PHP

PHP: Hypertext Preprocessor is a general-purpose programming language originally designed for web development. It was originally created by Rasmus Lerdorf in 1994; the PHP reference implementation is now produced by The PHP Group.

Below is the default coding template I use for PHP:

php coding template

```
<?php
/* Copyright <copyright owner> <home page URL> */

         /**
          * @file <file name>
          * @brief <bried information> \n
          * @author <author information>
     */

?>
```

Example)

php remark example

```
<?php
/* Copyright Chun Kang <http://qsok.com> */

         /**
          * @file /modules/api/register_user.php
          * @brief Registger user \n
          * @author Chun Kang (ck@qsok.com)
     */

?>
```

Regarding PHP common settings, please refer PHP configuration I commonly use

In the PHP, error_reporting does not need to be detail necessarily for security purpose, and short_open_tag could cause coding error easily, but needs to be used for backward compatibility in some cases.

Below is what I am using for developing/operating PHP application in my servers:

```
error_reporting = E_ALL & ~E_NOTICE
display_errors = On
display_startup_errors = On
short_open_tag = On
register_argc_argv = On
include_path = ".:.."
```

Other topics associated

- 32bits simple hash function in PHP — PHP provides a simple function can hash in 32bits - **hash**($algorithm, $key) - it will provide 8 bytes string as hash result.

- Array of arguments passed to script in PHP — **$argv** is an array of arguments passed to script in PHP, so you can know the parameters when user run the script
- Bob Jenkins' One-At-A-Time hashing algorithm in PHP — Bob Jenkins' One-At-A-Time hashing algorithm is simply known as **jenkins_hash**() providing 8 bytes long hash value.
- Call PHP-based server URL without its extenstion (.php) — If you develop software based in PHP but you don't know to show its extension(.php) like http://foo.com/auth instead of http://foo.com/auth.php, you can make your PHP page work without its extension by adding **ForceType application/x-httpd-php on .htaccess**
- Check CPU - is AMD or Intel? — In CentOS 7, CPU information is stored in /proc/cpuinfo, so you can implement to acquire CPU information in the system.
- Check if user is root or sudo in php — posix_getuid() returns non-zero if user is root or sudo, so you can warn user to use an appropriate permission if required.
- Checking XML element by preg_match — Below code shows how to check XML element by preg_match
- Clearing a PHP value/array — Sometimes, you will need to reset variables, and unset() will help you to do it.
- Count number of processes and do something in PHP — You can limit number of cron jobs, if you use PHP for DevOps software in CentOS 7, which can be easily implemented by checking process by shell_exec.
- Delete one element in PHP array — The **unset**() function in **PHP** resets any variable. If **unset**() is called inside a user-defined function, it unsets the local variables. If a user wants to **unset** the global variable inside the function, then he/she has to use $GLOBALS array to do so. The **unset**() function has no return value.
- Display an array as an HTML table with the index value in PHP — foreach($array as $key => $value) is useful when you need key and its value at the same time.
- Display the result of a MySQL query in an HTML table — You can obtain the results of a MySQL server query using the mysqli extension.
- Display the result of running python code by shell command (shell_exec) in PHP — I usually use shell_exec when I need to run python code. The benefit of shell_exec is returning the result as string, so I can use it for other purpose.
- Downgrade PHP from version 7.3 to 7.2 on CentOS — You can downgrade PHP from version 7.3. to 7.2 by yum and yum-config-manager.
- Easy Python Launcher based on PHP — You can run your python code easily based on your web browser if you have Apache+PHP server. Below code enable you to run your code by click
- Facebook scalable architecture beyond LAMP — Facebook is known as using LAMP as their basic platform for their website. One of the outstanding things Facebook doing very well is that they have their own strategy/architecture based on LAMP. LAMP stands for Linux + Apache + MySQL + PHP that is entirely free and open sourced software, so they can compete with their competitors in competitive operation cost with high-avail-abilities. In person, I think their strategy is pretty much innovative and its status tells their strategy is on the right direction.
- file_exists covers local and internet in PHP — file_exists() is useful function can check the availability of a file. Below customized function enables you to check file availability not only local but also remote file as well.
- file_get_contents with custom User-Agent in PHP — To put custom User-Agent value with file_get_contents, you will need to create a stream_content.
- Find files in PHP — **scandir**($path) returns the searched result in array. If you want to implement recursive scan for sub directories, you can implement it by checking filetype().
- Find image objects from the plain HTML text with PHP — DOMDocument object enables you to find the necessary HTML objects easily.
- Function as a part of parameter in PHP — PHP provides super easy feature can put a function as a part of function parameter. You can just put it like function exampleMethod($customFunc).
- Get array keys in PHP — array_keys() return array keys in a string array
- Get Client's IP Address in PHP — I had got client's IP address by $REMOTE_ADDR or $_SERVER['REMOTE_ADDR'], but it does not work on latest version of PHP.
- Get DB field name in PHP PDO from MySQL query result — PDO provides a method to get database field name (column name) from the query result.
- Get elements in GET, POST, COOKIE, SERVER as variable — The **extract**() Function is an inbuilt function in **PHP**. The **extract**() function does array to variable conversion. That is it converts array keys into variable names and array values into variable value. In other words, we can say that the **extract**() function imports variables from an array to the symbol table.
- Get file extension in PHP — pathinfo() in PHP enables you to have file extension easily.
- Get file path in PHP — **pathinfo**() provides various information like dirname, basename, extension, and full filename. You can get intended result by **pathinfo($filename)['dirname']**
- Get local IP Address in PHP on CentOS — The easiest way to get local IP Address in PHP on CentOS is using shell command like "**hostname -I**". I tried several methods to get it, but most of them returns the wrong answer.
- get_headers() with timeout — To set timeout in get_headers(), you will need to set stream_context_set_default(), so you can optimize communication cost as you designed.
- Hash value for a file — If you want to make a simple logic to get hash value for file, hash_file() gives you a straightforward solution.
- Heredoc widely used for SQL in PHP — Heredoc provides an alternative way of defining strings in PHP. They are particularly useful when we need to define a string over multiple lines. They work by defining an identifier that will mark the start and end of the string. The identifier can be any alphanumeric value following the same rules we'd use for variable names. One important thing to note with the identifier is that as of PHP 7.3 you need to make sure that it does not appear within the string itself.
- Install PHP on CentOS 7 — You need to run yum in order to use PHP after setup your CentOS 7. Even though you have httpd, PHP may need to be installed manually. For compatibility with various applications like PHPMyAdmin, PHP-based server management, you will need to install additional libraries.
- Install PHP on CentOS 8 — You need to run yum in order to use PHP after setup your CentOS 8. Even though you have httpd, PHP may need to be installed manually. For compatibility with various applications like PHPMyAdmin, PHP-based server management, you will need to install additional libraries.
- JSON file management in PHP — PHP provides strong solutions for JSON. **json_encode**($src_obj) converts obj to JSON text. **json_decode**($src_text) converts text string to JSON object.
- Loading a text file in PHP — PHP provides useful functions can read text file directly instead of open read close: **readfile**(), file_get_contents() - personally I prefer to use **file_get_contents**(), because it does not display anything after reading the content.
- MySQL combination with Heredoc in PHP — Heredoc is one of the useful solution to secure better readability when we code SQL Query in PHP
- No match for argument: php-mcrypt on CentOS 7 — **mcrypt** package is not included in the default CentOS repositories. It can be installed from the Remi repository.
- No match for argument: php-pecl-memcache on CentOS 7 — **php-pecl-memcache** package is not included in the default CentOS repositories. It can be installed from the Remi repository.
- No match for argument: php-tidy on CentOS 7 — **php-tidy** package is not included in the default CentOS repositories. It can be installed from the Remi repository.
- PHP configuration I commonly use — In the PHP, error_reporting does not need to be detail necessarily for security purpose, and short_open_tag could cause coding error easily, but needs to be used for backward compatibility in some cases.
- PHP does not show any errors

- PHP Frameworks — There are lots of frameworks available in PHP
- PHP Server RESTful API example to get message body — When call RESTful API containing JSON data in the body, server needs to take its content, but getting it by parameters may have some restrictions like data length or etc. To avoid that error, the easiest way can do it will be taking it from entity body directly. file_get_contents('php://input') will help you to solve it easily.
- PHP_SELF — PHP_SELF is a variable that returns the current script being executed. This variable returns the name and path of the current file (from the root folder). You can use this variable in the action field of the FORM or any place when you write a code dynamically.
- preg functions as replacement of ereg/eregi functions in PHP — In PHP 7.x, ereg functions are no longer supported. Based on my search on internet, ereg functions use a very limited regex flavor called POSIX ERE - meaning pref functions can cover more REGEX syntax as designed.
- Preparing production and staging environment in PHP project — When developing live service but requiring continuous maintenance /development in agile manner, codes needs to be handled by the operation environment (staging/production) - you may able to add development environment by your situation. In that case include_path is super useful.
- Queue implementation in PHP — To implement Queue, the easiest way can do it is using array() to store elements, and array_shift() to extract elements at the front of the Queue.
- Redis thread implementation in PHP without compilation on CentOS — Redis is a useful solution when design systems on server and required the minimum latency between the client and server for handling message. We can easily build a thread application if we use shell_exec based on & and wait. PHP also provides a solution named as pthreads. but unfortunately the pthreads extension cannot be used in a web server environment. Threading in PHP should therefore remain to CLI-based applications only.
- Removing memory limit in PHP — One of the case you need to remove memory limit in PHP will be that your application use memory dynamically and sometimes faces out of memory. In that case, you can simply remove memory limit in your PHP application by setting memory_limit=-1 in /etc /php.ini
- Resize image with keeping aspect ratio (width x height) in PHP — The code **cfolderResizeImage**() will enable you to resize the source image with keeping aspect ratio in PHP.
- Restart server daemon by checking dead processes in PHP — PHP is a good language to manage remote/local server. Below code shows how to restart daemons like httpd, mysqld, Confluence and Jira by PHP.
- Restore dumped MySQL data on CentOS 7 — Restoring dumped MySQL data is something painful. _restore_mysql.sh enables you to restore the dumped MySQL data easily.
- Run shell commands on remote server in SSH — I wanted to run a specific command through ssh protocol on my server. And this is super helpful to do some kind of remote processing by PHP easily, so I do not need to log on remote server.
- Run time consuming (background) tasks as post processing with no connection with browser client in PHP — You can flushe all response data to the client and finishes the request to run time consuming tasks with no connection with brower client by **fastcgi_finish_request**(). This allows for time consuming tasks to be performed without leaving the connection to the client open.
- Save a image file in PHP from a image in HTML that encoded in base64 — By copy & paste, we can embed image in the HTML that is mainly supported WYSWYG editor, but it may cause heavy loading time for the web browser. The web browser may work very well if such kind of image files are removed in the HTML. Below PHP code enables you to work more compactly.
- Saving a text file in PHP — **file_put_contents**($filename, $src) is useful function if you want to save a text file in PHP
- Secure and Scalable LAMP Service Architecture in AWS — LAMP means Linux + Apache + MySQL + PHP that is commonly used to develop web service for Start-ups that does not require paying license fee, because all of technologies are from open source.
- Set default timezone by date_default_timezone_set — When you want to set the default timezone used by all date/time functions, you can take one of two options: 1) put date_default_timezone_set() on your code, 2) put value in php.ini
- Simple function appending strings to a text file
- Simple PHP Web Shell — Simple & tiny PHP Web shell for executing unix commands from web page.
- substr($str, $start, $length) in PHP — **substr**() returns part of a string
- Thread implementation in PHP — A thread of execution is the smallest sequence of programmed instructions that can be managed independently by a scheduler, which is typically a part of the operating system. In the PHP, you can divide and conquer your planned process efficiently by Thread.
- Uncaught RedisException: Permission denied error in your PHP code — When you face an error Uncaught RedisException: Permission denied in your PHP code, you will need to check if that is working normally in the command line but happening in the web page. If it is, that is 100% caused by your linux security policy. In that case, you will need to turn SELinux policy off that located at **/etc/selinux/config**.
- Unicode Escape in PHP — PHP requires to implement a simple function to escape unicode. Some severs send uncode encoded string for supporting multi byte characters. Its encoded result will be something like \uBBF8\uC158 \uC774\uC2A4\uD0C4\uBD88 4: \uC775\uC2A4\uD2B8\uB9BC \uB370\uC774
- Upgrade PHP on CentOS 7/RHEL 7
- Upload multi selected files with single array variable in PHP — Array in HTML5 is useful when upload files with multi selection on file management dialog. Note that you should add tag (**multiple="multiple"**) to the "file" type element. In the PHP, you will be able to manage uploaded files easily by index.
- Upload multiple files serially and show its progress in real-time with XMLHttpRequest and PHP — Uploading files serially is strongly needed when uploading large files to the server, because it helps user to ensure the uploaded file is correctly delivered or not, so they can do the necessary things easily.
- Uploading a large sized file more than 2G — To upload a large sized file more than 2G in PHP, the OS must be based in 64bits instead of 32bits, and you should change some settings in **/etc/php.ini** and **LimitRequestBody**=0 in httpd.conf or virtualhost
- Using Redis in PHP on CentOS 7