Built in functions in PHP 3

As we discussed functions in previous lectures, so in this lecture we will continue to discuss following questions by using functions.

* How to check if variable is null?
* How to check if variable is a number?
* How to remove the first element from array?
* How to remove last element from array?
* How to remove duplicate values from array?
* How to delete an element from an array?
* How to insert elements to the beginning of array?
* How to insert elements at the end of array?
* How to change lower case letters to upper case?
* How to get values from an associative array not the string keys?

# is\_null()Function:

The (**is\_null**) is a built-in function which is used to check whether the variable is null or not. It only returns the boolean value.

**Syntax:**

is\_null($variable)

* $variable is used to pass a variable name to check whether it is NULL or not.
* It accepts only parameter as described in the syntax.
* Returns only true or false.

**Code:**

<?php

$name="Olivia";

if(is\_null($name)){

 echo "True <br>";

}else{

 echo "False <br>";

}

$stu\_name=Null;

if(is\_null($stu\_name)){

 echo "True";

}else{

 echo "False";

}

?>

**Output:**

False

True

Figure 1 illustrates the above mentioned code.



Figure

# is\_numeric ()Function:

The( **is\_numeric**) is a built-in function which is used to check whether the variable which is passed as a parameter is a number or a numeric string . It will also only return the boolean value.

**Syntax:**

is\_numeric($variable)

**Code:**

<?php

$num="12";

if(is\_numeric($num)){

 echo "True <br>";

}else{

 echo "False <br>";

}

?>

**Output:**

True

Here is the screen shot of the current program.

# 1.PNG

Figure

# array\_shift ()Function:

The (**array\_shift**) is a built-in function which is used to remove the first element from an array. It returns the array with the removed value. If the keys are numeric then all the keys get new values starting from 0 and which increases to 1.

**Syntax:**

array\_shift(array)

**Code:**

<?php

$names= array("Olivia","Robert","Mark","Isabella","Emma","Nick");

$remove=array\_shift($names);

print\_r($names);

echo " <br> The name which is removed is ".$remove ;

?>

**Output**

Array ( [0] => Robert [1] => Mark [2] => Isabella [3] => Emma [4] => Nick )
The name which is removed is Olivia

In the above example, a variable with name ($name) is initialized as an array having values in it.(array\_shift) is used here to remove the first value from the array.($remove) variable holds the removed element.Now we use (print\_r) function to print an array.

Here is the snippet of the code.



Figure

# array\_pop()Function:

The (array\_pop) is a built-in function which is used to remove the last element from an array. It returns the array with the removed value.

**Syntax:**

array\_pop (array)

**Code:**

<?php

$names= array("Olivia","Robert","Mark","Isabella","Emma","Nick");

$remove=array\_pop($names);

print\_r($names);

echo " <br> The name which is removed is ".$remove;

?>

**Output**

Array ( [0] => Olivia [1] => Robert [2] => Mark [3] => Isabella [4] => Emma )
The name which is removed is Nick

Here is the snapshot of the code.



Figure

# array\_unique()Function:

The array\_unique is a built-in function which is used to remove all the duplicate values from the array. Itreturns the array with only unique values.

**Code:**

<?php

$names= array("Olivia","Robert","Mark","Nick","Isabella","Emma","Nick");

$unique\_array=array\_unique($names);

print\_r($unique\_array);

?>

**Output**

Array ( [0] => Olivia [1] => Robert [2] => Mark [3] => Nick [4] => Isabella [5] => Emma )

Figure 5 illustrates the current program.



Figure

# array\_unset()Function:

The (**array\_unset**) is a built-in function which is used to delete an element from an array.

Here is the example to delete an element from an associative array.

**Code:**

<?php

$arr = array("a" => "Apple", "b" => "Ball", "c" => "Cat");

unset($arr["b"]);

print\_r($arr);

?>

**Output:**

Array ( [a] => Apple [c] => Cat )

Screenshot of the code.



Figure

Here is the example to delete an element from an indexed array.

**Code:**

<?php

$arr2 = array(1, 2, 3);

unset($arr2[0]);

print\_r($arr2);

?>

**Output:**

Array ( [1] => 2 [2] => 3 )

Figure 7 is the illustration of the code.



Figure

# array\_unshift()Function:

The (**array\_unshift**) is a built-in function which is used to insert one or more elements to the beginning of array. Indexed array starts from 0 but the associative array key remains the same and new inserted elements have always a numeric key.

 **Syntax:**

array\_unshift(array,value1,value2,value3,..)

**Code:**

<?php

$hobbies = array("Acting", "Drawing", "Music", "Films");

array\_unshift($hobbies, "Photography");

print\_r($hobbies);

?>

**Output**

Array ( [0] => Photography [1] => Acting [2] => Drawing [3] => Music [4] => Films )

Here is the snippet of the above mentioned program.

# 1.PNG

Figure

# array\_push()Function:

The (**array\_push**) is a built-in function which is used to insert elements at the end of array.Indexed array starts from 0 but the associative array key remains the same and new inserted elements has always a numeric key.

**Syntax:**

array\_push (array,value1,value2,value3,..)

**Code:**

<?php

$hobbies = array("Acting", "Drawing", "Music", "Films");

array\_push ($hobbies, "Photography");

print\_r($hobbies);

?>

**Output**

Array ( [0] => Acting [1] => Drawing [2] => Music [3] => Films [4] => Photography )

Here is the snapshot of the code.



Figure

# strtoupper() Function

The **strtoupper** function coverts the string into upper case. This function takes the string as parameter and converts the all lower case English alphabets into upper case English alphabets. All other numeric character and special character remains unchanged.

**Syntax:**

strtolower($string)

**Code:**

<?php

$str = "this is a program of strtoupper function";

echo strtoupper($str);

?>

**Output**

THIS IS A PROGRAM OF STRTOUPPER FUNCTION

In this program, PHP function (**strtoupper**) changes the lower-case letters into upper-case letter.

Here is the pictorial illustration of the current program.



Figure

# array\_values()Function:

The (**array\_values**) is a built-in function which is used to return all the values of the array but not the keys. In associative array when we want only the values to be printed then we use **array\_values** function. It accepts only one parameter to be passed which is an array.

**Syntax**

array\_values(array)

**Code:**

<?php

$hobbies= array("Olivia"=> "Acting","Robert"=>"Drawing","Mark"=>"Music","Nick"=>"Films","Isabella"=>"Photography");

 $values=array\_values($hobbies);

print\_r($values);

?>

**Output:**

Array ( [0] => Acting [1] => Drawing [2] => Music [3] => Films [4] => Photography )

Figure 11 shows the code in an editor.



Figure