and Directory configuration
DocumentRoot "${SRVROOT}/htdocs"
<Directory "${SRVROOT}/htdocs">
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>

# Logs (you can adjust the paths as needed)
ErrorLog "${SRVROOT}/logs/error.log"
CustomLog "${SRVROOT}/logs/access.log" common
```

## Step 6 :

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**'.

A screenshot of a Windows Command Prompt window. The command prompt shows the path to the httpd.exe binary and the command to test the configuration. The output of the command is 'Syntax OK'.

```
C:\Users\HP\Downloads\httpd-2.4.62-240904-win64-VS17\Apache24\bin>httpd.exe -t
Syntax OK
```

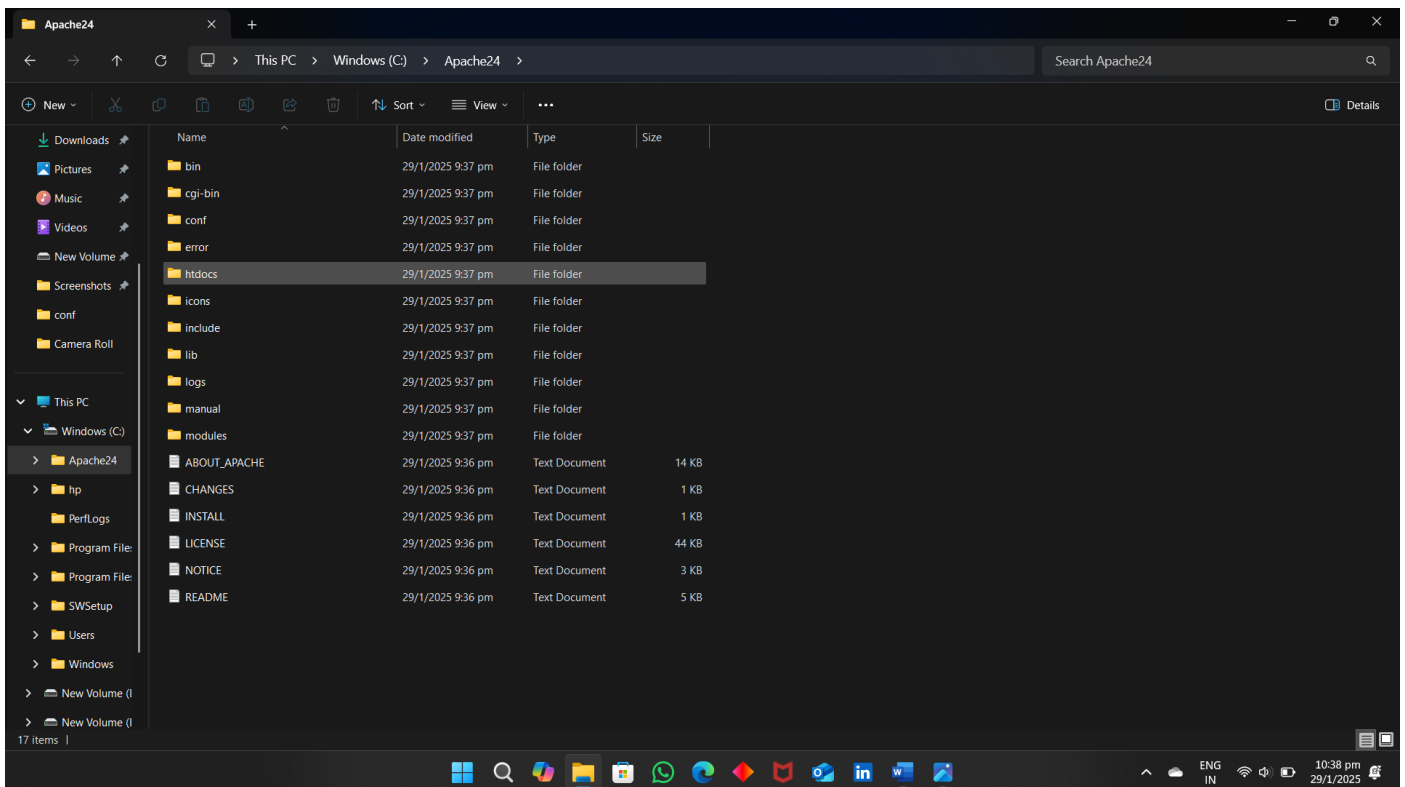
## Step 7 :

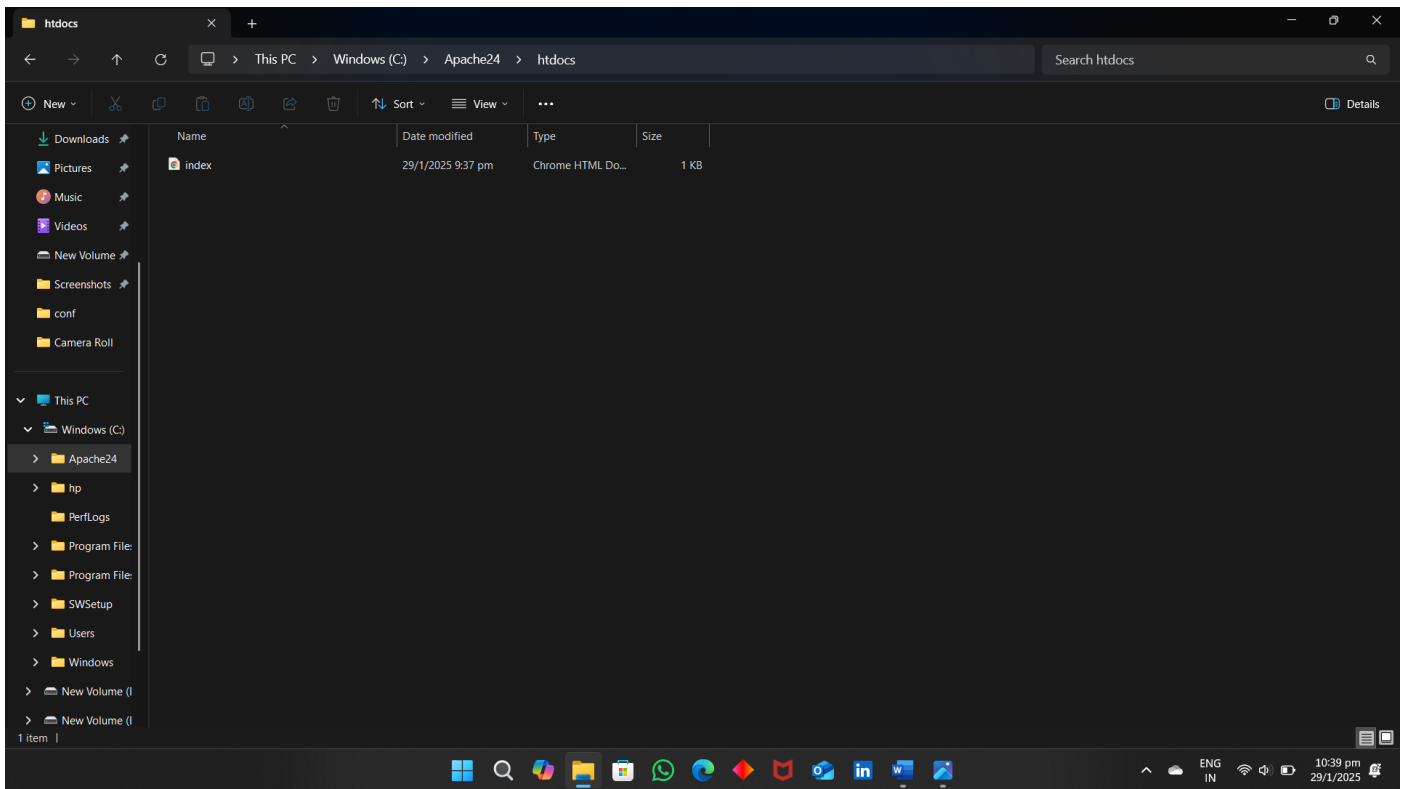
Run the command **httpd.exe -k start** to start the Apache server.

```
C:\Users\HP\Downloads\httpd-2.4.62-240904-win64-VS17\Apache24\bin>httpd.exe -k start
```

## Step8:

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 9 :

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

```
index.html | index.html.docker.txt | lambda_function.zip | dasan.zip | httpd.conf | httpd.conf | !DOCTYPE htr • +
File Edit View
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=
1.0">
  <title>Black Heart Emoji</title>
</head>
<body>
  <p>I'm always gonna love you 💜</p>
</body>
</html>
```

## Step 10 :



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.