## **Debugging PHP using Visual Studio Code and Xdebug**

Are you suffering from PHP debugging pain using var\_dump(); echo “foo”; die(); :-(

Let’s move to VS Code using PHP xDebug and have funny debugging.. :-)

1. **Visual Studio Code Installation** (source - <https://code.visualstudio.com/docs/setup/linux>)

i) Download latest version <https://go.microsoft.com/fwlink/?LinkID=760868>

ii) Installing the .deb package will automatically install the apt repository and signing key to enable auto-updating using the regular system mechanism.

sudo dpkg -i <file>.deb  
sudo apt-get install -f # Install dependencies

**OR**

The repository and key can also be installed manually with the following script:

curl https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > microsoft.gpg

sudo mv microsoft.gpg /etc/apt/trusted.gpg.d/microsoft.gpg

sudo sh -c 'echo "deb [arch=amd64] https://packages.microsoft.com/repos/vscode stable main" > /etc/apt/sources.list.d/vscode.list'

iii) Then update the package cache and install the package using:

sudo apt-get update  
 sudo apt-get install code # or code-insiders

iv) And then run: $code in terminal and your VC Code window open..

1. **Search the following VS Code Extensions and Install:**

PHP Debug - PHP Debug Adapter for Visual Studio Code

PHP IntelliSense (optional)

ESLint - VS Code ESLint extension (optional)

1. **After installing extensions, we overwrite User Settings in VS Code for that:**

Open User Settings in VS Code: press F1 and

search “user” and click on Preferences: Open User Settings option

In user settings add:

{

"php.validate.executablePath": "/usr/bin/php"

}

1. **Installing and configuring Xdebug extension for PHP**

Using Xdebug installation wizard ()

In terminal enter $php -i which gives you phpinfo in terminal

Copy terminal php info, open <https://xdebug.org/wizard.php> and pest in the box and click on Analyse My phpinfo() button, you get the instruction for installing as below:

These are instructions for PHP7.1 (You may get different version xdebug or commands depending on your PHP version, my is PHP7.1)

1. Download [xdebug-2.5.4.tgz](http://xdebug.org/files/xdebug-2.5.4.tgz)
2. Unpack the downloaded file with tar -xvzf xdebug-2.5.4.tgz
3. Run: cd xdebug-2.5.4
4. Run: phpize (See the [FAQ](https://xdebug.org/docs/faq#phpize) if you don't have phpize.
5. As part of its output it should show:
6. Configuring for:  
   ...  
   Zend Module Api No: 20160303  
   Zend Extension Api No: 320160303
7. If it does not, you are using the wrong phpize. Please follow [this FAQ entry](https://xdebug.org/docs/faq#custom-phpize) and skip the next step.
8. Run: ./configure
9. Run: make
10. Run: cp modules/xdebug.so /usr/lib/php/20160303
11. Update /etc/php/7.1/cli/php.ini and change the line
12. Zend\_extension = /usr/lib/php/20160303/xdebug.so

Follow the above instructions

But if you don’t have phpize here is the command for installing:

For 7.0 - sudo apt install php7.0-dev

For 7.1 - sudo apt install php7.1-dev

sudo service apache2 restart

1. **Update /etc/php/7.1/apache2/php.ini and add the lines**

[XDebug]

xdebug.remote\_enable = 1

xdebug.remote\_autostart = 1

xdebug.remote\_host = localhost

1. **Configuring xDebug with PHP7**

i) A simple apt-get command: sudo apt-get install php-xdebug

ii) You should check your PHP modules list and see xdebug in there now: php -m

iii) You will also need to restart Apache2: sudo service apache2 restart

iv) We enable stack trace manually: vim /etc/php/7.0/mods-available/xdebug.ini

And add line in file: xdebug.show\_error\_trace = 1

1. **Visual Studio Code debugger setting**

In VS Code go to debug icon and open launch.json by clicking debug setting icon

Add the below configuration in launch.json file and save the file

{

"version": "0.2.0",

"configurations": [

{

"name": "Listen for XDebug",

"type": "php",

"request": "launch",

"port": 9000,

"log": true

},

{

"name": "Launch",

"request": "launch",

"type": "php",

"program": "${file}",

"cwd": "${workspaceRoot}",

"externalConsole": false

}

]

}

1. **You have Done it…. Now start debugging.... :-)**

Note - If you stuck somewhere please check PHP version and you are adding xdebug configuration in same version php.ini