Experiment No. 2

Title: Arrays, Image and File handling functions, User
Defined function in PHP

Batch: **Roll No.:**   **Experiment No.:2**

Aim: Write PHP script for demonstrating use of arrays, functions, image and file handling functions.

Resources needed: Windows OS, Web Browser, Editor, and XAMPP Server

---

Pre Lab/ Prior Concepts:

Students should have prior knowledge of HTML/CSS/Basic Programming.

Theory:

What is PHP Arrays
Arrays are complex variables that allow us to store more than one value or a group of values under a single variable name.

Types of Arrays in PHP
There are three types of arrays that you can create. These are:
  1. Indexed array — An array with a numeric key.
  2. Associative array — An array where each key has its own specific value.
  3. Multidimensional array — An array containing one or more arrays within itself.

  1. Indexed Arrays Example
     ```php
     <?php
          echo "<br><b> --CREATING & DISPLAYING NUMERIC ARRAY--
          </b><br><br>";
          $languages = array("PHP", "JAVA", "PYTHON", "C++");
          $languages[0]="C";   //will overwrite "PHP"at index 0
          $languages[4]="PHP";//will add PHP at index 4
          var_dump($languages);
     ?>
     ```
var_dump() function dumps information about one or more variables. The information holds type and value of the variable(s).

  2. Associative Array Example
     ```php
     <?php
          echo "<br><br><b> --CREATING & DISPLAYING ASSOCIATIVE ARRAY-
          - </b><br>";
          $subjectcode=array(111=>"C",222=>"JAVA",333=>"PYTHON",444=>"C++",555=>"PHP");
          echo "<br><br><b> ACCESSING ASSOCIATIVE ARRAY USING KEY
          </b><br>";
          echo "SUBJECT with code 333 is ::".$subjectcode[333];
          //echo $subjectcode[777]; //throw php error
     ```

```php
    echo "<br><br><b> DISPLAYING ASSOCIATIVE ARRAY USING KEY
    THROUGH LOOPS</b><br>";
    foreach ($subjectcode as $code => $subvalue){
       echo "SubjectCode $code ::$subvalue  <br>";}
     echo "<br><b> Displaying  raw value of array</b><br>";
    print_r($subjectcode);
?>
```

print_r() function prints the information about a variable in a more human-readable way.

3. Multidimensional Array

```php
<?php
 echo "<br><b> --CREATING & DISPLAYING MULTIDIMENSIONAL ARRAY --
</b><br>";
 $courses = array(
    array("code" => "001", "subject" => "PIC","sem" => "First"),
    array( "code" => "003","subject" => "PYTHON", "sem" => "Third"),
    array( "code" => "004", "subject" => "Advance JAVA/PYTHON", "sem" => "Fourth")
  );
 echo "<b> Display multidimensional array</b><br>";
$keys = array_keys($courses);
for($i = 0; $i < count($courses); $i++) {
   echo $keys[$i] . "<br>";
   foreach($courses[$keys[$i]] as $key => $value) {
      echo $key . " : " . $value . "<br>";
   } }

?>
```

array_keys() function returns an array containing the keys.

Functions in PHP

1. PHP Built-in Functions
   -A function is a self-contained block of code that performs a specific task.
   -PHP has a huge collection of internal or built-in functions that you can call directly
   within your PHP scripts to perform a specific task, like gettype(), print_r(), var_dump,
   etc.
2. PHP User-Defined Functions
   a. Syntax for creating Functions
   ```php
   function functionName(){
      // Code to be executed
   }
   ```
   b.
   ```php
   function myFunc($oneParameter, $anotherParameter){
      // Code to be executed
   }
   ```
   c.
   ```php
   function myFunc($oneParameter, $anotherParameter=value){
      // Code to be executed
   }
   ```

```
//calling such functions
$myFunc("value1", "value2"); $myFunc("value1");// both will work
```

d. function myFunc($oneParameter, $anotherParameter){
    return $returnvalue;
     }

Built in Functions:
Built in functions are functions that comes along with PHP installation package. The built in functions are what make PHP a very efficient and productive scripting language. The built in functions can be classified into many categories.
1) String Funtions
2) Date and Time Functions
3) Math Functions
4) Image Handling Functions
5) File Handling Funtions

File Handling Functions:
File handling in PHP is similar as file handling is done by using any programming language like C. PHP has many functions to work with normal files.

Those functions are:

1) fopen() – This function is used to open a file. First parameter of fopen() contains name of the file which is to be opened and second parameter tells about mode in which file needs to be opened, e.g.,

```
<?php
$file = fopen("demo.txt",'w');
?>
```

Files can be opened in any of the following modes :

"w" – Opens a file for write only. If file not exist then new file is created and if file already exists then contents of file is erased.
"r" – File is opened for read only.
"a" – File is opened for write only. File pointer points to end of file. Existing data in file is preserved.
"w+" – Opens file for read and write. If file not exist then new file is created and if file already exists then contents of file is erased.
"r+" – File is opened for read/write.
"a+" – File is opened for write/read. File pointer points to end of file. Existing data in file is preserved. If file is not there then new file is created.
"x" – New file is created for write only.

2) fread() — After file is opened using fopen() the contents of data are read using fread(). It takes

two arguments. One is file pointer and another is file size in bytes, e.g.,

```php
<?php
$filename = "demo.txt";
$file = fopen( $filename, 'r' );
$size = filesize( $filename );
$filedata = fread( $file, $size );
?>
```

3) fwrite() – New file can be created or text can be appended to an existing file using fwrite() function. Arguments for fwrite() function are file pointer and text that is to written to file.

```php
<?php
$file = fopen("demo.txt", 'w');
$text = "Hello world\n";
fwrite($file, $text);
?>
```

4) fclose() – file is closed using fclose() function. Its argument is file which needs to be closed, e.g.,

```php
<?php
$file = fopen("demo.txt", 'r');
fclose($file);
?>
```

Image Handling Function:
imagecreate() returns an image identifier representing a blank image of specified size. In general, we recommend the use of imagecreatetruecolor() instead of imagecreate() so that image processing occurs on the highest quality image possible.

imagecreate (int $width , int $height)

```php
<?php
header("Content-Type: image/png");
$im = @imagecreate(110, 20)
    or die("Cannot Initialize new GD image stream");
$background_color = imagecolorallocate($im, 0, 0, 0);
$text_color = imagecolorallocate($im, 233, 14, 91);
imagestring($im, 1, 5, 5,  "A Simple Text String", $text_color);
imagepng($im);
imagedestroy($im);
```

?>

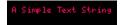The above example will output something similar to:

`A Simple Text String`

Image Upload using File Upload

The process of uploading a file follows these steps −

The user opens the page containing a HTML form featuring a text files, a browse button and a submit button.
The user clicks the browse button and selects a file to upload from the local PC.
The full path to the selected file appears in the text filed then the user clicks the submit button.
The selected file is sent to the temporary directory on the server.
The PHP script that was specified as the form handler in the form's action attribute checks that the file has arrived and then copies the file into an intended directory.
The PHP script confirms the success to the user.
An uploaded file could be a text file or image file or any document.

```php
<?php
  if(isset($_FILES['image'])){
    $errors= array();
    $file_name = $_FILES['image']['name'];
    $file_size =$_FILES['image']['size'];
    $file_tmp =$_FILES['image']['tmp_name'];
    $file_type=$_FILES['image']['type'];
    $file_ext=strtolower(end(explode('.',$_FILES['image']['name'])));

    $extensions= array("jpeg","jpg","png");

    if(in_array($file_ext,$extensions)=== false){
      $errors[]="extension not allowed, please choose a JPEG or PNG file.";
    }

    if($file_size > 2097152){
      $errors[]='File size must be excately 2 MB';
    }

    if(empty($errors)==true){
      move_uploaded_file($file_tmp,"images/".$file_name);
      echo "Success";
    }else{
      print_r($errors);
    }
```

```
 }
?>
<html>
  <body>

    <form action="" method="POST" enctype="multipart/form-data">
      <input type="file" name="image" />
      <input type="submit"/>
    </form>

  </body>
</html>
```

PHP GD

GD is an open-source code library that is required to create and manipulate images in PHP. It is used for creating PNG, JPEG, and GIF images. It is commonly used to create charts, graphics, thumbnails, etc, and website development is the most common application of GD
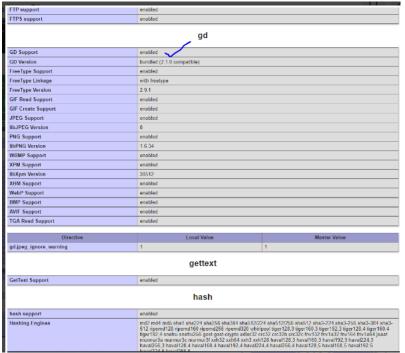
Installing PHP GD in Windows

To install the PHP GD follow the following steps:

Step 1: Install XAMPP in your windows system.
Step 2: Verify if GD is already installed or not. So to verify GD we need to follow the following steps:
- Open XAMPP and click on the start button in front of Apache and MySQL to start php server, and go to the admin.
- A web page will open. Go to the 'PHPInfo' option on the top of the page.
- A PHPInfo dashboard will open up.
- Scroll down the page and search for 'gd'. If present, GD is already installed
  - If 'gd' is not present on the phpinfo page, you can follow the next to install GD.

Step 3: Locate and open php.ini in your editor.

Step 4: Find **;extension=gd.**

Step 5: Remove semicolon from **;extension=gd** and save the file.
Step 6: Go to php folder. It is usually present in **C:\xampp**.
Step 7: Look for **php_gd.dll** in the ext folder.
Step 8: Copy php_gd.dll and paste it into the following folder.
    C:\Windows\System32
Step 9: Restart the XAMPP server. Now use the phpinfo() method as mentioned in step 2 to check whether the GD library is installed in the PHP server. It will show information about the PHP's configuration.

| FTP support | enabled |
| FTPS support | enabled |

gd

GD Support	enabled
GD Version	bundled (2.1.0 compatible)
FreeType Support	enabled
FreeType Linkage	with freetype
FreeType Version	2.9.1
GIF Read Support	enabled
GIF Create Support	enabled
JPEG Support	enabled
libJPEG Version	8
PNG Support	enabled
libPNG Version	1.6.34
WBMP Support	enabled
XPM Support	enabled
libXpm Version	30512
XBM Support	enabled
WebP Support	enabled
BMP Support	enabled
AVIF Support	enabled
TGA Read Support	enabled

Directive	Local Value	Master Value
gd.jpeg_ignore_warning	1	1

gettext

| GetText Support | enabled |

hash

| hash support | enabled |
| Hashing Engines | md2 md4 md5 sha1 sha224 sha256 sha384 sha512/224 sha512/256 sha512 sha3-224 sha3-256 sha3-384 sha3-512 ripemd128 ripemd160 ripemd256 ripemd320 whirlpool tiger128,3 tiger160,3 tiger192,3 tiger128,4 tiger160,4 tiger192,4 snefru snefru256 gost gost-crypto adler32 crc32 crc32b crc32c fnv132 fnv1a32 fnv164 fnv1a64 joaat murmur3a murmur3c murmur3f xxh32 xxh64 xxh3 xxh128 haval128,3 haval160,3 haval192,3 haval224,3 haval256,3 haval128,4 haval160,4 haval192,4 haval224,4 haval256,4 haval128,5 haval160,5 haval192,5 haval224,5 haval256,5 |

Output(Code with result Snapshot)

Questions:-

1. What is difference between array_merge and array_combine?
2. Write a program based on functions passing reference as parameter.
3. How can you display a file download dialog box using PHP?
4. What is the purpose of php.ini file?

Outcomes:

_____

_____

Conclusion: (Conclusion to be based on the objectives and outcomes achieved)

_____

_____

_____

_____

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of faculty in-charge with date

_____

References:

1. Instruction Link to install php GD –
https://www.geeksforgeeks.org/how-to-install-php-gd-in-windows/

Books:

1. Thomson PHP and MySQL Web Development Addison-Wesley Professional , 5th Edition 2016.
2. Peter MacIntyre, Kevin Tatroe Programming PHP O'Reilly Media, Inc, 4th Edition 2020
3. Frank M. Kromann Beginning PHP and MySQL: From Novice to Professional, Apress 1st Edition, 2018