Debugging NGINX errors:
To check what is causing the errors:

```
docker exec -it <container_id> cat /var/log/nginx/error.log
```

or

```
docker logs <container_id>
```

Ensure that the files inside your container are readable by Nginx:

```
sudo chmod -R 755 /path/to/your/static/files
sudo chown -R www-data:www-data /path/to/your/static/files
```

If you're serving static files, make sure your `nginx.conf` has the correct permissions:

```
server {
    listen 80;
    server_name localhost;

    location / {
        root /usr/share/nginx/html;
        index index.html;
        autoindex on;
    }

    location /static/ {
        root /app;
    }

    error_log /var/log/nginx/error.log warn;
    access_log /var/log/nginx/access.log;
}
```

restart the container

```
docker restart <container_id>
```

AWS Public IP has errors or page does not respond: Retry or Reload 3-4 times, some times it does not work on the first try so wait for a couple of minutes

AWS CLI ISSUES:
Make sure you have installed and configured CLI as per the instructions
When you create an IAM user and access key, make sure you attach full access policies to the user.

```
aws configure
```

Enter:

- **AWS Access Key ID**

- **AWS Secret Access Key**
- **Default region** (e.g., `us-east-1`)
- **Output format** (leave blank for default JSON

authenticate Docker with ECR
aws ecr get-login-password --region <your-region> | docker login --username AWS --password-stdin <aws_account_id>.dkr.ecr.<your-region>.amazonaws.com

Example:
aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 123456789012.dkr.ecr.us-east-1.amazonaws.com

Build and tag your repo
docker build -t my-repo .

docker tag my-repo:latest <aws_account_id>.dkr.ecr.<your-region>.amazonaws.com/my-repo:latest

Push the repo
docker push <aws_account_id>.dkr.ecr.<your-region>.amazonaws.com/my-repo:latest

COMMON WSL AND DOCKER ISSUES: (Follow the video in earlier announcement)

1. Install WSL 2

Ensure WSL 2 is installed and set as the default version:

1. Enable WSL and Virtual Machine Platform:

Unset
```
wsl --install
wsl --set-default-version 2
```

2. Install a Linux distribution (e.g., Ubuntu) from the Microsoft Store.

2. Install Docker Desktop

- During installation:
    - Select **WSL 2** as the backend.
    - Enable integration with your WSL distribution (e.g., Ubuntu).

3. Verify Docker Installation

Open your WSL terminal (e.g., Ubuntu) and run:

Unset

```
docker --version
docker run hello-world
```

Error 1: Docker Daemon Not Running

Symptoms: Running `docker` commands results in errors like:

Unset

```
Cannot connect to the Docker daemon at
unix:///var/run/docker.sock. Is the docker daemon running?
```

Fixes:

1. **Start Docker Desktop**: Ensure Docker Desktop is running on Windows. Docker in WSL relies on Docker Desktop as the daemon.
2. **Restart Docker**:
   - Restart Docker Desktop from the system tray.
   - Or use PowerShell:

Unset

```
Restart-Service com.docker.service
```

3. **Check Docker Service in WSL**: Ensure Docker service is running in WSL:

Unset

```
sudo service docker start
```

Error 2: WSL 2 is Not Enabled

Symptoms: Errors like:

Unset

```
WSL 2 requires an update to its kernel component.
```

Fixes:

1. **Install WSL 2 Kernel Update**: Download and install the kernel update from the Microsoft WSL 2 Update page.
2. **Set Default WSL Version**:

Unset

```
wsl --set-default-version 2
```

Error 3: Docker Not Recognized in WSL

Symptoms: Running `docker` results in:

Unset

```
Command 'docker' not found
```

Fixes:

1. **Check Integration**:
   - Open Docker Desktop settings.
   - Go to **Settings > Resources > WSL Integration**.
   - Ensure your distribution (e.g., Ubuntu) is checked.
2. **Restart WSL**:

Unset

```
wsl --shutdown
```

3. Then reopen your WSL terminal.

Error 4: Network Issues

Symptoms: Errors pulling images or accessing Docker Hub:

Unset

```
Network timed out while trying to connect to Docker Hub.
```

Fixes:

1. **Check Network Proxy**: If behind a proxy, configure Docker with proxy settings in Docker Desktop under **Settings > Resources > Proxies**.
2. **Reset Network**: Reset Docker's network settings via **Docker Desktop > Troubleshoot > Reset to factory defaults**.

Error 5: Volume Mounting Issues

Symptoms: Error messages like:

Unset

```
Mounts denied: The path <path> is not shared from Docker Desktop
```

Fixes:

1. **Share Drives in Docker Desktop**: Go to **Settings > Resources > File Sharing** and add the desired path.
2. **Check Permissions**: Ensure the WSL user has proper permissions for the directory:

Unset

```
sudo chmod -R 755 /path/to/directory
```

Error 6: Incompatible Docker Version

Symptoms: Errors related to Docker versions or features not available.

Fixes:

1. **Update Docker Desktop**: Ensure you are using the latest version of Docker Desktop.
2. **Check WSL Version**: Ensure WSL is set to version 2:

Unset

```
wsl -l -v
```

3. If your distribution is not version 2, upgrade it:

Unset

```
wsl --set-version <distro-name> 2
```
