**MANUAL DE COMO CONECTAR Visual Studio CODE CON XAMPP**

**Esto es para poder ver los scriptch o fuentes creadas en VSC en el servidor web de Xampp que es como la vería un cliente que se conecta al servidor**

VIDEO 1: <https://www.youtube.com/watch?v=W9cDh_sC2As>

VIDEO 2: https://www.youtube.com/watch?v=jHJGOAZG0oc

Los dos videos son casi iguales, solo se diferencian en que en uno configura unas settings de VSC y en el otro otras, pero al final es el mismo settings. Por si acaso verlos los dos.

Básicamente es configurar el setting.jason con la línea "php.validate.executablePath": "C:\\xampp\\php\\php.exe"

Luego instala 3 extensiones de php para VSC y otras 3 de mysql pero creo que las de mysql no hacen falta.

Luego crear una variable de entorno en Windows con la ruta c:\xampp\php.

Ya estaría conectado y en un proyecto php dentro de VSC con botón derecho se le da a php server serve proyect y se abre en el navegador web la página creada en php.

Para volver a lanzar hay que pararlo antes dándole a PHP STOP SERVER

RESUMEN DE LO QUE HACE ESTO: al instalar Xampp (X) se instala en el ordenador en local PHP, con lo que ya tenemos PHP en ejecutable. Solo nos falta un servidor para ejecutar un scripto php con el php.exe. Ese servidor es una extensión que se instala PHP Server. Al instalarla en los setting de VSC en la zona de Extensions/php server configuración aparecen nuevos parámetros como el puerto (3000), el path del php.exe. Con esto, lo que se hace es instalar un servidor web, la extensión de VSC PHP server, que coge el script y lo ejecuta en el puerto localhost/3000. No hace falta que X esta arrancado ni apache porque el servidor lo hace PHP server. Pero ese PHP server usa el php.exe que tiene X en su carpeta de instalación, que es para lo único que se necesita X.

Pero VSC usar la folder que tiene abierta para poder mapear el scritp con el PHP server, es decir, si el folder de VSC no es el del script que se quiere ejecutar en el PHP server, no va a funcionar porque no se ven, PHP server no encuentra el script para ejecutarlo y da error de no encontrado. Los servidores virtuales siempre usan una carpera del pc local para poder comunicarse con el pc que es el que tiene los script, esa carpeta en este caso es Folder de VSC.

Uno de los parámetros que hay en settings de VSC en la parte de extensión/PHP server es Relative Path: con este parametro se le indica una ruta local para mapear con el servidor PHP server, indicando una ruta, haríamos que php server cargue script que no este en el Folder actual.

**MUY IMPORTANTE: para ejecutar ficheros php en VSC primero debe estar abierta la folder del fichero a ejecutar en VSC. Ir File, close folder actual y opn folder donde esta el fichero .php. y así ya se puede usar lo del PHP SERVER:** server proyect **.**

Algo curioso: si ejecuto phpinfo() en un script ejecutado con el servidor de PHP server no me dice que tiene Xdebug, porque la extension de VSC PHP server no esta cargando el x\_debug que viene especificado en php.ini, usa php.exe que esta en X pero no esta usando php.ini. Php.ini si que lo usa Xampp y carga x-debug

Ahora bien para ejecutar script usando el servidor de Xampp hay que copiarlos en la carpeta C:\xampp\htdocs y escribir en el navegador web <http://localhost/script.php> ya que apache de Xampp escucha por el puerto 80 y esta mapeado a la carpeta C:\xampp\htdocs, no tengo que escribir <http://localhost/htdocs>, porque al estar mapeada la carpeta htdocs internamente en Xampp, el tiene el contenido de esa carpeta dentro de su servidor web y lo muestra por el puerto 80. Si tuviera mapeada la carpeta c:/pedrito, todo lo que yo metiera en pedrito, archivos HTML o script PHP estarían visibles para ejecutar por el puerto 80 de localhost.

**He hecho phpinfo() de** [**http://localhost/dashboard/phpinfo.php**](http://localhost/dashboard/phpinfo.php) **PHP corriendo en servidor apache de XAMPP y phpinfo() de** [**http://localhost:3000/prueba.php**](http://localhost:3000/prueba.php)**, SERVIDOR de VCS extensión PHP server. Puedo ver que unos esta en servidor apache y el otro en servidor llamado: Built-in HTTP server.**

En el parámetro Loaded Configuration File: apache tiene “C:\xampp\php\php.ini” y VSC tiene “C:\xampp\php\php.exe”, por eso no esta cargando xdebug.

Le pregunto a la IA y me dice esto:

Edita tu archivo phpinfo.php para mostrar el nombre del entorno de ejecución. Usa esto:

<?php

echo "<hr>";

echo "SAPI: " . php\_sapi\_name();

?>

Esto es lo que me sale: cli-server

Qué debes ver

* Si ves algo como: SAPI: apache2handler O SAPI: cgi-fcgi → ¡Perfecto! PHP se está ejecutando en modo servidor.
* **Pero si ves: SAPI: cli** → ¡Ahí está el problema! Estás ejecutando PHP en modo CLI (línea de comandos), no como un servidor web. Esto sugiere que el entorno en el que se está ejecutando phpinfo() no es un servidor real, o bien está usando PHP de forma incorrecta (como en modo CLI embebido, no como servidor web). La extensión PHP Server no es un servidor web completo como Apache o Nginx. Lo que hace es ejecutar PHP directamente con el servidor embebido de PHP: Y si no está correctamente configurado, puede acabar simplemente llamando a php en modo CLI, que solo ejecuta el archivo y lo devuelve como texto plano o sin procesar bien. Con php -S localhost:3000 se incia php en modo cli.

**Así que el problema es PHP server que no ejecuta completamente php.exe,**

**La IA dice:** El valor cli-server indica que estás ejecutando PHP con su **servidor embebido** (integrado en PHP desde la versión 5.4). Esto confirma que la extensión **PHP Server** de VS Code está usando el comando: php -S localhost:PORT, Y eso está bien, **funciona correctamente para desarrollo local**.

La extensión **PHP Server** de VS Code no está pasando correctamente el parámetro de configuración que le indica a PHP cuál archivo php.ini debe usar. Este comportamiento confirma que **PHP se está ejecutando**, pero **sin detectar ni usar el archivo php.ini**, porque la extensión lo invoca de forma incompleta o incorrecta.

**Interesante prueba. La IA me dice que haga esto:**

1. **Abre una terminal en la carpeta donde tienes tu prueba.php script que ejecutara phinfo()**

Ejecuta: **php -S localhost:8000** esto crea un servidor de php en el puerto 8000.

**Abre en el navegador:** [**http://localhost:8000/phpinfo.php**](http://localhost:8000/phpinfo.php)

**Ahora deberías ver:**

* **Toda la tabla de phpinfo()**
* **Y Loaded Configuration File correctamente apuntando a php.ini, como:**

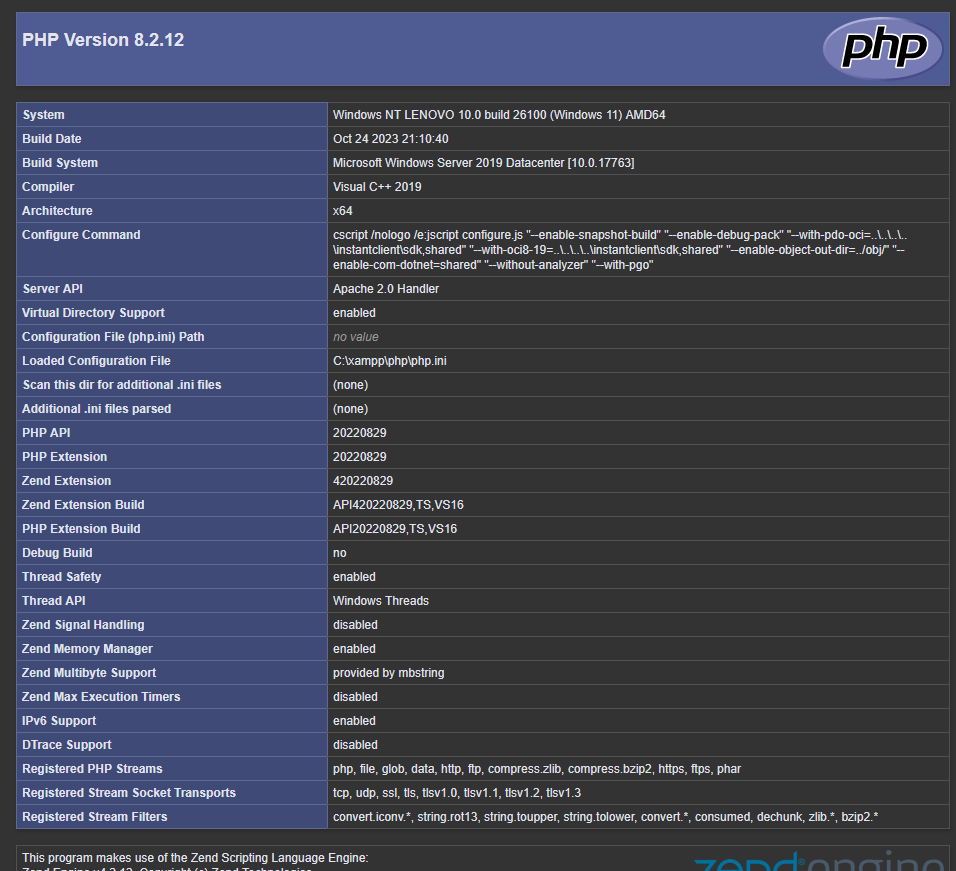
**Veo que el servidor que ha creado con el CMD** Es de tipo Built-in HTTP server igual que el de PHP server. Así que el problema no es php server si no la configuración de algún parámetro de VSC o PHP server

|  |  |
| --- | --- |
|  |  |
|  |  |

**COMO CONFIGURAR PHP DEBUG EN VSC (para depurar código paso a paso)**

VIDEO: <https://www.youtube.com/watch?v=QxRTw4GPk5Q>

SE INSTALA PHP DEBUG EN LAS EXTENSIONES DE VSC

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.EN LAS INSTRUCCIONES DE PHP DEBUG (PD) HAY UN LINK PARA USAR UN **ASISTENTE** QUE DICE QUE TIENES QUE HACER, PINCHAR EN EL LINK Y SALE UNA VENTANA COMO ESTA

PHP INFO

ASISTENTE

EN ESA VENTANA PEGAR EL TEXTO (CTRL+A) QUE TE DA LA FUNCION **PHPINFO()** EJECUTADA EN UN FICHERO PHP EN EL SERVIDOR DE PHP (CARPETA CONECTADA CON XAMPP) O ESTO EN EL NAVEGADOR <http://localhost/dashboard/phpinfo.php>

EN LAS INSTRUCCIONES TE DICE QUE DESCARGUES UNA DLL, HAY QUE DESCARGAR ESA EXACTAMENTE, YO H ESTADO 3 DIAS CON FALLOS Y ES PORQUE DESCARGUE OTRA.

SE DESCARGA, SE RENOMBRA, SE COPIA EN EL DIRECTORIO QUE DICE Y LUEGO EDITAR PHP.INI ESCRIBIENDO ESTO:

zend\_extension = C:\xampp\php\ext\php\_xdebug.dll

[XDebug]

xdebug.mode = debug

xdebug.start\_with\_request = yes

xdebug.remote\_port = 9003

xdebug.idekey = VSCODE

SE GUARDA SE REINICIA XAMPP Y

LUEGO VAMOS A ESTE BOTON, EL QUE ARRANCA LA DEPURACION DE PD(php DEBUGER) Y LA PRIMERA VEZ SALE QUE SE PINCHE EN CREAR ARCHIVO LAUNCH. JASON, DECIRLE QUE ES PARA ENTORNO PHP. EN LAUNCH.JASON

PONER EL PUESTO 9003 MISMO QUE EL PHP.INI

A LA HORA DE ARRANCAR EL DEBUG HAY QUE ELEGIR ESTA OPCION LISTEN FOR XDEBUG

Una captura de pantalla de un celular con texto e imagen

El contenido generado por IA puede ser incorrecto.SI SE BORRAR ESTE FICHERO LAUNCH.JASON OTRA VEX AL ARRANCAR PD TE PIDE

CREAR EL ARCHIVO LAUNCH.JASON

HAY QUE TENER EN CUENTA QUE PARA QUE SE VEAN LOS VALORES DE LAS VARIABLES EN EL DEGUG HAY QUE HACER PHP SERVER PROYEC CON EL BOTON DERECHO EN EL SCRITCH PHP CREADO, PARA CARGAR LA WEB DEL PROYECTO Y ASI YA SALEN LAS VARIABLES EN EL DEBUG.

**XAMPP**

XAMP es una multiherramienta, que lleva servidor web APACHE para PHP, porque PHP por si solo no tiene servidor web. Lleva mysql y más cosas.

XAMP (X) simula un servidor web en la 127.0.0.1 o localhost. Al escribir localhost (apache escucha el puerto 80. Para ejecutar scritp de php copiados en la carpeta C:\xampp\htdocs tengo que poner localhost:3000/nombre script, porque php escucha por el 3000, pero no sé dónde está esa configuración) en navegador web sale la página de bienvenida de X, que es el fichero index.php que hay en la carpeta C:\xampp\htdocs.

La carpeta C:\xampp\htdocs es la carpeta que el servidor web usa para contener sus páginas web creadas en PHP. Solo se puede ejecutar scritch php que este en esa carpeta (escribiendo localhost:3000/nombre script ) o en una subcarpeta que este dentro de C:\xampp\htdocs.

Nota: en la página de bienvenida de X hay un link arriba que pone phpinfo que es lo mismo que ejecutar la función de php phpinfo() para ver toda la info de PHP (antes hay que arrancar Apache en el panel de control de X. también hay otro link PHPmyadmin que va a la base de datos de Mysql, habiendo arrancado antes Mysql en el panel de control de X.

**Depurando PHP con Xdebug EN VSC y Docker (PHP CORRIENDO EN CONTENEDOR)**

LINK DE INSTRUCCIONES: <https://yoprogramo.com/2024/10/12/depurando-php-con-xdebug-y-docker/>

Crear la carpeta web, y dentro crear el archivo Docker docker-compose.yml.

version: '3'

services:

php:

image: yoprogramo/php8.2:1.0.2-xdebug

volumes:

- .:/var/www/Website

ports:

- 8080:80

extra\_hosts:

- host.docker.internal:host-gateway

Ojo con la última instrucción que hay que quitar las dobles comillas. Y en volúmenes quitar web, porque ya hemos creado la carpeta y estamos dentro ya, con lo que no tiene que crear nada. Si le ponemos .web creara la carpeta web dentro de web

Ejecutar ahí dentro Docker-compose ps -d para descargar la imagen.

Ya podemos ejecutar localhosto:8080/prueba.php para que ejecute phpinfo() y se puede ver que este contenedor tiene php con la librería xdebug .

Creo que debe estar el php que vayamos a depurar tiene que estar dentro de la carpeta www.

También podemos ver que en esta imagen no crea un contenedor con apache ni ningún servidor web, es porque ya lo lleva integrado la image, es una imagen con un servidor web y php y los dos configurados para que se vean.

Luego ir a VSC y añadir la línea a launch.json en la parte de Listen for Xdebug

"pathMappings": {

   "/var/www/Website": "${workspaceRoot}/web"}

Y ya está, no hace falta cargar el fichero Script en el navegador web para que se ejecute y poner ver las variables en el debug

Ni hace falta darle a PHP SERVER PROYEC CON EL BOTON DERECHO

Nota: gracias a esta línea añadida a launch.json, ahora con XAMPP no hace falta abrir el script en el navegador para que empiece a verse las variables. Realmente no hace falta arrancar XAMPP, no el contenedor de PHP creado con la imagen yoprogramo. No se qué ha pasado que me funciona el Xdebug sin arrancar nada y estando el archivo php a depurar en cualquier carpeta, lo único que tiene que haber es un launch.json con las siguientes líneas

{

    // Use IntelliSense to learn about possible attributes.

    // Hover to view descriptions of existing attributes.

    // For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387

    "version": "0.2.0",

    "configurations": [

        {

            "name": "Listen for Xdebug",

            "type": "php",

            "request": "launch",

            "port": 9003,

           "pathMappings": {

               "/var/www/Website/": "${workspaceFolder}/web"

        },

        {

            "name": "Launch currently open script",

            "type": "php",

            "request": "launch",

            "program": "${file}",

            "cwd": "${fileDirname}",

            "port": 0,

            "runtimeArgs": [

                "-dxdebug.start\_with\_request=yes"

            ],

            "env": {

                "XDEBUG\_MODE": "debug,develop",

                "XDEBUG\_CONFIG": "client\_port=${port}"

            }

        },

        {

            "name": "Launch Built-in web server",

            "type": "php",

            "request": "launch",

            "runtimeArgs": [

                "-dxdebug.mode=debug",

                "-dxdebug.start\_with\_request=yes",

                "-S",

                "localhost:0"

            ],

            "program": "",

            "cwd": "${workspaceRoot}",

            "port": 9003,

            "serverReadyAction": {

                "pattern": "Development Server \\(http://localhost:([0-9]+)\\) started",

                "uriFormat": "http://localhost:%s",

                "action": "openExternally"

            }

        }

    ]

}

Lo raro es que le cambio el número de puerto en launch.json y sigue funcionando. Está usando algo de otro sitio

Cuando quito esta línea en el launch.json  "pathMappings": {  "/var/www/Website/": "${workspaceFolder}/web" falla el debug.

NOTA: según donde se ponga el Breakpoint puede ser que no se inicie el dibug, porque hay líneas de código HTML que no son compatibles con debug.