ASSOCIATIVE ARRAY.

An array with a numeric index. Values are stored and accessed in linear fashion. Associative array – An array with strings as index. This stores element values in association with key values rather than in a strict linear index order.

PROGRAM:

```
<?php

echo "<br>" . "OUTPUT:" . "<br>";

$fruits = array('a'=>'apple','b'=>'banana','c'=>'pea');

echo "fruit[a] is " .$fruits['a'];

?>
```

OUTPUT: fruit[a] is apple

MODIFY ELEMENTS.

To modify elements in array.

PROGRAM:

```
<?php
echo "<br>" . "OUTPUT:" . "<br>";
$arr=array('a','b','c','d');
$arr[2]='z';
for($i=0;$i<count($arr);$i++)
{
echo"$arr[$i]"."<br>";
}
?>
```

OUTPUT:

A b z d

REMOVE ARRAY ELEMENTS

```
<?php
echo "<br>" . "OUTPUT:" . "<br>";
$arr=array('a','b','c','d');
unset($arr[2]);
for($i=0;$i<count($arr);$i++)
{
echo"$arr[$i]" . "<br>";
}
?>
```

OUTPUT:

A b

RETRIEVAL ARRAY SIZE.

```
<?php
$data=array('a','b','c');
echo "<br>" . "OUTPUT:" . "<br>";
echo "Array Size" . count($data) . "element";
?>
```

OUTPUT: Array Size3element

NESTED ARRAY.

```
<?php
echo "<br>" . "OUTPUT:" . "<br>";
$phonebook=array(
array('name'=>'ram','telno'=>'66665489878'),
array('name'=>'david','telno'=>'657665489878'));
print_r ( $phonebook[1]);
?>
```

OUTPUT:

Array ([name] => david [telno] => 657665489878)

PROCESS ARRAY WITH LOOP.

```
<?php
echo "<br>" . "OUTPUT:" . "<br>";
$city=array('kanpur','delhi','hydrabad');
for($i=0;$i<count($city);$i++)
print ($city[$i]."\n<br>");
print "<br>";
?>
 <?php
$city=array('kanpur','delhi','hydrabad');
foreach($city as $c)
print ($c."\n<br>");
print "<br>";
?>
```

```
<?php
$city=array('kanpur'=>'india','delhi'=>'india','hydrabad'=>'india');
foreach($city as $key=>$value)
print ("$value is in $key"."\n<br>");
print "<br>";
?>
```

OUTPUT:

```
kanpur
delhi
hydrabad

kanpur
delhi
hydrabad

india is in kanpur
india is in delhi
india is in hydrabad
```

FOREACH loop

This loop is used to iterate over arrays. For every counter of loop, an array element is assigned and the next counter is shifted to the next element.

PROGRAM:

```
<?php

echo "<br>" . "OUTPUT:" . "<br>";

$fruits = array('a'=>'apple','b'=>'banana','c'=>'pea');

foreach ($fruits as $value)

echo $value."<br>";

echo "<br>";

foreach ($fruits as $key => $value)

echo "$key: $value<br>";

?>
```

OUTPUT:

```
apple
banana
pea

a: apple
b: banana
c: pea
```

EXPLODE FUNCTION.

The explode function is used to "Split a string by a specified string into pieces i.e. it breaks a string into an array". The explode function in PHP allows us to break a string into smaller text with each break occurring at the same symbol. This symbol is known as the delimiter.

Using the explode command we will create an array from a string. The explode() function breaks a string into an array.

PROGRAM:

```
<?php

$arr="Twinker Tanker Tank";

$str=explode(" ",$arr);

echo "<br>" . "OUTPUT:" . "<br>";

print_r($str);

?>
```

OUTPUT: Array ([0] => Twinker [1] => Tanker [2] => Tank)

IMPLODE FUNCTION.

The implode function in PHP is used to "join elements of an array with a string".The implode() function returns a string from elements of an array. It takes an array of strings and joins them together into one string using a delimiter (string to be used between the pieces) of your choice. The implode function in PHP is easily remembered as "array to string", which simply means that it takes an array and returns a string.It rejoins any array elements and returns the resulting string, which may be put in a variable.

PROGRAM:

```
<?php
$arr=array('Twinker','Tanker','Tank');
$str=implode(" and ",$arr);
echo "<br>" . "OUTPUT:" . "<br>";
print("$str");
?>
```

OUTPUT: Twinker and Tanker and Tank.