Hosting a Website with a Custom Domain Using Termux

# 1. Get a Domain Name

- Buy a Domain: You can purchase a domain name from services like Namecheap, GoDaddy, or any domain registrar of your choice.  
- Get Free Domain: If you don’t want to pay for a domain, you can get free domains from services like Freenom.

# 2. Set Up Termux for Web Hosting

- Update and Upgrade Termux:  
First, make sure Termux is up to date:  
```  
pkg update && pkg upgrade  
```  
- Install a Web Server (Nginx):  
Install a lightweight web server like Nginx:  
```  
pkg install nginx  
```  
- Start Nginx:  
After installation, start the Nginx web server:  
```  
nginx  
```  
- Test Nginx:  
Open your browser and go to `http://localhost:8080` or your local IP address (`http://192.168.x.x:8080`) to check if the Nginx default page is working.  
- Set up Your Web Files:  
Your website files (HTML, CSS, JavaScript, etc.) can be placed in:  
```  
/data/data/com.termux/files/usr/share/nginx/html/  
```

# 3. Get a Static IP Address or Dynamic DNS

If you want to host a website that's accessible globally (not just on your local network), you’ll need a public IP address.  
  
- Use Dynamic DNS (DDNS): Services like No-IP or Dynu give you a domain that will update whenever your home IP address changes.  
- Get a Static IP from Your ISP: You can contact your Internet Service Provider (ISP) and ask for a static IP. This might involve additional costs.

# 4. Port Forwarding on Your Router

You need to forward ports from your router to your phone (Termux) to allow external access.  
  
- Log in to your router's admin panel (usually accessible via `192.168.1.1` or `192.168.0.1`).  
- Find the Port Forwarding section and forward HTTP (port 80) and HTTPS (port 443) traffic to the internal IP address of your Android device (where Termux is running).  
- For example, forward external port 80 to internal port `8080` (where Nginx runs on Termux).

# 5. Set Up DNS for Your Custom Domain

Once you've got your public IP (or dynamic DNS domain) and port forwarding set up, configure DNS settings for your domain:  
  
- Log in to Your Domain Registrar's Dashboard (e.g., Namecheap, GoDaddy).  
- Find the DNS Settings or Manage DNS section.  
- Add an A Record for your domain:  
 - Name: `@`  
 - Type: `A`  
 - Value: Your public IP address.  
 - TTL: Set it to a low value like 300.  
- If you're using a Dynamic DNS service, set a CNAME Record pointing to your Dynamic DNS domain instead of your public IP.

# 6. SSL (Optional but Recommended)

To serve your site over HTTPS, you should set up SSL. You can use Let's Encrypt for free SSL certificates.  
  
- Use ZeroSSL to generate an SSL certificate for free.  
- Once you have the SSL certificate, configure Nginx to serve your site with HTTPS by editing its configuration file.

# 7. Run Nginx in the Background

To ensure Nginx keeps running even if you close Termux, you can use tmux:  
```  
pkg install tmux  
tmux  
nginx  
```

# 8. Access Your Website via the Domain

After everything is set up, you should be able to access your website using your custom domain. It may take a few minutes for DNS changes to propagate.