**Q-1.** Deploy a website on localhost using either apache2 or Nginx. Create a DNS name for this website as ‘awesomeweb’. You can use any web template you want or can write your own simple HTML code.

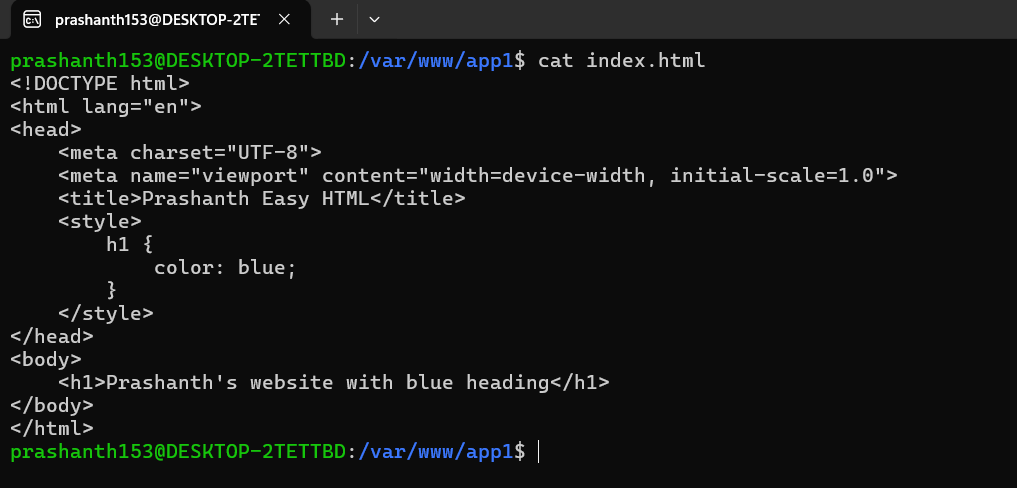
**Write the detailed documentation with the steps involved.**

***Create a index.html***

*This will create a simple HTML with the heading and put it in BLUE colour.*

|  |
| --- |
| *<!DOCTYPE html>*  *<html lang="en">*  *<head>*  *<meta charset="UTF-8">*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<title>Prashanth Easy HTML</title>*  *<style>*  *h1 {*  *color: blue;*  *}*  *</style>*  *</head>*  *<body>*  *<h1>Prashanth's website with blue heading</h1>*  *</body>*  *</html>* |

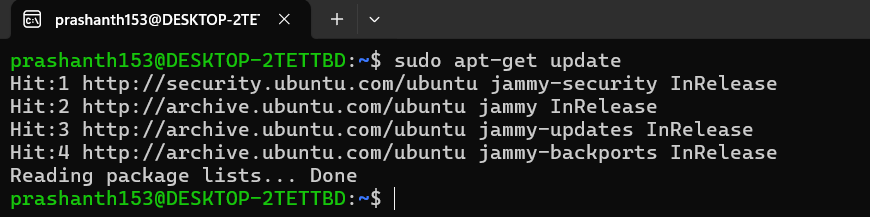
**Created the index.html within app1 folder in /var/www**



**WSL:**

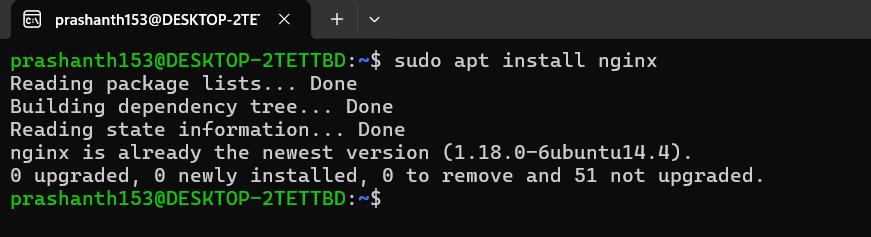
**Execute the following commands**

**sudo apt update:** This tells Linux to to update the list of available packages and their versions but does not install or upgrade any packages.



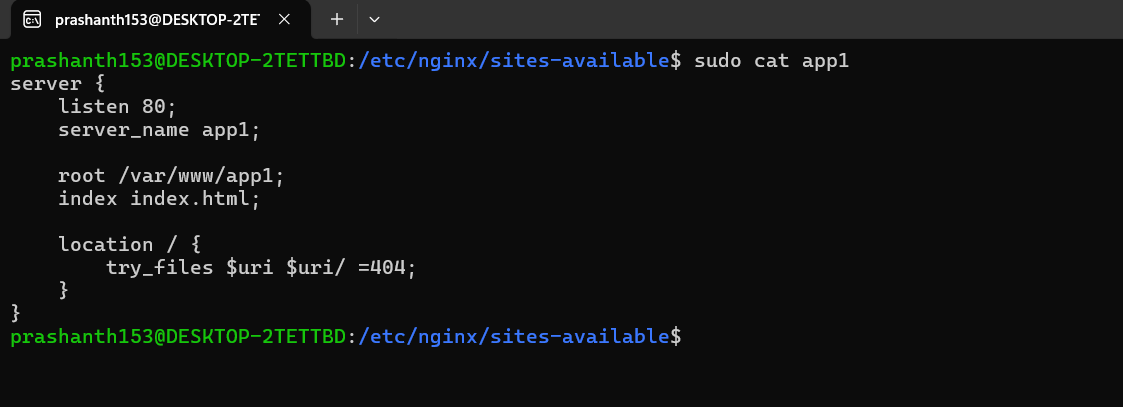
**sudo apt install nginx**

**Since nginx is already installed there was no further message on this**

****

**Create a new nginx configuration file in the sites-available folder:**

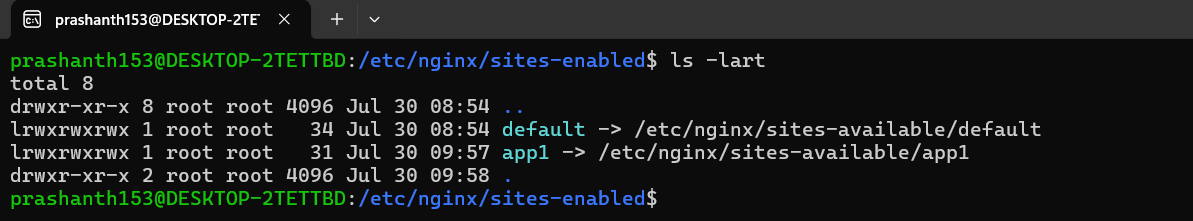
sudo nano /etc/nginx/sites-available/app1



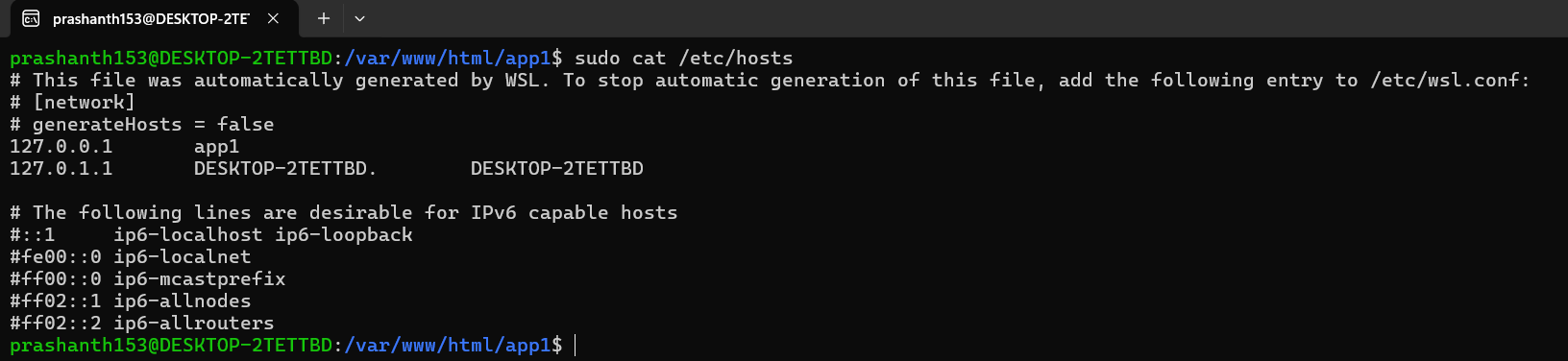
This will make sure when a request comes in with the Host header matching app1, Nginx will use this server block to process the request.

**Creates a symbolic link in the sites-enabled folder**

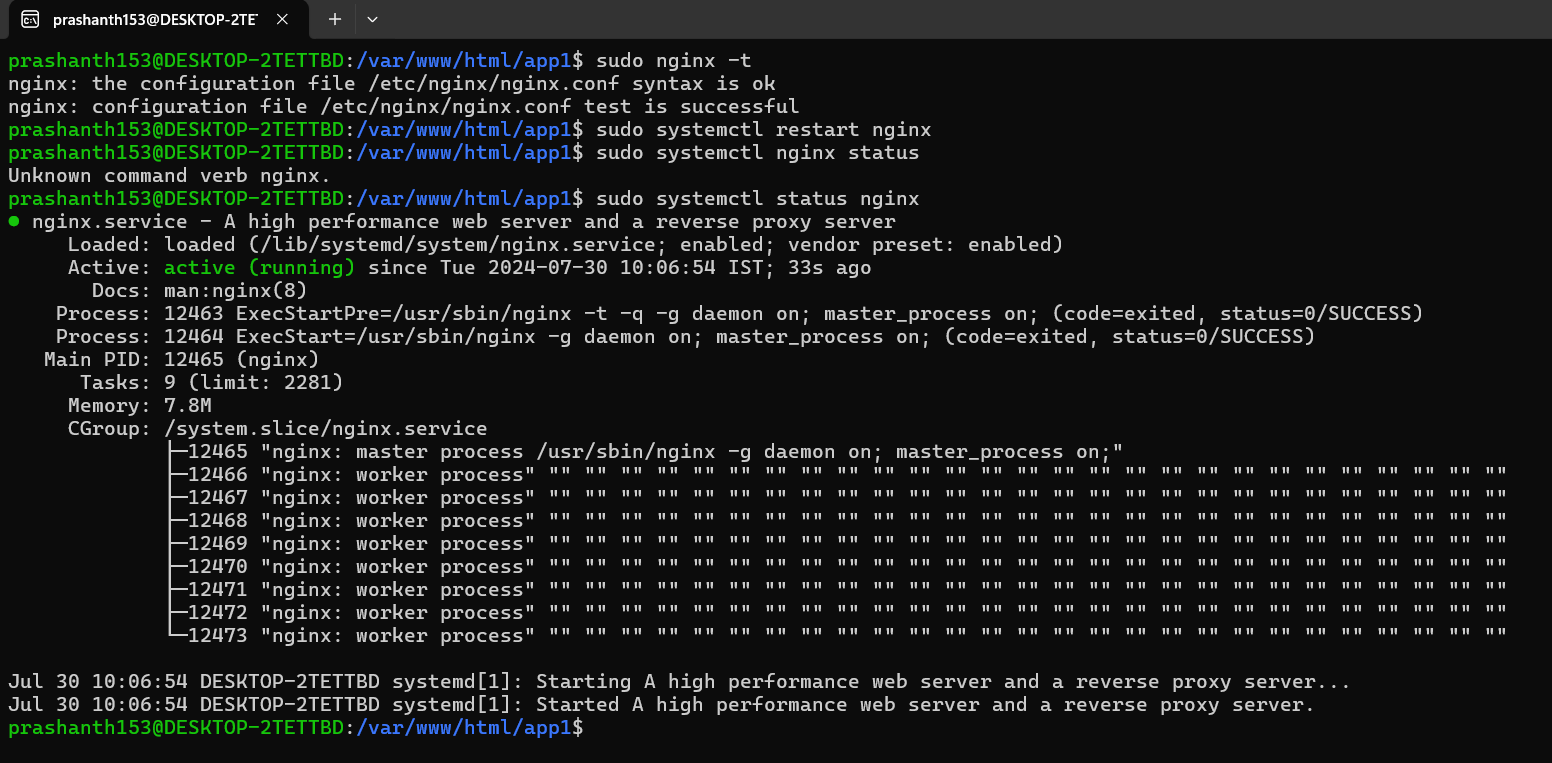
sudo ln -s /etc/nginx/sites-available/app1 /etc/nginx/sites-enabled/



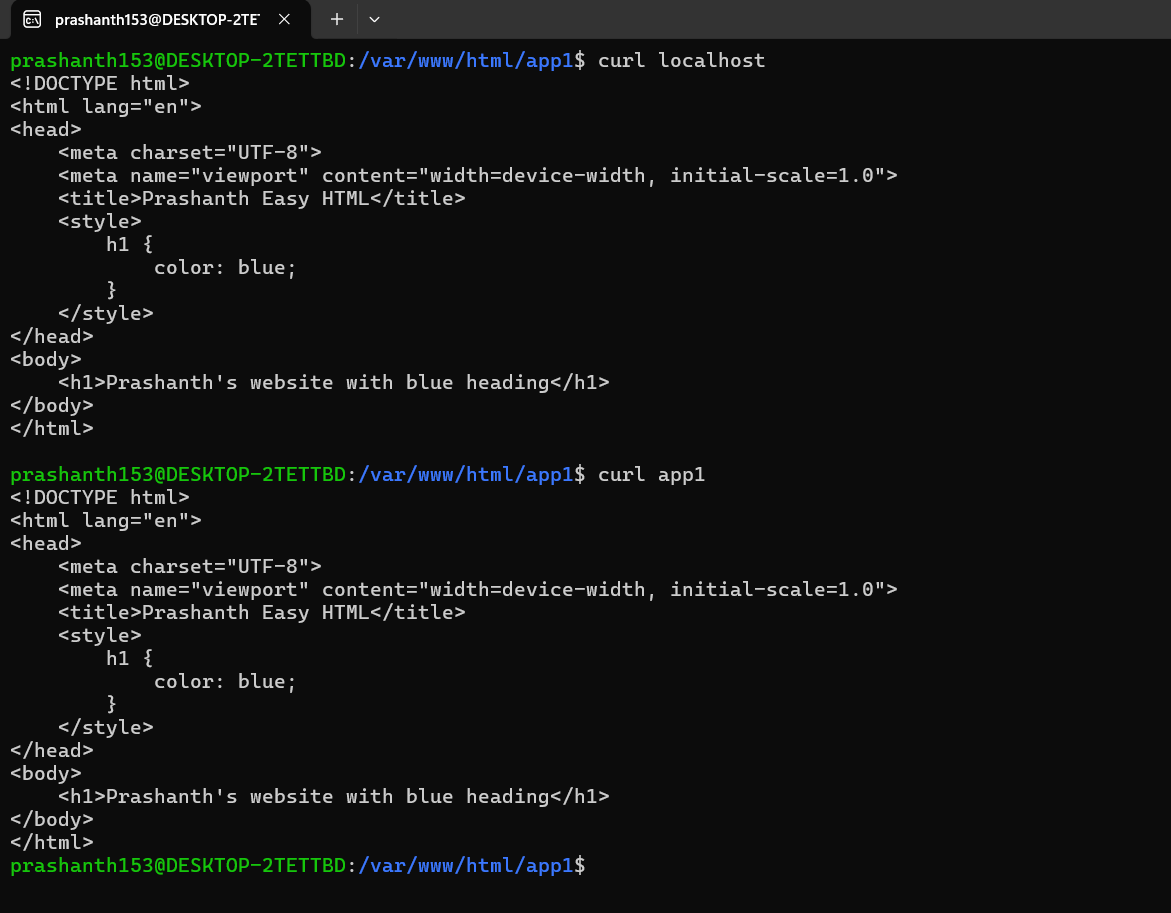
**Update etc/hosts file**



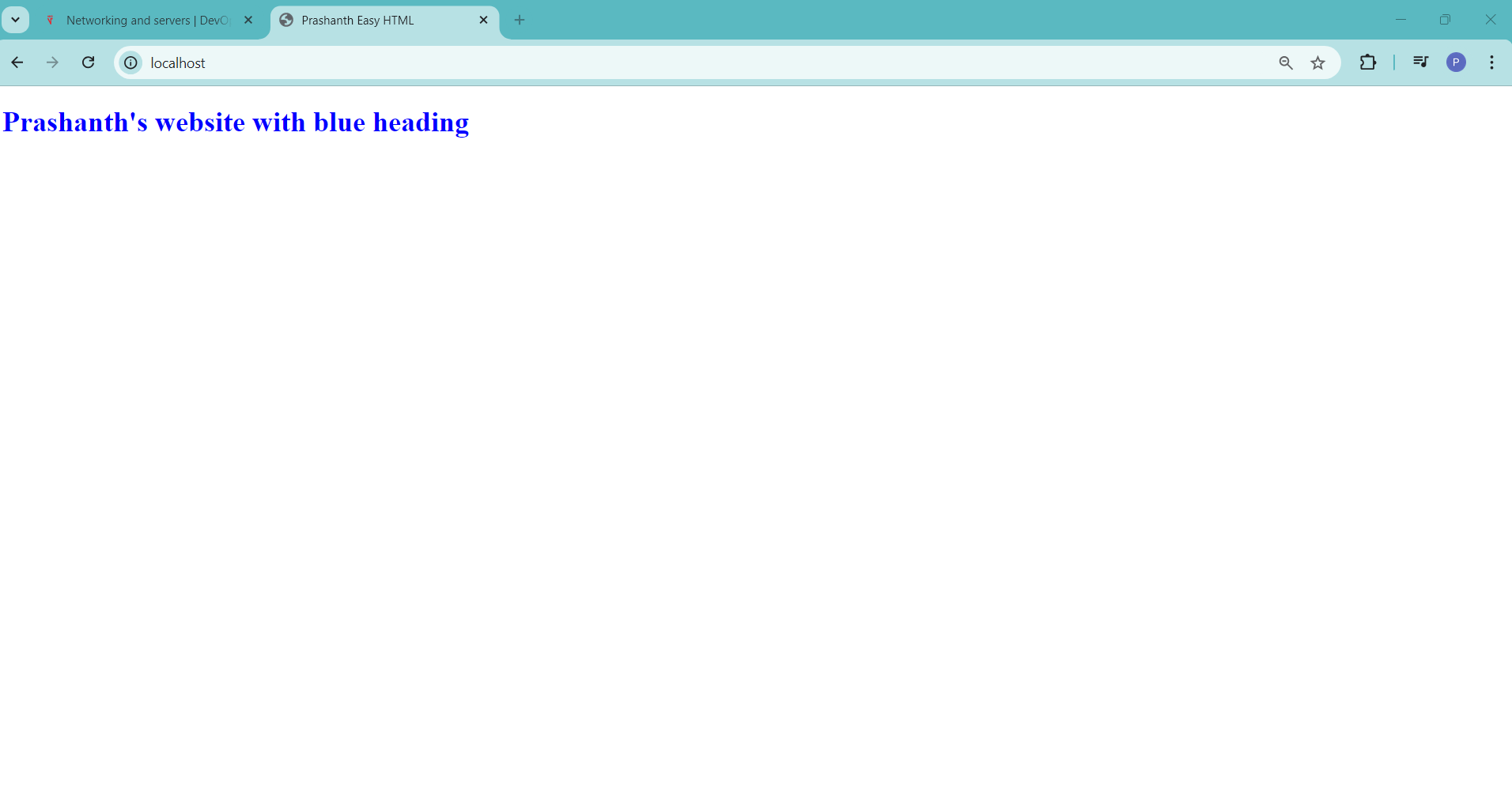
**Test nginx, restart and then check status**

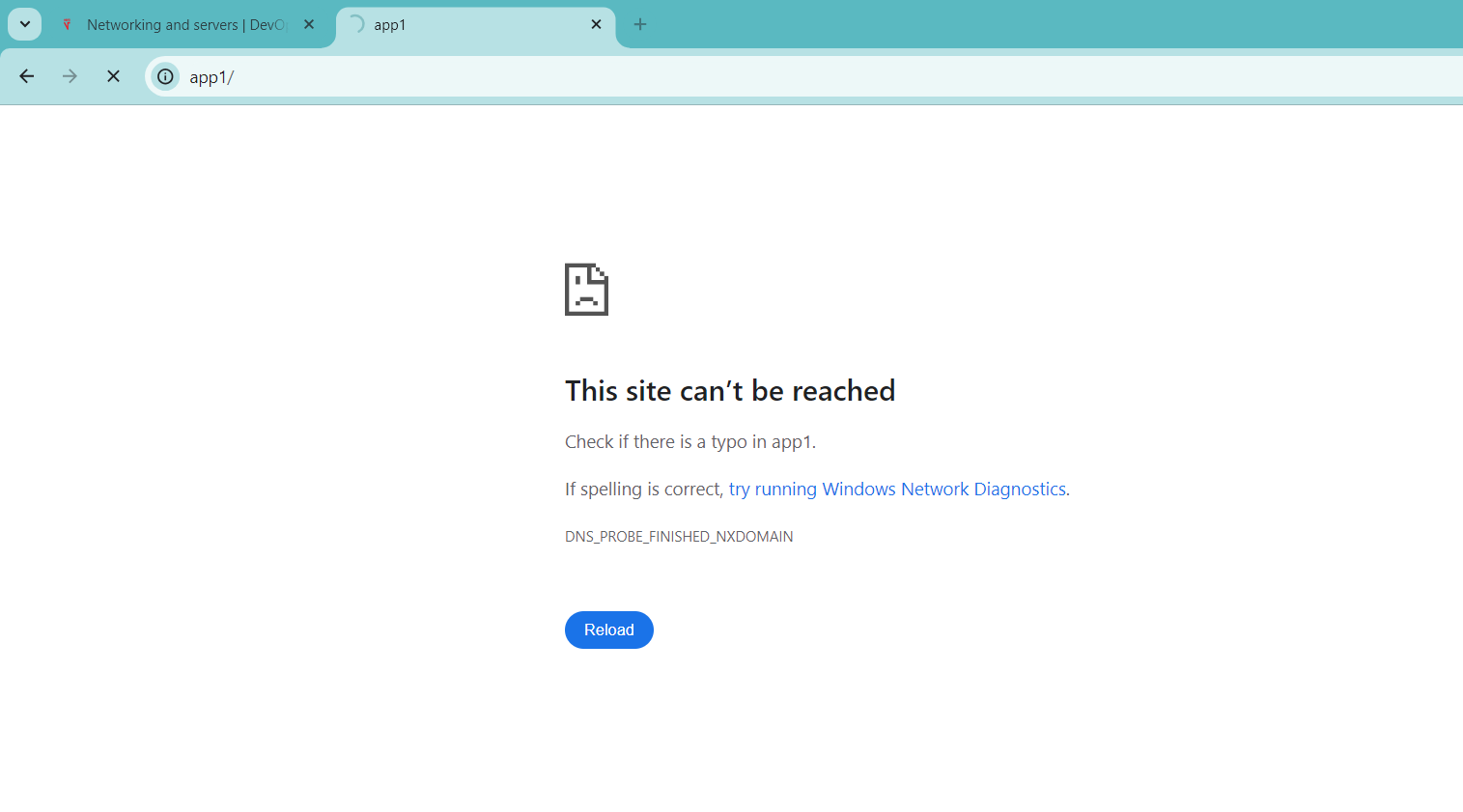


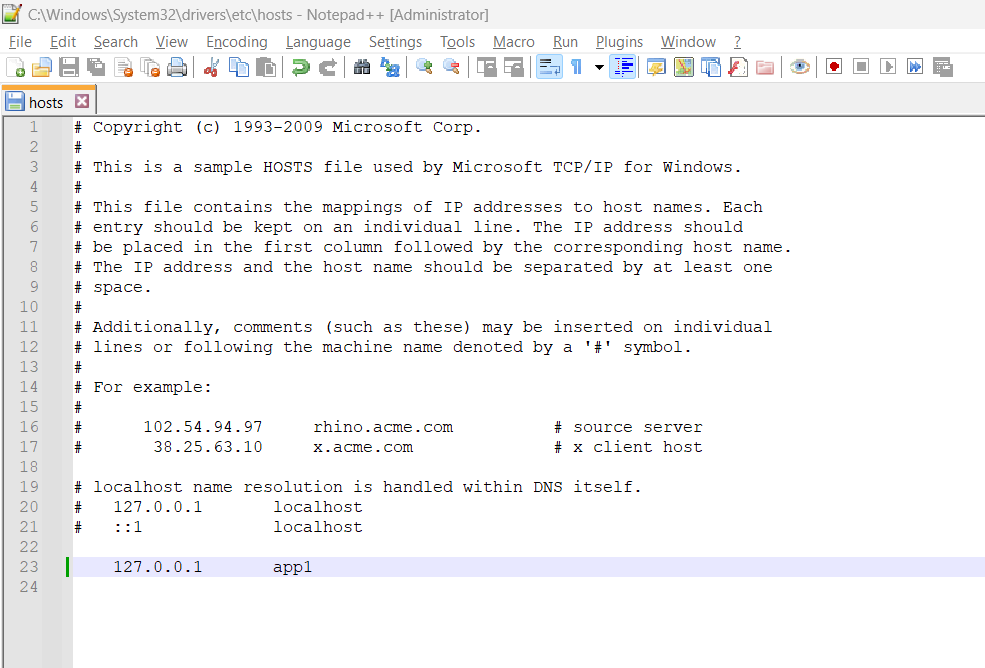
**Verify using curl command for localhost & app1**

****

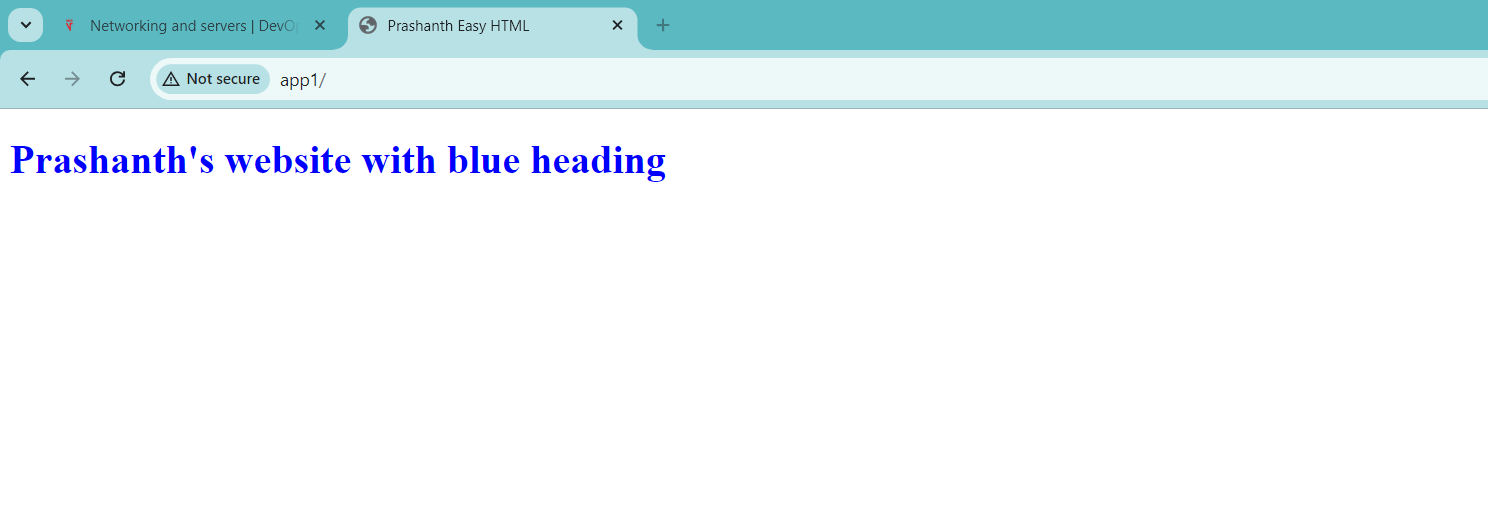
**Try with windows browser localhost and app1**

****

**app1 failed – so I edited the windows config file**

****

**Again I tried now and it worked**

****