PHP echo and print Statements

There are two basic ways to get the output in PHP:

- echo
- Print

Difference between echo and print

echo

- echo is a statement, which is used to display the output.
- echo can be used with or without parentheses.
- echo does not return any value.
- We can pass multiple strings separated by comma (,) in echo.
- echo is faster than print statement.

print

- print is also a statement, used as an alternative to echo at many times to display the output.
- print can be used with or without parentheses.
- print always returns an integer value, which is 1.
- Using print, we cannot pass multiple arguments.
- print is slower than echo statement.

PHP Control Structures and Loops

- The control structure controls the flow of code execution in application.
- PHP supports a number of different control structures:

1. if
2. else
3. elseif
4. switch
5. while
6. do-while
7. for
8. foreach

An array is a special variable, which can hold more than one value at a time.

Create an Array in PHP

In PHP, the array() function is used to create an array.

For example: If you have a list of items (a list of car names, for example), the array can be created as follows:

```
<?php
$cars = array("Volvo", "BMW", "Toyota");
echo "I like " . $cars[0] . ", " . $cars[1] . " and " . $cars[2] . ".";
?>
```

Output: I like Volvo, BMW and Toyota.

Count()

Get The Length of an Array - The count() Function

The count() function is used to return the length (the number of elements) of an array.

Eg:-

```
<?php
$cars = array("Volvo", "BMW", "Toyota");
echo count($cars);
?>
```

Output: 3

PHP Date/Time Functions

Function	Description
date_add()	- Adds days, months, years, hours, minutes, and seconds to a date
date_diff()	- Returns the difference between two dates
date_format()	- Returns a date formatted according to a specified format

PHP Function Arguments

Information can be passed to functions through arguments. An argument is just like a variable. Arguments are specified after the function name, inside the parentheses. You can add as many arguments as you want, just separate them with a comma. The following example has a function with one argument ($fname). When the familyName() function is called, we also pass along a name (e.g. Jani), and the name is used inside the function, which outputs several different first names, but an equal last name:

Example:

```
<?php
function familyName($fname) {
  echo "$fname <br>";
}
 familyName("Jani");
familyName("Hege");
familyName("Stale");
familyName("Kai Jim");
familyName("Borge");
?>
```

Result:

Jani
Hege
Stale
Kai Jim
Borge

- **Cookies**

  A mechanism for storing data in the remote browser and thus tracking or identifying return users

- **Sessions**

  It is support in PHP consists of a way to preserve certain data across subsequent accesses. This enables you to build more customized applications and increase the appeal of your web site.

Filters- Introduction

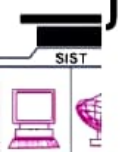
- PHP Filter is an extension that filters the data by either sanitizing or validating it
- It plays a crucial role in security of a website, especially useful when the data originates from unknown or foreign sources, like user supplied input
- For example data from a TML form

- There are mainly two types of filters
    1. **Validation**
    2. **Sanitization**

```php
<?php
// PHP program to validate URL

// Declare variable and initialize it to URL
$url = "https://www.geeksforgeeks.org";

// Use filter function to validate URL
if (filter_var($url, FILTER_VALIDATE_URL)) {
    echo("valid URL");
}
else {
    echo("Invalid URL");
}
```



Exception Handling in PHP

- An exception is unexpected program result that can be handled by the program itself. Exception Handling in PHP is almost similar to exception handling in all programming languages

PHP provides following specialized keywords for this purpose

- **try:** It represent block of code in which exception can arise
- **catch:** It represent block of code that will be executed when a particular exception has been thrown
- **throw:** It is used to throw an exception. It is also used to list the exceptions that a function throws, but doesn't handle itself
- **finally:** It is used in place of catch block or after catch block basically it is put for cleanup activity in PHP code