PHP Array

#PHP Notes

Array Function

List of Functions

- count ()
- array_count_values ()
- array_sum ()
- array_product ()
- array_reverse ()
- in_array ()
- array_rand ()
- array_unique ()
- array_merge ()
- array_search ()
- range ()
- sort ()
- rsort ()
- asort ()
- ksort () (Key Sort)
- krsort ( )
- shuffle ( )
- array_key_exists ( )
- array_change_key_case()
- array_combine()
- end ()
- compact()
- array_flip()
- array_diff()
- array_intersect()
- array_values ()
- array_push()
- array_pop()
- explode()
- implode()

array()

- Array() is used to create an array

  - **Numerical Array**
    - array(value1,value2,value3);

  - **Associative Array**
    - array(key=>value,

      key=>value,

      key=>value

      );

Numerical Array

count ()

- The **count()** array function in PHP will count the number of elements in an array.

Code :

```
<?php
 $arr = array('php','c','c++','java','Android');
 echo count($arr);
?>
```

Output : 5

array_count_values ()

- The function will return an associative array with the array values as keys, and their count is the value. It will only list distinct values as keys, and count all matches found for particular values in the array.
- Code :

```
<?php
$arr = array("C","php","c","android","php","java");
$newarr = array_count_values($arr);

foreach ($newarr as $key => $value) {
    echo "<br/>$key - <strong>$value</strong> ";
}
?>
```

```
C - 1
php - 2
c - 1
android - 1
java - 1
```

array_sum ()

- The array_sum() array function in PHP will calculate the sum of all values within an array.

- Syntax :

```
<?php
$myarray = array(1,2,3,4,5);
echo array_sum($myarray);
?>
```

- Output : 15

array_product ( )

- The **array_product()** array function in PHP will find the product of the target array by multiplying each value as it evaluates the array, compounding the values.

- **Code:**

```
 <?php
$myarray = array(1,2,3,4,5);
echo array_product($myarray);
?>
```

- **Output** : 120

array_reverse ( )

- The **array_reverse()** array function in PHP will reverse the elements in an array.

Code :

```
<?php
$arr = array('c', 'c++', 'java', 'android', 'php');
$revarr = array_reverse($arr);
print_r($revarr);
?>
```

Output :

```
Array
(
    [0] => php
    [1] => android
    [2] => java
    [3] => c++
    [4] => c
)
```

in_array ( )

- The **in_array()** array function in PHP is used when we wish to look inside of an array to see if a certain value exists. This function will return either "true“(1) or "false" when It runs.

- **Code :**

```
<?php
$arr = array('c', 'c++', 'java', 'android', 'php');
$temp = in_array('php',$arr);
echo $temp;
?>
```

- **Output :**

```
1
```

array_rand ( )

- The **array_rand()** array function in PHP will select one or more keys from an array at random.

```
<?php
$arr = array('c', 'c++', 'java', 'android', 'php');
$indexofarray = array_rand($arr);//Returns Array Index
echo $arr[$indexofarray];
?>
```

Select multiple keys at random

```
<?php
$arr = array('c', 'c++', 'java', 'android', 'php');
$indexofarray = array_rand($arr,2);//Returns Array Index

foreach ($indexofarray as $key => $value)
{
    echo "<br />$key - <strong>" . $arr[$value] . "</strong>";
}

?>
```

Output :

```
0 - php
1 - ios
```

array_unique ( )

- The **array_unique()** array function in PHP will remove all duplicate values from within an array.

Code :

```
<?php
$arr = array('c', 'c++','android', 'java', 'android', 'php');
print_r(array_unique($arr));
?>
```

Output :

```
Array
(
    [0] => c
    [1] => c++
    [2] => android
    [3] => java
    [5] => php
)
```

array_merge ( )

- The **array_merge()** array function in PHP will merge multiple arrays.

- **Code :**

```
<?php
$arr1 = array('c','c++','android','java','php');
$arr2 = array(10,20,30,40,50);

$NewArr = array_merge($arr1,$arr2);
print_r($NewArr);
?>
```

```
Array
(
    [0] => c
    [1] => c++
    [2] => android
    [3] => java
    [4] => php
    [5] => 10
    [6] => 20
    [7] => 30
    [8] => 40
    [9] => 50
)
```

array_search ( )

- The **array_search()** array function in PHP will search in an array for a specific value. If found it will return the corresponding key for that value. It is a good way to check to see if a value exists in an array.

- Code :

```
<?php
$myarr = array('c', 'c++','android', 'java', 'php');
$check = array_search('android', $myarr);
echo $check; // Return Index
?>
```

- Output :

```
2
```

range ( )

- The **range()** array function in PHP will create a new array using a specified range. You can use letter or number ranges.

```
<?php
$arr = range("11", "20");
foreach($arr as $key => $value)
{
    echo "<br /> $key - $value ";
}
?>
```

- $a = range('a','z');
- print_r($a);

```
$year = range(1900,2020);

echo "<select>";
foreach ($year as $value)
{
    echo
"<option>$value</option>";
}
echo "</select>";
```

sort ( )

- The **sort()** array function in PHP will sort an array by its values from lowest to highest. It will sort letters and numbers in an ascending order.

- Code :-

```
<?php
$arr = array(80,60,50,40,8,74);
sort($arr);
print_r($arr);
?>
```

- Output :

```
Array ( [0] => 8 [1] => 40 [2] => 50 [3] => 60 [4] => 74 [5] =>
80 )
```

rsort ( )

- The **rsort()** array function in PHP will sort an array in reverse or descending order. This function behaves opposite the sort() function.

```
<?php
$arr = array(80,60,50,40,8,74);
rsort($arr);
print_r($arr);
?>
```

- Array ([0] => 80 [1] => 74 [2] => 60 [3] => 50 [4] => 40 [5] => 8)

asort ( )

- The **asort()** array function in PHP will sort all of the elements in an array in ascending order according to the values. The indexes will be preserved.

- Code:

```
<?php
$arr = array(80,60,50,40,8,74);
asort($arr);
print_r($arr);
 ?>
```

- Array ([4] => 8 [3] => 40 [2] => 50 [1] => 60 [5] => 74 [0] => 80)

ksort ( ) (Key Sort)

- The **ksort()** array function in PHP will sort an array in ascending order according to the keys.

- Code :-

```
<?php
$arr = array(
    "Banana" => "Yellow",
    "Apple" => "Red",
    "Mango" => "Green");
ksort($arr);
foreach ($arr as $key => $value) {
    echo "<br />$key - $value ";
}
```

```
Apple - Red
Banana - Yellow
Mango - Green
```

krsort ( ) Key Reverse Sort

- The **krsort()** array function in PHP will sort an array in reverse according to the keys.

- **Code :**

```
<?php
$arr = array(
    "Banana" => "Yellow",
    "Apple" => "Red",
    "Mango" => "Green");
krsort($arr);
foreach ($arr as $key => $value) {
    echo "<br />$key - $value ";
}
```

```
Mango - Green
Banana - Yellow
Apple - Red
```

shuffle ( )

- The **shuffle()** array function in PHP will randomly shuffle the order of the elements in an array.

- Code:

```
<?php
$myArray = array("Football", "Baseball", "Hockey", "Tennis",
                 "Boxing");
shuffle($myArray);          // Shuffle the array
foreach ($myArray as $key => $value) {
    echo "<br /> $value ";
}
```

```
Hockey
Football
Baseball
Boxing
Tennis
```

Note:Output may be Different on each refresh

array_key_exists ( )

- The **array_key_exists()** array function in PHP will return either "true" or "false" when it runs.
- It takes two parameters to operate. The first parameter is the key that you wish to find out if it exists, and the second parameter is the target array to search in for that key or index.

- Code :

```
<?php
$arr = array("a" => "apple", "b" => "banana");
echo array_key_exists('a',$arr);
?>
```

- Output:
- 1

array_change_key_case()

- The array_change_key_case() function is used to change all keys of a given array to lower case or upper case.
- array_change_key_case (input_array, input_case)
- Either CASE_UPPER or CASE_LOWER.
  Default value : CASE_LOWER
- **Code:**

```php
<?php
$arr = array(
    "Banana" => "Yellow",
    "Apple" => "Red",
    "Mango" => "Green");
$uppercase = array_change_key_case($arr,CASE_UPPER);
print_r($uppercase);
?>
```

```
Array
(
   [BANANA] => Yellow
   [APPLE] => Red
   [MANGO] => Green
)
```

array_combine()

- The array_combine() function is used to creates an array by using one array for keys and another for its values.
- array_combine(array_keys, array_values)

- Code :

```
<?php
$arr1 = array("Banana" , "Apple" , "Mango");
$arr2 = array("Yellow","Red", "Green");
$NewVar = array_combine($arr1, $arr2);
print_r($NewVar);
?>
```

```
Array
(
    [Banana] => Yellow
    [Apple] => Red
    [Mango] => Green
)
```

- Note : Both Array Elements Must be Equal

end ()

- The **end()** array function in PHP will return the value of the last element in an array.

- Code :

```
<?php
$myarr = array('c', 'c++','android', 'java', 'php');
echo end($myarr);
?>
```

Output: php

compact()

- Create an array from variables and their values:

```php
<?php
 $name = "akash";
 $subject = "php";
 $arr = compact("name", "subject");
 print_r($arr);
?>
```

```
Array
(
    [name] => akash
    [subject] => php
)
```

array_flip()

- Flip all keys with their associated values in an array:

```
<?php
$arr = array("a" => "akash", "b" => "banana", "c" => "computer");
$fliparray = array_flip($arr);
print_r($fliparray);
?>
```

```
Array
(
    [akash] => a
    [banana] => b
    [computer] => c
)
```

array_diff()

- Compare the **values** of two arrays, and return the differences:

```
<?php
$a=array("akash","c","c++","java","php","android");
$b=array("akash","c","java","php");
$diff = array_diff($a, $b);
print_r($diff);
?>
```

```
Array
(
    [2] => c++
    [5] => android
)
```

array_intersect()

- The array function in PHP will create a new array from comparing arrays against each other to see if any values match.
- It will return an associative array with all of the matches found in the arrays you compare.

```
<?php
$arr1 = array(0=>"Sunday",1=>"Monday",2=>"Tuesday");
$arr2 = array(3=>"Tuesday",4=>"Sunday",5=>"Thursday");

print_r(array_intersect($arr1,$arr2));
?>
```

```
Array
(
    [0] => Sunday
    [2] => Tuesday
)
```

array_values ( )

- The array_values() array function in PHP will return a new array made up of the values from a target array.
- The resulting array will be a numerically indexed array.

```
<?php
$arr = array("i"=>"c","want"=>"c++","learn"=>"php");
$myarr = array_values($arr);
foreach($myarr as $key => $value) {
    echo " <br />$key - $value";
}
?>
```

```
0 - c
1 - c++
2 - php
```

array_push()

- The array_push() function inserts one or more elements to the end of a Numerical array.
- array_push(array,value1,value2...)

```
<?php
    $a=array("c","c++");
    array_push($a,"java","php");
    print_r($a);
?>
```

```
Array
(
   [0] => c
   [1] => c++
   [2] => java
   [3] => php
)
```

array_pop()

- The array_pop() function deletes the last element of an array.
- array_pop(*array*)

```
<?php
$a = array("c", "c++", "Java", "PHP", "ASP");
array_pop($a); //Remove
print_r($a);
array_pop($a); //Remove
print_r($a);
?>
```

explode()

- Explode function is used to break a string into an array.
- String to Array
- explode($delimiter, $string);

```
<?php
    $mystring = "I Love PHP Language";
    $arr =  explode(" ",$mystring);
    print_r($arr);
?>
```

```
Array
(
   [0] => I
   [1] => Love
   [2] => PHP
   [3] => Language
)
```

Array ([0] => I [1] => Love [2] => PHP [3] => Language)

implode()

- Php implode function is used to convert an array into string. It return output as a string.
- Array to String

```
<?php
    $arr = array("i","love","php","language");
    $mystring = implode(" ", $arr);
    echo $mystring;
?>
```

```
i love php language
```

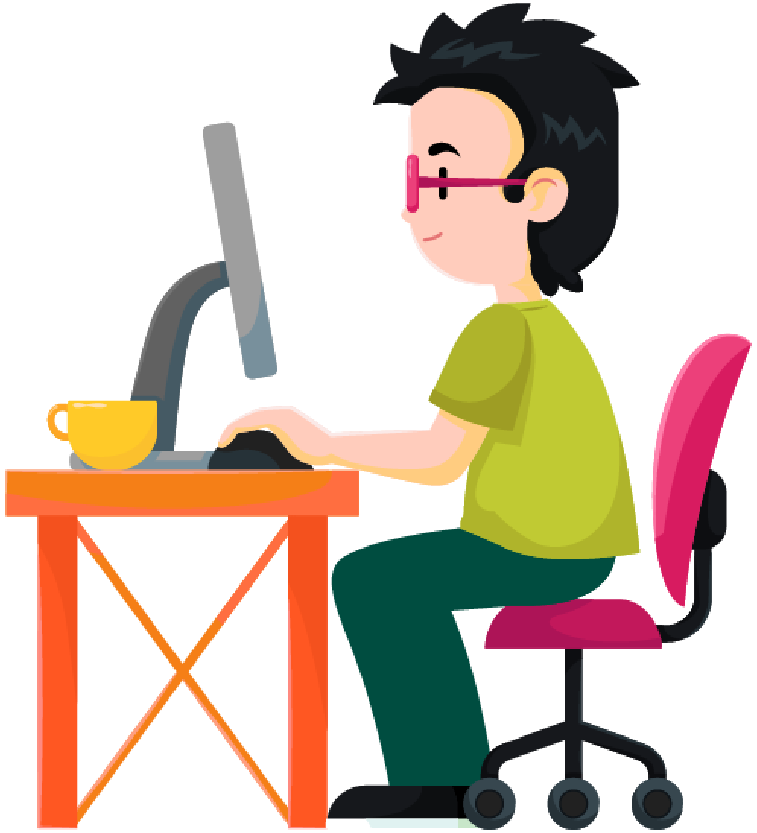
Example :

```
<form method="post">
        Programming :
 C<input type="checkbox" name="cbox[]" value="c">
 C++<input type="checkbox" name="cbox[]" value="c++">
 PHP<input type="checkbox" name="cbox[]" value="php">
       <input type="submit">
</form>
<?php
if($_POST)
{
    $programming= $_POST['cbox'];
    $string = implode(",", $programming);
    echo $string;
}
?>
```

Task

- Create one Associative Array Define A For , B For etc. Take a Value from user using get/post if user enter A check a is present or not in array if it is present then display respective value using array function.
- Take 5 Numbers from user store into array and print asc order and sum of all number.
- Take values from user in checkbox and print as a String.
- On button click print array value in shuffle order.

Get Exclusive Video Tutorials



www.aptutorials.com
https://www.youtube.com/user/Akashtips

Get More Details

www.akashsir.com

If You Liked It !

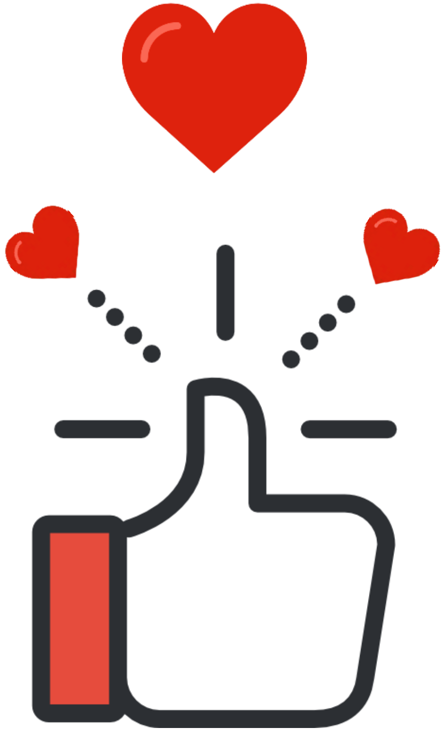
Rating Us Now

Just Dial

https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/079PXX79-XX79-170615221520-S5C4_BZDET

Sulekha

https://www.sulekha.com/akash-technolabs-navrangpura-ahmedabad-contact-address/ahmedabad



Connect With Me

Akash Padhiyar
#AkashSir

www.akashsir.com
www.akashtechnolabs.com
www.akashpadhiyar.com
www.aptutorials.com

# Social Info

akash.padhiyar

akashpadhiyar

Akash_padhiyar

+91 99786-21654

#AkashPadhiyar
#aptutorials