

## آشنایی با زبان برنامه نویسی PHP

پی اچ پی (PHP) یکی از پر طرفدارترین زبان‌های برنامه نویسی متن باز (Open Source) است که بیشتر برای طراحی برنامه‌های سمت سرور در وب سایت‌های پویا (Dynamic) مورد استفاده قرار می‌گیرد (مخفف Hypertext Preprocessor پیش‌پردازنده فرامتن) می‌باشد. در سال ۱۹۹۴ ایجاد شد. رسموس لردروف (Rasmus Lerdorf) ایجاد کننده اولیه آن بوده است ولی در طی زمان این زبان برنامه‌سازی، توسط سایر کارشناسان و برنامه‌نویسان تکمیل گردیده و به شکل کنونی آن درآمده است.

از زبان‌های اسکریپت‌نویسی تحت وب است. ساختار این زبان بسیار شبیه زبان سی (C) و زبان برنامه‌نویسی پرل (Perl) می‌باشد. شکل پذیری فوق العاده آن و نیز همگونی با اغلب بانک‌های اطلاعاتی از قبیل MySQL قدرت آن را افزایش داده است.

«مشهورترین نرم‌افزارهای (بسته‌های) ایجاد شده با PHP عبارت‌اند از:

phpBB , SMF , Mambo , Joomla , PHPNuke , mybb و مدیاویکی (MediaWiki)

سادگی استفاده از PHP و شباهت آن به زبان C و Perl و از نسخه ۵ آن به جاوا) باعث شده که اغلب برنامه‌نویسان با تجربه در کوتاه‌ترین زمان ممکن برنامه‌های کاملی را ایجاد کنند.

PHP تنها یک زبان اسکریپت‌نویسی نیست و با استفاده از PHP-GTK می‌توان برای طراحی برنامه‌های با ظاهر گرافیکی (GUI) و همچنین طراحی برنامه‌های خط فرمان شبیه پرل (Perl) یا پایتون (Python) از آن استفاده کرد.

PHP امکان استفاده از انواع مختلفی از پایگاه‌های داده را از جمله MySQL ، Oracle ، IBM DB ، Microsoft SQL Server ، PostgreSQL و SQLite با دستورهای ساده فراهم می‌سازد.

PHP روی بیشتر سیستم‌عامل‌های معروف از جمله لینوکس، یونیکس، ویندوز و Mac OSX و با اغلب کارگزارهای وب (Web Server) معروف، قابل اجراست.

### PHP یک تاریخچه است

PHP در آغاز جایگاه شخصی داشت. در ابتدای سال ۱۹۹۴ عموماً مانند زبان برنامه نویسی C به صورت Binary توسط برنامه نویسان دانمارکی /گرینلندی به نام رسموس لردروف نوشته شد. لردروف در اصل این ابزار را ایجاد کرد تا یک مجموعه کوچکی از اسکریپت را جایگزین کند. او سعی می‌کرد که صفحات شخصی را حفظ کند. این ابزار برای این استفاده می‌شد که وظایفی مانند نمایش دادن خلاصه تجربیات و حفظ کردن اینکه چقدر از این صفحه مورد استفاده قرار می‌گیرد را انجام دهد. او این دو را با مفسرش ترکیب کرد تا PHP / FI را ایجاد کند چون بیشتر کاربرد داشت PHP / FI. شامل پیاده سازی برای زبان برنامه نویسی C بود و می‌توانست با پایگاه داده ارتباط یابد. قابلیت ساخت ساده و پویا برای کاربردهای وب. لردروف PHP را در ۱۹۹۵ رها کرد تا کد آنرا بهبود ببخشد و اشکالات آنرا برطرف کند. این واگذاری به نسخه دوم PHP منجر شد و هم اکنون استفاده می‌شود. PHP. متشکل از شبیه سازی و دستکاری کردن متغیرها توانایی جاسازی HTML بود. ترکیب آن مانند پرل اما محدود تر، ساده تر و پایدار تر بود. زیو سوراسکی و اندی گاتمنز دو توسعه دهنده در تکنولوژی IIT اسرائیلی بودند که پارسر را در سال ۱۹۹۷ دوباره نوشتند و اساس PHP 3 را تشکیل دادند با تغییر دادن نام زبان به PHP بازگشتی: پیش‌پردازنده مافوق متن. گروه توسعه یافته رسماً PHP / FI 2 را در نوامبر ۱۹۹۷ بعد از دو ماه تجربه و آزمایش منتشر کرد. پس از آن آزمایش عمومی PHP 3 آغاز شد و زبان رسمی در ژوئن ۱۹۹۸ تشکیل شد. سوراسکی و گاتمنز گروهی را برای باز نویسی PHP تشکیل دادند و ماشین Zend را در ۱۹۹۹ تولید کردند. آنها همچنین تکنولوژی آن را در اسرائیل تاسیس کردند. در ۲۲ ماه مه ۲۰۰۰ PHP 4 که توسط ماشین Zend به کار آمده بود منتشر شد. در ۱۲ ژوئیه ۲۰۰۴ PHP 5 که توسط ماشین Zend ۲ نیرو گرفته بود منتشر شد PHP 5 دارای ویژگی‌هایی

مانند پشتیبانی از زبان شی گرا (برای پایگاه داده) و وسعت بسیار زیاد کارایی آن بود. نسخه ای که اخیرا توسط گروه PHP منتشر شده نسخه ۴ آن است. در آگوست ۲۰۰۸ این نسخه به ۹,۴,۴ توسعه یافت. در سال ۲۰۰۸ PHP 5 در دست توسعه بود. PHP 6 در کنار PHP 5 در دست توسعه است. تغییرات عمده آن شامل از بین بردن متغیر های رجیستری است PHP. کاملا از یونیکد یا رشته های چند بایتی حمایت نکرد. یونیکد در PHP 6 پشتیبانی خواهد شد. این نسخه در هر دو محیط ۳۲ و ۶۴ بیتی اجرا خواهد شد .

سیر تکامل انتشار نسخه های مختلف PHP از ابتدا تا کنون			
نسخه	نسخه / ویرایش جدید	تاریخ نشر	یادداشت ها
1.0	1.0.0	1995-06-08	این اولین نام کاربردی PHP بود. Personal Home Page Tools (PHP Tools)
2.0	2.0.0	1996-04-16	توسط سریعترین و ساده ترین ابزار ایجاد شد. برای ایجاد کردن صفحات وب بویا..
3.0	3.0.0	1998-06-06	توسعه آن از یک نفر به چند نفر تغییر کرد. زیو سوراسکی و اندی گاتمنز دوباره اساس این نسخه را نوشتند. rewrite the base for this version.
4.0	4.0.0	2000-05-22	سیستم تجزیه / اجرا در دو مرحله تحت عنوان ماشین Zend اضافه شد.
	4.1.0	2001-12-10	(GET, \$_POST, \$_SESSION\$_) متغیر های بسیار خوبی معرفی شدند.
	4.2.0	2002-04-22	متغیر های بدون استفاده با پیش فرض. holes in applications.
	4.3.0	2002-12-27	علاوه بر CGI CLI نیز معرفی شد.
	4.4.0	2005-07-11	صفحات مهمی برای اسکریپت phpsize و config-php اضافه شد.
	4.4.8	2008-01-03	چند پیشرفت ایمنی. منتشر شدن PHP 4.
4.4.9	2008-08-07	پیشرفت بیشتر و انتشار نسخه PHP 4.4.	
5.0	5.0.0	2004-07-13	ماشین Zend با امکان مدل شی گرای.
	5.2.0	2006-11-02	گسترش فیلتر با پیش فرض.
	6.0.0	نشر آینده	پشتیبانی از یونیکد یا متغیر های رجیستری.

## کاربرد:

PHP یک زبان اسکریپت نویسی برای وب است.

PHP عمدتاً بر روی سرور وب اجرا می شود.

PHP کد برنامه را به عنوان ورودی می گیرد و صفحات وب را به عنوان خروجی ایجاد می کند. همچنین می تواند برای اسکریپت کردن خط فرمان استفاده می شود.

PHP می تواند بر روی صفحات وب و سیستم های عامل و یا پایگاهها مستقر شود. همچنین می تواند به عنوان سیستم مدیریتی پایگاه داده استفاده شود. PHP رایگان بوده و گروه PHP منبع کاملی از کد آن را در اختیار کاربر قرار می دهد. همچنین می توان از آن برای کاربرد های شخصی استفاده کرد.

PHP عمدتاً مانند یک فیلتر عمل می کند. به این صورت که ورودی را به صورت متن یا دستور العمل های PHP می گیرد و خروجی را تولید می کند. خروجی آن غالباً از نوع HTML می باشد. این زبان می تواند کد باینری را برای ماشین Zend تولید کند.

از PHP عمدتاً در طراحی وب استفاده می شود.

## میزان بهینه سازی :

همزمان با اسکریپت کردن زبانها، ساختار الگو گونه برای نمایش ترتیب حوادث PHP به طور طبیعی به صورت کد مرجع خوانا برای تولید سرورهای وب نگه داری می شود . بنابر این ساختار الگو گونه برای PHP در زمان اجرا توسط ماشین PHP کامپایل خواهد شد. کامپایل کردن در زمان اجرا دفعات اجرای فایل آغاز گر را افزایش می دهد. چون یک مرحله اضافه به زمان اجرا می افزاید. ساختار الگو گونه PHP می تواند قبل از زمان اجرا توسط مترجم هایی درست شبیه زبانهای برنامه نویسی همچون C ترجمه شود. ( زبان برنامه سازی PHP برنامه دار است تا جائیکه برای تمدید کردن آن استفاده شود). کد می تواند کیفیت کد کامپایل شده را با کم کردن اندازه آن و ایجاد کردن تغییراتی که بتواند زمان اجرا را کاهش دهد و کارایی را بهبود ببخشد بالا برد و بهینه نماید. نوع کامپایلر PHP آنچنان است که اغلب فرصت برای بهینه سازی کد آن وجود دارد.

به طور مثال می توان برای بهینه سازی کد بهینه ساز Zend را نام برد. شتابگر PHP می تواند سطح کارایی را با ذخیره کردن نوع کامپایل شده ای از ساختار PHP در حافظه تسهیم شده بالا ببرد تا از هزینه های تجزیه و کامپایل کد در زمان اجرا جلوگیری شود. ایمنی: میزان ناامنی نرم افزار نوشتاری PHP که توسط نرم افزار های رایج دیگر در معرض آسیب پذیری است به این صورت می باشد: ۱۲ درصد در سال ۲۰۰۲ تا ۲۰ درصد در سال ۲۰۰۴ تا ۲۸ درصد در سال ۲۰۰۵ تا ۴۳ درصد در سال ۲۰۰۶ تا ۳۶ درصد در سال ۲۰۰۷ و ۳۳ درصد در بخشی از سال ۲۰۰۸ بیشتر از یک سوم آسیب پذیری نرم افزار PHP به تازگی صورت گرفته . اغلب آسیب ها با پیروی نکردن از قوانین برنامه نویسی ایجاد می شود و این آسیب ها مرتبط با نسخه قدیمی PHP است. یکی از مهمترین این نا امنی ها متغیر های رجیستری هستند که از سال ۲۰۰۲ در PHP نسخه ۴٫۲ بدون استفاده شده اند.

## ترکیب:

PHP می تواند قسمتی از یک کد HTML باشد PHP . کد را بدون تعیین حدود کردن تجزیه می کند. هر چیزی خارج از این محدوده به طور مستقیم به خروجی فرستاده می شود و توسط PHP تعیین نمی شوند. حدود رایج و مشترک <?> و <?php> است که به ترتیب بسته کردن و باز کردن می باشد. علائم کوتاه می تواند برای شروع کد به کار روند: =?< یا <?> و علامتی که برای پایان کد به کار می رود، >? می باشد. این علائم به طور رایج استفاده می شود. اما علائمی مانند <?%> یا <?%> کمتر مورد استفاده قرار می گیرند. به همین خاطر استفاده از علائم کوتاه و ASP کم شده است. هدف از این ها جدا کردن PHP و HTML می

باشد. متغیرهایی که قبل از آن‌ها علامت \$ به کار می‌رود لازم نیست مشخص شوند. بر خلاف نامها و توابع کلاس نام متغیرها به کوچک و بزرگ بودن حروف حساس است PHP. در حالت زبان form free با خط جدید و فاصله سرو کار دارد (به جز در رشته‌ها) و جمله‌ها با سمی‌کالن تمام می‌شوند PHP. دارای سه نوع روش برای توضیحات می‌باشد. از /\* \*/ و // برای توضیحات استفاده می‌شود. همچنین از # برای توضیحات کوتاه استفاده می‌شود PHP. در واژگان مانند زبانهای سطح بالا همچون C است if، شرطی - حلقه‌های while و for و توابع بازگشتی شبیه به زبان C و ++C است.

## شیوه کار برنامه

همانطور که گفته شد کد PHP می‌تواند بخشی از یک کد HTML باشد.

مثلاً در برنامه زیر تنها

```
<?php echo "Hello PHP !" ?>
```

به زبان PHP است و بقیه برنامه به زبان HTML نوشته شده است. برای جداسازی برنامه PHP و برنامه اچ‌تی‌ام‌ال باید قسمت PHP را بین <?php?> و <?> قرار داد. لازم به ذکر است که بجای تگ آغاز که همانا <?php?> است، می‌توان از حالت خلاصه شده آن که به صورت <?> است استفاده نمود، ولی این کار استاندارد نبوده و بستگی به تنظیمات هنگام نصب php دارد. بنابر این توصیه می‌شود که از همان حالت کامل استفاده شود.

```
<code>
  <pre>
  <html>
    <head>
      <title>Sample</title>
    </head>
    <body>
      <?php echo "Hello PHP!"; ?>
    </body>
  </html>
</pre>
</code>
```

با صدا زدن صفحه‌ای با متن بالا از سرور به وسیله مرورگر در صفحه "Hello PHP" را می‌بینیم.

## نوع داده:

PHP همه متغیرها را در یک محدوده مستقل ذخیره می‌کند. این محدوده به طور نمونه ۳۲ بیت اعداد علامتدار است. اعداد بدون علامت در حالت‌های معینی به مقادیر علامتدار تبدیل می‌شوند. این رفتار در زبان‌های برنامه نویسی متفاوت است. متغیرهای صحیح می‌توانند به صورت دهدهی (مثبت یا منفی)، هشت هشتی و شانزده شانزدهی مشخص شوند.

اعداد واقعی نیز در محدوده معینی ذخیره می‌شوند. آنها می‌توانند به دو صورت اعشاری و علمی استفاده شوند. PHP دارای نوع محلی به نام بولین می‌باشد که شبیه به بولین در زبان جاوا و ++C است. در نوع داده بولین مقدار غیر صفر به عنوان درست و مقدار صفر به عنوان نادرست استفاده می‌شود. همانطور که در پرل و ++C نیز همین طور است.

نوع داده null برای متغیرهایی که هیچ مقداری ندارند استفاده می‌شود. متغیرهایی با نوع مرجع به منابع خارجی اشاره می‌کنند. اینها نوعاً توسط توابع ایجاد می‌شوند و تنها می‌توانند با همان تابع مورد استفاده قرار گیرند. مانند پوشه‌ها و عکسها و پایگاه داده PHP. آرایه‌ها و رشته‌ها را هم حمایت می‌کند.

کتابخانه استاندارد (SPL) PHP به منظور حل مسائل استاندارد و دسترسی به داده‌های موثر و کلاسها به کار می‌رود.

نسخه ۵.۲ و قدیمی تر:

توابع بهترین توابع نیستند و تنها می‌توانند با نامشان صدا زده شوند. توابع تعریف شده توسط کاربر می‌توانند بیرون از نمونه اولیه ایجاد شوند. توابع می‌توانند در درون بلوک‌ها تعریف شوند. توابع فراخوانی شده باید از پرانتز استفاده کنند. به جز آرگومان zero توابع سازنده کلاس با اپراتور new فراخوانده می‌شوند. در اینجا پرانتزها اختیاری هستند. PHP توابع بی نام را که توسط دستور `function_create` ایجاد می‌شوند را هم پشتیبانی می‌کند. اگر چه درست نیستند چون توابع بی نام از اسم بر خوردار نیستند و توابع تنها می‌توانند توسط نامشان صدا زده شوند.

نسخه ۵.۲ و جدیدتر:

PHP از توابع درجه اول و توابع بی نام حمایت می‌کند که توسط ساختار زیر پشتیبانی می‌شوند:

```
function getAdder($x)
{
    Return function ($y) use ($x) {
        return $x + $y;
    };
}
```

در نمونه فوق تابع `getAdder()` یک پارامتر `$x` ایجاد می‌کند که آرگومان `$y` اضافی را می‌گیرد و آنرا به صدا زننده بر می‌گرداند.

## اشیاء:

برنامه نویسی شی گرا به PHP اضافه شد. اشیاء در PHP به طور کامل دوباره نوشته می شود. در نسخه قبلی PHP اشیاء مانند تایپ های اولیه به کار برده می شوند. اشکال این روش این بود که وقتی یک متغیر تعریف می شود تمام اشیاء کپی می شوند. در روش جدید اشیاء با handle بازگشت داده می شوند نه با مقدار. PHP متغیر های عضو خصوصی و حفاظت شده تعریف می کند. همچنین یک روش استاندارد برای معرفی سازنده و مخرب همانند زبان برنامه سازی ++C تعریف می کند Interface . های خاصی وجود دارد که به اشیاء اجازه می دهد بر روی زمان اجرا اثر بگذارد. اگر توسعه دهنده یک کپی از یک شیء با استفاده از کلمه محفوظ clone ایجاد کند ماشین Zend آنرا بررسی خواهد کرد که آیا روشی برای آن تعریف شده یا نه. اگر نشده بود یک Clone پیش فرض را صدا خواهد زد که ویژگی های آن شیء را کپی می کند. اگر تعریف شده بود برای ویژگی های لازم اشیاء ایجاد شده معتبر است. برای سادگی ماشین تابعی را تولید می کند که مشخصات شیء مرجع را وارد می کند که برنامه نویس می تواند با یک مقدار مشابه شیء مرجع آغاز به کار کند و تنها ویژگی هایی را که نیاز دارد تغییر دهد.

## روش نصب PHP

شما برای کار با PHP نیاز به یک سرور مجازی دارید که می توانید از IIS یا Apache استفاده کنید که ما آپاچی را توصیه می کنیم. همچنین به یک پایگاه داده نیاز دارید که برای این کار mysql بهترین گزینه است.

شما می توانید با نصب Wampserver به راحتی همه این گزینه ها را یکجا داشته باشید Wampserver. یک نرم افزار رایگان است که با نصب Apache ، PHP ، MySQL و PHP My Admin شما را از نصب تک تک این برنامه ها آسوده می کند.