V Skills

Skills for a secure future

Certified PHP Developer Sample Material

V-Skills Certifications

A Government of India
&
Government of NCT Delhi Initiative

V-Skills

Skills for a secure future

1. BASICS

1.1 Need for PHP

PHP is probably the most popular scripting language on the web. It is used to enhance web pages. With PHP, many things can be done like create username and password login pages, check details from a form, create forums, picture galleries, surveys, and a whole lot more

The need for PHP is growing by the day and is primarily due to various advantages as

- ✓ Open Source, readily available and dual-licensed.
- ✓ Very Easy to understand Syntax
- ✓ Interfaces very easily with Apache/MySQL
- ✓ Server side (no need for client installs).
- ✓ Many useful libraries for working with PDFs, graphics, etc.
- ✓ Lots of good books and on-line help (php.net is great)
- ✓ Platform agnostic, can run on Windows Linux or Mac servers. Also very scalable.
- ✓ Lots of hosting services have it ready to use, no special configuration (except for special security needs)
- ✓ Pretty easy to access other web-based tools through PHP (i.e. google maps, etc.)
- ✓ PHP is flexible. Use OOP or not. Use naming convention(s) or not
- ✓ It can be optimized, even "compiled" for performance closer to that of more established compiled languages

1.2 PHP Evolution

PHP 5.3, is the current version. PHP development can be traced back to 1994, when a developer named Rasmus Lerdorf first created a set of CGI scripts to monitor page views for his online résumé. This early version of PHP, named PHP/FI, was very basic in nature, although it had support for form input and the mSQL database. It lacked many of the security features and add-ons found in modern PHP versions.

Lerdorf later improved PHP/F1 1.0 and released it as PHP/FI 2.0, but it was only in 1997, when the developers Andi Gutmans and Zeev Suraski rewrote the PHP parser and released it as PHP 3.0, that the PHP popularity picked up. As PHP 3.0's syntax was more powerful and consistent, but it also introduced a new, extensible architecture that encouraged independent developers to create their own enhancements and extensions to the language. Needless to say, this only encouraged adoption of the language, and it wasn't long before PHP 3.0 began appearing on many thousands of Web servers.

The next iteration, PHP 4.0, was released in 2000. It offered a new engine, better performance and reliability, and built-in support for sessions and object-oriented features. PHP 4.x was the dominant version in use on the Internet's Web sites but, in July 2007, the PHP development team announced that PHP 4. x would no longer be supported after December 2007, paving the way for more widespread adoption of newer versions.

PHP 5.0, released in 2004, was a radical redesign of PHP 4.0, boasting a completely rewritten engine, a much-improved object model, and various security and performance improvements. Of particular interest to developers was the new object model, which now included support for such stalwarts of the OOP paradigm as abstract classes, destructors, multiple interfaces, and class type hints. PHP 5.0 also introduced various new and important tools: a common database access layer; Java-style exception handling; and an integrated database engine.

PHP 5.3, the most recent version, was released in January 2008. It improves on the new features first shown in PHP 5.0, and also attempts to correct some of the shortcomings noted by users of earlier versions. Some of the most noticeable improvements in this version are: support for namespaces; a cleaner and more secure environment for managing the variable space; built-in support for SQLite 3; and a new native driver for MySQL.

1.3 PHP Usage

PHP is used in web pages by embedding PHP in HTML as

```
<HTML>
<HEAD>Sample First PHP Script</HEAD>
<BODY>
The following prints "Hello, Vskills":
<?php
   print "Hello, Vskills";
?>
</BODY>
</HTML>
```

In this example, the PHP code sits embedded in HTML. Every time the PHP interpreter reaches a PHP open tag "<?php" , it runs the enclosed code up to the delimiting "?>" marker.

1.4 PHP Comments

Comments in PHP can be written in three different ways:

- ✓ **Multi Line**

```
/*  Multi Line comment Line 1
 *  Line 2
 */
```

- ✓ **Single Line**

```
// Single line comment
```

- ✓ **Single Line with #**

```
# This is a shell like comment which is limited to single line only.
```

1.5 PHP Script Life cycle

Every PHP script undergoes the following life cycle

- ✓ Users sends HTTP request for PHP page or script to execute from web browser
- ✓ Web server gets requests and directs all .php suffix URLs to PHP interpreter by invoking it.
- ✓ PHP interpreter parse and executes it. Database operations are executed if requested.
- ✓ Results are sent to user's web browser