

# برنامه نویسی تحت وب (با PHP)

مهندس حمید رضا نیرومند

<http://niroomand.ir>

## انواع برنامه نویسی تحت وب از لحاظ تبادل اطلاعات با کاربر:

### ۱- Static (ایستا)

در نوع ایستا اطلاعات چندانی از کاربر اخذ نمی‌شود یعنی کاربر فقط خواننده محتوای سایت است. این نوع سایت‌ها از دیتابیس استفاده نمی‌کنند بلکه فقط یک سری صفحات ثابت HTML هستند، این نوع سایت‌ها به همه کاربران یک نوع محتوا را نمایش می‌دهند.

مثال) سایت معرفی یک شرکت صنعتی، سایت معرفی یک شخصیت.

### ۲- Dynamic (پویا)

در نوع پویا، در هر لحظه از کاربر اطلاعاتی دریافت می‌شود و نسبت به اطلاعاتی که از کاربر دریافت می‌شود محتوای متفاوتی نمایش می‌دهد قطعا در این نوع سایت‌ها از دیتابیس استفاده می‌شود.

مثال) سایت‌های جیمیل، سایت‌های آموزشی، انجمن و...

### زبان‌های پویا:

۱- **Java**: یکی از بهترین زبان‌هاست. به خاطر ماشین JVM، یکبار کدنویسی می‌کنید و چند بار (بر روی بسترهای مختلف مثل ویندوز، لینوکس، موبایل، اینترنت و...) استفاده می‌کنید ولی درک آن سخت است و منابع آن کمتر است. بنابراین افراد کمتری به سراغ کدنویسی با Java می‌روند.

۲- **PHP**

۳- **ASP**

۴- و ...

## تفاوت‌های بین دو زبان PHP و ASP به طور خلاصه:

### :PHP

- PHP یک زبان رایگان و منبع باز (open source) است.
- در زبان PHP به خاطر رایگان بودن، کدهای آماده بسیار فراوان است.
- در PHP، IDE خوب و معتبر نداریم چون یک زبان Run-time است. (زبان Run-time است یعنی خطاها در زمان اجرا کشف می‌شوند)
- PHP بر روی windows, linux و MacOS قابل اجراست.
- Web Server مربوط به این زبان، Apache است.
- Data Base مربوط به این زبان، MySQL است.
- برای اجرای برنامه‌ها به زبان PHP مجموعاً به سه نرم افزار نیاز دارید: Apache-MySQL-PHP که باید از سایت‌های رسمی آن‌ها دانلود و روی سیستم یا سرور خود نصب کنید.

- به جای نصب کردن چند نرم افزار می توانید یک پکیج دانلود و نصب کنید فقط باید بستر خود را انتخاب کنید یعنی برای ویندوز می خواهید یا لینوکس یا مک؟
- به پکیجی که روی ویندوز نصب می شود در اصطلاح<sup>۱</sup> WAMP گفته می شود. به پکیج لینوکس، LAMP و به پکیج مک، MAMP گفته می شود.

## :ASP

- ASP یک زبان انحصاری است. (در انحصار مایکروسافت)
- ASP کدهای آماده کمتری دارد.
- ASP دارای محیط توسعه بسیار قوی به نام Visual Studio است.
- زبان ASP تنها بر روی Windows قابل اجراست.
- Web Server مربوط به این زبان IIS<sup>۲</sup> است.
- Data Base مربوط به این زبان SQL Server است.

\* برای استفاده از دیتابیس در سایت های پویا نیاز به یک DBMS است.

## :DBMS

- ۱- برای ASP : Access , SQL Server
- ۲- برای PHP : MySQL (محل دانلود: mysql.net)
- \* برای ایجاد و اجرای سایت های پویا نیاز به Webserver است.

## :Webserver

- ۱- برای ASP : IIS=Internet information Services (سرویس های اطلاعاتی اینترنتی)
- ۲- برای PHP : Apache (محل دانلود: Apache.org)

## وب سرور چیست؟

نرم افزاری است که امکان استفاده از پورت ۸۰ کامپیوتر را فراهم می کند پورت ۸۰ همانطور که در درس شبکه خواندید مربوط به پروتکل HTTP است در حقیقت این نرم افزار اجازه می دهد کاربران دیگر شبکه با وارد کردن IP یا یک Domain به سیستمی که وب سرور روی آن نصب است به آن متصل شده و سایتی که روی وب سرور هست را مشاهده کنند.

**نکته:** دو وب سرور IIS و Apache نمی توانند همزمان با پورت ۸۰ کار کنند.

<sup>۱</sup> WAMP = Windows Apache MySQL PHP

<sup>۲</sup> Internet Information Services

## Port چیست؟

برنامه‌ها و سرویس‌های مختلفی مثل مرورگر، ایمیل، برنامه چت و نرم افزارهای مختلف دیگر، از طریق یک اتصال شبکه کار می‌کنند. سؤال این است که وقتی یک Packet وارد کامپیوتر شما می‌شود کامپیوتر چگونه می‌فهمد که این Packet برای برنامه ایمیل است یا مرورگر یا دیگر برنامه‌ها؟

پاسخ: هر برنامه یا سرویس در دنیا یک شماره مختص و ثبت شده دارد که به آن شماره port گفته می‌شود. این شماره توسط مؤسسه IANA (مخفف Internet Assigned Number Authority به معنی مسؤول عدد نسبت داده شده اینترنتی) تعیین می‌شود. لیستی از پورت‌ها در آدرس مقابل قابل مشاهده است:

<http://www.iana.org/assignments/port-numbers>

به طور مثال شماره پورت برنامه Telnet عدد ۲۳ است، شماره پورت FTP عدد ۲۱، شماره پورت HTTP عدد ۸۰ و شماره پورت SMTP عدد ۲۵ است.

برخی از مهم‌ترین سرویس یا برنامه‌ها و شماره پورت آن‌ها:

Table 4-3 Commonly used TCP/IP port numbers

Port Number	Process Name	Protocol Used	Description
7	ECHO	TCP and UDP	Echo
20	FTP-DATA	TCP	File Transfer - Data
21	FTP	TCP	File Transfer-Control
22	SSH	TCP	Secure Shell
23	TELNET	TCP	Telnet
25	SMTP	TCP	Simple Mail Transfer Protocol
53	DNS	TCP and UDP	Domain Name System
69	TFTP	UDP	Trivial File Transfer Protocol
80	HTTP	TCP and UDP	World Wide Web HTTP
110	POP3	TCP	Post Office Protocol 3
119	NNTP	TCP	Network News Transport Protocol
143	IMAP	TCP	Internet Message Access Protocol
443	HTTPS	TCP	Secure implementation of HTTP

\* برای اجرای کدهای PHP نیاز به نرم‌افزار PHP است. (محل دانلود: PHP.net)

مشهورترین WAMP ها که برای نصب و استفاده در ویندوز آماده شده است:

۱- XAMPP = روی هم ویندوز و هم مک و هم لینوکس نصب می‌شود. سنگین است و کاربر پسند نیست.

۲- Bitnami

۳- EasyPHP

۴- UWAMP

۵- WAMP Server

۶- MoWes

## شروع کار با PHP:

پکیج WAMP خود را Extract و نصب کنید. به پوشه `www` رفته تمام فایل‌ها به جز پوشه `PHPmyadmin` را حذف کنید. سپس تمام پوشه‌های سایت خود را در این پوشه کپی کنید. از این پس باید برای دیدن سایت خود در مرورگر وارد کنید: `http://localhost`.

ساده ترین برنامه به زبان PHP:

```
<?php  
echo 'hello PHP!';  
?>
```

یک فایل متنی باز کنید و کدهای بالا را در آن تایپ کرده و با نام `index.php` در پوشه `www` ذخیره کنید. حالا به آدرس <http://localhost> بروید.

**متغیرها در زبان PHP:** در PHP متغیرها نیاز به تعریف یا مقداردهی اولیه ندارند. ضمن اینکه می‌توانید هر نوع کاراکتر یا عددی را به متغیر که با علامت دلار در ابتدایش مشخص می‌شود (مانند `$X`) نسبت دهید.

\* PHP از روی محتوای متغیر نوع متغیر را تشخیص می‌دهد.

ساختار کلی تعریف متغیر:

; مقدار اولیه = نام متغیر \$

مثال:

```
$X=5;  
$X= "Hamid";
```

برای اینکه زبان فارسی به درستی در خروجی نمایش داده شود باید دستور زیر را در قسمت `head` بنویسیم:

```
<meta charset="utf-8" >
```

## ساختار کلی تصمیم‌گیری یا کنترلی:

ساختار کلی ساختارهای کنترلی در PHP دقیقاً مانند زبان C است:

### ساختار if:

<pre>if ( شرط ) {     دستورات در صورت برقراری شرط }</pre>	<pre>if ( شرط ) {     دستورات در صورت برقراری شرط } else {     دستورات در صورت عدم برقراری شرط }</pre>	<pre>if ( شرط ) {     دستورات در صورت برقراری شرط اول } else if ( شرط دوم ) {     دستورات در صورت برقراری شرط دوم } else {     دستورات در صورت عدم برقراری شرطها }</pre>
---	--	--

### ساختار switch:

```
switch (متغیر شرط)
{
    case 'مقدار اول':
        دستورات در صورت برقراری شرط اول
        break;
    case 'مقدار دوم':
        دستورات در صورت برقراری شرط دوم
        break;
    default:
        دستورات در صورت عدم برقراری شرطها
        break;
}
```

**مثال** ) فرض کنید در فایل link.php هم افزودن لینک را نمایش دهیم و هم لیست لینک‌ها و هم حذف لینک‌ها با استفاده از پارامتر op که به این صفحه ارسال می‌شود. تصمیم بگیرید منظور کاربر از این load چه بوده مثلاً اگر op برابر با add بود چاپ کنید فرم افزودن لینک و...

```

$op=$ GET['op'];
switch ($op)
{
    case 'add':
        echo '<h1> افزودن لینک </h1>';
        break;
    case 'del':
        echo '<h1> حذف کن </h1>';
        break;
    default:
        echo '<h1> رخ داده است </h1>';
        break;
}

```

## دستور echo:

این دستور برای چاپ یک مقدار در خروجی استفاده می‌شود و به شکل‌های زیر قابل استفاده است:

echo (" ")      echo (' ')      echo " "      echo ' '

\*فرق بین " و ' در این است که در بین ' نمی‌توان از نام متغیر استفاده کرد و در صورت استفاده خود نام متغیر چاپ می‌شود و نه مقدار آن. در صورت نیاز به چاپ مقدار متغیر یا باید نام متغیر داخل " یا خارج از ' استفاده شود.

مثال) فرض کنید در بخش فایلی به نام links.php که لیست لینک‌ها را نشان می‌دهد از طریق نوار آدرس شماره صفحه را بگیرید و در یک متغیر بریزید و در صفحه بنویسید شما در صفحه n هستید و به جای n، شماره صفحه را چاپ کند.

می‌توان یکی از دو دستور echo زیر را نوشت تا مقدار متغیر را چاپ کند.

```

$page=$_GET['page'];
echo "shoma dar safhe $page hastid";
echo 'shoma dar safhe '.$page.' hastid';
shoma dar safhe 2 hastid

```

خروجی ←

## انواع حلقه‌ها در PHP:

### ساختار کلی While:

```
while (شرط)
{
    //Body Statements
}
```

### ساختار کلی do ... while:

```
do
{
    //Body Statements
}
while (شرط)
```

### ساختار کلی for:

```
( عملیات در هر دور ; شرط توقف ; عملیات اولیه )
for
{
    //Body Statements
}
```

## آرایه‌ها در PHP :

0	1	2	3	4
---	---	---	---	---

### ۱- شکل ساده:

```
$x=array();
$x[0] = "Hamidreza";
$x[1] = "Niroomand";
```

تفاوت تعریف آرایه با زبان C این است که در اینجا نیاز نیست ابتدا تعداد خانه‌های آرایه را تعیین کنیم. هر چقدر که بخواهیم می‌توانیم عنصر به آرایه اضافه کنیم.

### ۲- آرایه با اندیس دلخواه:

	First Name	Last Name	Father Name
Student			

```
نام اندیس " , مقدار اندیس اول" => "نام اندیس اول" = array نام آرایه $
( ... , مقدار اندیس دوم => "دوم" );

مثال:
$student=array ( "firstname" =>"Hamid Reza" , "Lastname"=>"Niroomand" , ... ) ;
```



## پیمایش آرایه‌ها:

\* حلقه foreach مختص پیمایش آرایه‌هاست.

```
(متغیر مقدار اندیس => متغیر نام اندیس as نام آرایه)
foreach
{
    عملیات روی آرایه
}
```

مثال) یک منوی select حاوی ماه‌های سال را به کمک حلقه foreach تولید کنید.

```
<select name="months">
$months=array("فروردین" , ... , "اسفند");
<?php
foreach ($months as $key => $month)
{
    $key2=($key+1);
    echo "<option value=\"\$key+1\" $month</option>";
}
?>
</select>
```

## توابع کار با آرایه:

مرتب سازی صعودی	sort ( نام آرایه );
مرتب سازی نزولی	rsort ( نام آرایه );
علاوه بر مرتب سازی صعودی اندیس‌ها را هم جابه جا می‌کند	asort ( نام آرایه );
علاوه بر مرتب سازی نزولی اندیس‌ها را هم جابه جا می‌کند	arsort ( نام آرایه );
فقط اندیس‌ها را بصورت صعودی مرتب می‌کند	ksort ( نام آرایه );
فقط اندیس‌ها را بصورت نزولی مرتب می‌کند	krsort ( نام آرایه );
عناصر آرایه را معکوس می‌کند	array_reverse ( نام آرایه );
به صورت رندم نمایش می‌دهد	shuffle ( نام آرایه );
تعداد عناصر آرایه را برمی‌گرداند	count ( نام آرایه );
برای وجود یا عدم وجود یک مقدار در بین عناصر آرایه	if (in_array ( "مقدار موردنظر" , نام آرایه ) )

## کار با فرم‌ها:

### تگ input:

برای قرار دادن انواع کنترل‌های دریافت اطلاعات از کاربران استفاده می‌شود و به وسیله خاصیت type می‌توانید نوع کنترل را تعیین کنید.

مثال) برای لاگین کردن به یک سایت:

```
<br> نام کاربری: <br>
<input type="text" name="username"><br>
کلمه عبور: <br>
<input type="password" name="password"><br>
<input type="submit" value="ورود به سیستم" name="login">
```

### تعدادی از انواع input:

<code>&lt;input type="text" name="username" &gt;</code>	برای متن ساده	text
<code>&lt;input type="radio" name="gender"&gt;</code>	برای چندگزینه‌ای	radio
<code>&lt;input type="password" name="password" &gt;</code>	برای رمز عبور	password
<code>&lt;input type="email" name="usremail" &gt;</code>	برای ایمیل	email
<code>&lt;input type="submit" name="send" &gt;</code>	برای ایجاد دکمه	submit

### فیلد عکس:

برای آپلود و دریافت عکس به کار می‌رود:

```
<input type="file" name="photo">
```

### فیلد checkbox:

این فیلد می‌تواند برای علاقه مندی‌ها به کار برود:

```
<input type="checkbox" name="fav[0]"> هنر
<input type="checkbox" name="fav[1]"> ورزش
<input type="checkbox" name="fav[2]"> موسیقی
```

### فیلد textarea:

در ای فیلد شما می‌توانید به میزان نامحدود کاراکتر وارد کنید این فیلد برای دریافت پیغام متنی و توضیحات به کار می‌رود

```
<textarea name="content" rows="10" cols="30"></textarea>
```

## فیلد select:

ابرای اینکه بتوانیم لیستی از موارد انتخابی داشته باشیم که هر مورد از لیست در تگ option نوشته می‌شود. به عنوان مثال می‌توان با این تگ تاریخ تولد را در فرم ثبت نام قرار داد(روز و ماه به صورت لیست باکس نشان داده می‌شود)

```
<select name="day">
<?php
$i=1;
while ($i<=31)
{
    echo "<option value=\"$i\">$i</option>";
    $i++;
}
?>
</select>
<select name="month">
<?php
$i=1;
while ($i<=12)
{
    echo "<option value=\"$i\">$i</option>";
    $i++;
}
?>
</select>
```

مثال دیگر میتوان اگر موارد انتخابی عددی نیست و تعداد کم هستند مثل مدرک تحصیلی:

```
<select name="grade">
    <option value="1">دیپلم</option>
    <option value="2">کار دانی</option>
    <option value="3" selected>لیسانس</option>
</select>
```

## تفاوت متود GET و POST:

متود GET اطلاعات را در نوار آدرس نمایش می‌دهد اطلاعات را از طریق نوار آدرس به صفحه پردازشگر می‌فرستد اما متود POST تغییری در آدرس اولیه سایت ایجاد نمی‌شود. GET برای فرم‌های جستجو مناسب است و POST برای سایت‌های login.

## احراز هویت کاربر:

برای احراز هویت کاربر دو امکان برای برنامه نویس در نظر گرفته شده:

Session -۱

Cookie -۲

هر دوی اینها یک فایل متنی ساده هستند که معمولاً نام کاربری و رمز عبور کاربر برای تشخیص او در مراجعات بعدی در این فایل‌ها ذخیره می‌شود.

## فرق Cookie و Session:

- ۱- Cookie روی سیستم کاربر ایجاد می‌شود اما Session روی سیستم سرور ایجاد می‌شود.
- ۲- Cookie امنیت پایین‌تری نسبت به Session دارد. بنابراین در کاربردهایی که امنیت داده‌ها خیلی مهم است (مانند سایت بانک‌ها) قطعاً از Session استفاده می‌شود اما در کاربردهایی که امنیت داده‌ها چندان مهم نیست، احتمالاً می‌توان از Cookie استفاده کرد.
- ۳- Cookie طول عمر دارد و اگر عمر آن به پایان نرسیده باشد با بستن مرورگر همچنان روی سیستم کاربر وجود دارد یعنی در مراجعات بعدی کاربر شناسایی شده و نیاز به وارد کردن مجدد نام کاربری و رمز عبور نیست. اما Session به محض بسته شدن مرورگر و حتی عدم فعالیت کاربر تا نهایتاً ۲۰ دقیقه خود به خود از روی سرور پاک می‌شود.

\* می‌توان با شمردن تعداد Session‌های روی سرور به تعداد کاربران آنلاین پی برد.

### :Cookie

#### ایجاد Cookie:

```
setcookie("محتوای کوکی","نام دلخواه کوکی");  
setcookie("linka_admin_username","admin",time()+860000);
```

#### حذف کوکی:

```
setcookie("محتوای کوکی"," ",time()-860000);  
setcookie("linka_admin_username"," ",time()-860000);
```

#### خواندن محتوا کوکی:

```
$_COOKIE ['نام کوکی موردنظر'];  
$username=$_COOKIE ['linka_admin_username'];
```

#### بررسی وجود کوکی:

```
if (isset($_COOKIE ['نام کوکی']))  
{  
    دستورات در صورت وجود کوکی  
}
```

### :Session

#### ایجاد Session:

قبل از هر نوع عملیات روی سشن باید ابتدای صفحه یعنی قبل از اینکه حتی یک کاراکتر به خروجی برود دستور زیر نوشته شود:

```
Session_start ();  
$_SESSION['نام دلخواه سشن']="محتوای سشن";  
$_SESSION['linka_admin_username']="admin";
```

#### خواندن محتوا سشن:

```
$_SESSION['نام موردنظر'];  
$username=$_SESSION['linka_admin_username'];
```

#### حذف سشن:

```
session_destroy();
```

بررسی وجود سشن:

```
if (isset($_SESSION['نام سشن']))  
{  
    دستورات در صورت وجود سشن  
}
```

## سناریوی لاگین کاربر به پنل مدیریت در یک CMS:

CMS=Content Management System

نرم افزارهای آماده تحت وبی هستند که زحمت کدنویسی را از روی دوش مدیران سایتها برداشته و آنها را مشغول محتوای سایتها می کند.

### CMSهای مشهور:

Wordpress Joomla Mambo	سایت سازمانی
Wordpress Drupal	سایت آموزشی
4images	گالری عکس
Prestashop OS Commerice	فروشگاه
PHPBB MyBB MySMF VBulletinBoard	انجمن گفتگو
PHP-Nuke	همه چیز با هم

### مزایای استفاده از CMS :

- کاهش هزینه ها: بسیاری از CMSها رایگان هستند و برخی نیز که غیر رایگان هستند نسبت به طراحی از صفر بسیار ارزان تر تمام می شوند.
- سیستم های آماده: آزمون خود را روی سایت های دیگران پس داده اند بنابراین باگ ها و خطاهای کمتری دارند.
- بسیاری از افراد در سراسر دنیا ماژول ها و امکانات مختلفی برای این سیستم نوشته اند و معمولاً به رایگان منتشر کرده اند که می توانید در سایت خود استفاده کنید.

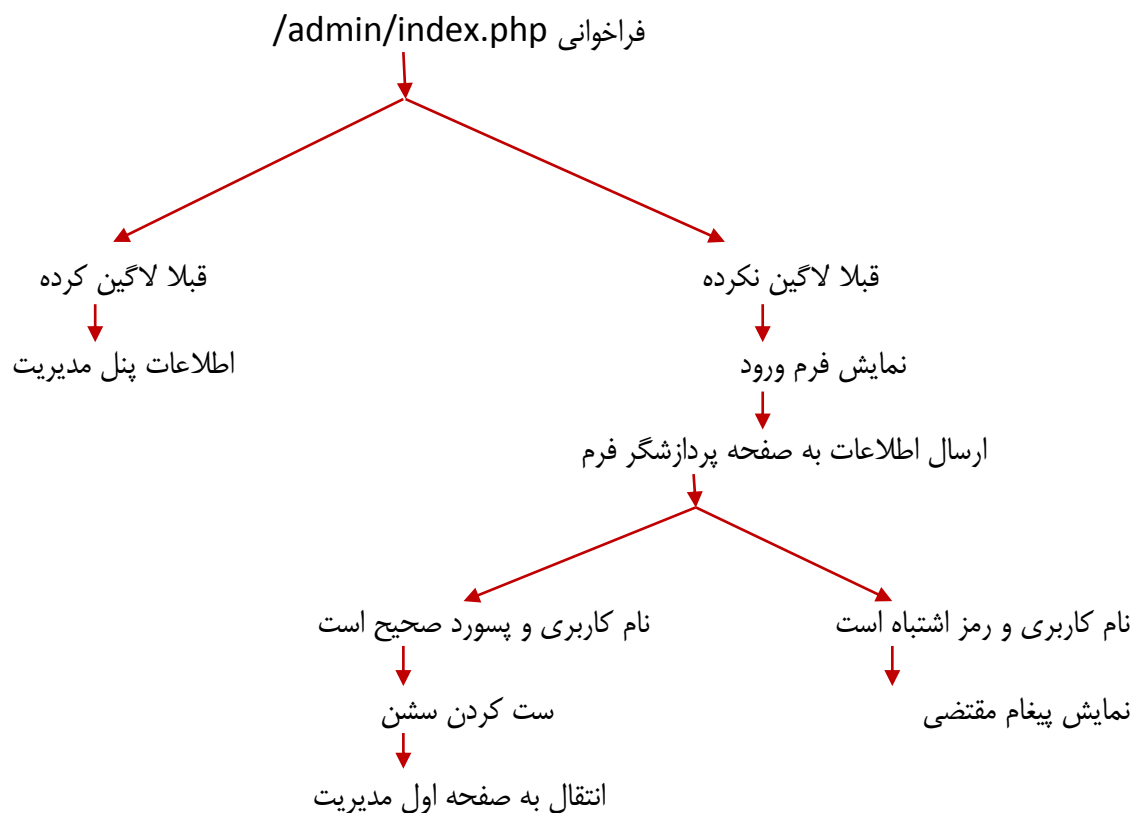
### معایب استفاده از CMS:

- ریسک بالا: از آنجا که معمولاً کد این سیستم ها در اختیار همه است، اگر باگ (حفره امنیتی) توسط یک هکر در این سیستم ها کشف شود تمام سایت هایی که از این CMSها استفاده می کنند در معرض خطر خواهند بود.
- مشکل در تغییرات اساسی: از آنجا که برنامه نویس یا خودتان این سیستم را طراحی نکرده اید تغییر دادن هر بخش از آن برای اینکه با خواسته شما مناسب شود بسیار سخت است.
- برای هر کاری نمی توان یک CMS پیدا کرد. ممکن است ایده ای که در ذهن شماست جدید باشد و هنوز CMS مورد نظر شما طراحی نشده باشد.

## پوشه‌ها و فایل‌های مشترک در اکثر CMSها:

admin/	پنل مدیریت سایت
includes/	کتابخانه‌ها و کلاس‌های طراحی شده یا استفاده شده در سایت
language/	فایل‌های ترجمه سایت به زبان‌های مختلف
theme/	قالب‌های مختلف سایت
config.php	اطلاعات پایه‌ای سایت ، اطلاعات اتصال به دیتابیس
index.php	صفحه اول سایت
main.php	بررسی هویت کاربر و امنیت، اطلاعات سشن و کوکی
header.php	محتویات ثابت بالای صفحات سایت مثل منوها و لوگوها
footer.php	محتویات ثابت پایین صفحات سایت مثل کپی، لیست و...
theme/default/ - style/ - js/ - img/	قالب پیش فرض صفحات شامل: ۱- فایل‌های CSS ۲- فایل JS ۳- عکس‌های قالب
login.php , logout.php	صفحه ورود به سایت و خروج از سایت

## سناریوی لاگین کاربر به پنل مدیریت:



توابع import کردن یک فایل به فایل دیگر:

تابع include:

```
include 'آدرس محلی فایل که باید اینپورت شود';  
include 'main.php';
```

تابع include\_once:

همان کار تابع include را انجام می‌دهد، با این تفاوت که دفعه دوم به بعد را نادیده می‌گیرد. Importهای تکراری یک فایل را دفعه دوم به بعد را نادیده می‌گیرد.

```
include_once 'main.php';
```

تابع require:

همان کار تابع include را انجام می‌دهد با این تفاوت که اگر فایل نامبرده شده به هر دلیلی پیدا نشد، PHP از اجرای ادامه کدها خودداری می‌کند چون به معنی نیاز داشتن است. این تابع در مباحث امنیتی گاهی بسیار مهم است؛ چرا که گاهی باید حتماً یک فایل خاص موجود باشد تا ادامه کدها اجرا شود.

```
require 'آدرس محلی فایل که باید اینپورت شود';
```

تابع require\_once:

همان کار تابع require را انجام می‌دهد با این تفاوت که importهای تکراری یک فایل را در دفعه دوم به بعد را نادیده می‌گیرد.

**مثال** ) اگر بخواهیم یک سایت که شامل صفحه لاگین (ورود به سیستم) و صفحه اول سایت که بعد از لاگین نمایش داده می‌شود را بسازیم باید این صفحات ایجاد و این کدها را بنویسیم:



## header.php:

```
<?php
<!DOCTYPE html>
<html>
<head>
    <title>linka</title>
    <meta charset="utf-8"> // برای پشتیبانی از زبان فارسی
    <link rel="stylesheet" type="text/css"
href=" ../themes/default/css/bootstrap.css"> // فایل را فراخوانی میکند
    <link rel="stylesheet" type="text/css"
href=" ../themes/default/css/style.css">
</head>
<body dir="rtl"> // برای راستچین شدن محتویات سایت
</body>
</html>
?>
```

صفحه اول که ادمین با وارد کردن نام کاربری و رمز صحیح می تواند به پنل خود وارد شود: login.php

```
<?php
// به ارث بردن فونت و استایل صفحه از هدر
include 'header.php';
?>
<form action="process.php" method="post">
    نام کاربری: <br>
    <input type="text" name="username"><br>
    کلمه عبور: <br>
    <input type="password" name="password"><br>
    <input type="submit" value="ورود به سیستم" name="login">
</form>
```

این صفحه عملیات پردازش صحیح بودن سشن را انجام میدهد: process.php

```
<?php
if (isset ($_POST ['login']))
{
    if ($_POST ['username']=="admin" && $_POST ['password']=="123") {
        session_start();
        $_SESSION['linka_admin_username']="admin"; //ایجاد سشن یوزر
        $_SESSION['linka_admin_password']="123"; //ایجاد سشن پسورد
        header("location:index.php"); //اگر یوزر با رمز بود به صفحه پنل برو
    }
    else{
        header ('refresh:3; url=login.php'); //اگر اشتباه بود بعد از ۳ ثانیه به صفحه لاگین برگرد
        echo "invalid your password!";
    }
}
?>
```

صفحه اول پنل مدیریت ادمین: index.php

```
<?php
session_start();
if (!isset($_SESSION ['linka_admin_username']) || !isset($_SESSION
['linka_admin_password'])) // اگر یوزر یا رمز اشتباه بود
{
    header("location:login.php");// برگرد به صفحه لاگین
}
else
{
    echo 'مدیر محترم خوش آمدی' . [<a href="logout.php">خروج</a>];
}
?>
```

## خروج از پنل مدیریت logout.php

```
<?php
session_start();
session_destroy();// خروج از پنل
header("location:login.php");// رفتن به صفحه لاگین
?>
```

## مخازن داده‌ای در زبان برنامه‌نویسی:

۱- متغیر

۲- ثوابت:

```
define ("مقدار ثابت","نام ثابت");
define ("_DBNAME","g1");
```

## این فایل باید در خارج از پوشه اصلی کدها نوشته شود config.php

```
<?php
define("_DBNAME", "g1");
define("_DBUSER", "dbuser");
define("_DBPASSWORD", "123456");
define("_DBHOST", "localhost");
?>
```

## مراحل کلی طراحی پایگاه داده:

### ۱- شناسایی موجودیت‌ها:

۱- چند نمونه در محیط دارد.

۲- می‌خواهیم اطلاعات درباره آن کسب کنیم.

۳- معمولاً بیش از یک صفت دارد.

مثل: لینک که خودش یک موجودیت است یا کاربران یا نظرها و ...

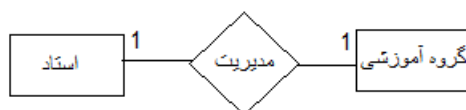
### ۲- شناسایی صفات موجودیت‌ها:

همان ویژگی‌های هر موجودیت است. مثل: آدرس لینک، تولتیپ لینک، متن لینک و...

### ۳- شناسایی رابطه بین موجودیت‌ها:

که به سه دسته تقسیم می‌شود: الف) 1-1 ب) 1-n ج) n-m

- رابطه ۱ به ۱: در این رابطه یک نمونه از موجودیت A با یک نمونه از موجودیت B در ارتباط است و یک نمونه از موجودیت B نیز با یک نمونه از موجودیت A در ارتباط است.



- رابطه ۱ به N: در این رابطه یک نمونه از موجودیت A با چند نمونه از موجودیت B در ارتباط است و یک نمونه از موجودیت B نیز با یک نمونه از موجودیت A در ارتباط است.



- رابطه N به M: در این رابطه یک نمونه از موجودیت A با چند نمونه از موجودیت B در ارتباط است و یک نمونه از موجودیت B نیز با چند نمونه از موجودیت A در ارتباط است.

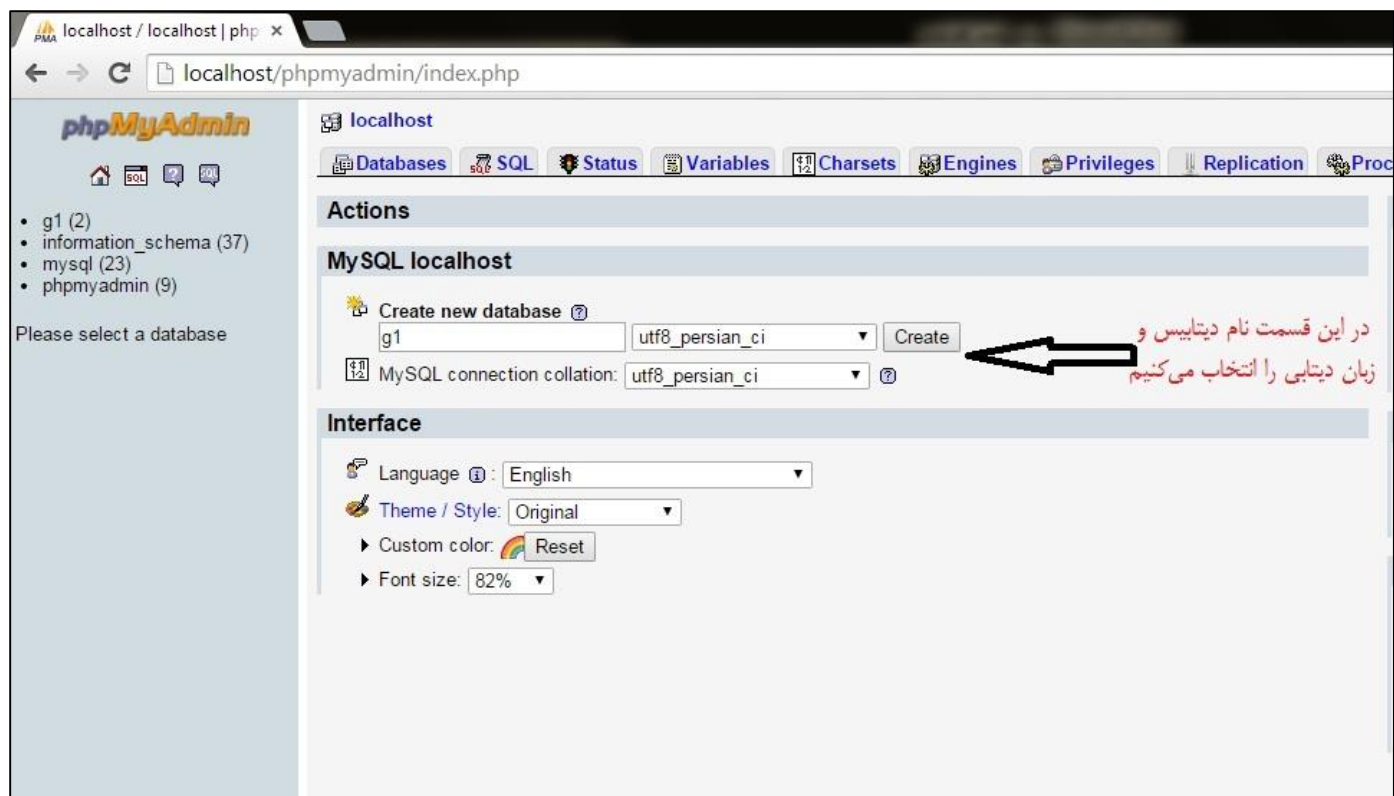


نکات مهم:

- در رابطه یک به یک کلید اصلی غیر الزامی در جدول سمت الزامی نگه داشته می‌شود.
- در رابطه یک به چند کلید اصلی سمت 1 در داخل جدول سمت n قرار می‌گیرد.
- در رابطه چند به چند برای رابطه یک جدول جداگانه در نظر می‌گیریم.

## ایجاد دیتابیس برای سایت:

پس از اینکه کدهای خود را در برنامه sublime نوشتید، برای اینکه اطلاعات و یوزرها و لیست کاربران در جایی ذخیره شود باید یک دیتابیس برای سایت خود ایجاد کنیم. پس از اجرای برنامه mowes باید در مرورگر خود در قسمت نوار آدرس کلمه localhost/phpmyadmin را تایپ کنید و سپس شما می‌توانید دیتابیس خود را ایجاد کنید.



پس از ایجاد دیتابیس باید جدول‌های لازم را درون آن ایجاد کنیم مانند جدول کاربران که نام و ایمیل و مشخصات لازم آنها در آن ذخیره شود.

phpMyAdmin

Database: group (1)

group (1)

users

Table 'group`.`users` has been created.

```
CREATE TABLE `group`.`users` (
  `uid` INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `ufname` VARCHAR(50) CHARACTER SET utf8 COLLATE utf8_persian_ci NOT NULL,
  `ulname` VARCHAR(50) CHARACTER SET utf8 COLLATE utf8_persian_ci NOT NULL,
  `uemail` VARCHAR(100) NULL,
  ENGINE = InnoDB
```

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> uid	int(11)			No	None	AUTO_INCREMENT	
<input type="checkbox"/> ufname	varchar(50)	utf8_persian_ci		No	None		
<input type="checkbox"/> ulname	varchar(50)	utf8_persian_ci		No	None		
<input type="checkbox"/> uemail	varchar(100)	latin1_swedish_ci		Yes	NULL		

Check All / Uncheck All With selected:

Print view Relation view Propose table structure Track table

Add 1 field(s) At End of Table At Beginning of Table After uid Go

Indexes:

Action	Keyname	Type	Unique	Packed	Field	Cardinality	Collation	Null	Comment
	PRIMARY	BTREE	Yes	No	uid	0	A		

جدول کاربران ایجاد  
شد و در سمت راست  
فیلدهای آن و نوع آنها  
را میبینید.

## انواع Data type در mysql :

نوع	نهایت طول	انواع رشته ای:
varchar	255 کاراکتر	
text	بینهایت	
int	0 تا 4000000	انواع عددی:
big int	بینهایت	
tiny int	0 تا 127	
float	اعداد کوچک اعشاری	
decimal	اعداد بزرگ اعشاری	
date	فقط تاریخ	انواع تاریخ:
time	فقط زمان	
date time	هم تاریخ هم زمان	
time stamp	تعداد ثانیه گذشته از تاریخ 1970/1/1 ساعت 00:00:00	

## راهنمایی برای نوع فیلدهای جدول links :

lid	ltext	lurl	ltitle	ladd_date	uid
کلید اصلی	متن لینک	آدرس لینک	متن تولتیپ	تاریخ افزودن	کلید خارجی از جدول کاربران
int	varchar	text	text	datetime	int

## مراحل کار با دیتابیس در PHP با استفاده از Mysql:

### ۱- اتصال به دیتابیس :

```
mysqli_connect (نام دیتابیس, پسر, نام کاربری, آدرس هاست) ;  
مثال:  
$link = mysqli_connect("myhost", "myuser", "mypassw", "mybd") ;
```

### ۲- تغییر یونیکد با استفاده از کد زیر:

```
mysqli_set_charset($link, "utf8") ;
```

### ۳- اجرای دستورات SQL با استفاده از خط دوم از کد زیر:

```
$query = "SELECT name FROM mytable";  
mysqli_query($link, $query) ;
```

## ساختار کلی درج در دیتابیس بر اساس SQL:

```
INSERT INTO `نام جدول` (`نام فیلد اول` , `نام فیلد دوم` , ...) VALUES  
("مقدار فیلد اول", "مقدار فیلد دوم", ...)
```

## لیست کردن داده‌های دیتابیس:

۱- اتصال به دیتابیس همان که در بالا گفته شده.

۲- اجرای کوئری زیر:

```
$result=mysqli_query ($link, "SELECT * FROM `table_name`");
```

۳- ریختن نتایج در یک آرایه:

```
$result=mysqli_fetch_array($result);
```

۴- چاپ نتایج با حلقه while به صورت زیر:

```
while ($result=mysqli_fetch_array($result,MYSQLI_ASSOC))  
{  
    //چاپ عناصر  
}
```

**مثال** وقتی که جداول را ایجاد کردیم با کدهای بالا می‌توانیم به دیتابیس وصل شده و دستورات خود را برای اضافه و لیست و حذف کردن انجام دهیم. این دستورات را در صفحه‌ای به نام جدولی که در دیتابیس ایجاد کرد ایم می‌نویسیم. در این جا نام صفحه `links.php` است. برای افزودن لینک

```
<?php  
switch ($_GET['op']) {  
    case 'add':  
        if(isset($_POST['add_link']))  
        {  
            $link=mysqli_connect(_DBHOST, _DBUSERNAME, _DBPASSWORD, _DBNAME);  
            mysqli_set_charset($link, "utf8");  
            $ltext=$_POST['ltext'];  
            $lurl=$_POST['lurl'];  
            $ltitle=$_POST['ltitle'];  
            $query="INSERT INTO  
`links`(`ltext`,`lurl`,`ltitle`,`ladd_date`,`lid`) VALUES  
('$ltext','$lurl','$ltitle',NOW(),1)";  
            $result=mysqli_query($link,$query);  
            if($result)
```

```

        {
            echo 'یک لینک با موفقیت ثبت شد';
        }
        else
        {
            echo 'مشکل در درج لینک به وجود آمد';
        }
    }

    echo '
    <h1>افزود لینک</h1>
    <form action="" method="post">
        متن لینک:<br>
        <input type="text" name="ltext"><br>
        آدرس لینک:<br>
        <input type="text" name="lurl" dir="ltr"><br>
        تولایپ لینک:<br>
        <input type="text" name="ltitle"><br>
        ثبت <input type="submit" name="add_link" value="ثبت
    لینک" class="btn btn-success">
    </form>
    ';
    break;
    case 'list':
        echo '
        <h1>لیست لینک ها</h1>';
        $link=
mysqli_connect(_DBHOST,_DBUSERNAME,_DBPASSWORD,_DBNAME);
        mysqli_set_charset($link, 'utf8');
        $query= "SELECT * FROM links";
        $result=mysqli_query($link,$query);

        echo '
        <table border ="1">
        <tr>
            <td>متن لینک</td>
            <td>آدرس لینک</td>
            <td>تولتپ لینک</td>
            <td>امکانات</td>
        </tr>
        ';
        while ($link=mysqli_fetch_array($result,MYSQLI_ASSOC)) {
            echo '
            <tr>
                <td>' . $link['ltext'] . '</td>
                <td>' . $link['lurl'] . '</td>
                <td>' . $link['ltitle'] . '</td>
                <td> <a
            href="links.php?op=delete&lid=' . $link['lid'] . '">
                </a></td>
            </tr>
            ';
        }
        echo '</table>';
    }

```



```

        break;
    case 'delete':
        $link=
mysqli_connect(_DBHOST,_DBUSERNAME,_DBPASSWORD,_DBNAME);
        mysqli_set_charset($link, 'utf8');
        $lid=$_GET['lid'];
        $result= mysqli_query($link,"DELETE FROM links WHERE
lid=$lid");
        if ($result)
        {
            echo "deleted successfully";

        }
        break;
    default:
        # code...
        break;
}
?>

```

## انواع الگوریتم‌های رمزنگاری:

**Reversible (برگشت پذیر):** داده‌های رمزگذاری شده را می‌توان به حالت اول برگرداند.

کاربرد: رمزگذاری داده‌هایی که می‌خواهیم اگر به سرقت رفت، بدون پسورد قابل خواندن نباشد و در مجموع، در جاهایی که کل داده در مقصد نیاز است.

مثال: رمزگذاری داده‌های روی فلش، لپ‌تاپ، داده‌های ارسالی از طریق شبکه و...

**Irreversible (برگشت ناپذیر):** داده‌های رمزگذاری شده را نمی‌توان به حالت اول برگرداند.

کاربرد: رمزگذاری داده‌هایی که به کل داده‌ها در مقصد نیاز نیست بلکه فقط گلچینی از داده‌ها که نماد کل داده باشد نیز کافی‌ست.

مثال: گرفتن یک چکیده از یک پسورد و ثبت در دیتابیس. گرفتن چکیده از یک فایل جهت بررسی صحت دریافت فایل در مقصد. (در بحث دانلود، با بررسی این چکیده در مقصد می‌توان فهمید فایل دانلود شده درست دانلود شده یا خیر)

## انواع الگوریتم‌های برگشت پذیر:

۱- **متقارن:** کلید رمزگذاری (یا Encryption) و کلید رمزگشایی (یا Decryption) یکسان است.

مثال:

DES = Data Encryption Standard

AES = Advanced Encryption System

۲- **نامتقارن:** کلید رمزگذاری با کلید رمزگشایی متفاوت است.

مثال: HTTPS , RSA

## پسورد امن:

۱- بیش از ۸ کاراکتر باشد.

۲- حداقل سه مورد از موارد زیر را شامل شود:

۱- حروف کوچک انگلیسی

۲- حروف بزرگ انگلیسی

۳- اعداد

۴- کاراکترهای غیر الفبایی

## توضیح الگوریتم برگشت‌ناپذیر یا همان Hash:

یک Hash که به آن Checksum، خلاصه پیام و یا اثرانگشت، نیز گفته می‌شود، فرآیندی است که با استفاده از روش‌های ریاضی، حجم یک جریان از داده را به صورت «یک‌طرفه» به یک طول ثابت کاهش می‌دهد (معمولاً ۱۲۸ و یا ۱۶۰ بیت). عملکرد hash، مشابه اثرانگشت یک شخص می‌باشد. اثرانگشت، پارامتری منحصر به فرد به منظور تشخیص هویت افراد بوده و در ادامه با استفاده از آن امکان دستیابی به سایر مشخصات افراد نظیر: رنگ چشم، قد، جنسیت و سایر موارد دلخواه، فراهم می‌گردد. اکثر توابع Hash از لحاظ رمزنگاری دارای عملکردی مشابه توابع رمزنگاری می‌باشند.

چند نکته مهم در مورد هش:

- کشف رشته اصلی از رشته هش شده‌ی آن (عملیات معکوس) به صورت کارا تقریباً غیر ممکن است.
- هر داده یک رشته هش شده کاملاً منحصر به فرد ایجاد می‌کند (احتمال یکی شدن خروجی‌های هش دو رشته متفاوت در الگوریتم MD5 یک در  $3.4028236692093846346337460743177e+38$  می‌باشد).

چون تعدادی از بیت‌های فایل رمزگذاری شده را گلیچین می‌کند پس قابل برگشت به حالت اول نیست چون بیت‌ها ناقص است. هر نوع داده‌ای به عنوان ورودی بگیرد تنها کد ۳۲ رقمی را می‌دهد حتی اگر فایل 1giga byte باشد باز هم ۳۲ بیت برمی‌گرداند. مثل پسورد کاربران در دیتابیس که با هش رمزگذاری می‌شود.

## هش کردن:

برای هش کردن از دستور زیر استفاده می‌کنیم:

```
md5 (رشته موردنظر) ;  
$pass=md5 ($_POST['pass']) ;
```

این قطعه کد پسورد کاربر را از فرم می‌گیرد و با الگوریتم md5 آن را hash کرده و در pass می‌ریزد.

الگوریتم دیگری به نام sha1 برای هش کردن وجود دارد:

```
sha1 (رشته موردنظر) ;
```

## توابع کار با رشته‌ها:

### تابع printf:

چاپ مقدار یک متغیر یا یک رشته در خروجی با توجه به نوع پارامتر بیان شده.

```
printf ("رشته", "نوع");  
printf ("%d", "2;DROP TABLE users");
```

\*حرف d برای اعداد صحیح و حرف f برای اعداد اعشاری است.

### تابع sprintf:

همان تابع printf است با این تفاوت که به جای چاپ در خروجی جواب را در یک متغیر می‌ریزد.

```
$lid=sprintf ("%d", $_GET['lid']);  
echo $lid;
```

این دستور پارامتری به نام lid را که انتظار می‌رود کاربر یک عدد صحیح را وارد کند به صورت امن از نوار آدرس گرفته و چاپ می‌کند.

نکته: این تابع کاربرد زیادی در امن کردن داده‌های ارسال شده از طرف کاربر دارد.

### تابع strtoupper:

این تابع رشته ورودی را به حروف بزرگ تبدیل می‌کند.

```
strtoupper ("رشته");  
$name=strtoupper ("hamidreza niroomand");  
echo $name;  
HAMIDREZA NIROOMAND ←خروجی
```

### تابع strtolower:

این تابع برعکس تابع strtoupper عمل می‌کند و حروف را به حروف کوچک تبدیل می‌کند.

```
strtolower ("رشته");  
$name=strtolower ("hamidreza niroomand");  
echo $name;  
hamidreza niroomand ←خروجی
```

### تابع ucfirst:

این تابع حرف اول کلمه اول را بزرگ و بقیه حروف را کوچک می‌کند.

```
ucfirst ("رشته");  
$name=ucfirst ("hamidreza niroomand");  
echo $name;  
Hamidreza niroomand ←خروجی
```

## تابع ucwords:

این تابع حروف اول کل عبارت را بزرگ می‌کند

```
ucwords ("رشته");  
$name=ucfirst ("hamidreza niroomand");  
echo $name;  
Hamidreza Nirromand ← خروجی
```

## تابع strstr:

رشته کامل را از جداکننده به بعد تکه می‌کند و در متغیر موردنظر می‌ریزد.

مثال) بررسی کنید ایمیلی که کاربر در نوار آدرس وارد می‌کند جیمیل است یا یاهو؟ کاربر در نوار آدرس این را تایپ می‌کند:

```
Localhost/index.php?email=niroomand@gmail.com  
strstr ("رشته جداکننده", "رشته کامل");  
  
$email=strstr ($_GET['email'], "@");  
echo $email;  
@gmail.com ← خروجی
```

## تابع strpos:

این تابع رشته دوم را در رشته اول جستجو می‌کند و اگر آن را یافت محل اولین وقوع را برمی‌گرداند و اگر نیافت مقدار null را برمی‌گرداند. پس تابع پرکاربردی برای بررسی وجود یک رشته در داخل رشته دیگر است.

مثال) در یک فرم، کاربر یک عبارت را در textarea تایپ می‌کند و یک کلمه را نیز در داخل یک input text تایپ می‌کند. با زدن دکمه Find نشان داده شود که آیا کلمه در عبارت وجود دارد یا خیر؟

```
strpos ("رشته جستجو", "رشته کامل");  
  
<form action=" " method="post">  
  Full String:<br><textarea name="fulltext"></textarea><br>  
  Search Query: <input type="q" name="q"> <br>  
  <input type="submit" name="search" value="Search!">  
</form>  
<?php  
if (isset($_POST['search'])) {  
    $res=strpos($_POST['fulltext'], $_POST['q']);  
    if (!empty($res)){  
        echo "حروف شما وجود دارد".$res;  
    }  
    else{  
        echo "حروف شما وجود ندارد";  
    }  
}  
?>
```

## تابع substr:

این تابع برای جدا کردن یک رشته از داخل یک رشته دیگر است.

```
strpos ("طول رشته جدایی", "نقطه شروع", "رشته کامل");
```

مثال) کد شهر را از داخل شماره تلفنی که کاربر در نوار آدرس وارد می‌کند، جدا کنید.

```
$tel=$_GET['tel'];  
echo substr ($tel,0,3);
```

**نکته:** بیان طول رشته اختیاری است و اگر طول رشته بیان نشود تا انتهای رشته کامل جدا خواهد شد.

مثال) ایمیل کاربر را وارد کنید و دامین ایمیل کاربر را بررسی کنید که yahoo.com یا gmail.com است اگر yahoo بود به yahoo.com و gmail اگر به gmail.com و اگر هیچ کدام نبود به سایت ایمیل لینک دهید.

```
<?php  
$email=$_GET['email'];  
$domain=substr($email, strpos($email,"@")+1);  
if ($domain=="gmail.com"){  
    echo '<a href=" http://google.com"> Google.com </a>';  
}elseif ($domain=="yahoo.com"){  
    echo '<a href=" http://yahoo.com">Yahoo.com</a>';  
}else{  
    echo '<a href=" http://$domain">$domain</a>';  
}  
?>
```

## تابع addslashes:

این تابع رشته‌ای را که کاربر وارد کرده را می‌گیرد و قبل از هر " و ' یک \ قرار می‌دهد. به این کار در اصطلاح escape کردن گفته می‌شود و هدف از این کار جلوگیری از هک‌های احتمالی و همینطور تداخل کوتیشن‌های رشته با کوتیشن‌های توابعی مانند echo و print است.

```
stripslashes (رشته);
```

## تابع stripslashes:

برعکس تابع بالایی کار می‌کند یعنی \ را از قبل از " و ' برمی‌دارد.

## تابع explode:

این تابع یک رشته را می‌گیرد و یک پارامتر جداکننده را نیز می‌گیرد و رشته را بنابر آن منفجر می‌کند. نتیجه این انفجار در یک آرایه قرار می‌گیرد و هر تکه از رشته‌ها در یکی از خانه‌های آرایه قرار می‌گیرد.

```
explode ("رشته کامل", "رشته جداکننده");
```

مثال) یک رشته را از ورودی بگیرید و تعداد کلمات آن را نمایش دهید.

```
<?php
$a=$_GET['a'];
$t=explode(" ",$a); //با هر فاصله کلمات را جدا میکند
echo count($t); //با این تابع تعداد خانه های آرایه را چاپ میکند
?>
```

## تابع implode:

عکس تابع explode عمل می کند و عناصر آرایه را به هم می چسباند و علامت بیان شده را بین آنها قرار می دهد.

مثال) اگر در آرایه مشخصات لینک قرار گرفته باشد که پسوند این فایلها CSV است و داده ها درون آن با کاما جدا می شود. می خواهیم این مشخصات را implode کنیم:

```
$linkinfo=array("shiva","abbasi","saveh");
echo implode(" ", $linkinfo);
```

## تابع htmlentities:

این تابع یک رشته را می گیرد و کاراکترهای خاص \*رشته را به معادل آن تبدیل میکند

\*کاراکترهای خاص: برخی کاراکترها ممکن است در html دچار اختلال کنند، در این صورت که کاربر انتظار دارد روی صفحه دیده شود اما نتیجه آن روی وب اعمال می شود که پردرسترین کاراکتر خاص < است این کاراکترها یک معادل دارند که برای نمایش آنها روس صفحه باید از معادل آنها استفاده کرد. برای مثال معادل < برابر با &lt; است و معادل > برابر با &gt; است.

مثال) خروجی برنامه زیر چیست؟

```
echo htmlentities("<h1>hello</h1>");
<h1>hello</h1> ← خروجی
```

## تابع trim:

این تابع یک رشته را می گیرد و فضای خالی میان رشته و ابتدا و انتهای رشته را حذف می کند. (در میان رشته، بیش از یک فاصله به یک فاصله تبدیل می شود و در ابتدا و انتها، حتی یک فاصله نیز حذف می شود)

مثال) خروجی برنامه زیر چیست؟

```
echo trim("jahad university");
jahad university ← خروجی
```

## تابع nl2br:

این تابع علامت newline یا \n را تشخیص داده و به تگ <br> تبدیل می‌کند. بیشترین کاربرد آن در نمایش متنی است که از دریافت می‌کند.

```
<form action=" " method="post">
    <textarea name="text"></textarea><br>
    <input type="submit" name="convert" value="convert!">
</form>
<?php
echo nl2br($_POST['text']);
?>
```

## تابع strlen:

یک رشته به عنوان ورودی می‌گیرد و طول رشته را برمی‌گرداند.

```
strlen "رشته";
```

برخی توابع رشته‌ای درمورد رشته‌های فارسی درست عمل نخواهد کرد مگر اینکه قبل از استفاده از توابع تابع زیر را فرا بخوانید و سپس به ابتدای توابع موردنظر mb\_ را اضافه کنید. (مثال)

```
mb_internal_encoding ('utf-8');
echo mb_strlen ("حمید رضا نیرومند");
16      خروجی ←
```



## تابع date:

باتوجه به فرمت بیان شده تاریخ را نمایش می‌دهد.

```
date ("فرمت");
```

\*تاریخ فعلی را با توجه به تاریخ لندن (نصف النهار مبدا) نشان می‌دهد. در فرمت می‌توان کاراکتر زد، برخی از آنها شامل:

\* J =روز را ۱ تا ۳۱

\* l = نام روز هفته

\* n = شماره ماه رقمی

\* F = نام ماه

\* y = دو رقم آخر سال

\* Y = چهار رقم سال

\* H = ساعت ۰۰ تا ۲۳

\* i = شماره دقیقه .. تا ۵۹ نشان می‌دهد.

## تعریف توابع در PHP :

ساختار کلی توابع:

```
function (لیست پارامترهای ورودی) نام تابع
{
    // عملیات توابع
}
```

توابع را می‌توان در هر جایی تعریف کرد و حتی میان یک تابع، تابع دیگری را تعریف کرد. اگر یک تابع داخل یک کلاس نوشته شود به آن متود گفته می‌شود و اگر خارج از کلاس تعریف شود فانکشن گفته می‌شود.

**مثال** یک تابع تعریف کنید که یک رشته کامل به عنوان پارامتر اول و یک رشته دیگر به عنوان پارامتر دریافت کند. اگر رشته اول در داخل رشته دوم وجود داشت **true** برگرداند و اگر نبود **false** برگرداند. یک فرم حاوی **text** طراحی کنید و کابر رشته کامل را در **textarea** و رشته‌ای که دنبال آن است را در **input text** وارد کند. آنها را گرفته و به تابع بفرستد و نسبت به پاسخ تابع پیغام مناسبی بفرستد.

```
<form action="" method="post">
    Full String:<br><textarea name="fulltext"></textarea><br>
    Search Query: <input type="q" name="q"> <br>
    <input type="submit" name="search" value="Search!">
</form>
<?php
if (isset($_POST['search'])) {
    $answer=Search($_POST['fulltext'], $_POST['q']);
    if ($answer) {
        echo "وجود دارد";
    }
    else{
        echo "وجود ندارد";
    }
}
function Search ($text,$query){
    if (strops ($text,$query))
        return true;
    else
        return false;
}
?>
```

## تعریف کلاس:

### ساختار کلی کلاس:

```
class نام کلاس{  
    data members  
    متودها  
}
```

### ساختار کلی شی:

```
$نام کلاس=new نام شی $;
```

مثال: یک متود به نام Welcome و یک کلاس به نام Student تعریف کنید که دارای سه دیتا-ممبر به نامهای \$stcode و \$fname و \$lname است. یک متود نیز در آن تعریف کنید که با فراخوانی آن از روی رقم آخر کد دانشجویی جنسیت دانشجو را تشخیص دهد (خانم=۰ و آقا=۱) و نسبت به جنسیت، خوش آمد بگوید. (جناب آقای یا سرکار خانم در ابتدای نام یا نام خانوادگی دانشجو نمایش دهد):

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Welcome</title>  
    <meta charset="utf-8">  
</head>  
<body>  
<?php  
$st = new Student();  
$st->stcode = "23324230";  
$st->fname = "zahra";  
$st->lname = "bostani";  
echo $st->Welcome();  
  
class Student  
{  
    public $stcode = "", $fname = "", $lname = "";  
    public function Welcome()  
    {  
        echo "خوش آمدید";  
        if (substr($this->stcode, -1) == 0) {  
            echo "سرکار خانم";  
        } else {  
            echo "جناب آقای";  
        }  
        echo $this->fname . ' ' . $this->lname;  
    }  
}  
?>  
</body>  
</html>
```

## SEO=Search Engine Optimization

به معنای «بهبود در موتور جستجو» است. که در اصل قواعدی است که تعیین می‌کند سایت خود را به چه صورت طراحی کنید تا در نتایج جستجو، در رتبه‌های بالاتری قرار بگیرد.

### راه‌های افزایش رتبه سایت:

#### ۱- از گذاشتن بنر خوش آمدگویی خود داری کنید

در بخش ورودی سایت از گذاشتن بنر های خوش آمد گویی و یا مواردی تحت عنوان «برای ورود به سایت اینجا را کلیک کنید» یا «لینک ورود به سایت» خود داری کنید زیرا ، پتانسیل بازگشت ویزیتور به سایت در همان ابتدایی کار به مقدار زیادی از دست خواهد رفت.

بدترین حالت هنگامی روی می دهد که در ورودی سایت، گرافیک یا انیمیشن "فلش" قرار گرفته که شامل لینک ورود یا enter در آن، در intro فلش است و دسترسی روباتهای موتور جستجوها را به آن عملاً غیر ممکن می سازد. حقیقت آن است که صفحه اصلی هر سایت، معمولاً بیشترین ویزیتور را داشته، بالاترین Rank را به همراه دارد و توسط روباتهای موتورهای جستجو بیشتر مورد دسترسی قرار می گیرد.

#### ۲- منوهای فلشی که قابل دستیابی توسط موتورهای جستجو نمی باشند

استفاده از منوهای فلش همراه با افکت های چشم نواز و متحرک در سیستم Navigation سایت، گرچه ممکن است زیبا و هیجان انگیز باشد، اما این تکنیک عملاً بررسی و فهرست بندی توسط روبات موتورهای جستجو را مختل کرده و عناوین و لینک های بکار برده شده در آن هیچگاه مورد بررسی و بالطبع صفحاتی که به آنها لینک داده شده مورد ارزش گذاری و فهرست بندی قرار نخواهد گرفت.

#### ۳- از Ajax بیش از حد استفاده نکنید

بسیاری از برنامه نویسان وب تلاش می کنند با استفاده بیش از حد از تکنولوژی Ajax، مخصوصاً در سیستم Navigation سایت، ویزیتورها را تحت تاثیر قرار می دهند، اما این امر یکی از بزرگترین اشتباهات SEO محسوب می شود.

توجه داشته باشید چون محتوی Ajax بصورت داینامیک Load می شوند، این محتوی، توسط spider موتورهای جستجو قابل رهگیری و فهرست بندی نخواهند بود.

نکته ی دیگر آنکه چون آدرس – URL – در قسمت هائی که از Ajax استفاده شده است، تغییر نمی کند، ویزیتور نمی تواند از این دست محتوی صفحات را برای بازدید دوستان بفرستد.

۴- عنوان هر پست باید به کل مطلب همان پست لینک شود یعنی وقتی موس را روی title متن قرار می دهیم بصورت لینک شده باشد نه متن خالی

#### ۵- تصاویری که در پرونده اچ تی ام ال قرار می گیرند دارای Alternate باشند

این مورد خیلی مهم هست. و دوتا مزیت اصلی داره. اولی اینکه اگر عکس لود نشه، متنی که در Alternate اون قرار دادید نمایش داده میشه مزیت بعدیش که مهم تر هست همون قضیه ی seo هست که خب مسلماً وقتی در موتورهای جستجو دنبال یک عکس میگردیم، این موتورها به Alternate عکس ها هم نگاه می کنند.

علاوه بر اسم عکس پس نام عکس هایی را هم که در سایت قرار می دهید درست انتخاب کنید

## ۶- عنوان اصلی هر مطلب را بهتر است **bold** کنید و بین تگهای **h1** و **h2** قرار دهید

### ۷- به موتورهای جستجو در شناسایی صفحات سایت کمک کنید

ساده ترین و بهترین راه حل برای شناساندن صفحات به موتورهای جستجو لینک به آنها در درون خود سایت است، در این میان نقش صفحه ای تحت عنوان **sitemap** پررنگتر است. صفحه ی **sitemap** می تواند شامل لینک تقریباً کل صفحات داخلی یک سایت باشد و این خود به یافتن آن صفحات و ایندکس شدن آنها در موتور جستجو کمک شایانی می کند. استفاده از سرویس رایگان **sitemap** گوگل می تواند از بهترین انتخاب ها باشد.

### ۸- لینک های شکسته را پیدا کنید

یکی از اشتباهاتی که خیلی از برنامه نویسان مرتکب می شوند این است که توسط کدهای جاوا اسکریپت عمل منتقل کردن لینک را به صفحه اصلی سایت

انجام می دهند و کد مورد نظر را در یک فایل **index.htm** قرار می دهند

در حالی که این امر باعث می شود موتور جستجو فقط فایل **index** را ببیند که داخل آن هیچ اطلاعاتی جز کد جاوا اسکریپت وجود ندارد. می توانید از امکانات **htaccess** برای **redirect** کردن بخش ها و صفحات جابجا شده و حتی کل یک سایت به آدرس جدید بهره ببرید.

مثلاً بوسیله فایل **htaccess** در **php** مبادرت به این امر بورزید.

مانند کد زیر :

```
redirect /index.htm http://www.yoursite.com/weblog/index.php
```

## ۹- بهتر است آخر هر پست کلمات کلیدی آن را بنویسید

## ۱۰- حتماً تگ **title** و متا تگ های زیر را در بخش **head** سایت قرار دهید

```
<title>عنوان سایت</title>
```

```
<meta name="keywords" content="کلمات کلیدی مرتبط با سایت">
```

```
<meta name="description" content="توضیح کوتاه در مورد سایت">
```

در انتخاب کلمات کلیدی کمی دقت نمایید ، هرچه را که به ذهنتان رسید فوراً به عنوان کلمه کلیدی قرار ندهید. تا حد امکان سعی کنید حدس بزنید کاربرانتان ممکن است چه عباراتی را جستجو کنند

## ۱۱- انواع روش های معرفی سایت به موتور های جستجو

الف) سرویس های رایگانی وجود دارند که شما با دادن آدرس سایت یا وبلاگ خود به این سایت، این سایت ها خودشان به طور خودکار سایت شما را در چندین موتور جستجو معروف ثبت می نمایند.

مانند سایت زیر :

<http://www.submitexpress.com/free-submission.html>

در این کار زیاده روی نکنید چون ممکن است نتایج منفی داشته باشد

ب) ثبت در دایرکتوری:

اگر بخواهم تعریفی از دایرکتوری داشته باشم می توان اینگونه تعریف کرد که دایرکتوری همان لینکدونی یا لینکستان های خودمان هست . یکی از معروف ترین این دایرکتوری ها سایت dmoz می باشد که وقتی شما سایتان را در این دایرکتوری ثبت کردید بسیاری از موتور های جستجو به سمت شما هجوم می آورند! و در نتیجه سبب افزایش Page Rank سایت یا وبلاگ شما می شود.

برای این کار اگر سایت شما فارسی هست با مراجعه به این بخش و انتخاب دسته مورد نظر و بعد کلیک بر روی پیشنهاد آدرس می توانید سایت یا وبلاگ خود را معرفی کنید.  
دایرکتوری های معروف :

<http://www.dmoz.org/>

<http://dir-link.com/>

## ۱۲- مدت زمان فعالیت یک آدرس وب domain

مدت زمانی که از ثبت یک domain مشخص می گذرد نیز، در رتبه بندی یک سایت در الگوریتم موتورهای جستجو اهمیت دارد. پس سن یک domain و تاریخچه فعال بودن آن یکی دیگر از ملاکهای موتورهای جستجو در ارزش گذاری صفحات آن آدرس خواهد بود.

## ۱۳ - صبور باشید!

امکان دارد کلیه بایدها و نبایدهای این نوشته را نکته به نکته در سایت خود رعایت کرده باشید، اما با این وجود هنوز صفحات شما در موتورهای جستجو، ایندکس و نمایش داده نشده باشند. علت چیست؟  
علت اینست که موفقیت در هر کاری، زمان لازم دارد. طبیعی است که برای ارزش گذاری، ایندکس شدن و نمایش صفحات سایت در موتورهای جستجو، مانند هر امر دیگری به زمان نیاز دارید، پس صبور باشید!  
لازم است بیاد داشته باشید محیط و ب یک دنیای رقابتی است و اگر سایت شما درباره ی موضوعاتی برپا شده و از کلمات کلیدی معینی استفاده می کند، در عین حال و همزمان هزاران و بلکه میلیونها صفحه ی دیگر با صفحه- سایت شما برای بدست آوردن مکان بهتر در نتایج موتورهای جستجو -SERB- در حال رقابت هستند.

## نمونه سؤالات کدنویسی:

توجه ۱: سؤالات تعریفی کاملاً مشخص هستند و نیازی به نمونه سؤال ندارند.

توجه ۲: سؤالات زیر، فقط نمونه سؤال هستند و ممکن است هیچ کدام به همین صورت در امتحان داده نشود اما بعید نیست که سؤالات امتحان شبیه به این ها باشد.

۱. کدهای مربوط به فرمی حاوی یک فیلد نام و نام خانوادگی و جنسیت را بنویسید. در صفحه پردازشگر، اطلاعات را بگیرید و با یک switch تصمیم بگیرید که شخص، خانم است یا آقا، اگر خانم بود چاپ کنید: سر کار خانم X Y خوش آمدید.

و اگر آقا بود، چاپ کنید: جناب آقای X Y خوش آمدید.

(به جای X Y باید نام و نام خانوادگی شخص چاپ شود)

۲. یک آرایه حاوی نام ماه‌های سال به فارسی (فرودین، اردیبهشت...) که از اندیس ۱ شروع شود ایجاد کنید. یک فرم طراح کنید که روی آن نوشته شده: ماه تولد خود را انتخاب کنید:

و مقابل آن یک منوی کرکره‌ای وجود دارد که ماه‌های سال را نمایش داده. (با حلقه foreach، عناصر آرایه را در قالب تگ option چاپ کنید. value در این تگ باید برابر با اندیس عناصر باشد) در صفحه‌ی پردازشگر، چاپ کنید: شما متولد ماه «X» هستید. (به جای X باید ماهی که کاربر انتخاب کرده چاپ شود.

۳. یک فرم لاگین طراحی کنید، در صفحه پردازشگر، اگر نام کاربری برابر با admin رمز عبور 123 بود، یک کوکی برای نام کاربری که ۳۰ روز عمر داشته باشد بسازید و نام کاربری را در آن ذخیره کنید و یک کوکی نیز برای رمز عبور که یک هفته عمر داشته باشد ایجاد کنید و هش شده‌ی رمز عبور را در آن ذخیره کنید.

۴. یک رشته انگلیسی از نوار آدرس بگیرید و موارد زیر را در مورد آن چاپ کنید:

a. طول رشته

b. تعداد کلمات رشته

c. آیا آن رشته یک ایمیل از سایت gmail.com است یا خیر؟

d. رشته با حروف کوچک انگلیسی

e. رشته به صورت «عبارت عنوانی»

۵. کدهای یک فرم حاوی نام کاربری و رمز عبور و یک صفحه‌ی پردازشگر که این اطلاعات را بگیرد و در دیتابیسی درج کند را بنویسید. کدهای لیست کردن داده‌ها را نیز بنویسید.
۶. قطعه کدی بنویسید که روزهای ماه (۱ تا ۳۱) را به عنوان گزینه‌های یک select چاپ کند.
۷. چند قطعه کد داده خواهد شد که باید نکات امنیتی که در آن‌ها رعایت نشده را بیان کنید...
۸. نکات امنیتی مربوط به توابع زیر را توضیح دهید:

- a. require
- b. htmlentities
- c. addslashes
- d. sha1
- e. sprintf
- f. ...و