

XAMPP: La Estación de Trabajo Local

1. ¿Qué es XAMPP? El Taller de Pruebas Personal ✂

XAMPP no es un lenguaje ni una base de datos; es un **paquete de software de código abierto** (open source) que te permite crear un **servidor web local** en tu propio ordenador. Es esencialmente tu **Taller de Pruebas** privado, donde puedes construir y ejecutar tus aplicaciones web (PHP y MySQL) sin necesidad de estar conectado a Internet.

El acrónimo XAMPP explica qué componentes incluye este paquete:

Letra	Componente	Función en el desarrollo web	Analogía
X	Cross-platform	Funciona en cualquier sistema operativo (Windows, macOS, Linux).	Compatibilidad universal.
A	Apache	El Servidor Web que procesa las solicitudes.	El motor que pone en marcha la página.
M	MySQL/ MariaDB	El Gestor de Base de Datos donde se guarda la información.	El Archivista (visto en el tema MySQL).
P	PHP	El Lenguaje de Scripting del lado del servidor.	El Chef (visto en el tema PHP).
P	Perl	Otro lenguaje de <i>scripting</i> (menos usado actualmente en web).	Herramienta de propósito general.

En resumen: XAMPP instala y configura todos los componentes esenciales para que el código PHP se ejecute y pueda hablar con la base de datos MySQL en tu máquina.

2. Instalación: Poniendo en Marcha el Taller

La instalación es sencilla, pero requiere atención en algunos puntos.

2.1. Descarga del Software

Visita el sitio web oficial de Apache Friends y descarga la versión adecuada para tu sistema operativo (Windows, macOS o Linux).

2.2. Proceso de Instalación (El Clásico "Siguiente, Siguiente...")

- Ejecutar el instalador.**
- Selección de Componentes:** Asegúrate de que al menos **Apache**, **MySQL** y **PHP** estén seleccionados.
- Ubicación de Instalación:** Se recomienda instalar XAMPP en la **raíz** de tu disco duro (ej. C:\xampp en Windows o /Applications/XAMPP en macOS) para evitar problemas de permisos.
- Finalizar:** Una vez terminada, el instalador te preguntará si deseas iniciar el Panel de Control (Control Panel).

2.3. Verificación de la Instalación

Para confirmar que todo funciona, abre tu navegador y escribe en la barra de direcciones:
`http://localhost`

Si ves la página de bienvenida de XAMPP, ¡el motor Apache está funcionando correctamente!

3. El Panel de Control: Manejando los Servicios

El **Panel de Control de XAMPP** es tu centro de mando. Desde aquí, inicias, detienes y gestionas los componentes principales (Apache y MySQL).

3.1. Iniciar/Detener Servicios

Para iniciar el entorno de desarrollo, debes hacer clic en el botón "**Start**" al lado de:

1. **Apache:** Para que tu servidor web esté activo.
2. **MySQL:** Para que la base de datos esté disponible para PHP.

Cuando un servicio se inicia correctamente, el nombre del módulo y sus LEDs asociados se vuelven **verdes**. Si fallan, se pondrán rojos o amarillos (ver sección de Errores).

3.2. Botones de Acceso Rápido

- **Explorer:** Abre directamente la carpeta principal de XAMPP.
 - **Shell:** Abre la consola de comandos para ejecutar comandos avanzados de MySQL o PHP.
 - **Admin (Apache):** Abre `http://localhost`.
 - **Admin (MySQL):** Abre `phpMyAdmin`.
-

4. La Carpeta Clave: `htdocs` (La Raíz de tu Proyecto) 📁

Dentro de la carpeta de instalación de XAMPP, hay una subcarpeta crítica: **`htdocs`**.

Concepto Fundamental: La carpeta `htdocs` es la **raíz del servidor web Apache**. Cualquier archivo PHP, HTML o CSS que quieras ejecutar o visitar a través de `http://localhost` **DEBE** estar dentro de `htdocs` o en una subcarpeta dentro de ella.

4.1. Creando tu Primer Proyecto

Para mantener la organización, nunca trabajes directamente en `htdocs`. Crea una subcarpeta para cada proyecto.

Ruta de ejemplo:

- `C:\xampp\htdocs\mi_primer_proyecto\`

Para visitar este proyecto en el navegador, usarías la URL:

- `http://localhost/mi_primer_proyecto/`

Si el archivo principal de tu proyecto se llama `index.php` o `index.html`, el navegador lo abrirá automáticamente.

5. phpMyAdmin: La Interfaz Gráfica para MySQL

Aunque MySQL se maneja con el lenguaje SQL (visto en el tema anterior), phpMyAdmin te ofrece una **interfaz gráfica (GUI)** para manejar tu base de datos sin escribir código SQL para las tareas básicas.

5.1. Acceso a phpMyAdmin

Desde el Panel de Control de XAMPP, haz clic en el botón "**Admin**" de la línea de MySQL. Se abrirá en tu navegador en la URL: `http://localhost/phpmyadmin/`.

5.2. Usos Fundamentales

- **Crear Bases de Datos:** Botón "New" o "Nueva" en el menú de la izquierda.
 - **Crear Tablas:** Definir los campos, tipos de datos y claves primarias para tus entidades.
 - **Insertar/Editar Registros:** Introducir datos manualmente en una tabla para hacer pruebas.
 - **Pestaña SQL:** Si sabes SQL, aquí puedes escribir y ejecutar directamente tus consultas (SELECT, INSERT, etc.).
-

6. Configuración de Puertos: El Conflicto Común

El problema más frecuente al iniciar XAMPP es el **Conflicto de Puertos**.

6.1. ¿Qué es un Puerto?

Los puertos son "canales" digitales a través de los cuales las aplicaciones se comunican. Por defecto, **Apache** usa el puerto **80**, y **MySQL** usa el puerto **3306**.

6.2. El Error: Puerto 80 Ocupado

A menudo, otra aplicación (como Skype, VMware o el propio **Servicio de Publicación Web de Windows**) ya está usando el puerto 80. Cuando intentas iniciar Apache, verás un error y la luz se quedará en rojo/amarillo.

6.3. Solución (Cambiando el Puerto de Apache)

La forma más limpia es cambiar el puerto por defecto de Apache a uno menos común (ej. **8080**).

1. En el Panel de Control de XAMPP, haz clic en el botón "**Config**" de la línea de Apache y selecciona **httpd.conf**.
2. Busca la línea: `Listen 80`
3. Cámbiala por: `Listen 8080`
4. Busca la línea: `ServerName localhost:80`

5. Cámbiala por: `ServerName localhost:8080`

6. **Guardar** el archivo y **Reiniciar** Apache.

¡Ojo! Si cambias el puerto a 8080, deberás acceder a tus proyectos usando `http://localhost:8080/` en lugar de solo `http://localhost/`.

7. Errores Comunes y Soluciones Rápidas 🛠️

Error Común	Síntoma y Causa	Solución
"Port 80 in use"	Apache no inicia (luz roja). Otra aplicación está usando el puerto 80.	Cambiar el puerto de Apache a 8080 (ver sección 6) o detener la aplicación en conflicto.
"Apache Shutdown Unexpectedly"	Apache se cierra inmediatamente después de iniciarse.	Suele ser un problema de permisos. Ejecuta el Panel de Control de XAMPP como Administrador .
MySQL no arranca	El puerto 3306 está siendo usado (ej. por una instalación previa de MySQL).	En el Panel de Control, haz clic en Config > my.ini y cambia <code>port = 3306</code> a <code>port = 3307</code> . Reinicia MySQL.
Access Denied en PHP	Tu código PHP no puede conectarse a MySQL.	Verifica que el usuario y contraseña en tu código PHP (<code>\$usuario = "root"; \$password = "";</code>) coincidan con los valores por defecto de XAMPP.
Error 404	No se encuentra la página (<code>http://localhost/archivo.php</code>).	Asegúrate de que el archivo esté guardado dentro de la carpeta htdocs y que la URL coincida exactamente con la ruta y el nombre del archivo (¡incluyendo mayúsculas/minúsculas!).

8. Despliegue de tu Primer Proyecto con PHP y MySQL

Una vez que XAMPP está instalado y los servicios están corriendo, puedes unir todo lo que has aprendido.

8.1. Estructura de Proyecto

1. **Crea la Carpeta:** Crea una carpeta en `C:\xampp\htdocs\mi_proyecto`.
2. **Archivos de Código:** Dentro, crea `index.php` (para el código PHP) y `style.css` (para el diseño).

8.2. Pasos para el Despliegue Local (Uniando todos los pilares)

1. **Iniciar XAMPP:** Asegúrate de que **Apache** y **MySQL** estén en verde.
2. **Crear la Base de Datos:** Abre **phpMyAdmin** (`http://localhost/phpmyadmin/`) y crea una Base de Datos llamada, por ejemplo, `proyecto_db`.
3. **Código PHP (index.php):** Escribe un script PHP que intente conectarse a esa Base de Datos.

```

<?php
// 1. Configuración de la Conexión (El 'root' por defecto no tiene contraseña)
$servidor = "localhost";
$usuario = "root";
$password = "";
$dbd = "proyecto_db";

// 2. Intentar la Conexión
$conexion = new mysqli($servidor, $usuario, $password, $dbd);

// 3. Imprimir el resultado de la conexión en HTML
echo "<!DOCTYPE html><html lang='es'><head><title>Test</title></head><body>";

if ($conexion->connect_error) {
    // Usa HTML y CSS para un mensaje claro
    echo "<h1 style='color: red;'>✗ ERROR de Conexión.</h1>";
    echo "<p>Revisa el Panel de Control y los datos de login.</p>";
} else {
    echo "<h1 style='color: green;'>✓ Conexión Exitosa. ¡Servidor Listo!</h1>";
    echo "<p>Tu motor (Apache) y tu archivista (MySQL) están hablando con PHP.</p>";
}

// 4. Cerrar la conexión y el HTML
$conexion->close();
echo "</body></html>";
?>

```

4. **Ejecutar en el Navegador:** Abre http://localhost/mi_proyecto/index.php (o la URL con el puerto que hayas configurado). Si todo es verde, ¡tu entorno local está funcionando perfectamente!