SOLUCIONES A LAS PRACTICAS DE APACHE

Virtual hosts:

Prueba 1: Directorios raíz distintos por dominio

Enunciado: Configura dos dominios locales (app1.local y app2.local) que sirvan contenido desde carpetas distintas.

- Objetivo: Validar que Apache sirve contenido desde diferentes DocumentRoot.
- Configuración esperada:

```
<VirtualHost *:80>
    ServerName app1.local
    DocumentRoot "C:/Apache24/htdocs/app1"
    <Directory "C:/Apache24/htdocs/app1">
        Require all granted
        </Directory>
        </VirtualHost>

<pr
```

- /VirtualHost>Resultado esperado: Al visitar http://app1.local y http://app2.local, se muestran contenidos distintos.
- Errores comunes: No añadir los dominios en C:\Windows\System32\drivers\etc\hosts.

```
Páginas:
```

<!-- app1/index.html -->

<h1>Bienvenido a App1</h1>

<!-- app2/index.html -->

<h1>Bienvenido a App2</h1>

C:\Windows\System32\drivers\etc\hosts

127.0.0.1 app1.local

127.0.0.1 app2.local

Por logs separados:

<VirtualHost *:80>

ServerName app1.local

DocumentRoot "C:/Apache24/htdocs/app1"

ErrorLog "logs/app1-error.log"

CustomLog "logs/app1-access.log" common

</VirtualHost>

<!-- app1/index.php -->

<?php echo "Log de acceso generado"; ?>

Con páginas de error personalizadas:

<VirtualHost *:80>

ServerName sitio1.local

DocumentRoot "C:/Apache24/htdocs/sitio1"

ErrorDocument 404 /errores/404.html

ErrorDocument 500 /errores/500.html

```
<Directory "C:/Apache24/htdocs/sitio1">
   Options Indexes FollowSymLinks
   AllowOverride All
   Require all granted
 </Directory>
</VirtualHost>
<VirtualHost *:80>
 ServerName sitio2.local
 DocumentRoot "C:/Apache24/htdocs/sitio2"
 ErrorDocument 404 /errores/404.html
 ErrorDocument 500 /errores/500.html
 <Directory "C:/Apache24/htdocs/sitio2">
   Options Indexes FollowSymLinks
   AllowOverride All
   Require all granted
 </Directory>
</VirtualHost>
```

mod_rewrite

1) Crea una regla en .htaccess que:

- a. Redirija cualquier acceso a /blog/123 hacia blog.php?id=123
- b. Solo si el método HTTP es GET
- c. Y si el usuario no accede desde una IP local (192.168.*.*)

Solución:

- RewriteEngine On
- RewriteCond %{REQUEST_METHOD} ^GET\$
- RewriteCond %{REMOTE_ADDR} !^192\.168\.
- RewriteRule ^blog/([0-9]+)\$ blog.php?id=\$1 [L]

Ejercicio 1: Redirección permanente

Enunciado: Redirige contacto.html a contacto.php con código 301.

Solución:

RewriteEngine On

RewriteRule ^contacto\.html\$ contacto.php [R=301,L]

Ejercicio 2: Forzar HTTPS

Enunciado: Redirige todo el tráfico HTTP a HTTPS.

V Solución:

RewriteEngine On

RewriteCond %{HTTPS} off

RewriteRule ^(.*)\$ https://%{HTTP_HOST}%{REQUEST_URI} [R=301,L]

Ejercicio 3: URL amigable para productos

Enunciado: Convierte /producto/123 en producto.php?id=123.

Solución:

RewriteEngine On

RewriteRule ^producto/([0-9]+)\$ producto.php?id=\$1 [L]

Ejercicio 4: Bloquear IP específica

Enunciado: Bloquea la IP 123.45.67.89.

✓ Solución:

RewriteEngine On

RewriteCond %{REMOTE_ADDR} ^123\.45\.67\.89\$

RewriteRule ^.*\$ - [F,L]

Ejercicio 5: Redirección por hora

Enunciado: Redirige a cerrado.html si la hora es entre 00:00 y 06:00.

Solución:

RewriteEngine On

RewriteCond %{TIME_HOUR} ^0[0-6]\$
RewriteRule ^.*\$ cerrado.html [L]

Ejercicio 6: Redirección por navegador

Enunciado: Si el navegador es Internet Explorer, redirige a nosoportado.html.

✓ Solución:

RewriteEngine On

RewriteCond %{HTTP_USER_AGENT} "MSIE"

RewriteRule ^.*\$ no-soportado.html [L]

Ejercicio 7: Redirección por subdominio

Enunciado: Si el subdominio es blog.ejemplo.com, redirige a

blog/index.php.

✓ Solución:

RewriteEngine On

RewriteCond %{HTTP_HOST} ^blog\.ejemplo\.com\$

RewriteRule ^\$ blog/index.php [L]

Ejercicio 8: Redirección por query string

Enunciado: Si la URL contiene ?promo=navidad, redirige a navidad.html.

Solución:

RewriteEngine On

RewriteCond %{QUERY_STRING} promo=navidad

RewriteRule ^.*\$ navidad.html? [L]

Ejercicio 9: Redirección por idioma

Enunciado: Si el navegador está en francés, redirige a fr/index.html.

Solución:

RewriteEngine On

RewriteCond %{HTTP:Accept-Language} ^fr

RewriteRule ^\$ fr/index.html [L]

Ejercicio 10: Redirección por método POST

Enunciado: Si la petición es POST a /registro, redirige a procesar.php.

Solución:
RewriteEngine On

RewriteCond %{REQUEST_METHOD} POST RewriteRule ^registro\$ procesar.php [L]

Ejercicio 11: Tenemos una URL amigable con dos parámetros y queremos convertirla a una querystring: producto/123/abc → producto.php?num=123&letra=abc

✓ Solución:

RewriteEngine On RewriteRule ^producto/([0-9]+)/([a-z]+)\$ producto.php?numero=\$1&letra=\$2 [L]