

Nginx Cheat Sheet

♦ Install & Service Management

```
sudo apt update
```

```
sudo apt install nginx -y      # Install Nginx (Debian/Ubuntu)
```

```
sudo systemctl start nginx     # Start Nginx
```

```
sudo systemctl stop nginx      # Stop Nginx
```

```
sudo systemctl restart nginx   # Restart Nginx
```

```
sudo systemctl status nginx    # Check status
```

Default paths:

- Config: `/etc/nginx/nginx.conf`
- Site configs: `/etc/nginx/sites-available/`
- Enabled sites: `/etc/nginx/sites-enabled/`
- Default root: `/var/www/html`

♦ Basic Commands

```
nginx -t      # Test config syntax
```

```
nginx -s reload  # Reload Nginx
```

```
nginx -v      # Show version
```

◆ Basic Web Server

Default server block (`/etc/nginx/sites-available/default`)

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

Enable site:

```
sudo ln -s /etc/nginx/sites-available/default  
/etc/nginx/sites-enabled/  
  
sudo systemctl reload nginx
```

◆ Reverse Proxy

```
server {  
  
    listen 80;
```

```
server_name myapp.com;

location / {

    proxy_pass http://127.0.0.1:5000;    # Flask app

    proxy_set_header Host $host;

    proxy_set_header X-Real-IP $remote_addr;

}

}
```

◆ Load Balancing

```
upstream backend {

    server 127.0.0.1:5000;

    server 127.0.0.1:5001;

}

server {

    listen 80;

    server_name myapp.com;

    location / {

        proxy_pass http://backend;

    }

}
```

```
}
```

◆ SSL (with Let's Encrypt)

Install certbot:

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

Obtain certificate:

```
sudo certbot --nginx -d myapp.com -d www.myapp.com
```

◆ Useful Logs

```
tail -f /var/log/nginx/access.log    # Requests log
```

```
tail -f /var/log/nginx/error.log     # Errors log
```