Date:2021-01-27

* Intro. Of array:
  + - Array is a collection of data.
    - We can store a multiple data in single variablr.
    - Syntax:
      * $varname = array(‘ele1’,’ele2’,……elen);
    - For access data:
      * + $varname[index];
    - Example:
      * <?php
      * //here we create a array of food
      * $food = array('Samosa','Kachori','pizza');
      * //for accessing a perticular element from array..
      * echo "<br>$food[0]";
      * echo "<br>$food[2]<br>";
      * //for add a element in a array.
      * $food[count($food)] = "Dhosha";
      * //for see a whole array with key and value..
      * print\_r($food);
      * ?>
* Associative array:
  + - Associative array is a allow us to change key.
    - So we can set a key by our self.
    - We can set a key as a string. Or a numerical.
    - Syntax:
      * $varname = array(“key”=>value,”key2”=>value);
    - Example:
      * <?php
      * //associative array contain a key and value pair.
      * //creating a array with personal info.
      * $personal\_info = array('name' => 'jay' ,'education' => 'MCA' , 'age' => 21);
      * //for accessing a value.
      * echo $personal\_info['name'];
      * echo "<br>".$personal\_info['age']."<br>";
      * //now we add a new element in a personal\_info
      * $personal\_info['job'] = 'developer';
      * //for print whole array
      * print\_r($personal\_info);
      * ?>
* Multi-dimensional array:
  + - Example:
      * <?php
      * //we can create a n-dimensional array in php.
      * //create an array for student.
      * $student = array('name' => 'jay', 'hobby'=>
      * array('Cricket' ,'Basketball' ,'Running'),
      * 'result'=>
      * array('bca' => 81 , 'mca' => 80)
      * );
      * //code for access a element from an array
      * echo "<br>".$student['name'];
      * echo "<br>".$student['hobby'][0];
      * echo "<br>".$student['result']['bca'];
      * echo "<br>".$student['result']['mca'];
      * //for access whole array
      * echo "<br>";
      * print\_r($student);
      * echo "<br>";
      * print\_r($student['hobby']);
      * echo "<br>";
      * print\_r($student['result']);
      * ?>
* Foreach statement:
  + - Syntax:
      * foreach(varname as varname){

//code

}

* + - Example:
      * <?php
      * //foreach for single array.
      * $info = array('name' => 'jay','age' => 21,'job' => 'student' );
      * foreach ($info as $element) {
      * echo "$element<br>";
      * }
      * echo "<br><br>";
      * //code of foreach for a multidimensional array
      * $student = array('hobby'=>
      * array('Cricket' ,'Basketball' ,'Running'),
      * 'result'=>
      * array('bca' => 81 , 'mca' => 80)
      * );
      * foreach ($student as $info => $inner\_array) {
      * echo "<br>$info";
      * foreach ($inner\_array as $items) {
      * echo "<br>$items";
      * }
      * }
      * ?>
* Include and required function:
  + - Include and required are used to include a php file in current file.
    - If some peace of code are same in many pages then instead of write a code multiple time we can include that file in current file.
    - But difference between a include and required is if file is not exists then include give error and run the remaining code.
    - But if we use a required then it kill wall page and give an error…
    - Example of required:
      * <?php
      * //here we add file using a required.
      * require 'multiple.php';
      * echo "This is req.php file.";
      * //$content is initialized in multiple.php
      * echo "<br>".$content;
      * ?>
    - Example of include:
      * <?php
      * //add a file multiple.php using include.
      * include "multiple.php";
      * //$content is initialized in multiple.php
      * echo $content;
      * echo "<br>THis is include.php fire.";
      * ?>
* Include\_once and required\_once:
  + - Include\_once and required\_once both are work same like a include and required but difference is if file is already included then it does not include again.
* Expression matching (preg\_match()):
  + - It is used for match a string.
    - Example:
      * <?php
      * //function which return true if string match else false.
      * function iscontain($string){
      * if(preg\_match('/@/', $string)){
      * return true;
      * } else {
      * return false;
      * }
      * }
      * $str = "This is @ jay";
      * if (iscontain($str)) {
      * echo "$str contain @ sign...";
      * } else {
      * echo "$str doesnot contain @ sign...";
      * }
      * ?>
* String function – length:
  + - We use strlen() for count a character of string.
    - Example:
      * <?php
      * //strlen() for count a no of characters.
      * $str = "This is new india.";
      * echo strlen($str);
      * //get all character using string.
      * for ($i=0; $i < strlen($str) ; $i++) {
      * echo $str[$i];
      * }
      * ?>
* String function-upper and lower case:
  + - strtolower(str) – for lower case.
    - strtoupper(str) – for upper case.
    - Example:
      * <?php
      * if (isset($\_GET['name']) && !empty($\_GET['name'])) {
      * $name = $\_GET['name'];
      * $lower\_name = strtolower($name);
      * if ($lower\_name == 'jay') {
      * echo "YOu are good person $name";
      * }
      * }
      * ?>
      * <form method="get" action="#">
      * <input type="text" name="name">
      * <br>
      * <input type="submit" name="submit">
      * </form>
* String position:
  + - strops(‘arg1’,’arg2’,[arg3]) for string position.
    - arg1: a string in which you find.
    - arg2: which character you want to find.
    - arg3:from which position you want to find.
    - Ex:
      * <?php
      * $str = "This is a string. is this a string.";
      * $offset = 0;
      * $find = 'is';
      * $length = strlen($find);
      * while ($str\_pos = strpos($str, $find, $offset)) {
      * echo "<br>$find is find at position $str\_pos.";
      * $offset = $str\_pos + $length;
      * }
      * ?>
* Replacing a part of string:
  + - substr\_replace(arg1,arg2,arg3,arg4);
      * arg1:main string
      * arg2:new string which we want to replace.
      * arg3:starting
      * arg4:ending
    - Example:
      * + <?php
        + $string = "This is a php learning. a good person";
        + $new\_str = substr\_replace($string, "developing" ,14 ,9);
        + echo "$new\_str";
        + ?>
      * str\_replace(arg1,arg2,arg3):
        + arg1:find a string
        + arg2:replace a sring
        + arg3:actual string
        + Example:

<?php

//replace a string with one word.

$string = 'hello ,this is a jay piparava from marwadi university.';

echo str\_replace('hello', 'hi', $string);

//but when we want to find a one or more string with more replacement then....

$str1 = 'Hello , i am jay piparava. i am in marwadi university. my job is student.';

$find = array('Hello' ,'marwadi university' ,'student');

$replace = array('Hi', 'Cybercom creation', 'Intern');

echo "<br>".str\_replace($find, $replace, $str1);

?>