Important notes about basic php:

Php Comments

--Php has mainly three types of comments e.g.

#Use this method to indicate an inline comment. You cannot include any executable code on the same line

// Use this method to indicate an inline comment. This can appear on its own line of code or added at the end of an executable line.

/\*

This indicates a multiline comment block. Do not place any executable code within this block of text.

\*/

Echo/Print

* Print and Echo both used for printing purpose.
* Print method can return a true/false (1/0) value.
* Echo does not return a value, but has been considered as a faster executed command.
* Example

$val = Print “LearnPHP7”;

Echo $val; //It will return 1, because it assumes the print work, since ‘print’ returns true/false (1/0) value, so it prints value 1, not LearnPHP7

* Using “echo” we can print multiple values but same is not possible with “print”.
* Print(‘I am at’,’LearnPHP7’);
* Echo(‘I am at’,’LearnPHP7);
* Comma(,) will not work in ‘print’ but works in ‘echo’
* Print function takes only one string at a time while as echo function takes more string at a time.
* Echo function is faster than print function.

Single Quotation and Double Quotation

-If we use single quotation in a statement then it will print the whole statement in output not parse the line?

-If we use double quotation in a statement then the parse each line and print the value of the variable.

🡪If we want to use a variable with a string without separation then we use curly brackets {} to separate the string.

1. php automatically convert a string into integer value like in this php will convert '12' into integer number

**Most Popular Array Functions in PHP:**

Count()🡪How many elements are present in the array.

Max()🡪Determine the maximum value from the array.

Min()🡪Determine the minimum value from the array.

Sort()🡪Sort the array in ascending order.

Rsort()🡪Sort the array in descending order.

Implode()🡪 Turn an array into string In other words combined values together to get a string

Explode()🡪 takes a string every time it takes the user divider as a new object in array

in\_array()🡪determines whether a element is present in array or not.

//Explode split the string in divider place the object as a new element in array

//Explode is extremly useful in comma separtor list

1. Boolean is a own type is always true or false.

NULL:

1. Null is fancy terms for nothing not zeros or not has a value.

Empty:

Empty values are : “”, null, 0, 0.0, “0”, false, array().

Type Juggling:

PHP converted one type to another type for us. For example if we want to add an integer with a string php converted the string in integer return the Boolean true and returns the value. This is referred as type juggling.

Type Casting:

Convert from one type to another type.

Type casting is done in two ways:

settype($var, “integer”)

(integer) $var

Common Data Type in PHP:

String,int,integer,float,array,bool,Boolean.

Type Casting:

1. More often we used two types of function for type casting one is for set the type named SETTYPE()🡪Conversion is done in place and another one is GETTYPE() to get the data type.
2. Custom type conversion is done if we assigned the conversion in a variable.
3. Custom type conversion is not a permanent conversion.

Constants:

1. Variable can change but constant can never change.

If Statement:

1. Always use curly brackets if even statement is in a single line it increases more readability.
2. Always apply indent ability.

Continue

1. Continue says as like go to the next element.
2. Is more useful if we want to skip several statements in the if condition.
3. Continue (1)🡪it defines the first loop or it denotes the loop from where it is originated.

Continue (2)🡪it defines the previous loop or parent of the of the continuous loop.

Break is also same as the continue loop at this point.

Array Pointers:

1. Is more useful to work with databases.

Array Pointers are

Current()🡪 current pointer value

Next()🡪 move the pointer forward

Previous🡪prev()🡪 move the pointer backward

Reset()🡪 move the pointer to the first element

End()🡪 move the pointer to the last element

Argument Advantages:

1. Arguments give flexibility and reusability of the code.

List()

1. list is the powerful tool to pack a variable and immediately unpack the variable.

**Common Problems:**

1. No output at all

🡪Try to access an HTML page

🡪Make sure web server is running

If this working then we

🡪Try to access a PHP page

For confirming php is working run following code.

<? php phpinfo(); ?>

If above tests are successful then problems is probably in our code.

🡪Make sure display errors are on and configured.

1. Most common php errors

🡪Typos: misspelled variable name; commented out code.

🡪 Missing semicolons at end of line

🡪Missing closing brace :})]

🡪Missing closing quote: “‘

🡪Case-sensitive variable names: $myvar vs. $myVar

🡪= vs. ==

1. PHP Errors

* Fatal Errors

🡪PHP understood the code but could not execute it

🡪Most common cause of failure is php call an undefined class or function.

* Syntax Errors

🡪PHP could not understood or process the code.

🡪PHP could not understand what you want to do.

🡪This occurs mainly missing semicolon, mismatch variable names, Typo, quotation mark, parenthesis etc..

🡪This type error is like that unexpected error.

* Warnings

🡪PHP found a problem, but was able to recover.

🡪Non failure

🡪Warnings occur in case of mathematical operation, incorrect number of arguments, and incorrect path to a file or you have don’t permission to access the file.

* Notices

🡪PHP is offering advice.

🡪Something smells bad.

🡪Good indicator of bug or sloppy programming.

1. Debugging and Troubleshooting

* Echo $variable //variable value
* Print\_r($array) //print readable array
* Gettype($variable) //variable type
* Var\_dump($variable) //variable type and value
* Get\_defined\_vars(); //array of defined variables
* Debug\_backtrace(); //show backtrace

Links and Urls

1. There are three ways to get information or data from users

* URLs/Links 🡪 GET request
* Forms 🡪POST ||
* Cookies 🡪COOKIE

1. COOKIE is not a request but it’s the way to access the cookie request.
2. Browser cookies restored on browser.
3. One of important things is that we can send lot of data by sending something query string.
4. Query string comes after page name e.g.

<a href="second\_page.php**?id=1**"><?php echo $link\_name;?></a>

**Query Parameter**:

Query parameter is part of the url it’s come after the question mark e.g. somepage.php?page=2

If we want to pass more than one variable via url then we use & percent like somepage.php?category = 7 & page=2

Real time example of query parameter is like that <http://google.com/search?q=php>

Super global variable

* “Super global” for short
* Always available in all scopes
* Assigned by PHP before processing page code
* There are 9 super global variables all together

1. $\_GET is a httpd method is used to pass value to the url.

**UrlEncode:**

* Letters, numbers, underscore, and dash are unchanged.
* Reserved characters become % + 2-digit hexadecimal.
* Spaces become “+”

**RawUrlEncode:**

* Letters, numbers, underscore, and dash are unchanged.
* Reserved characters become % + 2-digit hexadecimal.
* Spaces become “%20”

**UrlEncode vs. RawUrlEncode:**

* Rawurlencode the path

-Path is the part before the “?”

-Spaces must be encoded as %20

* Urlencode the query string

-Query string is the part after the “?”

-Spaces are better encoded as “+”

* Rawurlencode is more compatible generally
* URLs use letters, numbers, underscore and dash but some characters are not used in the urls which is known as reserved characters in urls which have special meaning in urls.
* We encode the reserved characters from urls without any affecting the function in url.
* Encoding is just for GET request encode is used for url encoding.
* Everything in left question mark is used rawurlencode() and everything in right in question mark is used urlencode().

**Encoding for HTML:**

* HTML has some reserved words like < > & “ which has a special meaning in html but if we want to this in browser exactly in same way then we need to encode them.
* In PHP it is done in two ways:

Htmlspecialchars() 🡪encoding only 4 reserved chars.

Htmlentities() 🡪encoding all types of reserved chars and words like trademark issues, currency, reserved chars etc.

Including and Requiring Files:

* If we need same code in a file which is located in another file then we can add the file or code of the file in our present file by using include function which is much better instead of copying code because copying code sometimes arise some bugs or looking arrangement is decresed.
* Include() is useful in following areas:

Include

Functions🡪because we can define them once and only once like form functions, database functions, general functions etc.

Layout sections🡪header, footer, sidebar i.e. navigational enhancement etc.

Reusable HTML/PHP code🡪banner ads, page analytics, and little snippets like images.

CSS and JavaScript.

* Include() there are three variations:

Require()🡪is same thing as include but it raise fatal errors if the file can’t be found it says like that look really it required not able to go forward if can’t find this file. Include() doesn’t do that include will try to find the file but it can’t find the file it can go to forward any way that may cause fail in later we try to call a function or class or something which is undefined.

Include\_once()🡪keeps an array of files to the files that his already included so as an includes the files it just add the file path to an array if we asked to included file again no it will because it is seems that it is included before. This great to way including functions because can’t define functions more than once without getting an error.

Require\_once()🡪same.

This four functions well organized our code.