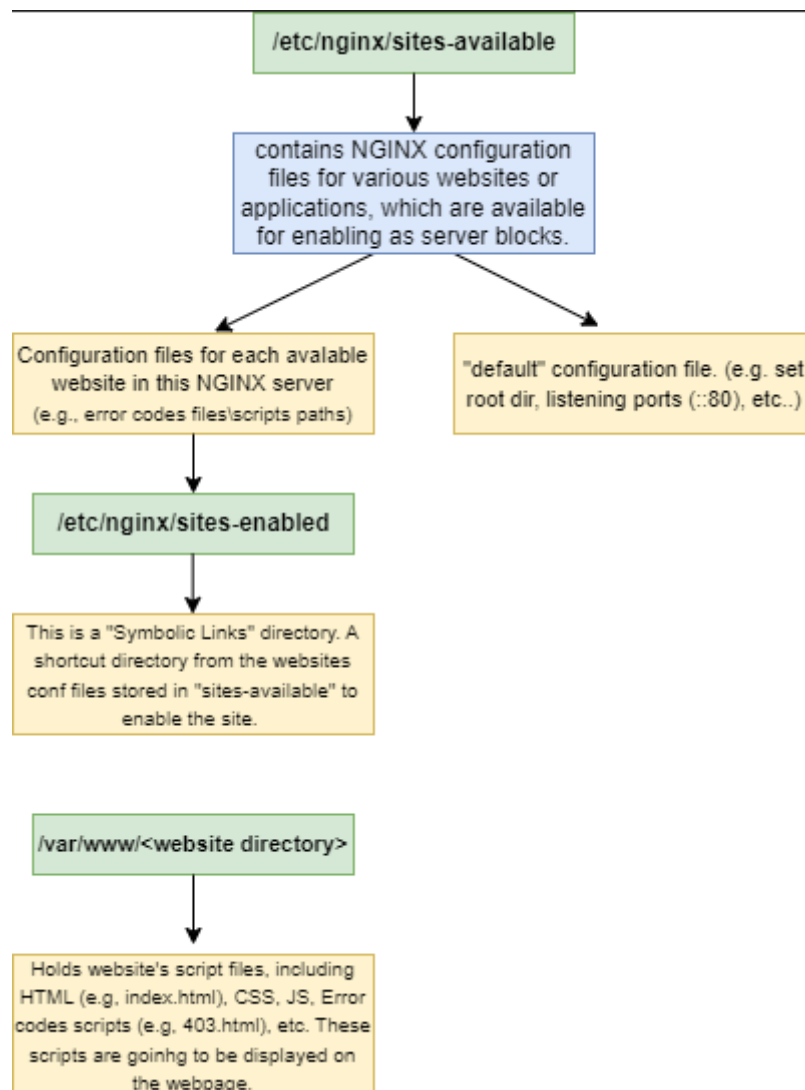


# NGINX Basic Setup



1. Install:  
***sudo apt install nginx -y***
2. Start and enable NGINX service:  
***sudo systemctl start nginx***  
***sudo systemctl enable nginx***
3. Verify installation success. Open web browser and connect to local host:  
[\*\*\*http://your\\_server\\_ip\*\*\*](http://your_server_ip)  
You should see the Nginx welcome page.

# Basic Configuration

Objective: Configure Nginx to serve a static website.

1. Navigate to configuration directory:  
**`cd /etc/nginx/sites-available`**
2. Create a configuration file for your website:  
**`sudo vi /etc/nginx/sites-available/<mywebsite>`**

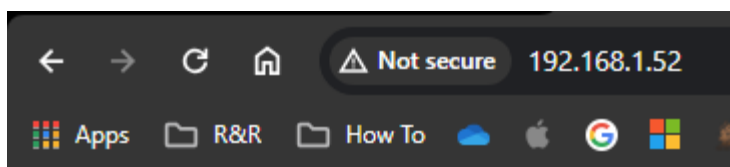
Basic configuration template:

```
server {  
    listen 80;  
    server_name your_domain_or_IP;  
    root /var/www/mywebsite;  
    index index.html;  
    location / {  
        try_files $uri $uri/ =404;  
    }  
}
```

## Description:

- listen 80;; Nginx listens on port 80.
  - server\_name your\_domain\_or\_IP; Replace your\_domain\_or\_IP with your domain name or IP address.
  - root /var/www/mywebsite;; The root directory where the website files are located.
  - index index.html;; The default file to serve.
  - location / { try\_files \$uri \$uri/ =404; }; Try to serve the requested URI, if not found, return a 404 error.
3. Create a Symbolic Link (shortcut) to enable the site:  
**`sudo ln -s /etc/nginx/sites-available/<mywebsite> /etc/nginx/sites-enabled/`**
  4. Create the website directory and a (index) sample file:  
**`sudo mkdir -p /var/www/mywebsite`**

**`echo "<html><body><h1>Hello, World</h1></body></html>" | sudo tee /var/www/mywebsite/index.html`**



# Hello, Nginx

# Website Error handling

This example will demonstrate error code 403, the error code for a forbidden page.

1. Create a "Restricted" directory with a dummy file in it:

```
sudo mkdir /var/www/<mywebsite>/restricted
```

```
sudo touch /var/www/<mywebsite>/restricted/index.html
```

```
sudo echo "This is a restricted area." > /var/www/<mywebsite>/restricted/index.html
```

2. Create an error code script:

```
vi /var/www/<mywebsite>/403.html
```

**Example file content:**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>403 Forbidden</title>
  <style>
    body { font-family: Arial, sans-serif; text-align: center; padding: 50px; }
    h1 { font-size: 50px; }
    p { font-size: 20px; }
  </style>
</head>
<body>
  <h1>403 Forbidden</h1>
  <p>Sorry, you don't have permission to access this page.</p>
  <p><a href="/">Go to Home Page</a></p>
</body>
</html>
```

3. Edit website configuration file:

```
sudo vi /etc/nginx/sites-available/<mywebsite>
```

4. See file example:

```
server {
  listen 80;
  server_name 192.168.1.52; # Server address

  root /var/www/mywebsite; # Path to website web pages (scripts)
  index index.html;

  location / {
    try_files $uri $uri/ =404;
  }
```

```

location /restricted { # the path of the target directory to restrict
    deny all; # Role - Deny all access
}

error_page 403 /403.html; # Error code type / error code script name

location = /403.html { # Location in the website directory
    internal;
}
}

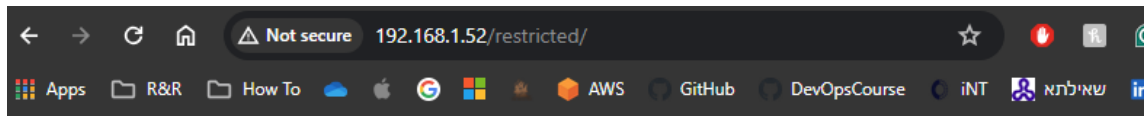
```

- Restart Service and check service:  
***systemctl restart nginx***

***nginx -t***

**Output:**

nginx: the configuration file /etc/nginx/nginx.conf syntax is **ok**  
 nginx: configuration file /etc/nginx/nginx.conf test is **successful**



## 403 Forbidden

Sorry, you don't have permission to access this page.

[Go to Home Page](#)

## Adding a second web application

- Create a second configuration file:  
***sudo nano /etc/nginx/sites-available/<mysecondwebsite>***
- File content:

```

server {
    listen 8080;

```

```
server_name 192.168.1.52;

root /var/www/mysecondwebsite;
index index.html;

location / {
    try_files $uri $uri/ =404;
}
}
```

**NOTE!** Unlike the previous website's configuration file which listens to port 80, this configuration file listens to port 8080

3. Create a Symbolic Link (shortcut) to enable the site:  
***sudo ln -s /etc/nginx/sites-available/<mysecondwebsite> /etc/nginx/sites-enabled/***

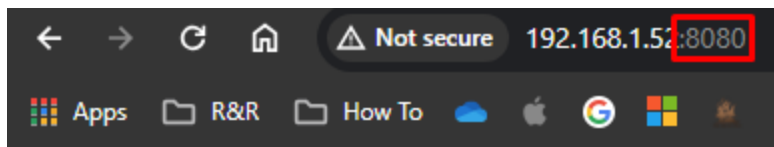
4. Create the website directory and a (index) sample file:  
***sudo mkdir -p /var/www/<mysecondwebsite>***

```
echo "<html><body><h1>Hello, World</h1></body></html>" | sudo tee
/var/www/<myseconwebsite>/index.html
```

5. Check and restart service:

```
sudo nginx -t
```

```
sudo systemctl restart nginx
```



# Hello, my second website