But first —
# 1. Allow SSH first (to avoid locking yourself out)
sudo ufw allow OpenSSH
# 2. Enable the firewall
sudo ufw enable
# 3. Check status
sudo ufw status verbose

✅ COMPLETE WEBSITE SETUP CHECKLIST

🔷 PHASE 1: 🖥️ Prepare Ubuntu Server

Step 1: Connect to your server

bash Copy Edit

```
ssh root@your_server_ip
```

Step 2: Update packages

bash Copy Edit

```
sudo apt update && sudo apt upgrade -y
```

Step 3: Install Nginx

bash Copy Edit

```
sudo apt install nginx -y
```

Step 4: Enable UFW and allow Nginx

bash Copy Edit

```
sudo ufw allow 'Nginx Full'
sudo ufw enable
sudo ufw status
```

✅ This opens ports **80** and **443** in your Ubuntu **internal firewall**.

🔷 PHASE 2: 🔥 Open Ports in Cloud Firewall

> ⚠️ Without this, external users **cannot access** your server.

Step 5: Go to your cloud provider firewall panel (example: DigitalOcean)

- Navigate to your droplet's **Firewall Settings**.
- Under **Inbound Rules**, add:

Protocol	Port	Source
TCP	22	Your IP or 0.0.0.0/0
TCP	80	0.0.0.0/0
TCP	443	0.0.0.0/0

- Attach this firewall to your Droplet and **save**.

Phase 3:

- **Links your domain**
- **Enables HTTPS via Cloudflare**
- **Controls how Cloudflare interacts with your Nginx server**

Without this phase, even if Nginx is working perfectly:

- Users can't reach your website by domain name
- Or they may get a security warning or timeout

🔷 PHASE 3: 🌐 Configure Domain in Cloudflare

Step 6: Add your domain to Cloudflare

Step 7: Point A record

- Go to **DNS tab**
- Add a record:

Type	Name	Content (your server IP)	Proxy
A	bash	157.230.xxx.xxx	☁ Proxied

✅ This maps `bash.english-welsh.net` to your server.

🔷 PHASE 4: ⚙️ Create Nginx Site Config

Saved memory full

Step 8: Create config file

bash Copy Edit

```
sudo nano /etc/nginx/sites-available/bash.english-welsh.net
```

Step 9: Paste this configuration

nginx Copy Edit

```
server {
    listen 80;
    listen [::]:80;
    server_name bash.english-welsh.net;

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

    if ($http_x_forwarded_proto = 'http') {
        return 301 https://$host$request_uri;
    }

    gzip on;
    gzip_types text/plain text/css application/javascript application/json text/javascript;
    gzip_min_length 1024;
    gzip_comp_level 6;
    gzip_proxied any;
    gzip_vary on;

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

    location ~ ^/english-to-(\w+)-meaning-(.+)$ {
        rewrite ^/english-to-(\w+)-meaning-(.+)$ /cache/meaning-of-$2.html break;
    }

    location = /favicon.ico {
        try_files /favicon.ico =204;
    }
}
```

Then save (`Ctrl + X` , then `Y` , then `Enter`)

🔷 PHASE 5: 🟢 Enable the Site

Step 10: Enable the site

bash Copy Edit

```
sudo ln -s /etc/nginx/sites-available/bash.english-welsh.net /etc/nginx/sites-enabled/
```

Step 11: Remove default config (optional)

bash Copy Edit

```
sudo rm /etc/nginx/sites-enabled/default
```

Step 12: Test config

bash Copy Edit

```
sudo nginx -t
```

✅ It must return: "syntax is ok" and "test is successful"

Step 13: Reload Nginx

bash Copy Edit

```
sudo systemctl reload nginx
```

🔷 PHASE 6: ✅ Verify Nginx is Working

Step 14: Test locally

bash Copy Edit

```
curl -I http://localhost
```

You should see `301 Moved Permanently` or `200 OK`

🔷 PHASE 7: 🤗 Optional – Add SSL (Certbot)

Only if you want HTTPS from your **own cert** (not just Cloudflare Flexible SSL)

Step 15: Install Certbot

bash Copy Edit

```
sudo apt install certbot python3-certbot-nginx -y
```

Step 16: Issue cert

bash Copy Edit

```
sudo certbot --nginx -d bash.english-welsh.net
```

Step 17: Set Cloudflare SSL mode

- Go to **Cloudflare → SSL/TLS**
- Set SSL Mode to: **Full (Strict)**

If some time curl is not working or any problem occurs..then **to check using a normal index.html file**

ls -l /var/www/html
Add this,
echo "<h1>Hello from bash.english-welsh.net</h1>" | sudo tee /var/www/html/index.html

Set permission
sudo chown -R www-data:www-data /var/www/html

sudo chmod -R 755 /var/www/html

Configure

sudo nano /etc/nginx/sites-available/default
root /var/www/html;
index index.html index.htm;

Restart

sudo nginx -t
sudo systemctl reload nginx

Retest

curl -I http://159.203.181.96

Now you should see:

```
HTTP/1.1 200 OK
And in browser… http://159.203.181.96/
```

Hello from bash.english-welsh.net

Step 18: From your **local computer**: aved memory full ⓘ

bash Copy Edit

```
curl -I http://bash.english-welsh.net
```

You should get:

arduino Copy Edit

```
HTTP/1.1 301 Moved Permanently
Location: https://bash.english-welsh.net/
```

Then test HTTPS:

bash Copy Edit

```
curl -I https://bash.english-welsh.net
```

And open in browser:

arduino Copy Edit

```
https://bash.english-welsh.net
```

✅✅ Deployment Complete

You now have:

- Nginx serving your site
- DNS configured via Cloudflare
- Firewall open on ports 80 + 443 (cloud + ufw)
- Gzip compression enabled
- Optional SSL (Let's Encrypt or Cloudflare Universal SSL)