ENRIQUE CRESPO RAMIREZ

Práctica 1: Apache + PHP y Nginx en WSL

Objetivo

Configurar en **WSL (Ubuntu)**:

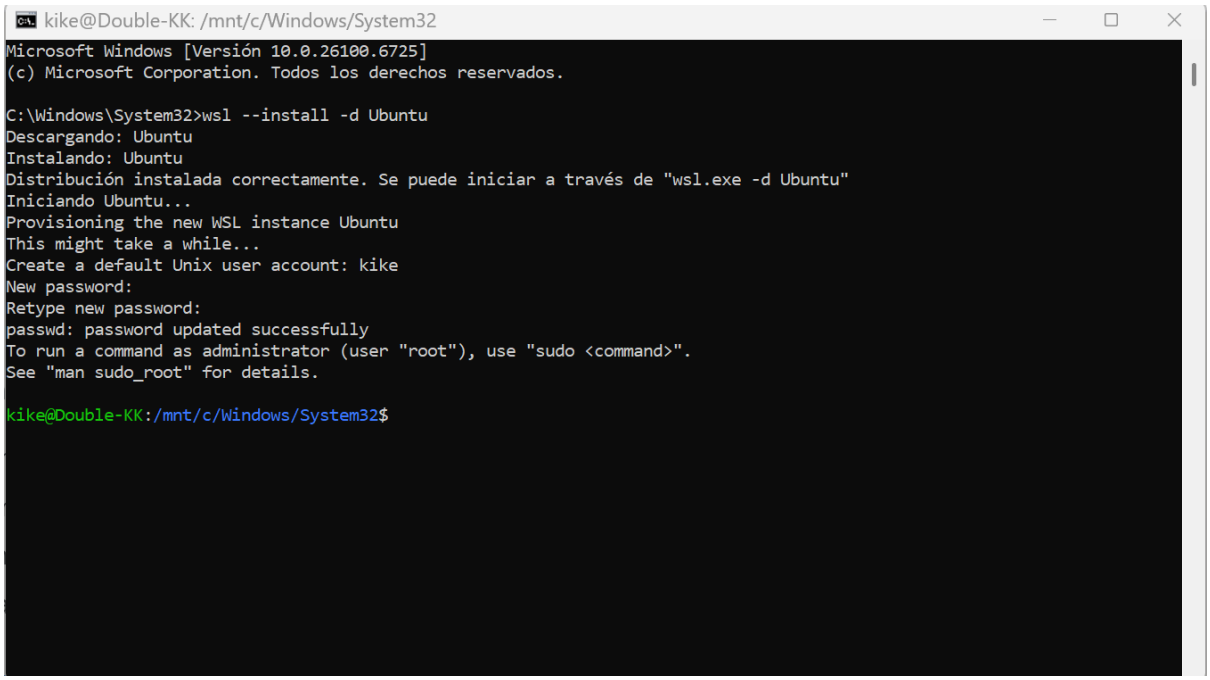
1. Un servidor **Apache** que sirva **PHP (info.php)**.
2. Un servidor **Nginx** que sirva una **página HTML**.
3. Verificar, comparar y documentar con **capturas**.

Entorno

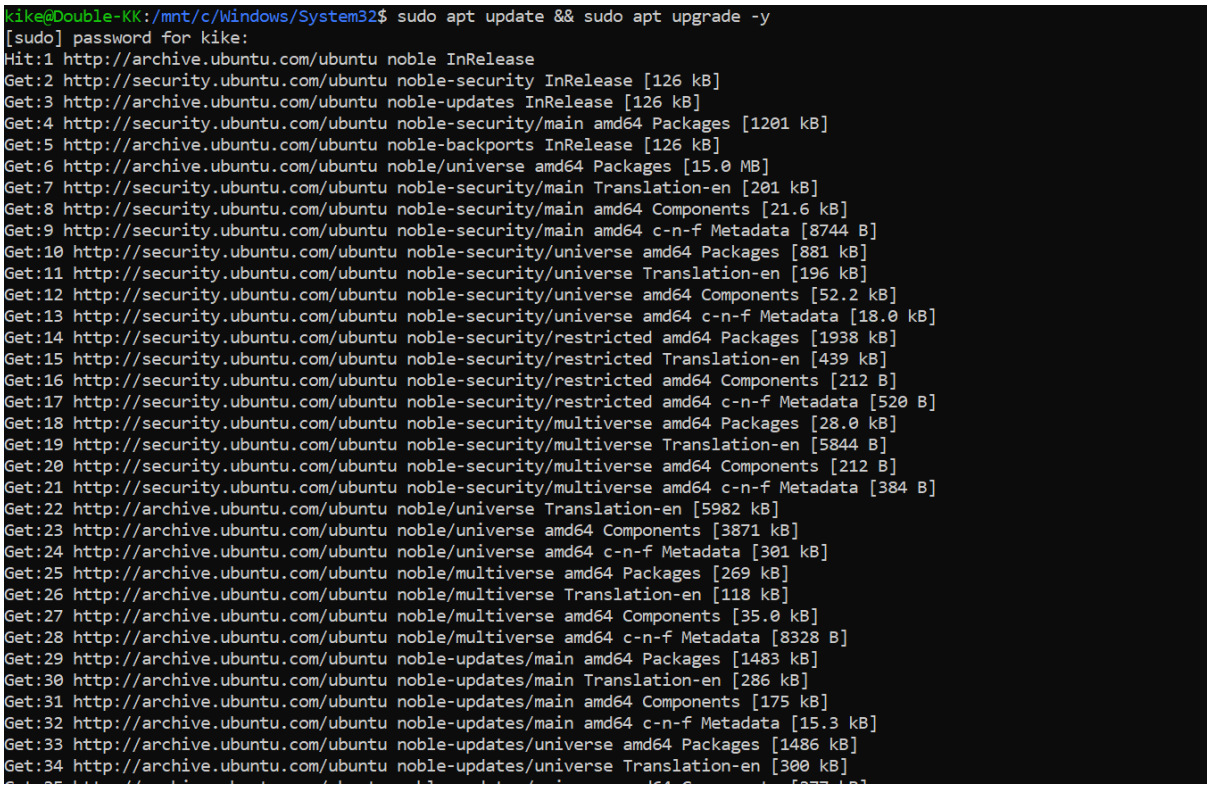
- **Windows 10/11** con **WSL 2** y **Ubuntu** instalado.
- Terminal: Ubuntu en WSL.

1 Instalación de WSL

con comando wsl --install -d Ubuntu



`apt update && apt upgrade` pone al día índices y paquetes, evitando errores de dependencias.



sudo apt install -y curl lsb-release ca-certificates curl te permite probar desde terminal las páginas servidas

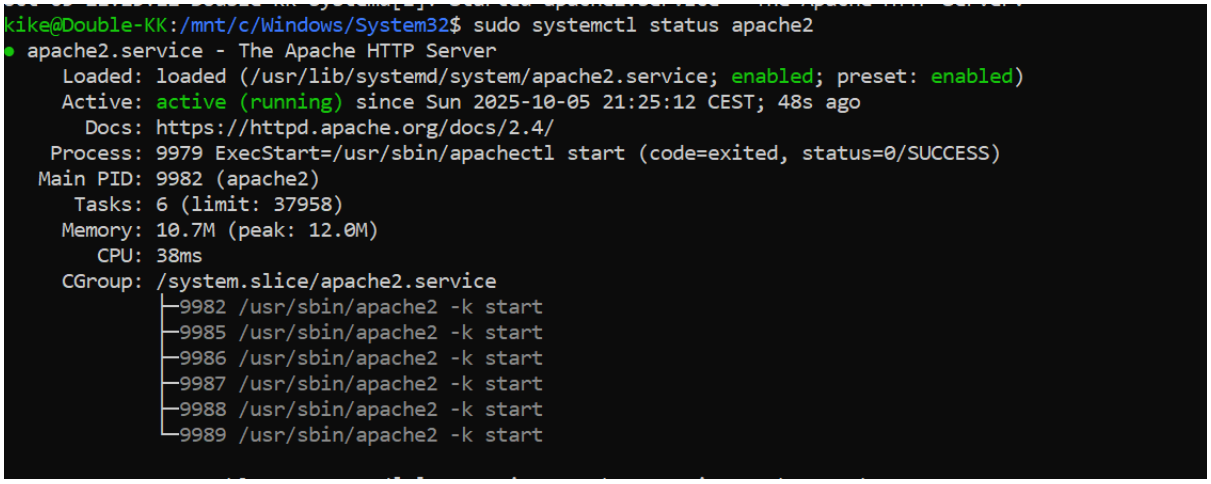
```
kike@Double-KK:/mnt/c/Windows/System32$ sudo apt install -y curl lsb-release ca-certificates
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (8.5.0-2ubuntu10.6).
curl set to manually installed.
lsb-release is already the newest version (12.0-2).
lsb-release set to manually installed.
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
The following package was automatically installed and is no longer required:
  libllvm19
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
kike@Double-KK:/mnt/c/Windows/System32$
```

2 Apache + PHP

- **Apache** es un servidor web popular y estable; **PHP** permite páginas **dinámicas**, frente a HTML estático. Ambos son “conceptos clave” citados en tu documento.
- `service/systemctl start` **levanta el servicio**; `systemctl status apache2` te da la evidencia de que está **activo** para la captura obligatoria.
- `info.php` con `phpinfo()` es la **prueba de fuego**: muestra versión y configuración de PHP. La práctica exige **captura del navegador/`curl`** mostrando esa página.

```
kike@Double-KK:/mnt/c/Windows/System32$ sudo apt install -y apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
```

```
kike@Double-KK:/mnt/c/Windows/System32$ sudo apt install -y php libapache2-mod-php
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
```



3 Nginx + HTML

Antes de Nginx, **deténer Apache** para liberar el puerto 80.

sudo service apache2 stop # o sudo systemctl stop apache2

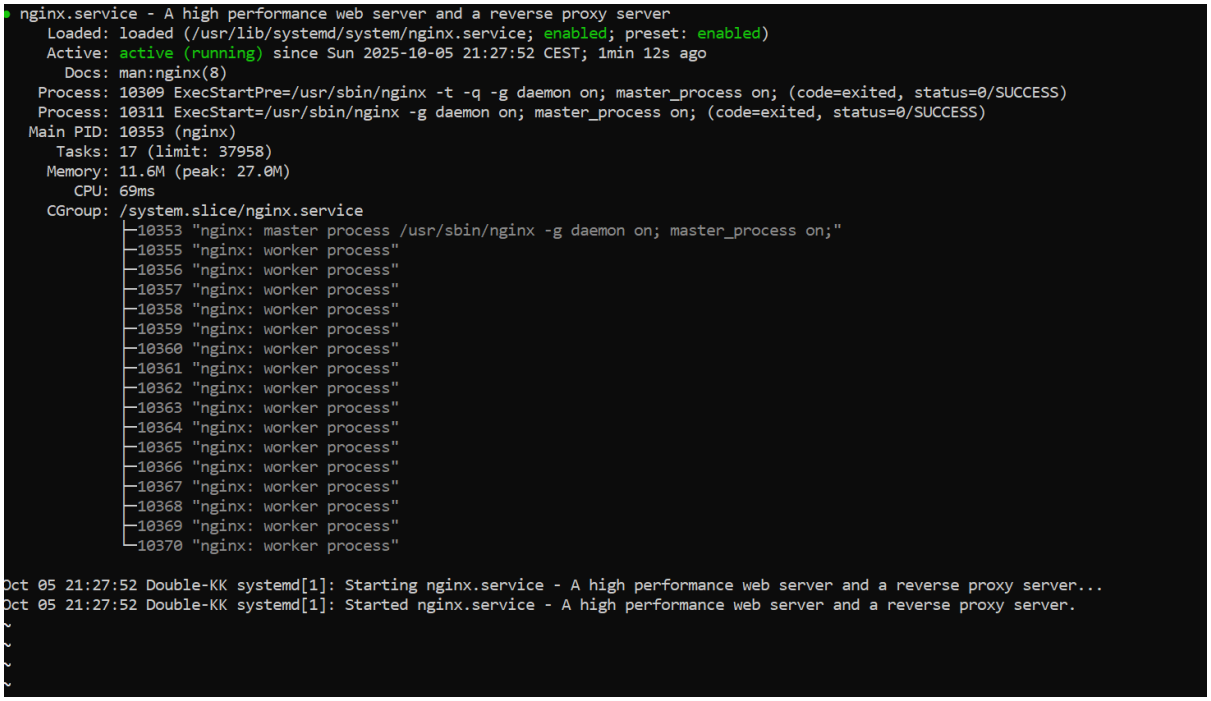
sudo apt install -y nginx

sudo service nginx start o sudo systemctl start nginx

sudo systemctl status nginx

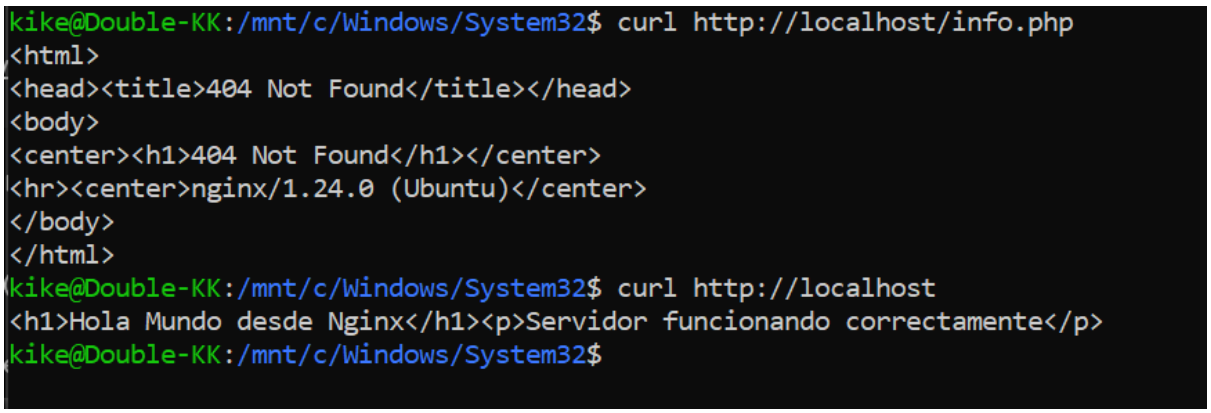
echo "<h1>Hola Mundo desde Nginx</h1> Servidor funcionando correctamente " | sudo tee /var/www/html/index.html

curl http://localhost



- **Apache y Nginx ambas escuchan en 80** por defecto; si no paras Apache, Nginx no podrá iniciar.
- Nginx es un servidor **ligero y muy rápido** para contenido estático; la práctica te pide **crear un HTML simple** y comprobarlo con `curl`/navegador.

Verificación y comparación



- El servidor activo es **Nginx**, no Apache.
- Nginx no tiene configurado PHP, por eso no interpreta el archivo `info.php` y devuelve error 404.
- Esto demuestra que **Apache está detenido**, y **Nginx está sirviendo el puerto 80** correctamente.

- El archivo `/var/www/html/index.html` creado en el paso anterior se muestra correctamente.
- Verifica que **Nginx funciona como servidor de contenido estático**.

IMPORTANTE ERROR 404:

4 Qué ocurrió con el error 404

Cuando ejecutaste:

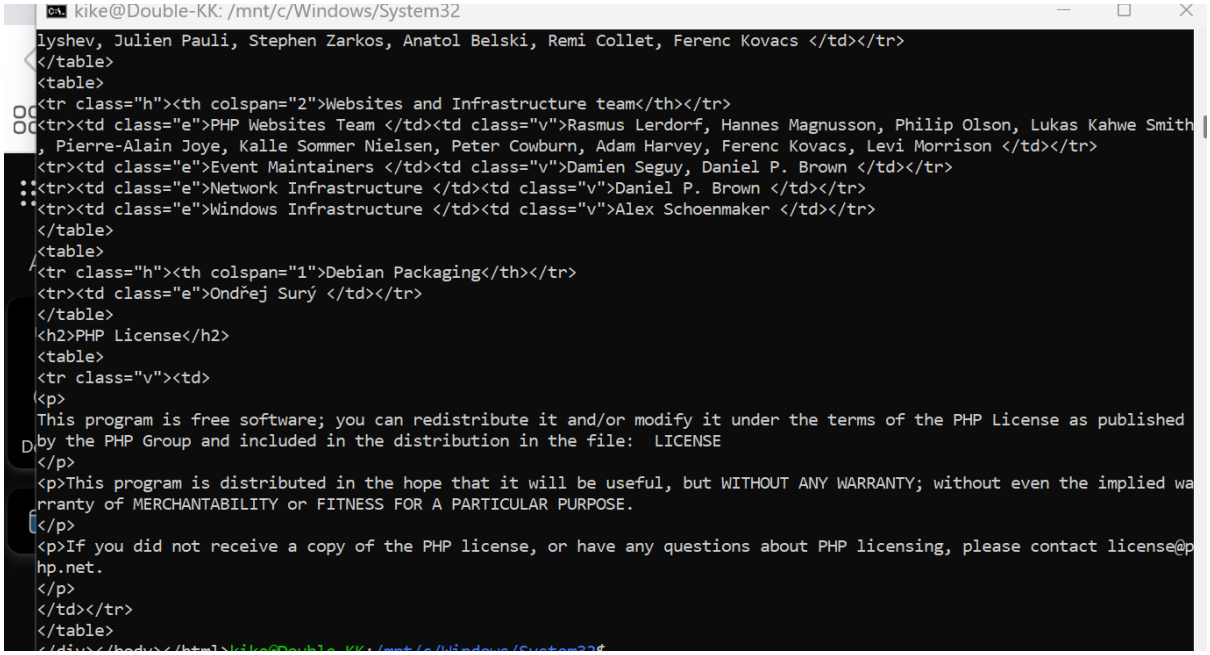
```
curl http://localhost/info.php
```

...el servidor Apache respondió con:

```
404 Not Found
The requested URL was not found on this server.
Apache/2.4.58 (Ubuntu) Server at localhost Port 80
```

Esto significa que **Apache sí estaba funcionando**, pero **no encontró el archivo solicitado (`info.php`)** en la carpeta raíz del sitio web.

Durante la verificación del servidor Apache, se intentó acceder a la página `info.php` y se obtuvo un error **404 Not Found**.
Este error no indicaba un fallo del servidor, sino que el archivo solicitado no existía en el directorio raíz (`/var/www/html/`).
El problema se debió a que, tras la instalación de Nginx, el archivo `info.php` había sido eliminado o reemplazado.
Para solucionarlo, se volvió a crear el archivo mediante el comando:



PHP Version 8.3.6

System	Linux Double-KK 6.6.87.2-microsoft-standard-WSL2 #1 SMP PREEMPT_DYNAMIC Thu Jun 5 18:30:46 UTC 2025 x86_64
Build Date	Jul 14 2025 18:30:55
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.3/apache2
Loaded Configuration File	/etc/php/8.3/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.3/apache2/conf.d
Additional .ini files parsed	/etc/php/8.3/apache2/conf.d/10-opcache.ini, /etc/php/8.3/apache2/conf.d/10-pdo.ini, /etc/php/8.3/apache2/conf.d/20-calendar.ini, /etc/php/8.3/apache2/conf.d/20-ctype.ini, /etc/php/8.3/apache2/conf.d/20-exif.ini, /etc/php/8.3/apache2/conf.d/20-ffi.ini, /etc/php/8.3/apache2/conf.d/20-fileinfo.ini, /etc/php/8.3/apache2/conf.d/20-ftp.ini, /etc/php/8.3/apache2/conf.d/20-gettext.ini, /etc/php/8.3/apache2/conf.d/20-iconv.ini, /etc/php/8.3/apache2/conf.d/20-phar.ini, /etc/php/8.3/apache2/conf.d/20-posix.ini, /etc/php/8.3/apache2/conf.d/20-readline.ini, /etc/php/8.3/apache2/conf.d/20-shmop.ini, /etc/php/8.3/apache2/conf.d/20-sockets.ini, /etc/php/8.3/apache2/conf.d/20-sysvmsg.ini, /etc/php/8.3/apache2/conf.d/20-sysvsem.ini, /etc/php/8.3/apache2/conf.d/20-sysvshm.ini, /etc/php/8.3/apache2/conf.d/20-tokenizer.ini
PHP API	20230831
PHP Extension	20230831
Zend Extension	420230831
Zend Extension Build	API420230831,NTS
PHP Extension Build	API20230831,NTS
Debug Build	no
Thread Safety	disabled


localhost
Inicio | ThePower FP -...
Inicio | Microsoft 365...
Hola Mundo desde Nginx
Servidor funcionando correctamente