**يتم تنفيذ البرنامج النصي PHP على الخادم، ويتم إرسال نتيجة HTML عادي إلى المستعرض.**

**PHP بناء الجملة الأساسية**

**يمكن وضع البرنامج النصي PHP أي مكان في المستند.**

**يبدأ البرنامج النصي PHP مع <فب؟ وينتهي ؟> :**

**<?php  
// PHP code goes here  
?>**

**ملحق الملف الافتراضي لملفات PHP هي ". فب".**

**ملف PHP يحتوي عادة علامات HTML، وبعض التعليمات البرمجية البرمجة PHP.**

**أدناه، لدينا مثال على ملف PHP بسيط، مع نصي PHP التي تستخدم وظيفة PHP المدمج في "صدى" لإخراج النص "مرحبا العالم!" على صفحة ويب:**

**مثال**

**<!DOCTYPE html>  
<html>  
<body>  
  
<h1>My first PHP page</h1>  
  
<?php  
echo "Hello World!";  
?>  
  
</body>  
</html>**[**تشغيل يل المثال »**](http://www.w3schools.com/php/showphp.asp?filename=demo_syntax)

**ملاحظة: يتم إنهاء البيانات بواسطة PHP منقوطة (؛). علامة إغلاق من كتلة من التعليمات البرمجية PHP يعني تلقائيا أيضا فاصلة منقوطة (لذلك لم يكن لديك ليكون فاصلة منقوطة إنهاء السطر الأخير من كتلة PHP).**

**تعليقات في PHP**

**تعليق في كود PHP هو خط الذي لا يقرأ / المنفذة كجزء من البرنامج. الغرض منه فقط هو أن تقرأ من قبل شخص يقوم بتحرير رمز!**

**تعليقات مفيدة ل:**

* **أن تدع الآخرين فهم ما تقومون به - تعليقات السماح المبرمجين الأخرى فهم ما كنت تفعل في كل خطوة (إذا كنت تعمل في مجموعة)**
* **لتذكير نفسك ما فعلت - شهدت معظم المبرمجين العودة إلى عملهم بعد سنة أو سنتين في وقت لاحق والحاجة إلى إعادة الرقم الى ما فعلوه. يمكن تعليق أذكركم ما كنت أفكر عندما كتبت رمز**

**PHP يدعم ثلاث طرق للالتعليق:**

**مثال**

**<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
// This is a single line comment  
  
# This is also a single line comment  
  
/\*  
This is a multiple lines comment block  
that spans over more than  
one line  
\*/  
?>  
  
</body>  
</html>تشغيل سبيل المثال »**

**PHP حالة الحساسية**

**في PHP، وجميع الدالات المعرفة من قبل المستخدم، والطبقات، والكلمات الرئيسية (على سبيل المثال إذا، وإلا، في حين، صدى، الخ) حساسة لحالة الأحرف.**

**في المثال أدناه، كل التصريحات صدى الثلاث أدناه هي قانونية (والمساواة):**

**مثال**

**<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
ECHO "Hello World!<br>";  
echo "Hello World!<br>";  
EcHo "Hello World!<br>";  
?>  
  
</body>  
</html>**[**تشغيل سبي الثال »**](http://www.w3schools.com/php/showphp.asp?filename=demo_syntax_case1)

**ولكن؛ في PHP، وجميع المتغيرات حساسة لحالة الأحرف.**

**في المثال أدناه، لن يؤدي إلا إلى بيان أول عرض قيمة متغير اللون $ (هذا لأن $ اللون، لون $، و $ اللون تعامل على أنها ثلاثة متغيرات مختلفة):**

**مثال**

**<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
$color="red";  
echo "My car is " . $color . "<br>";  
echo "My house is " . $COLOR . "<br>";  
echo "My boat is " . $coLOR . "<br>";  
?>  
  
</body>  
</html>**[**تشغيل سبيل المثال »**](http://www.w3schools.com/php/showphp.asp?filename=demo_syntax_case2)

# PHP 5 مقدمة

**ويتم تنفيذ البرامج النصية PHP على الخادم.**

## ما يجب أن تعرفه بالفعل

**قبل المتابعة يجب أن يكون لديك فهم أساسي من الإجراءات التالية:**

* **HTML**
* **CSS**
* **جافا سكريبت**

**إذا كنت ترغب في دراسة هذه الموضوعات الأولى، تجد الدروس على موقعنا على**[الصفحة الرئيسية](http://www.w3schools.com/default.asp)**.**

## ما هي PHP؟

* **PHP هو اختصار ل "PHP التشعبي قبل المعالج"**
* **PHP هي تستخدم على نطاق واسع، مفتوح المصدر لغة البرمجة**
* **ويتم تنفيذ البرامج النصية PHP على الخادم**
* **PHP لا تكلف شيئا، فهو حر لتحميل واستخدام**

|  |  |
| --- | --- |
| **مذكرة** | **PHP بسيطة للمبتدئين .  كما يقدم PHP العديد من الميزات المتقدمة للمبرمجين المحترفين.** |

## ما هو ملف PHP؟

* **يمكن أن تحتوي على ملفات PHP النص، HTML، CSS، وجافا سكريبت، ورمز PHP**
* **يتم تنفيذ كود PHP على الخادم، ويتم إرجاع النتيجة إلى المتصفح كما سهل HTML**
* **ملفات PHP لديهم التمديد ". فب"**

## ما يمكن PHP هل؟

* **PHP يمكن أن تولد دينامية محتوى الصفحة**
* **PHP يمكن إنشاء أو فتح، وقراءة، وكتابة، والملفات وثيق على ملقم**
* **PHP يمكن جمع بيانات النموذج**
* **PHP يمكن إرسال واستقبال ملفات تعريف الارتباط**
* **PHP يمكن إضافة أو حذف أو تعديل البيانات في قاعدة البيانات الخاصة بك**
* **PHP يمكن تقييد المستخدمين من الوصول إلى بعض الصفحات على موقع الويب الخاص بك**
* **PHP يمكن تشفير البيانات**

**مع PHP كنت لا تقتصر على إخراج HTML. يمكنك إخراج الصور، وملفات PDF، وحتى أفلام فلاش. يمكنك أيضا إخراج أي نص، مثل XHTML و XML.**

## لماذا PHP؟

* **PHP يعمل على منصات مختلفة (ويندوز، لينكس، ويونيكس، وماك OS X، الخ)**
* **PHP متوافق مع جميع تقريبا الخوادم المستخدمة اليوم (اباتشي، IIS، الخ)**
* **PHP تدعم مجموعة واسعة من قواعد البيانات**
* **PHP مجانية. تحميل البرنامج من الموارد PHP الرسمي:**[www.php.net](http://www.php.net/)
* **PHP من السهل تعلم ويعمل بكفاءة على جانب الملقم**

# PHP 5 التثبيت

## ما الذي أحتاج إليه؟

**للبدء في استخدام PHP، يمكنك:**

* **العثور على استضافة المواقع على الشبكة مع دعم PHP و MySQL**
* **تثبيت ملقم على جهاز الكمبيوتر الخاص بك على شبكة الإنترنت، ومن ثم تثبيت PHP و MySQL**

## استخدام استضافة المواقع على الشبكة مع دعم PHP

**إذا نشطت الخادم الخاص بك دعم PHP لا تحتاج إلى القيام بأي شيء.**

**مجرد خلق بعض ملفات PHP، ووضعها في الدليل الويب الخاص بك، وسيقوم الملقم تحليل لهم تلقائيا بالنسبة لك.**

**أنت لا تحتاج إلى تجميع أي شيء أو تثبيت أي أدوات إضافية.**

**لأن PHP مجانية، ومعظم المضيفين على الشبكة تقديم الدعم PHP.**

## إعداد PHP على جهاز الكمبيوتر الخاص بك

**ومع ذلك، إذا لا يدعم الخادم الخاص بك PHP، يجب عليك:**

* **تثبيت خادم الويب**
* **تثبيت PHP**
* **تثبيت قاعدة بيانات، مثل الخلية**

**موقع PHP الرسمي (PHP.net) لديه تعليمات التثبيت لPHP:**<http://php.net/manual/en/install.php>

|  |  |
| --- | --- |
| **مذكرة** | **نصيحة: للحصول على PHP وتشغيلها على الفور لنظام التشغيل Windows، يمكنك:**  [تحميل WebMatrix](http://www.microsoft.com/web/webmatrix/) |

# PHP 5 متغيرات

**المتغيرات هي "حاويات" لتخزين المعلومات:**

## مثال

**<?php  
$x=5;  
$y=6;  
$z=$x+$y;  
echo $z;  
?>**

## مثل الكثير من الجبر

**س = 5   
ص = 6   
ض = س + ص**

**في الجبر نستخدم الحروف (مثل x) لعقد القيم (مثل 5).**

**من التعبير ض = س + ص أعلاه، يمكننا حساب قيمة z لتكون 11.**

**في PHP وتسمى هذه الرسائل المتغيرات.**

|  |  |
| --- | --- |
| **مذكرة** | **التفكير في المتغيرات كما الحاويات لتخزين البيانات.** |

## متغيرات PHP

**كما هو الحال مع الجبر، والمتغيرات PHP يمكن استخدامها لاجراء القيم (س = 5) أو عبارات (ض = س + ص).**

**متغير يمكن أن يكون لها اسم مختصر (مثل x و y) أو اسم وصفي أكثر (العمر، carname، total\_volume).**

**قواعد للمتغيرات PHP:**

* **متغير يبدأ بعلامة دولار، تليها اسم المتغير**
* **يجب أن يبدأ اسم المتغير بحرف أو الشرطة السفلية**
* **لا يمكن بدء اسم متغير مع عدد**
* **يمكن أن يحتوي اسم المتغير الوحيد الأحرف الرقمية ألفا والسفلية (من الألف إلى الياء، 0-9، و\_)**
* **أسماء المتغيرات حساسة لحالة الأحرف ($ $ y و Y هما متغيرات مختلفة)**

|  |  |
| --- | --- |
| **مذكرة** | **تذكر أن متغيرات PHP تتحسس.** |

## خلق (تصريح) متغيرات PHP

**PHP لديه أي أمر لإعلان متغير.**

**يتم إنشاء متغير لحظة تعيين أول قيمة لها:**

## مثال

**<?php  
$txt="Hello world!";  
$x=5;  
$y=10.5;  
?>**

**بعد إعدام البيانات المذكورة أعلاه، المتغير النص سيعقد قيمة أهلا بالعالم! ، المتغير س سيعقد قيمة 5 ، والمتغير ذ سيعقد قيمة 10.5 .**

**ملاحظة: عند تعيين قيمة نصية إلى متغير، ووضع علامات الاقتباس حول القيمة.**

## PHP هو نوع فضفاضة اللغة

**في المثال أعلاه، لاحظ أن لم يكن لدينا لنقول PHP أي نوع بيانات المتغير.**

**PHP تلقائيا بتحويل المتغير إلى نوع البيانات الصحيح، وهذا يتوقف على قيمته.**

**بلغات أخرى مثل C، C + +، جافا و، يجب على مبرمج تعلن اسم ونوع المتغير قبل استخدامه.**

## متغيرات PHP نطاق

**في PHP، المتغيرات يمكن تعريف أي مكان في البرنامج النصي.**

**نطاق متغير هو جزء من السيناريو حيث يمكن الرجوع إليها المتغير / المستخدمة.**

**PHP لديه ثلاثة نطاقات متغير مختلفة:**

* **محلي**
* **عالمي**
* **ساكن**

## نطاق المحلية والعالمية

**أعلن متغير خارج وظيفة لديها النطاق العالمي ويمكن الوصول إليها فقط خارج وظيفة.**

**متغير أعلن في وظيفة لديها النطاق المحلي ويمكن الوصول إليها فقط في تلك الوظيفة.**

**المثال التالي باختبار المتغيرات مع نطاق المحلية والعالمية:**

## مثال

**<?php  
$x=5; // global scope  
  
function myTest()  
{  
$y=10; // local scope  
echo "<p>Test variables inside the function:<p>";  
echo "Variable x is: $x";  
echo "<br>";  
echo "Variable y is: $y";  
}   
  
myTest();  
  
echo "<p>Test variables outside the function:<p>";  
echo "Variable x is: $x";  
echo "<br>";  
echo "Variable y is: $y";  
?>**

**في المثال أعلاه أن هناك متغيرين $ $ x و y و وظيفة myTest (). $ x هو متغير العالمي منذ تم تعريفه خارج الدالة و$ y غير متغير محلي منذ إنشائه داخل الدالة.**

**عندما كنا إخراج القيم من اثنين من المتغيرات داخل الدالة myTest ()، فإنه يطبع قيمة $ ذ كما هو المعلن محليا، ولكن لا يمكن طباعة قيمة $ س منذ إنشائه خارج الدالة.**

**ثم، ونحن عندما خرج قيم المتغيرين خارج (وظيفة myTest)، فإنه يطبع قيمة $ س، ولكن لا يمكن طباعة قيمة $ ذ نظرا لأنه هو المتغير المحلي ويتم إنشاؤه داخل myTest (وظيفة) .**

|  |  |
| --- | --- |
| **مذكرة** | **هل يمكن أن يكون المتغيرات المحلية بنفس الاسم في وظائف مختلفة، وذلك لأن المتغيرات المحلية وتعترف بها سوى وظيفة التي تم الإعلان عنها.** |

## PHP الكلمة العالمية

**يتم استخدام الكلمة العالمية للوصول متغير عمومي من داخل دالة.**

**للقيام بذلك، استخدم الكلمة العالمي قبل المتغيرات (داخل الدالة):**

## مثال

**<?php  
$x=5;  
$y=10;  
  
function myTest()  
{  
global $x,$y;  
$y=$x+$y;  
}  
  
myTest();  
echo $y; // outputs 15  
?>**

**أيضا بتخزين جميع PHP المتغيرات العمومية في مجموعة يسمى $ GLOBALS [ *مؤشر* ]. في مؤشر يحمل اسم المتغير. هذه المجموعة هو أيضا يمكن الوصول إليها من خلال وظائف، ويمكن استخدامها لتحديث المتغيرات العالمية مباشرة.**

**المثال أعلاه يمكن إعادة كتابة مثل هذا:**

## مثال

**<?php  
$x=5;  
$y=10;  
function myTest()  
{  
$GLOBALS['y']=$GLOBALS['x']+$GLOBALS['y'];  
}   
  
myTest();  
echo $y; // outputs 15  
?>**

## PHP الكلمة الأساسية ثابتة

**عادة، عند اكتمال وظيفة / تنفيذ، يتم حذف كافة متغيراته. ومع ذلك، في بعض الأحيان نريد متغير محلي غير المراد حذفه. نحن في حاجة إليها للعمل أكثر من ذلك.**

**للقيام بذلك، استخدم ساكنة الكلمة عند أول بتعريف المتغير:**

## مثال

**<?php  
  
function myTest()  
{  
static $x=0;  
echo $x;  
$x++;  
}  
  
myTest();  
myTest();  
myTest();  
  
?>**

**ثم، في كل مرة يتم استدعاء الدالة، هذا المتغير سوف لا تزال لديها المعلومات التي ترد من آخر مرة تم فيها استدعاء الدالة.**

**ملاحظة: المتغير لا يزال المحلية إلى وظيفة.**

# PHP 5 متغيرات

**المتغيرات هي "حاويات" لتخزين المعلومات:**

## مثال

**<?php  
$x=5;  
$y=6;  
$z=$x+$y;  
echo $z;  
?>**

## مثل الكثير من الجبر

**س = 5   
ص = 6   
ض = س + ص**

**في الجبر نستخدم الحروف (مثل x) لعقد القيم (مثل 5).**

**من التعبير ض = س + ص أعلاه، يمكننا حساب قيمة z لتكون 11.**

**في PHP وتسمى هذه الرسائل المتغيرات.**

|  |  |
| --- | --- |
| **مذكرة** | **التفكير في المتغيرات كما الحاويات لتخزين البيانات.** |

## متغيرات PHP

**كما هو الحال مع الجبر، والمتغيرات PHP يمكن استخدامها لاجراء القيم (س = 5) أو عبارات (ض = س + ص).**

**متغير يمكن أن يكون لها اسم مختصر (مثل x و y) أو اسم وصفي أكثر (العمر، carname، total\_volume).**

**قواعد للمتغيرات PHP:**

* **متغير يبدأ بعلامة دولار، تليها اسم المتغير**
* **يجب أن يبدأ اسم المتغير بحرف أو الشرطة السفلية**
* **لا يمكن بدء اسم متغير مع عدد**
* **يمكن أن يحتوي اسم المتغير الوحيد الأحرف الرقمية ألفا والسفلية (من الألف إلى الياء، 0-9، و\_)**
* **أسماء المتغيرات حساسة لحالة الأحرف ($ $ y و Y هما متغيرات مختلفة)**

|  |  |
| --- | --- |
| **مذكرة** | **تذكر أن متغيرات PHP تتحسس.** |

## خلق (تصريح) متغيرات PHP

**PHP لديه أي أمر لإعلان متغير.**

**يتم إنشاء متغير لحظة تعيين أول قيمة لها:**

## مثال

**<?php  
$txt="Hello world!";  
$x=5;  
$y=10.5;  
?>**

**بعد إعدام البيانات المذكورة أعلاه، المتغير النص سيعقد قيمة أهلا بالعالم! ، المتغير س سيعقد قيمة 5 ، والمتغير ذ سيعقد قيمة 10.5 .**

**ملاحظة: عند تعيين قيمة نصية إلى متغير، ووضع علامات الاقتباس حول القيمة.**

## PHP هو نوع فضفاضة اللغة

**في المثال أعلاه، لاحظ أن لم يكن لدينا لنقول PHP أي نوع بيانات المتغير.**

**PHP تلقائيا بتحويل المتغير إلى نوع البيانات الصحيح، وهذا يتوقف على قيمته.**

**بلغات أخرى مثل C، C + +، جافا و، يجب على مبرمج تعلن اسم ونوع المتغير قبل استخدامه.**

## متغيرات PHP نطاق

**في PHP، المتغيرات يمكن تعريف أي مكان في البرنامج النصي.**

**نطاق متغير هو جزء من السيناريو حيث يمكن الرجوع إليها المتغير / المستخدمة.**

**PHP لديه ثلاثة نطاقات متغير مختلفة:**

* **محلي**
* **عالمي**
* **ساكن**

## نطاق المحلية والعالمية

**أعلن متغير خارج وظيفة لديها النطاق العالمي ويمكن الوصول إليها فقط خارج وظيفة.**

**متغير أعلن في وظيفة لديها النطاق المحلي ويمكن الوصول إليها فقط في تلك الوظيفة.**

**المثال التالي باختبار المتغيرات مع نطاق المحلية والعالمية:**

## مثال

**<?php  
$x=5; // global scope  
  
function myTest()  
{  
$y=10; // local scope  
echo "<p>Test variables inside the function:<p>";  
echo "Variable x is: $x";  
echo "<br>";  
echo "Variable y is: $y";  
}   
  
myTest();  
  
echo "<p>Test variables outside the function:<p>";  
echo "Variable x is: $x";  
echo "<br>";  
echo "Variable y is: $y";  
?>**

**في المثال أعلاه أن هناك متغيرين $ $ x و y و وظيفة myTest (). $ x هو متغير العالمي منذ تم تعريفه خارج الدالة و$ y غير متغير محلي منذ إنشائه داخل الدالة.**

**عندما كنا إخراج القيم من اثنين من المتغيرات داخل الدالة myTest ()، فإنه يطبع قيمة $ ذ كما هو المعلن محليا، ولكن لا يمكن طباعة قيمة $ س منذ إنشائه خارج الدالة.**

**ثم، ونحن عندما خرج قيم المتغيرين خارج (وظيفة myTest)، فإنه يطبع قيمة $ س، ولكن لا يمكن طباعة قيمة $ ذ نظرا لأنه هو المتغير المحلي ويتم إنشاؤه داخل myTest (وظيفة) .**

|  |  |
| --- | --- |
| **مذكرة** | **هل يمكن أن يكون المتغيرات المحلية بنفس الاسم في وظائف مختلفة، وذلك لأن المتغيرات المحلية وتعترف بها سوى وظيفة التي تم الإعلان عنها.** |

## PHP الكلمة العالمية

**يتم استخدام الكلمة العالمية للوصول متغير عمومي من داخل دالة.**

**للقيام بذلك، استخدم الكلمة العالمي قبل المتغيرات (داخل الدالة):**

## مثال

**<?php  
$x=5;  
$y=10;  
  
function myTest()  
{  
global $x,$y;  
$y=$x+$y;  
}  
myTest();  
echo $y; // outputs 15  
?>**

**أيضا بتخزين جميع PHP المتغيرات العمومية في مجموعة يسمى $ GLOBALS [ *مؤشر* ]. في مؤشر يحمل اسم المتغير. هذه المجموعة هو أيضا يمكن الوصول إليها من خلال وظائف، ويمكن استخدامها لتحديث المتغيرات العالمية مباشرة.**

**المثال أعلاه يمكن إعادة كتابة مثل هذا:**

## مثال

**<?php  
$x=5;  
$y=10;  
function myTest()  
{  
$GLOBALS['y']=$GLOBALS['x']+$GLOBALS['y'];  
}   
myTest();  
echo $y; // outputs 15  
?>**

## PHP الكلمة الأساسية ثابتة

**عادة، عند اكتمال وظيفة / تنفيذ، يتم حذف كافة متغيراته. ومع ذلك، في بعض الأحيان نريد متغير محلي غير المراد حذفه. نحن في حاجة إليها للعمل أكثر من ذلك.**

**للقيام بذلك، استخدم ساكنة الكلمة عند أول بتعريف المتغير:**

## مثال

**<?php  
  
function myTest()  
{  
static $x=0;  
echo $x;  
$x++;  
}  
myTest();  
myTest();  
myTest();  
?>**

**ثم، في كل مرة يتم استدعاء الدالة، هذا المتغير سوف لا تزال لديها المعلومات التي ترد من آخر مرة تم فيها استدعاء الدالة.**

**ملاحظة: المتغير لا يزال المحلية إلى وظيفة.**

# PHP 5 بناء الجملة

**يتم تنفيذ البرنامج النصي PHP على الخادم، ويتم إرسال نتيجة HTML عادي إلى المستعرض.**

## PHP بناء الجملة الأساسية

**يمكن وضع البرنامج النصي PHP أي مكان في المستند.**

**يبدأ البرنامج النصي PHP مع <فب؟ وينتهي ؟> :**

**<?php  
// PHP code goes here  
?>**

**ملحق الملف الافتراضي لملفات PHP هي ". فب".**

**ملف PHP يحتوي عادة علامات HTML، وبعض التعليمات البرمجية البرمجة PHP.**

**أدناه، لدينا مثال على ملف PHP بسيط، مع نصي PHP التي تستخدم وظيفة PHP المدمج في "صدى" لإخراج النص "مرحبا العالم!" على صفحة ويب:**

## مثال

**<!DOCTYPE html>  
<html>  
<body>  
  
<h1>My first PHP page</h1>  
<?php  
echo "Hello World!";  
?>  
</body>  
</html>**

**ملاحظة: يتم إنهاء البيانات بواسطة PHP منقوطة (؛). علامة إغلاق من كتلة من التعليمات البرمجية PHP يعني تلقائيا أيضا فاصلة منقوطة (لذلك لم يكن لديك ليكون فاصلة منقوطة إنهاء السطر الأخير من كتلة PHP).**

## تعليقات في PHP

**تعليق في كود PHP هو خط الذي لا يقرأ / المنفذة كجزء من البرنامج. الغرض منه فقط هو أن تقرأ من قبل شخص يقوم بتحرير رمز!**

**تعليقات مفيدة ل:**

* **أن تدع الآخرين فهم ما تقومون به - تعليقات السماح المبرمجين الأخرى فهم ما كنت تفعل في كل خطوة (إذا كنت تعمل في مجموعة)**
* **لتذكير نفسك ما فعلت - شهدت معظم المبرمجين العودة إلى عملهم بعد سنة أو سنتين في وقت لاحق والحاجة إلى إعادة الرقم الى ما فعلوه. يمكن تعليق أذكركم ما كنت أفكر عندما كتبت رمز**

**PHP يدعم ثلاث طرق للالتعليق:**

## مثال

**<!DOCTYPE html>  
<html>  
<body>  
<?php  
// This is a single line comment  
# This is also a single line comment  
/\*  
This is a multiple lines comment block  
that spans over more than  
one line  
\*/  
?>  
</body>  
</html>**

## PHP حالة الحساسية

**في PHP، وجميع الدالات المعرفة من قبل المستخدم، والطبقات، والكلمات الرئيسية (على سبيل المثال إذا، وإلا، في حين، صدى، الخ) حساسة لحالة الأحرف.**

**في المثال أدناه، كل التصريحات صدى الثلاث أدناه هي قانونية (والمساواة):**

## مثال

**<!DOCTYPE html>  
<html>  
<body>  
<?php  
ECHO "Hello World!<br>";  
echo "Hello World!<br>";  
EcHo "Hello World!<br>";  
?>  
</body>  
</html>**

**ولكن؛ في PHP، وجميع المتغيرات حساسة لحالة الأحرف.**

**في المثال أدناه، لن يؤدي إلا إلى بيان أول عرض قيمة متغير اللون $ (هذا لأن $ اللون، لون $، و $ اللون تعامل على أنها ثلاثة متغيرات مختلفة):**

## مثال

**<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
$color="red";  
echo "My car is " . $color . "<br>";  
echo "My house is " . $COLOR . "<br>";  
echo "My boat is " . $coLOR . "<br>";  
?>  
</body>  
</html>**

# PHP 5 الصدى والطباعة البيانات

**في PHP هناك طريقتان الأساسية للحصول على الإخراج: الصدى والطباعة.**

**في هذا البرنامج التعليمي نستخدم صدى (والطباعة) في المثال تقريبا. لذلك، يحتوي هذا الفصل أكثر من ذلك بقليل من المعلومات حول تلك التصريحات إخراج اثنين.**

## صدى PHP وطباعة البيانات

**هناك بعض الاختلافات بين الصدى والطباعة:**

* **صدى - يمكن إخراج واحدة أو أكثر من السلاسل**
* **الطباعة - يمكن إخراج سلسلة واحدة فقط، وتعود دائما 1**

**نصيحة: صدى وهامشيا مقارنة أسرع لطباعة كما الصدى لا يرجع أي قيمة.**

## وPHP بيان الصدى

**الصدى هو بناء اللغة، ويمكن استخدامها مع أو بدون أقواس: صدى صدى أو ().**

**سلاسل العرض**

**يوضح المثال التالي كيفية عرض سلاسل مختلفة مع الأمر echo (كما لاحظت أن السلاسل يمكن أن تحتوي على علامات HTML):**

## مثال

**<?php  
echo "<h2>PHP is fun!</h2>";  
echo "Hello world!<br>";  
echo "I'm about to learn PHP!<br>";  
echo "This", " string", " was", " made", " with multiple parameters.";  
?>**

**متغيرات العرض**

**يوضح المثال التالي كيفية عرض سلاسل والمتغيرات مع الأمر الصدى:**

## مثال

**<?php  
$txt1="Learn PHP";  
$txt2="W3Schools.com";  
$cars=array("Volvo","BMW","Toyota");  
  
echo $txt1;  
echo "<br>";  
echo "Study PHP at $txt2";  
echo "My car is a {$cars[0]}";  
?>**

## بيان PHP الطباعة

**الطباعة هو أيضا بناء اللغة، ويمكن استخدامها مع أو بدون أقواس: الطباعة أو الطباعة ().**

**سلاسل العرض**

**يوضح المثال التالي كيفية عرض سلاسل مختلفة مع الأمر الطباعة (كما لاحظت أن السلاسل يمكن أن تحتوي على علامات HTML):**

## مثال

**<?php  
print "<h2>PHP is fun!</h2>";  
print "Hello world!<br>";  
print "I'm about to learn PHP!";  
?>**

**متغيرات العرض**

**يوضح المثال التالي كيفية عرض سلاسل والمتغيرات مع الأمر الطباعة:**

## مثال

**<?php  
$txt1="Learn PHP";  
$txt2="W3Schools.com";  
$cars=array("Volvo","BMW","Toyota");  
  
print $txt1;  
print "<br>";  
print "Study PHP at $txt2";  
print "My car is a {$cars[0]}";  
?>**

# PHP أنواع البيانات

**سلسلة، صحيح، أرقام النقطة العائمة، منطقية، صفيف، كائن، NULL.**

## سلاسل PHP

**سلسلة هو سلسلة من الأحرف، مثل "أهلا بالعالم!".**

**سلسلة يمكن أن يكون أي نص داخل الاقتباس. يمكنك استخدام علامات الاقتباس المفردة أو المزدوجة:**

## مثال

**<?php   
$x = "Hello world!";  
echo $x;  
echo "<br>";   
$x = 'Hello world!';  
echo $x;  
?>**

## PHP الأعداد الصحيحة

**عدد صحيح هو رقم بدون العشرية.**

**قواعد صحيحة:**

* **يجب أن يكون عدد صحيح رقم واحد على الأقل (0-9)**
* **لا يمكن أن تحتوي على عدد صحيح فاصلة أو الفراغات**
* **يجب أن لا يكون عدد صحيح نقطة العشرية**
* **عدد صحيح يمكن أن تكون إما إيجابية أو سلبية**
* **يمكن تحديد الأعداد الصحيحة في ثلاثة أشكال: عشرية (على 10)، ست عشري (على أساس 16 - مسبوقة ب 0X) أو ثماني (8 المستندة - مسبوقة ب 0)**

**في المثال التالي وسوف نقوم باختبار أرقام مختلفة. وvar\_dump PHP (وظيفة) بإرجاع نوع البيانات والقيمة من المتغيرات:**

## مثال

**<?php   
$x = 5985;  
var\_dump($x);  
echo "<br>";   
$x = -345; // negative number   
var\_dump($x);  
echo "<br>";   
$x = 0x8C; // hexadecimal number  
var\_dump($x);  
echo "<br>";  
$x = 047; // octal number  
var\_dump($x);  
?>**

## أرقام النقطة العائمة PHP

**عدد النقطة العائمة هو رقم مع نقطة العشرية أو عدد في شكل الأسي.**

**في المثال التالي وسوف نقوم باختبار أرقام مختلفة. وvar\_dump PHP (وظيفة) بإرجاع نوع البيانات والقيمة من المتغيرات:**

## مثال

**<?php   
$x = 10.365;  
var\_dump($x);  
echo "<br>";   
$x = 2.4e3;  
var\_dump($x);  
echo "<br>";   
$x = 8E-5;  
var\_dump($x);  
?>**

# PHP الثوابت

**الثوابت لا تختلف عن المتغيرات إلا أنه بمجرد أن يتم تعريف انهم لا يمكن تغييرها أو غير معروف**

## PHP الثوابت

**الثابت هو معرف (اسم) عن قيمة بسيطة. لا يمكن تغيير القيمة خلال البرنامج النصي.**

**اسم ثابت صالحة يبدأ بحرف أو تسطير (أي علامة $ قبل اسم ثابت).**

**ملاحظة: على عكس المتغيرات، والثوابت هي عالمية تلقائيا عبر النصي بأكمله.**

## تعيين ثابت PHP

**لتعيين ثابت، استخدم تحديد (وظيفة) - يستغرق المعلمات الثلاث: تحدد المعلمة الأولى اسم ثابت، ويعرف المعلمة الثانية من قيمة ثابتة، ويحدد ما إذا كانت المعلمة الثالث اختياري يجب أن يكون اسم ثابت حالة الأحرف. الإعداد الافتراضي هو false.**

**على سبيل المثال يخلق أدناه المستمر لحالة الأحرف ، مع قيمة "مرحبا بكم في W3Schools.com!":**

## مثال

**<?php  
define("GREETING", "Welcome to W3Schools.com!");  
echo GREETING;  
?>**

**على سبيل المثال يخلق أدناه ثابت حالة الأحرف ، مع قيمة "مرحبا بكم في W3Schools.com!":**

## مثال

**<?php  
define("GREETING", "Welcome to W3Schools.com!", true);  
echo greeting;  
?>**

# PHP مشغلي

**ويبين هذا الفصل مشغلي المختلفة التي يمكن استخدامها في البرامج النصية PHP.**

## PHP الحسابي مشغلي

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator** | **Name** | **Example** | **Result** |
| **+** | **Addition** | **$x + $y** | **Sum of $x and $y** |
| **-** | **Subtraction** | **$x - $y** | **Difference of $x and $y** |
| **\*** | **Multiplication** | **$x \* $y** | **Product of $x and $y** |
| **/** | **Division** | **$x / $y** | **Quotient of $x and $y** |
| **%** | **Modulus** | **$x % $y** | **Remainder of $x divided by $y** |

**المثال التالي يوضح نتائج مختلفة باستخدام العوامل الحسابية المختلفة:**

## مثال

**<?php   
$x=10;   
$y=6;  
echo ($x + $y); // outputs 16  
echo ($x - $y); // outputs 4  
echo ($x \* $y); // outputs 60  
echo ($x / $y); // outputs 1.6666666666667   
echo ($x % $y); // outputs 4   
?>**

## مشغلي PHP التعيين

**يتم استخدام عوامل التعيين PHP لإرسال قيمة إلى متغير.**

**عامل التعيين الأساسية في PHP هو "=". فهذا يعني أن المعامل الأيسر يحصل لتعيين قيمة التعبير الاحالة على اليمين.**

|  |  |  |
| --- | --- | --- |
| **Assignment** | **Same as...** | **Description** |
| **x = y** | **x = y** | **The left operand gets set to the value of the expression on the right** |
| **x += y** | **x = x + y** | **Addition** |
| **x -= y** | **x = x - y** | **Subtraction** |
| **x \*= y** | **x = x \* y** | **Multiplication** |
| **x /= y** | **x = x / y** | **Division** |
| **x %= y** | **x = x % y** | **Modulus** |

**المثال التالي يوضح نتائج مختلفة باستخدام عوامل التعيين مختلفة:**

## مثال

**<?php   
$x=10;   
echo $x; // outputs 10  
$y=20;   
$y += 100;  
echo $y; // outputs 120  
$z=50;  
$z -= 25;  
echo $z; // outputs 25  
$i=5;  
$i \*= 6;  
echo $i; // outputs 30  
$j=10;  
$j /= 5;  
echo $j; // outputs 2  
$k=15;  
$k %= 4;  
echo $k; // outputs 3  
?>**

## PHP سلسلة مشغلي

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator** | **Name** | **Example** | **Result** |
| **.** | **Concatenation** | **$txt1 = "Hello" $txt2 = $txt1 . " world!"** | **Now $txt2 contains "Hello world!"** |
| **.=** | **Concatenation assignment** | **$txt1 = "Hello" $txt1 .= " world!"** | **Now $txt1 contains "Hello world!"** |

**المثال التالي يوضح نتائج استخدام مشغلي سلسلة:**

## مثال

**<?php  
$a = "Hello";  
$b = $a . " world!";  
echo $b; // outputs Hello world!   
  
$x="Hello";  
$x .= " world!";  
echo $x; // outputs Hello world!   
?>**

## مشغلي PHP الاضافة / إنقاص

|  |  |  |
| --- | --- | --- |
| **Operator** | **Name** | **Description** |
| **++$x** | **Pre-increment** | **Increments $x by one, then returns $x** |
| **$x++** | **Post-increment** | **Returns $x, then increments $x by one** |
| **--$x** | **Pre-decrement** | **Decrements $x by one, then returns $x** |
| **$x--** | **Post-decrement** | **Returns $x, then decrements $x by one** |

**المثال التالي يوضح نتائج مختلفة باستخدام مختلف مشغلي الزيادة / إنقاص:**

## مثال

**<?php  
$x=10;   
echo ++$x; // outputs 11  
  
$y=10;   
echo $y++; // outputs 10  
  
$z=5;  
echo --$z; // outputs 4  
  
$i=5;  
echo $i--; // outputs 5  
?>**

## PHP مقارنة مشغلي

**يتم استخدام عوامل المقارنة PHP لمقارنة قيمتين (رقم أو سلسلة):**

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator** | **Name** | **Example** | **Result** |
| **==** | **Equal** | **$x == $y** | **True if $x is equal to $y** |
| **===** | **Identical** | **$x === $y** | **True if $x is equal to $y, and they are of the same type** |
| **!=** | **Not equal** | **$x != $y** | **True if $x is not equal to $y** |
| **<>** | **Not equal** | **$x <> $y** | **True if $x is not equal to $y** |
| **!==** | **Not identical** | **$x !== $y** | **True if $x is not equal to $y, or they are not of the same type** |
| **>** | **Greater than** | **$x > $y** | **True if $x is greater than $y** |
| **<** | **Less than** | **$x < $y** | **True if $x is less than $y** |
| **>=** | **Greater than or equal to** | **$x >= $y** | **True if $x is greater than or equal to $y** |
| **<=** | **Less than or equal to** | **$x <= $y** | **True if $x is less than or equal to $y** |

**المثال التالي يوضح نتائج مختلفة من استخدام بعض عوامل المقارنة:**

## مثال

**<?php  
$x=100;   
$y="100";  
  
var\_dump($x == $y);  
echo "<br>";  
var\_dump($x === $y);  
echo "<br>";  
var\_dump($x != $y);  
echo "<br>";  
var\_dump($x !== $y);  
echo "<br>";  
  
$a=50;  
$b=90;  
  
var\_dump($a > $b);  
echo "<br>";  
var\_dump($a < $b);  
?>**

## PHP مشغلي المنطقية

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator** | **Name** | **Example** | **Result** |
| **and** | **And** | **$x and $y** | **True if both $x and $y are true** |
| **or** | **Or** | **$x or $y** | **True if either $x or $y is true** |
| **xor** | **Xor** | **$x xor $y** | **True if either $x or $y is true, but not both** |
| **&&** | **And** | **$x && $y** | **True if both $x and $y are true** |
| **||** | **Or** | **$x || $y** | **True if either $x or $y is true** |
| **!** | **Not** | **!$x** | **True if $x is not true** |

## مشغلي PHP صفيف

**وتستخدم شركات مجموعة بي إتش بي لمقارنة صفائف:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator** | **Name** | **Example** | **Result** |
| **+** | **Union** | **$x + $y** | **Union of $x and $y (but duplicate keys are not overwritten)** |
| **==** | **Equality** | **$x == $y** | **True if $x and $y have the same key/value pairs** |
| **===** | **Identity** | **$x === $y** | **True if $x and $y have the same key/value pairs in the same order and of the same types** |
| **!=** | **Inequality** | **$x != $y** | **True if $x is not equal to $y** |
| **<>** | **Inequality** | **$x <> $y** | **True if $x is not equal to $y** |
| **!==** | **Non-identity** | **$x !== $y** | **True if $x is not identical to $y** |

**المثال التالي يوضح نتائج مختلفة من استخدام مشغلي مجموعة مختلفة:**

## مثال

**<?php  
$x = array("a" => "red", "b" => "green");   
$y = array("c" => "blue", "d" => "yellow");   
$z = $x + $y; // union of $x and $y  
var\_dump($z);  
var\_dump($x == $y);  
var\_dump($x === $y);  
var\_dump($x != $y);  
var\_dump($x <> $y);  
var\_dump($x !== $y);  
?>**

# PHP إذا ... آخر ... ELSEIF البيانات

**وتستخدم عبارات الشرطي لتنفيذ إجراءات مختلفة استنادا إلى ظروف مختلفة.**

## PHP عبارات الشرطي

**في كثير من الأحيان عند كتابة التعليمات البرمجية، وتريد لتنفيذ إجراءات مختلفة لاتخاذ قرارات مختلفة. يمكنك استخدام عبارات الشرطية في التعليمات البرمجية للقيام بذلك.**

**في PHP لدينا العبارات الشرطية التالية:**

* **إذا البيان - ينفذ بعض التعليمات البرمجية فقط إذا كان الشرط المحدد غير صحيح**
* **إذا ... بيان آخر - ينفذ بعض التعليمات البرمجية إذا كان الشرط صحيحا ورمز آخر إذا كان الشرط غير صحيح**
* **إذا ... ELSEIF .... بيان آخر - يختار واحدة من عدة كتل من التعليمات البرمجية ليتم تنفيذها**
* **التبديل بيان - يختار واحدة من العديد من كتل من التعليمات البرمجية ليتم تنفيذها**

## PHP - وإذا بيان

**يتم استخدام عبارة if لتنفيذ بعض التعليمات البرمجية فقط إذا كان الشرط المحدد غير صحيح .**

### بناء الجملة

**if (*condition*)  
  { *code to be executed if condition is true*;}**

**المثال أدناه إرادة الإخراج "لقد يوم جيد!" إذا كان الوقت الحالي (ساعة) أقل من 20:**

## مثال

**<?php  
$t=date("H");  
if ($t<"20")  
  {  
  echo "Have a good day!";  
  }  
?>**

## PHP - وإذا ... بيان آخر

**استخدام إذا .... بيان آخر لتنفيذ بعض التعليمات البرمجية إذا كان الشرط صحيحا ورمز آخر إذا كان الشرط غير صحيح .**

### بناء الجملة

**if (*condition*)  
 {  
  *code to be executed if condition is true;*  
 }  
else  
 {  
  *code to be executed if condition is false;*}**

**المثال أدناه إرادة الإخراج "لقد يوم جيد!" إذا كان الوقت الحالي هو أقل من 20، و "ليلة سعيدة!" على خلاف ذلك:**

## مثال

**<?php  
$t=date("H");  
if ($t<"20")  
  {  
  echo "Have a good day!";  
  }  
else  
  {  
  echo "Have a good night!";  
  }  
?>**

## PHP - وإذا ... ELSEIF .... بيان آخر

**استخدام ELSEIF .... إذا ... بيان آخر ل اختيار واحد من عدة كتل من التعليمات البرمجية ليتم تنفيذها .**

### بناء الجملة

**if (*condition*)  
  {  
  *code to be executed if condition is true;*}  
elseif (*condition*)  
  {  
  *code to be executed if condition is true;*}else  
  {  
  *code to be executed if condition is false;* }**

**المثال أدناه إرادة الإخراج "هل لديك صباح الخير!" إذا كان الوقت الحالي هو أقل من 10، و"لقد يوم جيد!" إذا كان الوقت الحالي هو أقل من 20. وإلا فإنه سيتم إخراج "ليلة سعيدة!":**

## مثال

**<?php  
$t=date("H");  
if ($t<"10")  
  {  
  echo "Have a good morning!";  
  }  
elseif ($t<"20")  
  {  
  echo "Have a good day!";  
  }  
else  
  {  
  echo "Have a good night!";  
  }  
?>**

## PHP - بيان التبديل(جملة الswitch)

**وسيتم شرح البيان التبديل في الفصل التالي.**

# PHP التبديل بيان

**يتم استخدام عبارة التبديل لتنفيذ إجراءات مختلفة استنادا إلى ظروف مختلفة.**

## بيان التبديل PHP

**استخدام العبارة التحول إلى اختيار واحد من كتل كثيرة من رمز ليتم تنفيذها .**

### بناء الجملة

**switch (*n*)  
{  
case *label1:*  
  *code to be executed if n=label1;*  
  break;  
case *label2:*  
  *code to be executed if n=label2;*  
  break;  
case *label3:*  
  *code to be executed if n=label3;*  
  break;  
...  
default:  
  *code to be executed if n is different from all labels;*  
}**

**هذه هي الطريقة التي يعمل: أولا لدينا تعبير واحد *ن* (في معظم الأحيان متغير)، التي يتم تقييمها مرة واحدة. ثم تتم مقارنة قيمة التعبير مع القيم لكل حالة في الهيكل. إذا كان هناك تطابق، يتم تنفيذ كتلة من التعليمات البرمجية المقترنة هذه الحالة. استخدام فاصل لمنع رمز من الترشح في القضية التالية تلقائيا. و الافتراضي يستخدم بيان إذا تم العثور على أي تطابق.**

## مثال

**<?php  
$favcolor="red";  
switch ($favcolor)  
{  
case "red":  
  echo "Your favorite color is red!";  
  break;  
case "blue":  
  echo "Your favorite color is blue!";  
  break;  
case "green":  
  echo "Your favorite color is green!";  
  break;  
default:  
  echo "Your favorite color is neither red, blue, or green!";  
}  
?>**

# PHP ل حلقات

**PHP لحلقات تنفيذ كتلة من التعليمات البرمجية لعدد محدد من المرات.**

## وPHP للحلقة

**يتم استخدام لحلقة عندما كنت تعرف مسبقا كم مرة يجب تشغيل البرنامج النصي.**

### بناء الجملة

**for (*init counter; test counter; increment counter*)  
  {  
  *code to be executed;*  
  }**

**المعلمات:**

* ***الحرف الأول عداد* : تهيئة قيمة العداد حلقة**
* ***اختبار العداد* : تم تقييم لكل حلقة التكرار. إذا كان يقيم إلى TRUE، تواصل الحلقة. إذا كان يقيم إلى FALSE، تنتهي الحلقة.**
* ***مكافحة الزيادة* : زيادة قيمة العداد حلقة**

**يعرض المثال التالي أعداد 0-10:**

## مثال

**<?php   
for ($x=0; $x<=10; $x++)  
  {  
  echo "The number is: $x <br>";  
  }   
?>**

## حلقة PHP فورياش(foreach)

**حلقة فورياش يعمل فقط على المصفوفات، ويستخدم لتنفيذ حلقات عبر كل زوج مفتاح / قيمة في صفيف.**

### بناء الجملة

**foreach ($*array*as$*value*)  
  {  
  *code to be executed;*  
  }**

**لكل حلقة التكرار، يتم تعيين قيمة عنصر صفيف الحالي لقيمة دولار، ويتم نقل المؤشر مجموعة تلو الآخر، حتى تصل إلى عنصر صفيف الماضي.**

**يوضح المثال التالي حلقة التي سيتم إخراج القيم من مجموعة معينة ($ ألوان):**

## مثال

**<?php   
$colors = array("red","green","blue","yellow");   
foreach ($colors as $value)  
  {  
  echo "$value <br>";  
  }  
?>**

# PHP وظائف(function)

**القوة الحقيقية للPHP يأتي من وظائفها، ولديها أكثر من 1000 وظائف المضمنة.**

## وظائف PHP معرف المستخدم

**إلى جانب وظائف PHP مدمج، يمكننا خلق الوظائف الخاصة بنا.**

**وظيفة هو كتلة من البيانات التي يمكن استخدامها مرارا وتكرارا في البرنامج.**

**وسوف يتم تنفيذ وظيفة مباشرة عند تحميل الصفحة.**

**وسيتم تنفيذ وظيفة بواسطة استدعاء الدالة.**

## إنشاء معرف المستخدم وظيفة في PHP

**مستخدم تعريف الدالة الإعلان يبدأ بكلمة "وظيفة":**

### بناء الجملة

**function *functionName*()  
{  
*code to be executed*;  
}**

**ملاحظة: اسم وظيفة يمكن أن تبدأ بحرف أو تسطير (وليس رقم).**

**نصيحة: إعطاء وظيفة وهو الاسم الذي يعكس ما لا وظيفة!**

|  |  |
| --- | --- |
| **مذكرة** | **تذكر أن أسماء ظيفة حساسة لحالة الأحرف.** |

**في المثال أدناه، ونحن إنشاء دالة باسم "writeMsg ()". متعرج فتح ({) يدل على بداية رمز الدالة وقوس الإغلاق متعرجة (}) تشير إلى نهاية وظيفة. مخرجات وظيفة "أهلا بالعالم!".لاستدعاء الدالة، مجرد كتابة اسمها:**

## مثال

**<?php  
function writeMsg()  
{  
echo "Hello world!";  
}  
  
writeMsg(); // call the function  
?>**

## وسيطات PHP وظيفة

**يمكن تمرير المعلومات إلى وظائف من خلال الحجج. حجة هو تماما مثل متغير.**

**يتم تحديد الحجج بعد اسم الدالة، داخل الأقواس. يمكنك إضافة العديد من الحجج على النحو الذي تريد، فقط تفريق بينها بفاصلة.**

**المثال التالي لديه وظيفة مع وسيطة واحدة ($ FNAME). عندما يتم استدعاء الدالة familyName ()، ونحن أيضا تمرير اسم (على سبيل المثال جاني)، ويستخدم اسم داخل الدالة، والتي بإخراج عدة مختلفة الأسماء الأولى، ولكن على قدم المساواة والاسم الأخير:**

## مثال

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

**المثال التالي لديه وظيفة مع اثنين من الحجج ($ $ FNAME والسنة):**

## مثال

**<?php  
function familyName($fname,$year)  
{  
echo "$fname Refsnes. Born in $year <br>";  
}  
  
familyName("Hege","1975");  
familyName("St�le","1978");  
familyName("Kai Jim","1983");  
?>**

## PHP افتراضي سيطة القيمة

**يوضح المثال التالي كيفية استخدام المعلمة الافتراضية. إذا كنا استدعاء الدالة setHeight () بدون وسائط فإنه يأخذ القيمة الافتراضية كوسيطة:**

## مثال

**<?php  
function setHeight($minheight=50)  
{  
echo "The height is : $minheight <br>";  
}  
  
setHeight(350);  
setHeight(); // will use the default value of 50  
setHeight(135);  
setHeight(80);  
?>**

## وظائف PHP - القيم وبالعودة

**السماح لدالة ترجع قيمة، استخدم العبارة العودة:**

## مثال

**<?php  
function sum($x,$y)  
{  
$z=$x+$y;  
return $z;  
}  
  
echo "5 + 10 = " . sum(5,10) . "<br>";  
echo "7 + 13 = " . sum(7,13) . "<br>";  
echo "2 + 4 = " . sum(2,4);  
?>**

# PHP الفرز صالحة(الترتيب في php)

**العناصر في صفيف يمكن فرزها في أبجدي أو عددي النظام، أو تنازلي تصاعدي.**

## PHP - ترتيب وظائف لصالحة

**في هذا الفصل، ونحن سوف تذهب من خلال وظائف مجموعة بي إتش بي النوع التالي:**

* **صفائف الفرز في ترتيب تصاعدي - نوع ()**
* **صفائف الفرز في ترتيب تنازلي - rsort ()**
* **asort () - صفائف النقابي الفرز في ترتيب تصاعدي، وفقا لقيمة**
* **ksort () - صفائف النقابي الفرز في ترتيب تصاعدي، وفقا لمفتاح**
* **arsort () - صفائف النقابي الفرز في ترتيب تنازلي، وفقا لقيمة**
* **krsort () - صفائف النقابي الفرز في ترتيب تنازلي، وفقا لمفتاح**

## نوع صفيف في ترتيب تصاعدي - نوع ()

**المثال التالي يفرز عناصر $ سيارات مجموعة بترتيب أبجدي تصاعدي:**

## مثال

**<?php  
$cars=array("Volvo","BMW","Toyota");  
sort($cars);  
?>**

**المثال التالي يفرز عناصر من مجموعة أرقام دولار في الصعود الترتيب العددي:**

## مثال

**<?php  
$numbers=array(4,6,2,22,11);  
sort($numbers);  
?>**

## نوع صفيف في ترتيب تنازلي ترتيب - rsort ()

**المثال التالي يفرز عناصر $ السيارات المصفوفة في ترتيب تنازلي حسب الحروف الأبجدية:**

## مثال

**<?php  
$cars=array("Volvo","BMW","Toyota");  
rsort($cars);  
?>**

**المثال التالي يفرز عناصر من مجموعة أرقام دولار في تنازلي الترتيب العددي:**

## مثال

**<?php  
$numbers=array(4,6,2,22,11);  
rsort($numbers);  
?>**

## نوع صفيف في ترتيب تصاعدي، وفقا لقيمة - asort ()

**المثال التالي يفرز مجموعة النقابي في ترتيب تصاعدي، وفقا للقيمة:**

## مثال

**<?php  
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");  
asort($age);  
?>**

## نوع صفيف في ترتيب تصاعدي، وفقا لمفتاح - ksort ()

**المثال التالي يفرز مجموعة النقابي في ترتيب تصاعدي، وفقا لمفتاح:**

## مثال

**<?php  
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");  
ksort($age);  
?>**

## نوع صفيف في ترتيب تنازلي ترتيب، وفقا لقيمة - arsort ()

**المثال التالي يفرز مجموعة النقابي في ترتيب تنازلي، وفقا للقيمة:**

## مثال

**<?php  
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");  
arsort($age);  
?>**

## نوع صفيف في ترتيب تنازلي ترتيب، وفقا لمفتاح - krsort ()

**المثال التالي يفرز مجموعة النقابي في ترتيب تنازلي، وفقا لمفتاح:**

## مثال

**<?php  
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");  
krsort($age);  
?>**

## كاملة PHP صفيف المرجعي

**للإشارة كاملة من جميع وظائف مجموعة، انتقل إلى لدينا على أكمل**[مجموعة بي إتش بي المرجعي](http://www.w3schools.com/php/php_ref_array.asp)**.**

**يحتوي على المرجع وصفا موجزا، وأمثلة على الاستخدام، ل**

*PHP 5 بناء الجملة*

*يتم تنفيذ البرنامج النصي PHP على الخادم، ويتم إرسال نتيجة HTML عادي إلى المستعرض.*

*PHP بناء الجملة الأساسية*

*يمكن وضع البرنامج النصي PHP أي مكان في المستند.*

*يبدأ البرنامج النصي PHP مع <فب؟ وينتهي ؟> :*

*<?php  
// PHP code goes here  
?>*

*ملحق الملف الافتراضي لملفات PHP هي ". فب".*

*ملف PHP يحتوي عادة علامات HTML، وبعض التعليمات البرمجية البرمجة PHP.*

*أدناه، لدينا مثال على ملف PHP بسيط، مع نصي PHP التي تستخدم وظيفة PHP المدمج في "صدى" لإخراج النص "مرحبا العالم!" على صفحة ويب:*

*مثال*

*<!DOCTYPE html>  
<html>  
<body>  
  
<h1>My first PHP page</h1>  
  
<?php  
echo "Hello World!";  
?>  
  
</body>  
</html>*

*ملاحظة: يتم إنهاء البيانات بواسطة PHP منقوطة (؛). علامة إغلاق من كتلة من التعليمات البرمجية PHP يعني تلقائيا أيضا فاصلة منقوطة (لذلك لم يكن لديك ليكون فاصلة منقوطة إنهاء السطر الأخير من كتلة PHP).*

*تعليقات في PHP*

*تعليق في كود PHP هو خط الذي لا يقرأ / المنفذة كجزء من البرنامج. الغرض منه فقط هو أن تقرأ من قبل شخص يقوم بتحرير رمز!*

*تعليقات مفيدة ل:*

* *أن تدع الآخرين فهم ما تقومون به - تعليقات السماح المبرمجين الأخرى فهم ما كنت تفعل في كل خطوة (إذا كنت تعمل في مجموعة)*
* *لتذكير نفسك ما فعلت - شهدت معظم المبرمجين العودة إلى عملهم بعد سنة أو سنتين في وقت لاحق والحاجة إلى إعادة الرقم الى ما فعلوه. يمكن تعليق أذكركم ما كنت أفكر عندما كتبت رمز*

*PHP يدعم ثلاث طرق للالتعليق:*

*مثال*

*<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
// This is a single line comment  
  
# This is also a single line comment  
  
/\*  
This is a multiple lines comment block  
that spans over more than  
one line  
\*/  
?>  
  
</body>  
</html>*

*PHP حالة الحساسية*

*في PHP، وجميع الدالات المعرفة من قبل المستخدم، والطبقات، والكلمات الرئيسية (على سبيل المثال إذا، وإلا، في حين، صدى، الخ) حساسة لحالة الأحرف.*

*في المثال أدناه، كل التصريحات صدى الثلاث أدناه هي قانونية (والمساواة):*

*مثال*

*<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
ECHO "Hello World!<br>";  
echo "Hello World!<br>";  
EcHo "Hello World!<br>";  
?>  
  
</body>  
</html>*

*ولكن؛ في PHP، وجميع المتغيرات حساسة لحالة الأحرف.*

*في المثال أدناه، لن يؤدي إلا إلى بيان أول عرض قيمة متغير اللون $ (هذا لأن $ اللون، لون $، و $ اللون تعامل على أنها ثلاثة متغيرات مختلفة):*

*مثال*

*<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
$color="red";  
echo "My car is " . $color . "<br>";  
echo "My house is " . $COLOR . "<br>";  
echo "My boat is " . $coLOR . "<br>";  
?>  
  
</body>  
</html>*

*PHP التعامل مع نموذج*

*وPHP superglobals \_GET دولار و \_POST تستخدم لجمع البيانات النموذج.*

*PHP - نموذج HTML بسيط*

*يعرض المثال التالي نموذج HTML بسيطة مع اثنين من إدخال الحقول وزر الإرسال:*

*مثال*

*<html>  
<body>  
  
<form action="welcome.php" method="post">  
Name: <input type="text" name="name"><br>  
E-mail: <input type="text" name="email"><br>  
<input type="submit">  
</form>  
  
</body>  
</html>*

*عندما يملأ المستخدم الاستمارة أعلاه وينقر على زر إرسال، يتم إرسال بيانات النموذج للتجهيز لملف PHP المسمى "welcome.php". يتم إرسال بيانات النموذج مع أسلوب HTTP POST.*

*لعرض البيانات المقدمة هل يمكن ببساطة أردد كل المتغيرات. و"welcome.php" يبدو مثل هذا:*

*<html>  
<body>  
  
Welcome <?php echo $\_POST["name"]; ?><br>  
Your email address is: <?php echo $\_POST["email"]; ?>  
  
</body>  
</html>*

*الإخراج يمكن أن يكون شيئا مثل هذا:*

*Welcome John  
Your email address is john.doe@example.com*

*كما يمكن تحقيق نفس النتيجة باستخدام طريقة GET HTTP:*

*مثال*

*<html>  
<body>  
  
<form action="welcome\_get.php" method="get">  
Name: <input type="text" name="name"><br>  
E-mail: <input type="text" name="email"><br>  
<input type="submit">  
</form>  
  
</body>  
</html>*

*و"welcome\_get.php" يبدو مثل هذا:*

*<html>  
<body>  
  
Welcome <?php echo $\_GET["name"]; ?><br>  
Your email address is: <?php echo $\_GET["email"]; ?>  
  
</body>  
</html>*

*رمز أعلاه هو بسيط جدا. ومع ذلك، فإن أهم شيء مفقود. تحتاج إلى التحقق من صحة بيانات النموذج إلى حماية السيناريو الخاص بك من الشيفرات الخبيثة.*

|  |  |
| --- | --- |
| *مذكرة* | *اعتقد الأمن عند معالجة أشكال PHP!*  *هذه الصفحة لا تحتوي على أي التحقق من صحة النموذج، لكنه يظهر فقط كيف يمكنك إرسال واسترجاع بيانات النموذج.*  *ومع ذلك، فإن الصفحات التالية تبين كيفية معالجة أشكال PHP مع الأمن في الاعتبار! التحقق من صحة بيانات النموذج السليم للالمهم لحماية النموذج الخاص بك من المتسللين والمتطفلين!* |

*الحصول على مقابل وظيفة*

*كلا GET و POST إنشاء صفيف (مثل مجموعة (مفتاح => قيمة، key2 => VALUE2، key3 => value3، ...)). هذه المجموعة يحمل أزواج مفتاح / قيمة، حيث المفاتيح هي أسماء عناصر التحكم في النماذج والقيم هي إدخال البيانات من المستخدم.*

*كلا GET و POST كما يتم التعامل مع \_GET دولار و \_POST. هذه هي superglobals، الأمر الذي يعني أنها دائما يمكن الوصول إليها، بغض النظر عن نطاق - ويمكنك الوصول إليها من أي وظيفة أو الطبقة أو ملف دون الحاجة إلى القيام بأي شيء خاص.*

*$ \_GET هو مجموعة من المتغيرات التي تم تمريرها إلى البرنامج النصي الحالي عبر المعلمات URL.*

*$ \_POST هو مجموعة من المتغيرات التي تم تمريرها إلى البرنامج النصي الحالي عن طريق الأسلوب POST HTTP.*

*عند استخدام GET؟*

*المعلومات المرسلة من نموذج مع أسلوب GET هو مرئية للجميع (يتم عرض كافة أسماء وقيم المتغيرات في عنوان URL). GET لديها أيضا حدود لكمية المعلومات لإرسال. القيد هو حوالي 2000 حرفا. ومع ذلك، لأنه يتم عرض المتغيرات في عنوان URL، فمن الممكن أن المرجعية في الصفحة. وهذا يمكن أن يكون مفيدا في بعض الحالات.*

*GET يمكن استخدامها لإرسال البيانات غير الحساسة.*

*ملاحظة: GET يجب ألا تستخدم أبدا لإرسال كلمات المرور أو غيرها من المعلومات الحساسة!*

*عند استخدام وظيفة؟*

*المعلومات المرسلة من نموذج مع الأسلوب POST هي غير مرئية للآخرين (هي جزء لا يتجزأ عن أسماء / القيم داخل الجسم من طلب HTTP) ولديه لا حدود على كمية من المعلومات لإرسال.*

*وعلاوة على ذلك وظيفة يدعم وظائف متقدمة مثل دعم متعددة الأجزاء المدخل الثنائي أثناء تحميل الملفات إلى الخادم.*

*ومع ذلك، لأنه لم يتم عرض المتغيرات في عنوان URL، فإنه ليس من الممكن أن المرجعية في الصفحة.*

|  |  |
| --- | --- |
| *مذكرة* | *المطورين يفضلون وظيفة لإرسال بيانات النموذج.* |

*المقبل؛ دعونا نرى كيف يمكننا معالجة PHP يشكل طريقة آمنة!*

*PHP التحقق من صحة النموذج*

*هذا والفصول التالية تظهر كيفية استخدام PHP للتحقق من صحة بيانات النموذج.*

*PHP التحقق من صحة النموذج*

|  |  |
| --- | --- |
| *مذكرة* | *اعتقد الأمن عند معالجة أشكال PHP!*  *سوف تظهر هذه الصفحات كيفية معالجة أشكال PHP مع الأمن في الاعتبار. التحقق من صحة بيانات النموذج السليم للالمهم لحماية النموذج الخاص بك من المتسللين والمتطفلين!* |

*شكل HTML سوف نعمل في في هذه الفصول، ويحتوي على حقول الإدخال المختلفة: مطلوب والحقول الاختيارية النص، أزرار الراديو، وزر الإرسال:*

*قواعد التحقق من صحة النموذج أعلاه هي كما يلي:*

|  |  |
| --- | --- |
| *Field* | *Validation Rules* |
| *Name* | *Required. + Must only contain letters and whitespace* |
| *E-mail* | *Required. + Must contain a valid email address (with @ and .)* |
| *Website* | *Optional. If present, it must contain a valid URL* |
| *Comment* | *Optional. Multi-line input field (textarea)* |
| *Gender* | *Required. Must select one* |

*أولا نحن سوف ننظر في كود HTML عادي للنموذج:*

*حقول النص*

*الاسم، البريد الإلكتروني، والحقول الموقع هي عناصر إدخال النص، وحقل التعليق هو ناحية النص. رمز HTML يبدو مثل هذا:*

*Name: <input type="text" name="name">  
E-mail: <input type="text" name="email">  
Website: <input type="text" name="website">  
Comment: <textarea name="comment" rows="5" cols="40"></textarea>*

*أزرار الراديو*

*الحقول بين الجنسين هي أزرار الراديو ورمز HTML يبدو مثل هذا:*

*Gender:  
<input type="radio" name="gender" value="female">Female  
<input type="radio" name="gender" value="male">Male*

*نموذج العنصر*

*رمز HTML من النموذج يبدو مثل هذا:*

*<form method="post" action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>">*

*عندما يتم إرسال النموذج، يتم إرسال بيانات النموذج مع أسلوب = "آخر".*

|  |  |
| --- | --- |
| *مذكرة* | *ما هو $ \_SERVER ["PHP\_SELF"] متغير؟  و$ \_SERVER ["PHP\_SELF"] هو متغير السوبر العالمية التي تقوم بإرجاع اسم ملف البرنامج النصي التنفيذ حاليا.* |

*لذلك، و$ \_SERVER ["PHP\_SELF"] يرسل بيانات النموذج المقدم إلى الصفحة نفسها، بدلا من القفز إلى صفحة مختلفة. بهذه الطريقة، سيكون المستخدم الحصول على رسائل خطأ على نفس الصفحة مثل النموذج.*

|  |  |
| --- | --- |
| *مذكرة* | *ما هو htmlspecialchars (وظيفة)؟  ظيفة وhtmlspecialchars () تحويل أحرف خاصة إلى كيانات HTML. هذا يعني أنه سيحل محل الأحرف HTML مثل <و> مع <و>. هذا يمنع المهاجمين من استغلال رمز HTML أو عن طريق حقن شفرة جافا سكريبت (هجمات البرمجة عبر الموقع) في النماذج.* |

*ملاحظة كبيرة على PHP نموذج أمان*

*يمكن استخدام $ \_SERVER ["PHP\_SELF"] متغير من قبل قراصنة!*

*إذا تم استخدام PHP\_SELF في الصفحة الخاصة بك ثم يمكن للمستخدم إدخال الخط المائل (/) وبعد ذلك بعض الكتابة عبر الموقع (XSS) لتنفيذ أوامر.*

|  |  |
| --- | --- |
| *مذكرة* | *عبر موقع البرمجة النصية (XSS) هو نوع من ثغرة أمنية الكمبيوتر التي توجد عادة في تطبيقات الويب. XSS تمكن المهاجمين لحقن نصي من جانب العميل في صفحات الويب ينظر إليها من قبل مستخدمين آخرين.* |

*تفترض لدينا النموذج التالي في صفحة باسم "test\_form.php":*

*<form method="post" action="<?php echo $\_SERVER["PHP\_SELF"];?>">*

*الآن، إذا يقوم المستخدم بإدخال عنوان URL العادي في شريط العنوان مثل "http://www.example.com/test\_form.php"، سيتم ترجمة التعليمات البرمجية أعلاه إلى:*

*<form method="post" action="test\_form.php">*

*حتى الان جيدة جدا.*

*ومع ذلك، والنظر في أن المستخدم بإدخال URL التالي في شريط العنوان:*

*http://www.example.com/test\_form.php/%22%3E%3Cscript%3Ealert('hacked')%3C/script%3E*

*في هذه الحالة، سيتم ترجمة التعليمات البرمجية أعلاه إلى:*

*<form method="post" action="test\_form.php"/><script>alert('hacked')</script>*

*هذا الرمز يضيف العلامة النصي وأمر حالة تأهب. وعند تحميل الصفحة، وسوف يتم تنفيذ شفرة جافا سكريبت (لن يرى المستخدم مربع تنبيه). هذا هو مجرد مثال بسيط وغير ضارة كيف يمكن استغلال المتغير PHP\_SELF.*

*يكون على بينة من أن أي شفرة جافا سكريبت يمكن إضافتها داخل علامة <SCRIPT>! يمكن للهاكر إعادة توجيه المستخدم إلى ملف على خادم آخر، وهذا الملف يمكن أن تعقد الشيفرات الخبيثة التي يمكن أن تغير المتغيرات العالمية أو إرسال النموذج إلى عنوان آخر لحفظ بيانات المستخدم، على سبيل المثال.*

*كيفية تجنب $ \_SERVER ["PHP\_SELF"] المآثر؟*

*يمكن تجنبها $ \_SERVER ["PHP\_SELF"] مآثر باستخدام htmlspecialchars (وظيفة).*

*يجب أن رمز شكل تبدو مثل هذا:*

*<form method="post" action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>">*

*وظيفة htmlspecialchars () تحويل أحرف خاصة إلى كيانات HTML. الآن إذا كان المستخدم يحاول استغلال متغير PHP\_SELF، فإنه سيؤدي إلى الإخراج التالي:*

*<form method="post" action="test\_form.php/&quot;&gt;&lt;script&gt;alert('hacked')&lt;/script&gt;">*

*واستغلال فشل المحاولة، ويتم ذلك أي ضرر!*

*تحقق من صحة بيانات النموذج مع PHP*

*أول شيء سنفعله هو لتمرير جميع المتغيرات من خلال htmlspecialchars PHP في (وظيفة).*

*عندما نستخدم htmlspecialchars (وظيفة)، ثم إذا حاول المستخدم أن يقدم ما يلي في حقل النص:*

*<SCRIPT> location.href ('http://www.hacked.com') </ النصي>*

*- هذا لن يعدم، لأنه يتم حفظها كما هرب HTML رمز، مثل هذا:*

*<SCRIPT> location.href ('http://www.hacked.com') </ النصي>*

*رمز الآن آمنة ليتم عرضها على الصفحة أو داخل البريد الإلكتروني.*

*وسوف نقوم أيضا قيام اثنين من أكثر الأشياء عندما يقوم المستخدم بإرسال النموذج:*

1. *تجريد الأحرف غير الضرورية (مساحة إضافية، التبويب، السطر الجديد) من البيانات إدخال المستخدم (مع PHP تقليم (وظيفة))*
2. *إزالة الخطوط المائلة العكسية (\) من البيانات إدخال المستخدم (مع stripslashes PHP (وظيفة))*

*الخطوة التالية هي إنشاء وظيفة من شأنها أن تفعل كل فحص بالنسبة لنا (والذي هو أكثر ملاءمة من كتابة نفس رمز مرارا وتكرارا).*

*سوف نقوم بتسمية ظيفة test\_input ().*

*الآن، يمكننا التحقق من كل $ \_POST متغير مع test\_input (وظيفة)، ونظرة النصي مثل هذا:*

*مثال*

*<?php  
// define variables and set to empty values  
$name = $email = $gender = $comment = $website = "";  
  
if ($\_SERVER["REQUEST\_METHOD"] == "POST")  
{  
  $name = test\_input($\_POST["name"]);  
  $email = test\_input($\_POST["email"]);  
  $website = test\_input($\_POST["website"]);  
  $comment = test\_input($\_POST["comment"]);  
  $gender = test\_input($\_POST["gender"]);  
}  
  
function test\_input($data)  
{  
  $data = trim($data);  
  $data = stripslashes($data);  
  $data = htmlspecialchars($data);  
  return $data;  
}  
?>*

*لاحظ أنه في بداية البرنامج النصي، ونحن تحقق ما إذا كان قد تم إرسال النموذج باستخدام $ \_SERVER ["REQUEST\_METHOD"]. إذا كان REQUEST\_METHOD هو وظيفة، ثم تم إرسال النموذج - وينبغي التحقق من صحة ذلك. إذا لم يتم تقديمه، تخطي المصادقة وعرض نموذج فارغ.*

*ومع ذلك، في المثال أعلاه، جميع حقول الإدخال اختيارية. البرنامج النصي يعمل بشكل جيد حتى إذا كان المستخدم لا تدخل أي بيانات.*

*والخطوة التالية هي لجعل حقول الإدخال المطلوبة، وإنشاء رسائل الخطأ إذا لزم الأمر.*

*PHP أشكال - حقول مطلوبة*

*هذا الفصل تظهر كيفية جعل حقول الإدخال المطلوبة، وإنشاء رسائل الخطأ إذا لزم الأمر.*

*PHP - حقول مطلوبة*

*من الجدول قواعد التحقق من الصحة في الصفحة السابقة، ونحن نرى أن "الاسم"، "البريد الإلكتروني"، وحقول "الجنس" مطلوبة. هذه المجالات لا يمكن أن تكون فارغة ويجب أن تملأ في شكل HTML.*

|  |  |
| --- | --- |
| *Field* | *Validation Rules* |
| *Name* | *Required. + Must only contain letters and whitespace* |
| *E-mail* | *Required. + Must contain a valid email address (with @ and .)* |
| *Website* | *Optional. If present, it must contain a valid URL* |
| *Comment* | *Optional. Multi-line input field (textarea)* |
| *Gender* | *Required. Must select one* |

*في الفصل السابق، كانت جميع حقول الإدخال الاختيارية.*

*في التعليمات البرمجية التالية واضاف لدينا بعض متغيرات جديدة: $ nameErr، emailErr $، $ genderErr، ومبلغ websiteErr. وسوف تعقد هذه المتغيرات الخطأ رسائل الخطأ للحقول المطلوبة. واضاف لدينا أيضا إذا بيان آخر لكل $ \_POST متغير. هذا يتحقق إذا كان المتغير $ \_POST فارغة (مع فارغة وظيفة PHP ()). إذا كانت فارغة، يتم تخزين رسالة خطأ في خطأ المتغيرات المختلفة، وإذا لم يكن فارغا، فإنه يرسل البيانات من خلال إدخال المستخدم test\_input (وظيفة):*

*<?php  
// define variables and set to empty values  
$nameErr = $emailErr = $genderErr = $websiteErr = "";  
$name = $email = $gender = $comment = $website = "";  
  
if ($\_SERVER["REQUEST\_METHOD"] == "POST")  
{  
  
  if (empty($\_POST["name"]))  
    {$nameErr = "Name is required";}  
  else  
    {$name = test\_input($\_POST["name"]);}  
  
  if (empty($\_POST["email"]))  
    {$emailErr = "Email is required";}  
  else  
    {$email = test\_input($\_POST["email"]);}  
  
  if (empty($\_POST["website"]))  
    {$website = "";}  
  else  
    {$website = test\_input($\_POST["website"]);}  
  
  if (empty($\_POST["comment"]))  
    {$comment = "";}  
  else  
    {$comment = test\_input($\_POST["comment"]);}  
  
  if (empty($\_POST["gender"]))  
    {$genderErr = "Gender is required";}  
  else  
    {$gender = test\_input($\_POST["gender"]);}  
}  
?>*

*PHP - عرض رسائل الخطأ*

*ثم في شكل HTML، ونضيف إلى السيناريو قليلا بعد كل حقل المطلوب، الذي يقوم بإنشاء رسالة الخطأ الصحيح إذا لزم الأمر (أي إذا حاول المستخدم لإرسال النموذج دون ملء الحقول المطلوبة):*

*مثال*

*<form method="post" action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>">  
  
Name: <input type="text" name="name">  
<span class="error">\* <?php echo $nameErr;?></span>  
<br><br>  
E-mail:  
<input type="text" name="email">  
<span class="error">\* <?php echo $emailErr;?></span>  
<br><br>  
Website:  
<input type="text" name="website">  
<span class="error"><?php echo $websiteErr;?></span>  
<br><br>  
<label>Comment: <textarea name="comment" rows="5" cols="40"></textarea>  
<br><br>  
Gender:  
<input type="radio" name="gender" value="female">Female  
<input type="radio" name="gender" value="male">Male  
<span class="error">\* <?php echo $genderErr;?></span>  
<br><br>  
<input type="submit" name="submit" value="Submit">   
  
</form>*

*والخطوة التالية هي للتحقق من صحة البيانات المدخلة، وهذا هو "هل يحتوي حقل الاسم الحروف وبيضاء فقط؟"، و "هل يحتوي حقل البريد الإلكتروني بناء جملة عنوان بريد إلكتروني صالح؟"، وإذا ملء " لا يحتوي الحقل الموقع URL صالح؟ "*

*PHP أشكال - التحقق من صحة البريد الإلكتروني وURL*

*هذا الفصل تظهر كيفية التحقق من صحة الأسماء، ورسائل البريد الإلكتروني، وعناوين المواقع.*

*PHP - التحقق من صحة اسم*

*رمز أدناه يوضح طريقة بسيطة لمعرفة ما اذا كان يحتوي على حقل الاسم فقط الحروف والمسافات. إذا كانت قيمة حقل الاسم غير صالح، ثم تخزين رسالة خطأ:*

*$name = test\_input($\_POST["name"]);  
if (!preg\_match("/^[a-zA-Z ]\*$/",$name))  
  {  
  $nameErr = "Only letters and white space allowed";   
  }*

|  |  |
| --- | --- |
| *مذكرة* | *وظيفة preg\_match () يبحث سلسلة لنمط، والعودة الحقيقية في حالة وجود نمط وكاذبة خلاف ذلك.* |

*PHP - التحقق من صحة البريد الإلكتروني*

*رمز أدناه يوضح طريقة بسيطة لمعرفة ما اذا كان بناء جملة عنوان البريد الإلكتروني غير صحيح. إذا بناء الجملة عنوان البريد الإلكتروني غير صالح، ثم تخزين رسالة خطأ:*

*$email = test\_input($\_POST["email"]);  
if (!preg\_match("/([\w\-]+\@[\w\-]+\.[\w\-]+)/",$email))  
  {  
  $emailErr = "Invalid email format";   
  }*

*PHP - التحقق من صحة URL*

*رمز أدناه يوضح طريقة لمعرفة ما اذا كان عنوان URL بناء جملة غير صالح (هذا التعبير العادي كما يسمح شرطات في URL). إذا بناء الجملة عنوان URL غير صالح، ثم تخزين رسالة خطأ:*

*$website = test\_input($\_POST["website"]);  
if (!preg\_match("/\b(?:(?:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?=~\_|!:,.;]\*[-a-z0-9+&@#\/%=~\_|]/i",$website))  
  {  
  $websiteErr = "Invalid URL";   
  }*

*PHP - التحقق من صحة الاسم والبريد الإلكتروني، وعنوان URL*

*الآن، والسيناريو يبدو مثل هذا:*

*مثال*

*<?php  
// define variables and set to empty values  
$nameErr = $emailErr = $genderErr = $websiteErr = "";  
$name = $email = $gender = $comment = $website = "";  
  
if ($\_SERVER["REQUEST\_METHOD"] == "POST")  
{  
  if (empty($\_POST["name"]))  
    {$nameErr = "Name is required";}  
  else  
    {  
    $name = test\_input($\_POST["name"]);  
    // check if name only contains letters and whitespace  
    if (!preg\_match("/^[a-zA-Z ]\*$/",$name))  
      {  
      $nameErr = "Only letters and white space allowed";   
      }  
    }  
  
  if (empty($\_POST["email"]))  
    {$emailErr = "Email is required";}  
  else  
    {  
    $email = test\_input($\_POST["email"]);  
    // check if e-mail address syntax is valid  
    if (!preg\_match("/([\w\-]+\@[\w\-]+\.[\w\-]+)/",$email))  
      {  
      $emailErr = "Invalid email format";   
      }  
    }  
  
  if (empty($\_POST["website"]))  
    {$website = "";}  
  else  
    {  
    $website = test\_input($\_POST["website"]);  
    // check if URL address syntax is valid (this regular expression also allows dashes in the URL)  
    if (!preg\_match("/\b(?:(?:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?=~\_|!:,.;]\*[-a-z0-9+&@#\/%=~\_|]/i",$website))  
      {  
      $websiteErr = "Invalid URL";   
      }  
    }  
  
  if (empty($\_POST["comment"]))  
    {$comment = "";}  
  else  
    {$comment = test\_input($\_POST["comment"]);}  
  
  if (empty($\_POST["gender"]))  
    {$genderErr = "Gender is required";}  
  else  
    {$gender = test\_input($\_POST["gender"]);}  
}  
?>*

*والخطوة التالية هي لإظهار كيفية منع النموذج من إفراغ جميع حقول الإدخال عندما يقوم المستخدم بإرسال النموذج.*

*PHP أشكال - التحقق من صحة البريد الإلكتروني وURL*

*هذا الفصل تظهر كيفية التحقق من صحة الأسماء، ورسائل البريد الإلكتروني، وعناوين المواقع.*

*PHP - التحقق من صحة اسم*

*رمز أدناه يوضح طريقة بسيطة لمعرفة ما اذا كان يحتوي على حقل الاسم فقط الحروف والمسافات. إذا كانت قيمة حقل الاسم غير صالح، ثم تخزين رسالة خطأ:*

*$name = test\_input($\_POST["name"]);  
if (!preg\_match("/^[a-zA-Z ]\*$/",$name))  
  {  
  $nameErr = "Only letters and white space allowed";   
  }*

|  |  |
| --- | --- |
| *مذكرة* | *وظيفة preg\_match () يبحث سلسلة لنمط، والعودة الحقيقية في حالة وجود نمط وكاذبة خلاف ذلك.* |

*PHP - التحقق من صحة البريد الإلكتروني*

*رمز أدناه يوضح طريقة بسيطة لمعرفة ما اذا كان بناء جملة عنوان البريد الإلكتروني غير صحيح. إذا بناء الجملة عنوان البريد الإلكتروني غير صالح، ثم تخزين رسالة خطأ:*

*$email = test\_input($\_POST["email"]);  
if (!preg\_match("/([\w\-]+\@[\w\-]+\.[\w\-]+)/",$email))  
  {  
  $emailErr = "Invalid email format";   
  }*

*PHP - التحقق من صحة URL*

*رمز أدناه يوضح طريقة لمعرفة ما اذا كان عنوان URL بناء جملة غير صالح (هذا التعبير العادي كما يسمح شرطات في URL). إذا بناء الجملة عنوان URL غير صالح، ثم تخزين رسالة خطأ:*

*$website = test\_input($\_POST["website"]);  
if (!preg\_match("/\b(?:(?:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?=~\_|!:,.;]\*[-a-z0-9+&@#\/%=~\_|]/i",$website))  
  {  
  $websiteErr = "Invalid URL";   
  }*

*PHP - التحقق من صحة الاسم والبريد الإلكتروني، وعنوان URL*

*الآن، والسيناريو يبدو مثل هذا:*

*مثال*

*<?php  
// define variables and set to empty values  
$nameErr = $emailErr = $genderErr = $websiteErr = "";  
$name = $email = $gender = $comment = $website = "";  
  
if ($\_SERVER["REQUEST\_METHOD"] == "POST")  
{  
  if (empty($\_POST["name"]))  
    {$nameErr = "Name is required";}  
  else  
    {  
    $name = test\_input($\_POST["name"]);  
    // check if name only contains letters and whitespace  
    if (!preg\_match("/^[a-zA-Z ]\*$/",$name))  
      {  
      $nameErr = "Only letters and white space allowed";   
      }  
    }  
  
  if (empty($\_POST["email"]))  
    {$emailErr = "Email is required";}  
  else  
    {  
    $email = test\_input($\_POST["email"]);  
    // check if e-mail address syntax is valid  
    if (!preg\_match("/([\w\-]+\@[\w\-]+\.[\w\-]+)/",$email))  
      {  
      $emailErr = "Invalid email format";   
      }  
    }  
  
  if (empty($\_POST["website"]))  
    {$website = "";}  
  else  
    {  
    $website = test\_input($\_POST["website"]);  
    // check if URL address syntax is valid (this regular expression also allows dashes in the URL)  
    if (!preg\_match("/\b(?:(?:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?=~\_|!:,.;]\*[-a-z0-9+&@#\/%=~\_|]/i",$website))  
      {  
      $websiteErr = "Invalid URL";   
      }  
    }  
  
  if (empty($\_POST["comment"]))  
    {$comment = "";}  
  else  
    {$comment = test\_input($\_POST["comment"]);}  
  
  if (empty($\_POST["gender"]))  
    {$genderErr = "Gender is required";}  
  else  
    {$gender = test\_input($\_POST["gender"]);}  
}  
?>*

*والخطوة التالية هي لإظهار كيفية منع النموذج من إفراغ جميع حقول الإدخال عندما يقوم المستخدم بإرسال النموذج.*

# *PHP متعددة الأبعاد صفائف*

*يمكن أن تحتوي أيضا على مجموعة مجموعة أخرى كقيمة، والذي بدوره يمكن أن تعقد المصفوفات الأخرى كذلك. في مثل هذه الطريقة التي يمكن إنشاء صفائف يومين أو ثلاثة الأبعاد:*

## *مثال*

*<?php  
// A two-dimensional array:  
$cars = array  
  (  
  array("Volvo",100,96),  
  array("BMW",60,59),  
  array("Toyota",110,100)  
  );  
?>*

## *PHP - متعددة الأبعاد صفائف*

*وهناك مجموعة متعددة الأبعاد هي مجموعة تحتوي على واحد أو أكثر من المصفوفات.*

*في مجموعة متعددة الأبعاد، كل عنصر في صفيف الرئيسية يمكن أيضا أن يكون صفيف. وكل عنصر في مجموعة فرعية يمكن أن يكون صفيف، وهلم جرا.*

### *مثال*

*في هذا المثال نقوم بإنشاء مجموعة متعددة الأبعاد، مع مفاتيح معرف تعيينه تلقائيا:*

*$families = array  
  (  
  "Griffin"=>array  
  (  
  "Peter",  
  "Lois",  
  "Megan"  
  ),  
  "Quagmire"=>array  
  (  
  "Glenn"  
  ),  
  "Brown"=>array  
  (  
  "Cleveland",  
  "Loretta",  
  "Junior"  
  )  
  );*

*ان مجموعة أعلاه تبدو هذه إذا خطية إلى الإخراج:*

*Array  
(  
[Griffin] => Array  
  (  
  [0] => Peter  
  [1] => Lois  
  [2] => Megan  
  )  
[Quagmire] => Array  
  (  
  [0] => Glenn  
  )  
[Brown] => Array  
  (  
  [0] => Cleveland  
  [1] => Loretta  
  [2] => Junior  
  )  
)*

### *مثال 2*

*دعونا نحاول عرض قيمة واحدة من مجموعة أعلاه:*

*echo "Is " . $families['Griffin'][2] .   
" a part of the Griffin family?";*

*رمز أعلاه إرادة الإخراج:*

*Is Megan a part of the Griffin family?*

# *PHP التسجيل () وظيفة*

*يتم استخدام الدالة تاريخ PHP () لتهيئة الوقت و / أو التاريخ.*

## *وPHP التسجيل () وظيفة*

*وظيفة تاريخ PHP () تنسيقات طابع زمني إلى تاريخ أكثر قابلية للقراءة والوقت.*

|  |  |
| --- | --- |
| *مذكرة* | *A الطابع الزمني هو سلسلة من الأحرف، تدل على التاريخ و / أو الوقت الذي حدث حدث معين.* |

### *بناء الجملة*

*date(format,timestamp)*

|  |  |
| --- | --- |
| *Parameter* | *Description* |
| *format* | *Required. Specifies the format of the timestamp* |
| *timestamp* | *Optional. Specifies a timestamp. Default is the current date and time* |

## *PHP التسجيل () - تنسيق التسجيل*

*المطلوب تنسيق المعلمة في وظيفة التاريخ () يحدد كيفية تنسيق التاريخ / الوقت.*

*وهنا بعض الشخصيات التي يمكن استخدامها:*

* *د - يمثل يوم من الشهر (01-31)*
* *م - يمثل الشهر (01-12)*
* *Y - يمثل السنة (في أربعة أرقام)*

*قائمة بجميع الأحرف التي يمكن استخدامها في تنسيق المعلمة، يمكن العثور عليها في PHP لدينا تاريخ المرجعية،*[*وظيفة التاريخ ()*](http://www.w3schools.com/php/func_date_date.asp)*.*

*"." الشخصيات الأخرى، مثل "/"، أو "-" يمكن أيضا أن يدرج بين الحروف لإضافة تنسيق إضافية:*

*<?php  
echo date("Y/m/d") . "<br>";  
echo date("Y.m.d") . "<br>";  
echo date("Y-m-d");  
?>*

*الناتج من التعليمات البرمجية أعلاه يمكن أن يكون شيئا مثل هذا:*

*2009/05/11  
2009.05.11  
2009-05-11*

## *PHP التسجيل () - إضافة الطابع الزمني*

*اختياري الطابع الزمني معلمة في وظيفة التاريخ () يحدد طابع زمني. إذا كنت لم تحدد الطابع الزمني، سيتم استخدام التاريخ والوقت الحالي.*

*وظيفة mktime () بإرجاع الطابع الزمني يونكس للتاريخ.*

*يحتوي على الطابع الزمني يونكس عدد الثواني بين عصر يونكس (1 يناير 1970 00:00:00 GMT) والوقت المحدد.*

### *بناء الجملة من أجل mktime ()*

*mktime(hour,minute,second,month,day,year,is\_dst)*

*للذهاب يوم واحد في المستقبل ونحن ببساطة إضافة واحد إلى حجة يوم mktime ():*

*<?php  
$tomorrow = mktime(0,0,0,date("m"),date("d")+1,date("Y"));  
echo "Tomorrow is ".date("Y/m/d", $tomorrow);  
?>*

*الناتج من التعليمات البرمجية أعلاه يمكن أن يكون شيئا مثل هذا:*

*Tomorrow is 2009/05/12*

## *كاملة PHP التسجيل المرجعي*

*للإشارة كاملة من جميع وظائف حتى الآن، الذهاب الى موقعنا*[*PHP التسجيل الكامل المرجعي*](http://www.w3schools.com/php/php_ref_date.asp)*.*

*يحتوي على المرجع وصفا موجزا، وأمثلة على الاستخدام، لكل وظيفة!*

# *PHP تضمين ملفات*

## *PHP تشمل وتتطلب بيانات*

*في PHP، يمكنك إدراج محتويات ملف PHP في ملف واحد PHP آخر قبل خادم ينفذ ذلك.*

*وتشمل وتتطلب وتستخدم البيانات لإدراج رموز مفيدة مكتوبة في ملفات أخرى، في تدفق التنفيذ.*

*وتشمل وتتطلب متطابقة، إلا بناء على الفشل:*

* *تتطلب سينتج خطأ فادح (E\_COMPILE\_ERROR) ووقف النصي*
* *وتشمل سوف تنتج فقط تحذير (E\_WARNING) وسيستمر البرنامج النصي*

*لذلك، إذا كنت تريد تنفيذ على المضي قدما وعرض المستخدمين الإخراج، حتى لو كانت تشمل الملف مفقود، واستخدام وتشمل. خلاف ذلك، في حالة الإطار، CMS أو المعقدة لتطبيق الترميز PHP، استخدم دائما تتطلب لتشمل ملف مفتاح لتدفق التنفيذ. وهذا يساعد على تجنب تعريض الأمن للخطر التطبيق الخاص بك والنزاهة، فقط في ملف القضية مفتاح واحد مفقود بطريق الخطأ.*

*بما في ذلك ملفات يوفر الكثير من العمل. هذا يعني أنه يمكنك إنشاء ملف رأس أو تذييل أو القائمة الموحدة لجميع صفحات موقعك. ثم، عندما يحتاج رأس إلى تحديث، يمكنك تحديث فقط وتشمل رأس الملف.*

## *بناء الجملة*

*include 'filename';  
  
or  
  
require 'filename';*

## *PHP تشمل وتتطلب بيان*

### *مثال الأساسية*

*نفترض أن لديك ملف الرأس القياسية، ودعا "header.php على". لتضمين ملف الرأس في الصفحة، واستخدام تشمل / تتطلب ما يلي:*

*<html>  
<body>  
  
<?php include 'header.php'; ?>  
<h1>Welcome to my home page!</h1>  
<p>Some text.</p>  
  
</body>  
</html>*

### *مثال 2*

*نفترض لدينا ملف القائمة الموحدة التي ينبغي استخدامها على كافة الصفحات.*

*"menu.php":*

*echo '<a href="/default.php">Home</a>  
<a href="/tutorials.php">Tutorials</a>  
<a href="/references.php">References</a>  
<a href="/examples.php">Examples</a>   
<a href="/about.php">About Us</a>   
<a href="/contact.php">Contact Us</a>';*

*وينبغي أن تشمل جميع الصفحات في موقع ويب هذا الملف القائمة. هنا هو كيف يمكن القيام بذلك:*

*<html>  
<body>  
  
<div class="leftmenu">  
<?php include 'menu.php'; ?>  
</div>  
  
<h1>Welcome to my home page.</h1>  
<p>Some text.</p>  
  
</body>  
</html>*

### *المثال 3*

*نفترض لدينا ملف التضمين مع بعض المتغيرات المعرفة ("vars.php"):*

*<?php  
$color='red';  
$car='BMW';  
?>*

*ثم متغيرات يمكن استخدامها في ملف الدعوة:*

*<html>  
<body>  
  
<h1>Welcome to my home page.</h1>  
<?php include 'vars.php';  
echo "I have a $color $car"; // I have a red BMW  
?>  
  
</body>  
</html>*

# *PHP معالجة الملف*

*يتم استخدام الدالة الدالة fopen () لفتح الملفات في PHP.*

## *فتح الملف*

*يتم استخدام الدالة الدالة fopen () لفتح الملفات في PHP.*

*المعلمة الأولى من هذه الدالة يحتوي على اسم الملف المراد فتحه وتعين المعلمة الثانية في الوضع الذي يجب فتح الملف:*

*<html>  
<body>  
  
<?php  
$file=fopen("welcome.txt","r");  
?>  
  
</body>  
</html>*

*يمكن فتح الملف في واحدة من الأوضاع التالية:*

|  |  |
| --- | --- |
| *Modes* | *Description* |
| *r* | *Read only. Starts at the beginning of the file* |
| *r+* | *Read/Write. Starts at the beginning of the file* |
| *w* | *Write only. Opens and clears the contents of file; or creates a new file if it doesn't exist* |
| *w+* | *Read/Write. Opens and clears the contents of file; or creates a new file if it doesn't exist* |
| *a* | *Append. Opens and writes to the end of the file or creates a new file if it doesn't exist* |
| *a+* | *Read/Append. Preserves file content by writing to the end of the file* |
| *x* | *Write only. Creates a new file. Returns FALSE and an error if file already exists* |
| *x+* | *Read/Write. Creates a new file. Returns FALSE and an error if file already exists* |

*ملاحظة: إذا كانت وظيفة الدالة fopen () غير قادر على فتح الملف المحدد، تقوم بإرجاع 0 (كاذبة).*

### *مثال*

*المثال التالي بإنشاء رسالة إذا كان الدالة fopen (وظيفة) غير قادر على فتح الملف المحدد:*

*<html>  
<body>  
  
<?php  
$file=fopen("welcome.txt","r") or exit("Unable to open file!");  
?>  
  
</body>  
</html>*

## *إغلاق الملف*

*يتم استخدام الدالة fclose () لإغلاق ملف مفتوح:*

*<?php  
$file = fopen("test.txt","r");  
  
//some code to be executed  
  
fclose($file);  
?>*

## *تحقق النهائي من الملف*

*الشيكات وظيفة feof () إذا تم الوصول إلى "نهاية الملف" (EOF).*

*وظيفة feof () مفيد لحلقات عبر البيانات من طول مجهولة.*

*ملاحظة: لا يمكنك قراءة من الملفات التي تم فتحها في ث، لذلك، ووضع خ!*

*if (feof($file)) echo "End of file";*

## *قراءة الخط الملف بواسطة الخط*

*يتم استخدام الدالة fgets () لقراءة سطر واحد من ملف.*

*ملاحظة: بعد مكالمة لهذه الوظيفة قد نقل مؤشر الملف إلى السطر التالي.*

### *مثال*

*المثال التالي يقرأ ملف خط سطرا، حتى يتم الوصول إلى نهاية الملف:*

*<?php  
$file = fopen("welcome.txt", "r") or exit("Unable to open file!");  
//Output a line of the file until the end is reached  
while(!feof($file))  
  {  
  echo fgets($file). "<br>";  
  }  
fclose($file);  
?>*

## *قراءة الأحرف الملف بواسطة حرف*

*يتم استخدام الدالة fgetc () لقراءة حرف واحد من ملف.*

*ملاحظة: بعد مكالمة لهذه الوظيفة يتحرك مؤشر الملف إلى الحرف التالي.*

### *مثال*

*يقرأ المثال التالي حرف الملف بواسطة حرف، حتى يتم الوصول إلى نهاية الملف:*

*<?php  
$file=fopen("welcome.txt","r") or exit("Unable to open file!");  
while (!feof($file))  
  {  
  echo fgetc($file);  
  }  
fclose($file);  
?>*

## *PHP نظام الملفات المرجعي*

*للإشارة كاملة من وظائف PHP الملفات، زيارة موقعنا على*[*نظام الملفات المرجعي PHP*](http://www.w3schools.com/php/php_ref_filesystem.asp)*.*

*PHP تحميل الملف*

*مع PHP، فمن الممكن لتحميل الملفات إلى الخادم.*

*إنشاء نموذج تحميل الملف-*

*للسماح للمستخدمين بتحميل الملفات من نموذج يمكن أن تكون مفيدة جدا.*

*ننظر في شكل HTML التالي لتحميل الملفات:*

*<html>  
<body>  
  
<form action="upload\_file.php" method="post"  
enctype="multipart/form-data">  
<label for="file">Filename:</label>  
<input type="file" name="file" id="file"><br>  
<input type="submit" name="submit" value="Submit">  
</form>  
  
</body>  
</html>*

*لاحظ التالية حول شكل HTML أعلاه:*

* *السمة enctype من العلامة <form> يحدد نوع المحتوى الذي لاستخدامها عند تقديم النموذج. يستخدم "متعددة الأجزاء / النموذج البيانات" عندما يتطلب شكلا البيانات الثنائية، مثل محتويات الملف، وسيتم تحميلها*
* *نوع = "ملف" سمة من العلامة <input> يحدد أن الإدخال يجب أن تتم معالجتها كملف. على سبيل المثال، عند عرضها في المتصفح، وسوف يكون هناك زر استعراض بجانب حقل الإدخال*

*ملاحظة: السماح للمستخدمين بتحميل ملفات مخاطرة أمنية كبيرة. يسمح فقط للمستخدمين الموثوق بها لأداء تحميل الملف.*

*إنشاء وتحميل سكربت*

*يحتوي على "upload\_file.php" ملف رمز لتحميل ملف:*

*<?php  
if ($\_FILES["file"]["error"] > 0)  
  {  
  echo "Error: " . $\_FILES["file"]["error"] . "<br>";  
  }  
else  
  {  
  echo "Upload: " . $\_FILES["file"]["name"] . "<br>";  
  echo "Type: " . $\_FILES["file"]["type"] . "<br>";  
  echo "Size: " . ($\_FILES["file"]["size"] / 1024) . " kB<br>";  
  echo "Stored in: " . $\_FILES["file"]["tmp\_name"];  
  }  
?>*

*باستخدام مجموعة بي إتش بي $ \_FILES العالمية يمكنك تحميل الملفات من كمبيوتر العميل إلى الملقم البعيد.*

*المعلمة الأولى هو نموذج المدخلات واسم الفهرس الثاني إما أن يكون "اسم"، "اكتب"، "حجم"، "tmp\_name" أو "خطأ". مثل هذا:*

* *$ \_FILES ["ملف"] ["الاسم"] - اسم الملف الذي تم تحميله*
* *$ \_FILES ["ملف"] ["نوع"] - نوع الملف الذي تم تحميله*
* *$ \_FILES ["ملف"] ["حجم"] - حجم بالبايت من الملف الذي تم تحميله*
* *$ \_FILES ["ملف"] ["tmp\_name"] - اسم نسخة مؤقتة من الملف المخزنة على الخادم*
* *$ \_FILES ["ملف"] ["خطأ"] - رمز الخطأ الناتج عن تحميل الملف*

*هذا هو وسيلة بسيطة للغاية لتحميل الملفات. لأسباب أمنية، يجب أن تضيف قيود على ما يسمح للمستخدم تحميل.*

*القيود المفروضة على تحميل*

*في هذا السيناريو نضيف بعض القيود إلى تحميل الملف. يمكن للمستخدم تحميل GIF، JPEG، وبابوا نيو غينيا الملفات، ويجب أن يكون حجم الملف أقل من 20 كيلو بايت.:*

*<?php  
$allowedExts = array("gif", "jpeg", "jpg", "png");  
$temp = explode(".", $\_FILES["file"]["name"]);  
$extension = end($temp);  
if ((($\_FILES["file"]["type"] == "image/gif")  
|| ($\_FILES["file"]["type"] == "image/jpeg")  
|| ($\_FILES["file"]["type"] == "image/jpg")  
|| ($\_FILES["file"]["type"] == "image/pjpeg")  
|| ($\_FILES["file"]["type"] == "image/x-png")  
|| ($\_FILES["file"]["type"] == "image/png"))  
&& ($\_FILES["file"]["size"] < 20000)  
&& in\_array($extension, $allowedExts))  
  {  
  if ($\_FILES["file"]["error"] > 0)  
    {  
    echo "Error: " . $\_FILES["file"]["error"] . "<br>";  
    }  
  else  
    {  
    echo "Upload: " . $\_FILES["file"]["name"] . "<br>";  
    echo "Type: " . $\_FILES["file"]["type"] . "<br>";  
    echo "Size: " . ($\_FILES["file"]["size"] / 1024) . " kB<br>";  
    echo "Stored in: " . $\_FILES["file"]["tmp\_name"];  
    }  
  }  
else  
  {  
  echo "Invalid file";  
  }  
?>*

*حفظ الملف الذي تم تحميله*

*الأمثلة أعلاه إنشاء نسخة مؤقتة من الملفات التي تم تحميلها في مجلد temp PHP على الخادم.*

*الملفات المنسوخة مؤقت يختفي عندما ينتهي البرنامج النصي. لتخزين الملف الذي تم تحميله نحن بحاجة إلى نسخه إلى موقع مختلف:*

*<?php  
$allowedExts = array("gif", "jpeg", "jpg", "png");  
$temp = explode(".", $\_FILES["file"]["name"]);  
$extension = end($temp);  
if ((($\_FILES["file"]["type"] == "image/gif")  
|| ($\_FILES["file"]["type"] == "image/jpeg")  
|| ($\_FILES["file"]["type"] == "image/jpg")  
|| ($\_FILES["file"]["type"] == "image/pjpeg")  
|| ($\_FILES["file"]["type"] == "image/x-png")  
|| ($\_FILES["file"]["type"] == "image/png"))  
&& ($\_FILES["file"]["size"] < 20000)  
&& in\_array($extension, $allowedExts))  
  {  
  if ($\_FILES["file"]["error"] > 0)  
    {  
    echo "Return Code: " . $\_FILES["file"]["error"] . "<br>";  
    }  
  else  
    {  
    echo "Upload: " . $\_FILES["file"]["name"] . "<br>";  
    echo "Type: " . $\_FILES["file"]["type"] . "<br>";  
    echo "Size: " . ($\_FILES["file"]["size"] / 1024) . " kB<br>";  
    echo "Temp file: " . $\_FILES["file"]["tmp\_name"] . "<br>";  
  
    if (file\_exists("upload/" . $\_FILES["file"]["name"]))  
      {  
      echo $\_FILES["file"]["name"] . " already exists. ";  
      }  
    else  
      {  
      move\_uploaded\_file($\_FILES["file"]["tmp\_name"],  
      "upload/" . $\_FILES["file"]["name"]);  
      echo "Stored in: " . "upload/" . $\_FILES["file"]["name"];  
      }  
    }  
  }  
else  
  {  
  echo "Invalid file";  
  }  
?>*

*السيناريو أعلاه الشيكات إذا كان الملف موجودا بالفعل، إذا لم يحدث ذلك، فإنه ينسخ الملف إلى مجلد يسمى "تحميل".*

# *PHP الكوكيز*

*وغالبا ما تستخدم الكوكيز لتحديد مستخدم.*

## *ما هو كوكي؟*

*وغالبا ما تستخدم الكوكيز لتحديد مستخدم. ملف تعريف الارتباط هو ملف صغير أن يضمن الخادم على جهاز الكمبيوتر الخاص بالمستخدم. في كل مرة تطلب نفس الكمبيوتر صفحة مع متصفح، فإنه سيتم إرسال ملف تعريف الارتباط أيضا. مع PHP، يمكنك على حد سواء إنشاء واسترداد القيم الكعكة.*

## *كيفية إنشاء ملف تعريف الارتباط؟*

*يتم استخدام الدالة setcookie () لتعيين ملف تعريف الارتباط.*

*ملاحظة: يجب أن تظهر وظيفة setcookie () قبل علامة <HTML>.*

### *بناء الجملة*

*setcookie(name, value, expire, path, domain);*

### *المثال 1*

*في المثال أدناه، سوف نقوم بإنشاء ملف تعريف الارتباط المسمى "المستخدم" وتعيين قيمة "أليكس بورتر" لذلك. ونحن أيضا تحديد أن الكعكة يجب أن تنتهي بعد ساعة واحدة:*

*<?php  
setcookie("user", "Alex Porter", time()+3600);  
?>  
  
<html>  
.....*

*ملاحظة: يتم URLencoded قيمة الكعكة تلقائيا عند إرسال ملف تعريف الارتباط، وتلقائيا فك الشفرة عندما تلقى (لمنع URLencoding، واستخدام setrawcookie () بدلا من ذلك).*

### *مثال 2*

*يمكنك أيضا تعيين وقت انتهاء الصلاحية من الكعكة بطريقة أخرى. قد يكون أسهل من استخدام ثواني.*

*<?php  
$expire=time()+60\*60\*24\*30;  
setcookie("user", "Alex Porter", $expire);  
?>  
  
<html>  
.....*

*يتم تعيين في المثال أعلاه وقت انتهاء الصلاحية لمدة شهر ( 60 ثانية \* 60 دقيقة \* 24 ساعة \* 30 يوما ).*

## *كيفية استرداد قيمة كوكي؟*

*يتم استخدام متغير PHP $ \_COOKIE لاسترداد قيمة الكعكة. في المثال أدناه، فإننا استرداد قيمة الكعكة المسماة "المستخدم" وعرضه على الصفحة:*

*<?php  
// Print a cookie  
echo $\_COOKIE["user"];  
  
// A way to view all cookies  
print\_r($\_COOKIE);  
?>*

*في المثال التالي نستخدم isset (وظيفة) لمعرفة إذا كان قد تم تعيين ملف تعريف ارتباط:*

*<html>  
<body>  
  
<?php  
if (isset($\_COOKIE["user"]))  
  echo "Welcome " . $\_COOKIE["user"] . "!<br>";  
else  
  echo "Welcome guest!<br>";  
?>  
  
</body>  
</html>*

## *كيفية حذف ملفات تعريف الارتباط؟*

*عند حذف ملفات تعريف الارتباط يجب أن أؤكد أن تاريخ انتهاء الصلاحية هو في الماضي.*

*حذف سبيل المثال:*

*<?php  
// set the expiration date to one hour ago  
setcookie("user", "", time()-3600);  
?>*

## *ما إذا كان المتصفح لا يدعم الكوكيز؟*

*إذا يتعامل التطبيق الخاص بك مع المتصفحات التي لا تدعم ملفات تعريف الارتباط، وسوف تضطر إلى استخدام أساليب أخرى لتمرير المعلومات من صفحة واحدة إلى أخرى في التطبيق الخاص بك. أسلوب واحد هو لتمرير البيانات من خلال أشكال (موصوفة أشكال ومدخلات المستخدم في هذا البرنامج التعليمي في وقت سابق).*

*يمر النموذج أدناه لإدخال المستخدم إلى "welcome.php" عندما ينقر المستخدم على زر "إرسال":*

*<html>  
<body>  
  
<form action="welcome.php" method="post">  
Name: <input type="text" name="name">  
Age: <input type="text" name="age">  
<input type="submit">  
</form>  
  
</body>  
</html>*

*استرداد القيم في "welcome.php" ملف مثل هذا:*

*<html>  
<body>  
  
Welcome <?php echo $\_POST["name"]; ?>.<br>  
You are <?php echo $\_POST["age"]; ?> years old.  
  
</body>  
</html>*

*PHP الدورات*

*ويستخدم PHP متغير جلسة لتخزين المعلومات عن، أو تغيير الإعدادات لجلسة عمل المستخدم. متغيرات جلسة تعقد المعلومات حول مستخدم واحد واحد، ومتاحة لجميع الصفحات في تطبيق واحد.*

*متغيرات جلسة PHP*

*عندما كنت تعمل مع تطبيق، يمكنك فتحه، والقيام ببعض التغييرات ثم تقوم بإغلاقه. هذا هو مثل الكثير من الدورة. يعرف جهاز الكمبيوتر من أنت. كان يعرف عند بدء تشغيل التطبيق وعند النهاية. ولكن على شبكة الانترنت أن هناك مشكلة واحدة: لا يعرف خادم الويب من أنت وماذا تفعل لأن عنوان HTTP لا يحتفظ الدولة.*

*دورة PHP يحل هذه المشكلة عن طريق السماح لك لتخزين معلومات المستخدم على ملقم لاستخدامها لاحقا (أي المستخدم، سلع التسوق، الخ). ومع ذلك، معلومات جلسة العمل هو مؤقت وسيتم حذف بعد أن ترك المستخدم الموقع. إذا كنت في حاجة الى تخزين دائم قد ترغب في تخزين البيانات في قاعدة بيانات.*

*جلسات العمل عن طريق إنشاء معرف فريد (UID) لكل زائر وتخزين المتغيرات على أساس هذا UID. إما يتم تخزين UID في الكوكيز أو يتم نشر في URL.*

*بدء الدورة PHP*

*قبل أن تتمكن من تخزين معلومات المستخدم في الدورة PHP الخاصة بك، يجب أن تبدأ أولا في الدورة.*

*ملاحظة: يجب أن تظهر وظيفة session\_start () قبل علامة <HTML>:*

*<?php session\_start(); ?>  
  
<html>  
<body>  
  
</body>  
</html>*

*سوف رمز أعلاه تسجيل جلسة عمل المستخدم مع الخادم، يسمح لك البدء في توفير المعلومات للمستخدم، وتعيين UID للجلسة الخاصة بهذا المستخدم.*

*تخزين متغير الدورة*

*الطريقة الصحيحة لتخزين واسترداد متغيرات جلسة العمل هو استخدام المتغير $ \_SESSION PHP:*

*<?php  
session\_start();  
// store session data  
$\_SESSION['views']=1;  
?>  
  
<html>  
<body>  
  
<?php  
//retrieve session data  
echo "Pageviews=". $\_SESSION['views'];  
?>  
  
</body>  
</html>*

*الإخراج:*

*Pageviews=1*

*في المثال أدناه، ونحن إنشاء بسيط صفحة وجهات النظر المضادة. الشيكات الدالة isset () إذا كان قد تم بالفعل تعيين "وجهات النظر" متغير. إذا تم تعيين "وجهات النظر"، يمكننا زيادة العداد لدينا. إذا "وجهات النظر" لا وجود لها، ونحن خلق "وجهات النظر" متغير، وتعيينها إلى 1:*

*<?php  
session\_start();  
  
if(isset($\_SESSION['views']))  
$\_SESSION['views']=$\_SESSION['views']+1;  
else  
$\_SESSION['views']=1;  
echo "Views=". $\_SESSION['views'];  
?>*

*تدمير الدورة*

*إذا كنت ترغب في حذف بعض البيانات الدورة، يمكنك استخدام ضبطه () أو session\_destroy (وظيفة).*

*يتم استخدام الدالة ضبطه () لتحرير متغير جلسة محدد:*

*<?php  
session\_start();  
if(isset($\_SESSION['views']))  
  unset($\_SESSION['views']);  
?>*

*يمكنك أيضا تدمر تماما الدورة عن طريق استدعاء session\_destroy (وظيفة):*

*<?php  
session\_destroy();  
?>*

*ملاحظة: session\_destroy ()فقد جميع البيانات المخزنة الدورة الخاصة ب سيتم إعادة تعيين جلسة العمل الخاصة بك وسوف تفقد جميع ا*

*PHP إرسال رسائل البريد الإلكتروني*

*PHP تسمح لك لإرسال رسائل البريد الإلكتروني مباشرة من البرنامج النصي.*

*البريد PHP () وظيفة*

*يتم استخدام وظيفة البريد PHP () لإرسال رسائل البريد الإلكتروني من داخل النصي.*

*بناء الجملة*

*mail(to,subject,message,headers,parameters)*

|  |  |
| --- | --- |
| *Parameter* | *Description* |
| *to* | *Required. Specifies the receiver/receivers of the email* |
| *subject* | *Required. Specifies the subject of the email. Note: This parameter cannot contain any newline characters* |
| *message* | *Required. Defines the message to be sent. Each line should be separated with a LF (\n). Lines should not exceed 70 characters* |
| *headers* | *Optional. Specifies additional headers, like From, Cc, and Bcc. The additional headers should be separated with a CRLF (\r\n)* |
| *parameters* | *Optional. Specifies an additional parameter to the sendmail program* |

*ملاحظة: للحصول على وظائف البريد لتكون متوفرة، يتطلب PHP نظام تثبيت الكتروني والعمل. يتم تعريف البرنامج ليتم استخدامها من قبل إعدادات التكوين في ملف php.ini. قراءة المزيد من خلال موقعنا*[*المرجعي PHP البريد*](http://www.w3schools.com/php/php_ref_mail.asp)*.*

*PHP بسيطة البريد الإلكتروني*

*أبسط طريقة لإرسال رسالة بريد إلكتروني مع PHP هو ارسال رسالة نصية.*

*في المثال أدناه علينا أولا بتعريف المتغيرات (دولار ل، $ الموضوع، رسالة دولار، دولار من، $ رؤوس)، ثم نستخدم المتغيرات في البريد (وظيفة) لإرسال البريد الإلكتروني:*

*<?php  
$to = "someone@example.com";  
$subject = "Test mail";  
$message = "Hello! This is a simple email message.";  
$from = "someonelse@example.com";  
$headers = "From:" . $from;  
mail($to,$subject,$message,$headers);  
echo "Mail Sent.";  
?>*

*نموذج البريد PHP*

*مع PHP، يمكنك إنشاء نموذج الملاحظات-على موقع الويب الخاص بك. المثال أدناه يرسل رسالة نصية إلى عنوان البريد الإلكتروني المحدد:*

*<html>  
<body>  
  
<?php  
if (isset($\_REQUEST['email']))  
//if "email" is filled out, send email  
  {  
  //send email  
  $email = $\_REQUEST['email'] ;  
  $subject = $\_REQUEST['subject'] ;  
  $message = $\_REQUEST['message'] ;  
  mail("someone@example.com", $subject,  
  $message, "From:" . $email);  
  echo "Thank you for using our mail form";  
  }  
else  
//if "email" is not filled out, display the form  
  {  
  echo "<form method='post' action='mailform.php'>  
  Email: <input name='email' type='text'><br>  
  Subject: <input name='subject' type='text'><br>  
  Message:<br>  
  <textarea name='message' rows='15' cols='40'>  
  </textarea><br>  
  <input type='submit'>  
  </form>";  
  }  
?>  
  
</body>  
</html>*

*هذه هي الطريقة التي يعمل المثال أعلاه:*

* *أولا، تحقق إذا يتم تعبئة حقل إدخال البريد الإلكتروني من*
* *إذا لم يتم تعيينها (مثل عندما زار الصفحة الأولى)؛ إخراج نموذج HTML*
* *إذا تم تعيين (بعد تعبئة النموذج)؛ إرسال البريد الإلكتروني من النموذج*
* *عند الضغط يقدم بعد تعبئة النموذج، وإعادة تحميل الصفحة، يرى أن يتم تعيين إدخال البريد الإلكتروني، ويرسل البريد الإلكتروني*

*ملاحظة: هذا هو أبسط وسيلة لإرسال البريد الإلكتروني، لكنها ليست آمنة. في الفصل التالي من هذا البرنامج التعليمي يمكنك قراءة المزيد عن نقاط الضعف في البرامج النصية البريد الإلكتروني، وكيفية التحقق من صحة إدخال المستخدم لجعله أكثر أمانا.*

*PHP المرجعي البريد*

*لمزيد من المعلومات حول دالة PHP الإلكتروني ()، زيارة موقعنا*[*المرجعي البريد PHP*](http://www.w3schools.com/php/php_ref_mail.asp)*.*

*PHP معالجة خطأ*

[*«السابق*](http://www.w3schools.com/php/php_secure_mail.asp)

[*الفصل التالي »*](http://www.w3schools.com/php/php_exception.asp)

*الخطأ الافتراضي في التعامل مع PHP بسيط جدا. يتم إرسال رسالة خطأ مع اسم الملف، رقم السطر ورسالة تصف الخطأ إلى المستعرض.*

*معالجة خطأ PHP*

*عند إنشاء البرامج النصية وتطبيقات الويب، ومعالجة الأخطاء هو جزء مهم. إذا التعليمات البرمجية يفتقر خطأ تدقيق رمز، البرنامج قد تبدو غير مهنية للغاية وكنت قد تكون مفتوحة لمخاطر أمنية.*

*هذا البرنامج التعليمي يحتوي على بعض من الخطأ الأكثر شيوعا فحص الأساليب في PHP.*

*ونحن سوف تظهر مختلف أساليب معالجة الأخطاء:*

* *بسيطة "يموت ()" البيانات*
* *الأخطاء المخصصة ومشغلات الخطأ*
* *التقرير عن الخطأ*

*معالجة خطأ الأساسية: استخدام يموت (وظيفة)*

*يوضح المثال الأول سيناريو بسيط الذي يفتح ملف نصي:*

*<?php  
$file=fopen("welcome.txt","r");  
?>*

*حالة عدم وجود الملف قد تحصل خطأ مثل هذا:*

*Warning: fopen(welcome.txt) [function.fopen]: failed to open stream:  
No such file or directory in C:\webfolder\test.php on line 2*

*لمنع المستخدم من الحصول على رسالة خطأ مثل واحد أعلاه، ونحن اختبار ما إذا كان الملف موجودا قبل أن نحاول الوصول إليه:*

*<?php  
if(!file\_exists("welcome.txt"))  
  {  
  die("File not found");  
  }  
else  
  {  
  $file=fopen("welcome.txt","r");  
  }  
?>*

*الآن إذا لم يكن الملف موجودا تحصل على خطأ مثل هذا:*

*File not found*

*رمز أعلاه هو أكثر كفاءة من التعليمات البرمجية السابقة، لأنه يستخدم آلية معالجة الأخطاء بسيطة لوقف البرنامج النصي بعد الخطأ.*

*ومع ذلك، ببساطة وقف البرنامج النصي ليست دائما الطريق الصحيح للذهاب. دعونا نلقي نظرة على وظائف PHP بديل عن الأخطاء المناولة.*

*إنشاء معالج خطأ مخصص*

*إنشاء معالج خطأ مخصصة بسيط جدا. نحن ببساطة إنشاء دالة خاصة التي يمكن استدعاؤها عند حدوث خطأ في PHP.*

*يجب أن تكون هذه الوظيفة قادرة على التعامل مع ما لا يقل عن اثنين من المعلمات (مستوى خطأ ورسالة الخطأ) ولكن لا يمكن أن يقبل ما يصل إلى خمسة المعلمات (اختياري: الملف، خط العدد، والسياق الخطأ):*

*بناء الجملة*

*error\_function(error\_level,error\_message,  
error\_file,error\_line,error\_context)*

|  |  |
| --- | --- |
| *Parameter* | *Description* |
| *error\_level* | *Required. Specifies the error report level for the user-defined error. Must be a value number. See table below for possible error report levels* |
| *error\_message* | *Required. Specifies the error message for the user-defined error* |
| *error\_file* | *Optional. Specifies the filename in which the error occurred* |
| *error\_line* | *Optional. Specifies the line number in which the error occurred* |
| *error\_context* | *Optional. Specifies an array containing every variable, and their values, in use when the error occurred* |

*مستويات تقرير الخطأ*

*هذه المستويات التقرير عن الخطأ هي أنواع مختلفة من الخطأ معالج الأخطاء المعرفة من قبل المستخدم يمكن استخدامها من أجل:*

|  |  |  |
| --- | --- | --- |
| *Value* | *Constant* | *Description* |
| *2* | *E\_WARNING* | *Non-fatal run-time errors. Execution of the script is not halted* |
| *8* | *E\_NOTICE* | *Run-time notices. The script found something that might be an error, but could also happen when running a script normally* |
| *256* | *E\_USER\_ERROR* | *Fatal user-generated error. This is like an E\_ERROR set by the programmer using the PHP function trigger\_error()* |
| *512* | *E\_USER\_WARNING* | *Non-fatal user-generated warning. This is like an E\_WARNING set by the programmer using the PHP function trigger\_error()* |
| *1024* | *E\_USER\_NOTICE* | *User-generated notice. This is like an E\_NOTICE set by the programmer using the PHP function trigger\_error()* |
| *4096* | *E\_RECOVERABLE\_ERROR* | *Catchable fatal error. This is like an E\_ERROR but can be caught by a user defined handle (see also set\_error\_handler())* |
| *8191* | *E\_ALL* | *All errors and warnings (E\_STRICT became a part of E\_ALL in PHP 5.4)* |

*الآن يتيح إنشاء دالة لمعالجة الأخطاء:*

*function customError($errno, $errstr)  
  {  
  echo "<b>Error:</b> [$errno] $errstr<br>";  
  echo "Ending Script";  
  die();  
  }*

*رمز أعلاه هي وظيفة معالجة الأخطاء بسيطة. عندما يتم تشغيله، فإنه يحصل على مستوى الخطأ ورسالة خطأ. بعد ذلك بإخراج مستوى الخطأ ورسالة وإنهاء النصي.*

*الآن بعد أن قمنا بإنشاء وظيفة معالجة الأخطاء نحن بحاجة إلى أن تقرر متى يجب تشغيله.*

*تعيين معالج خطأ*

*معالج الأخطاء الافتراضية لPHP هو الذي بني في معالج الأخطاء. نحن ذاهبون لجعل وظيفة فوق معالج خطأ الافتراضي لمدة النصي.*

*فمن الممكن لتغيير معالج خطأ لتطبيق فقط بعض الأخطاء، وبهذه الطريقة يمكن التعامل مع البرنامج النصي أخطاء مختلفة بطرق مختلفة. ومع ذلك، في هذا المثال نحن نذهب لاستخدام لدينا معالج خطأ مخصصة لجميع الأخطاء:*

*set\_error\_handler("customError");*

*حيث أننا نريد الدالة المخصصة جهدنا لمعالجة كافة الأخطاء، وset\_error\_handler () يحتاج فقط معلمة واحدة، يمكن إضافة المعلمة الثانية لتحديد مستوى الخطأ.*

*مثال*

*اختبار معالج خطأ خلال محاولة لإخراج المتغير الذي لا وجود له:*

*<?php  
//error handler function  
function customError($errno, $errstr)  
  {  
  echo "<b>Error:</b> [$errno] $errstr";  
  }  
  
//set error handler  
set\_error\_handler("customError");  
  
//trigger error  
echo($test);  
?>*

*الناتج من التعليمات البرمجية أعلاه ينبغي أن يكون شيئا من هذا القبيل:*

*Error: [8] Undefined variable: test*

*يؤدي الى خطأ*

*في السيناريو حيث يمكن للمستخدمين إدخال البيانات من المفيد أن تؤدي الأخطاء عند حدوث المدخلات غير قانوني. في PHP، ويتم ذلك من قبل trigger\_error (وظيفة).*

*مثال*

*في هذا المثال يحدث خطأ إذا كان "اختبار" المتغير هو أكبر من "1":*

*<?php  
$test=2;  
if ($test>1)  
{  
trigger\_error("Value must be 1 or below");  
}  
?>*

*الناتج من التعليمات البرمجية أعلاه ينبغي أن يكون شيئا من هذا القبيل:*

*Notice: Value must be 1 or below  
in C:\webfolder\test.php on line 6*

*يمكن أن تسبب خطأ في أي مكان كنت ترغب في السيناريو، وذلك بإضافة المعلمة الثانية، يمكنك تحديد ما أثار مستوى الخطأ.*

*أنواع الخطأ المحتملة:*

* *E\_USER\_ERROR - فادح خطأ وقت التشغيل المستخدم لتوليد. الأخطاء التي لا يمكن استردادها من. وأوقف تنفيذ البرنامج النصي*
* *E\_USER\_WARNING - غير قاتلة للإنذار وقت التشغيل المستخدم لتوليد. لم يتم وقف تنفيذ البرنامج النصي*
* *E\_USER\_NOTICE - افتراضي. المستخدم لتوليد إشعار وقت التشغيل. العثور على البرنامج النصي شيء يمكن أن يكون خطأ، ولكن يمكن أن يحدث أيضا عند تشغيل البرنامج النصي عادة*

*مثال*

*في هذا المثال يحدث E\_USER\_WARNING إذا كان "اختبار" المتغير هو أكبر من "1". في حالة حدوث E\_USER\_WARNING سوف نستخدم لدينا معالج خطأ مخصصة وإنهاء البرنامج النصي:*

*<?php  
//error handler function  
function customError($errno, $errstr)  
  {  
  echo "<b>Error:</b> [$errno] $errstr<br>";  
  echo "Ending Script";  
  die();  
  }  
  
//set error handler  
set\_error\_handler("customError",E\_USER\_WARNING);  
  
//trigger error  
$test=2;  
if ($test>1)  
  {  
  trigger\_error("Value must be 1 or below",E\_USER\_WARNING);  
  }  
?>*

*الناتج من التعليمات البرمجية أعلاه ينبغي أن يكون شيئا من هذا القبيل:*

*Error: [512] Value must be 1 or below  
Ending Script*

*الآن بعد أن تعلمنا لخلق الأخطاء الخاصة بنا وكيفية تحريك لهم، دعونا نلقي نظرة على تسجيل الخطأ.*

*خطأ تسجيل*

*افتراضيا، PHP يرسل سجل خطأ في نظام تسجيل للملقم أو ملف، اعتمادا على كيفية إعداد تكوين error\_log في ملف php.ini. باستخدام error\_log (وظيفة) يمكنك إرسال سجلات الخطأ إلى ملف محدد أو وجهة بعيدة.*

*يمكن إرسال رسائل الخطأ لنفسك عن طريق البريد الإلكتروني أن يكون وسيلة جيدة للحصول على إخطار من أخطاء محددة.*

*إرسال رسالة خطأ عن طريق البريد الإلكتروني*

*في المثال التالي سوف نقوم بإرسال بريد إلكتروني مع رسالة خطأ وإنهاء السيناريو، في حالة حدوث خطأ معينة:*

*<?php  
//error handler function  
function customError($errno, $errstr)  
  {  
  echo "<b>Error:</b> [$errno] $errstr<br>";  
  echo "Webmaster has been notified";  
  error\_log("Error: [$errno] $errstr",1,  
  "someone@example.com","From: webmaster@example.com");  
  }  
  
//set error handler  
set\_error\_handler("customError",E\_USER\_WARNING);  
  
//trigger error  
$test=2;  
if ($test>1)  
  {  
  trigger\_error("Value must be 1 or below",E\_USER\_WARNING);  
  }  
?>*

*الناتج من التعليمات البرمجية أعلاه ينبغي أن يكون شيئا من هذا القبيل:*

*Error: [512] Value must be 1 or below  
Webmaster has been notified*

*والبريد الواردة من التعليمات البرمجية أعلاه يبدو مثل هذا:*

*Error: [512] Value must be 1 or below*

*هذا لا ينبغي أن تستخدم مع جميع الأخطاء. يجب أن يتم تسجيل الأخطاء العادية على الخادم باستخدام نظام تسجيل PHP الافتراضي.*

*PHP معالجة استثناء*

*وتستخدم استثناءات لتغيير التدفق الطبيعي للبرنامج نصي في حالة حدوث الخطأ المحدد.*

*ما هو استثناء*

*مع PHP 5 جاءت وجوه المنحى طريقة جديدة للتعامل مع الأخطاء.*

*يستخدم معالجة الاستثناء لتغيير التدفق الطبيعي للتنفيذ التعليمات البرمجية في حالة حدوث الخطأ المحدد (استثنائية) حالة. ويسمى هذا الشرط استثناء وهذا ما يحدث عادة عندما يتم تشغيل استثناء:*

* *يتم حفظ الدولة الرمز الحالي*
* *سوف بتنفيذ التعليمات البرمجية التحول إلى معرفة مسبقا (حسب الطلب) وظيفة معالج استثناء*
* *اعتمادا على الحالة، قد معالج ثم استئناف التنفيذ من الدولة رمز المحفوظة، إنهاء تنفيذ البرنامج النصي أو مواصلة البرنامج النصي من موقع مختلف في التعليمات البرمجية*

*ونحن سوف تظهر مختلف أساليب معالجة الأخطاء:*

* *استخدام الأساسية للاستثناءات*
* *إنشاء معالج استثناء مخصصة*
* *استثناءات متعددة*
* *إعادة رمي استثناء*
* *وضع معالج استثناء المستوى الأعلى*

*ملاحظة: يجب استخدام الاستثناءات فقط مع حالات الخطأ، وينبغي ألا تستخدم للانتقال إلى مكان آخر في التعليمات البرمجية عند نقطة محددة.*

*استخدام الأساسية للاستثناءات*

*عندما يتم طرح استثناء، لن يتم تنفيذ التعليمات البرمجية التالية لها، وسوف PHP محاولة للعثور على مطابقة "الصيد" كتلة.*

*إذا لم يتم اكتشاف استثناء، ستصدر خطأ فادح مع رسالة "استثناء غير مسك".*

*يتيح محاولة لرمي استثناء دون اللحاق به:*

*<?php  
//create function with an exception  
function checkNum($number)  
  {  
  if($number>1)  
    {  
    throw new Exception("Value must be 1 or below");  
    }  
  return true;  
  }  
  
//trigger exception  
checkNum(2);  
?>*

*سوف رمز أعلاه تحصل على خطأ مثل هذا:*

*Fatal error: Uncaught exception 'Exception'  
with message 'Value must be 1 or below' in C:\webfolder\test.php:6  
Stack trace: #0 C:\webfolder\test.php(12):  
checkNum(28) #1 {main} thrown in C:\webfolder\test.php on line 6*

*محاولة، ورمي والتقاط*

*لتجنب الخطأ من المثال أعلاه، نحن بحاجة إلى إنشاء التعليمات البرمجية المناسبة لمعالجة استثناء.*

*يجب أن رمز الاستثناء المناسبة ما يلي:*

1. *محاولة - وظيفة باستخدام استثناء يجب أن يكون في كتلة "محاولة". إذا لم تؤدي إلى استثناء، وستواصل رمز كالمعتاد. ولكن إذا بتشغيل استثناء، استثناء هو "طرح"*
2. *رمي - هذه هي الطريقة التي يؤدي استثناء. كل "رمي" يجب أن يكون "الصيد" واحد على الأقل*
3. *قبض - A "الصيد" كتلة باسترداد استثناء وبإنشاء كائن يحتوي على المعلومات استثناء*

*يتيح محاولة لاشعال استثناء مع رمز صالح:*

*<?php  
//create function with an exception  
function checkNum($number)  
  {  
  if($number>1)  
    {  
    throw new Exception("Value must be 1 or below");  
    }  
  return true;  
  }  
  
//trigger exception in a "try" block  
try  
  {  
  checkNum(2);  
  //If the exception is thrown, this text will not be shown  
  echo 'If you see this, the number is 1 or below';  
  }  
  
//catch exception  
catch(Exception $e)  
  {  
  echo 'Message: ' .$e->getMessage();  
  }  
?>*

*سوف رمز أعلاه تحصل على خطأ مثل هذا:*

*Message: Value must be 1 or below*

*وأوضح سبيل المثال:*

*رمز أعلاه يطرح استثناء وأدرك أنه:*

1. *يتم إنشاء وظيفة checkNum (). فإنه يتحقق ما إذا كان العدد أكبر من 1. إذا كان كذلك، يتم طرح استثناء*
2. *يتم استدعاء الدالة checkNum () في كتلة "محاولة"*
3. *يتم طرح استثناء ضمن الدالة checkNum ()*
4. *في "الصيد" كتلة retrives الاستثناء ويخلق كائن ($ ه) الذي يحتوي على معلومات استثناء*
5. *وردد رسالة الخطأ من استثناء بالدعوة $ الإلكترونية> getMessage () من الكائن استثناء*

*ومع ذلك، طريقة واحدة للالتفاف على "كل رمي يجب أن يكون الصيد" القاعدة هي لتعيين معالج استثناء المستوى الأعلى لمعالجة الأخطاء التي تنزلق من خلال.*

*إنشاء فئة استثناء مخصص*

*إنشاء معالج استثناء العرف هو بسيط جدا. نحن ببساطة إنشاء فئة خاصة مع الوظائف التي يمكن استدعاؤها عند حدوث استثناء في PHP. يجب أن تكون الطبقة امتدادا للطبقة استثناء.*

*الطبقة استثناء مخصصة يرث خصائص من الدرجة استثناء PHP ويمكنك إضافة وظائف مخصصة لذلك.*

*يتيح إنشاء فئة استثناء:*

*<?php  
class customException extends Exception  
  {  
  public function errorMessage()  
    {  
    //error message  
    $errorMsg = 'Error on line '.$this->getLine().' in '.$this->getFile()  
    .': <b>'.$this->getMessage().'</b> is not a valid E-Mail address';  
    return $errorMsg;  
    }  
  }  
  
$email = "someone@example...com";  
  
try  
  {  
  //check if  
  if(filter\_var($email, FILTER\_VALIDATE\_EMAIL) === FALSE)  
    {  
    //throw exception if email is not valid  
    throw new customException($email);  
    }  
  }  
  
catch (customException $e)  
  {  
  //display custom message  
  echo $e->errorMessage();  
  }  
?>*

*الطبقة الجديدة هي نسخة من الطبقة استثناء القديمة مع إضافة وظيفة errorMessage (). نظرا لأنه هو نسخة من الطبقة القديمة، وأنه يرث خصائص وأساليب من الطبقة القديمة، يمكننا استخدام أساليب الفئة استثناء مثل getLine () وgetFile () وgetMessage ().*

*وأوضح سبيل المثال:*

*رمز أعلاه يطرح استثناء وأدرك أنه مع استثناء فئة مخصصة:*

1. *يتم إنشاء customException () فئة امتدادا للطبقة استثناء القديمة. بهذه الطريقة فإنه يرث كل الأساليب والخصائص من فئة استثناء القديمة*
2. *يتم إنشاء وظيفة errorMessage (). هذه الدالة تقوم بإرجاع رسالة خطأ إذا كان عنوان البريد الإلكتروني غير صالح*
3. *يتم تعيين البريد الإلكتروني المتغير $ إلى سلسلة ليس عنوان بريد إلكتروني صالح*
4. *في "محاولة" يتم تنفيذ كتلة ويتم طرح استثناء منذ عنوان البريد الإلكتروني غير صالح*
5. *في "الصيد" كتلة catches الاستثناء ويعرض رسالة خطأ*

*استثناءات متعددة*

*فمن الممكن لبرنامج نصي لاستخدام استثناءات متعددة للتحقق من ظروف متعددة.*

*فمن الممكن استخدام عدة إذا .. كتل آخر، التبديل، أو عش استثناءات متعددة. ويمكن لهذه الاستثناءات استخدام فئات مختلفة استثناء والعودة رسائل خطأ مختلفة:*

*<?php  
class customException extends Exception  
{  
public function errorMessage()  
{  
//error message  
$errorMsg = 'Error on line '.$this->getLine().' in '.$this->getFile()  
.': <b>'.$this->getMessage().'</b> is not a valid E-Mail address';  
return $errorMsg;  
}  
}  
  
$email = "someone@example.com";  
  
try  
  {  
  //check if  
  if(filter\_var($email, FILTER\_VALIDATE\_EMAIL) === FALSE)  
    {  
    //throw exception if email is not valid  
    throw new customException($email);  
    }  
  //check for "example" in mail address  
  if(strpos($email, "example") !== FALSE)  
    {  
    throw new Exception("$email is an example e-mail");  
    }  
  }  
  
catch (customException $e)  
  {  
  echo $e->errorMessage();  
  }  
  
catch(Exception $e)  
  {  
  echo $e->getMessage();  
  }  
?>*

*وأوضح سبيل المثال:*

*رمز أعلاه اختبارات شرطين ويطرح استثناء إذا لم يتم استيفاء أي من الشروط:*

1. *يتم إنشاء customException () فئة امتدادا للطبقة استثناء القديمة. بهذه الطريقة فإنه يرث كل الأساليب والخصائص من فئة استثناء القديمة*
2. *يتم إنشاء وظيفة errorMessage (). هذه الدالة تقوم بإرجاع رسالة خطأ إذا كان عنوان البريد الإلكتروني غير صالح*
3. *يتم تعيين متغير $ البريد الإلكتروني إلى سلسلة هذا هو عنوان بريد إلكتروني صالح، ولكن يحتوي على السلسلة "المثال"*
4. *في "محاولة" يتم تنفيذ كتلة وليس إلقاء استثناء على الشرط الأول*
5. *الشرط الثاني يطلق استثناء منذ البريد الإلكتروني يحتوي على "سبيل المثال" سلسلة*
6. *في "الصيد" كتلة catches الاستثناء ويعرض رسالة خطأ الصحيح*

*إذا كانت استثناء القيت من customException الدرجة ولم تكن هناك customException الصيد، إلا أن الصيد الاستثناء قاعدة، والاستثناء أن يتم التعامل معها هناك.*

*الاستثناءات من رماة إعادة*

*في بعض الأحيان، عندما يتم طرح استثناء، قد ترغب في التعامل معها بشكل مختلف من الطريقة القياسية. فمن الممكن أن يقوم بطرح استثناء للمرة الثانية في غضون "الصيد" كتلة.*

*وهناك سيناريو ينبغي إخفاء أخطاء النظام من المستخدمين. قد تكون أخطاء النظام الهامة لالمبرمج، ولكن هو من لا مصلحة للمستخدم. لجعل الأمور أسهل بالنسبة للمستخدم يمكنك إعادة رمي استثناء مع رسالة سهلة الاستعمال:*

*<?php  
class customException extends Exception  
  {  
  public function errorMessage()  
    {  
    //error message  
    $errorMsg = $this->getMessage().' is not a valid E-Mail address.';  
    return $errorMsg;  
    }  
  }  
  
$email = "someone@example.com";  
  
try  
  {  
  try  
    {  
    //check for "example" in mail address  
    if(strpos($email, "example") !== FALSE)  
      {  
      //throw exception if email is not valid  
      throw new Exception($email);  
      }  
    }  
  catch(Exception $e)  
    {  
    //re-throw exception  
    throw new customException($email);  
    }  
  }  
  
catch (customException $e)  
  {  
  //display custom message  
  echo $e->errorMessage();  
  }  
?>*

*وأوضح سبيل المثال:*

*رمز أعلاه الاختبارات إذا كان عنوان بريد إلكتروني يحتوي على السلسلة "المثال" في ذلك، وإذا كان كذلك، والاستثناء هو إعادة طرح:*

1. *يتم إنشاء customException () فئة امتدادا للطبقة استثناء القديمة. بهذه الطريقة فإنه يرث كل الأساليب والخصائص من فئة استثناء القديمة*
2. *يتم إنشاء وظيفة errorMessage (). هذه الدالة تقوم بإرجاع رسالة خطأ إذا كان عنوان البريد الإلكتروني غير صالح*
3. *يتم تعيين متغير $ البريد الإلكتروني إلى سلسلة هذا هو عنوان بريد إلكتروني صالح، ولكن يحتوي على السلسلة "المثال"*
4. *في "محاولة" يحتوي على كتلة أخرى "محاولة" كتلة لتجعل من الممكن لإعادة رمي استثناء*
5. *يتم تشغيل استثناء منذ البريد الإلكتروني يحتوي على "سبيل المثال" سلسلة*
6. *في "الصيد" كتلة يمسك استثناء وإعادة رميات واحد "customException"*
7. *القبض على "customException" ويعرض رسالة خطأ*

*إذا لم يتم القبض على استثناء في "محاولة" كتلة الحالي، فإنه سيتم البحث عن كتلة الصيد على "مستويات أعلى".*

*تعيين استثناء معالج أعلى مستوى*

*وظيفة set\_exception\_handler () يحدد دالة معرفة من قبل المستخدم للتعامل مع جميع الاستثناءات غير مسك.*

*<?php  
function myException($exception)  
{  
echo "<b>Exception:</b> " , $exception->getMessage();  
}  
  
set\_exception\_handler('myException');  
  
throw new Exception('Uncaught Exception occurred');  
?>*

*الناتج من التعليمات البرمجية أعلاه ينبغي أن يكون شيئا من هذا القبيل:*

*Exception: Uncaught Exception occurred*

*في التعليمات البرمجية أعلاه لم يكن هناك "الصيد" كتلة. بدلا من ذلك، أثار معالج الاستثناء المستوى الأعلى. هذه الوظيفة ينبغي أن تستخدم لالتقاط الاستثناءات غير مسك.*

*قواعد للاستثناءات*

* *قد تكون محاطة التعليمات البرمجية في كتلة المحاولة، للمساعدة في الاستثناءات المحتملة الصيد*
* *كل محاولة كتلة أو "رمي" يجب أن يكون واحد على الأقل المقابلة كتلة catch*
* *كتل الصيد متعددة يمكن استخدامها لالتقاط فئات مختلفة من الاستثناءات*
* *استثناءات يمكن طرح (أو إعادة طرح) في كتلة catch داخل كتلة المحاولة*

*وهناك قاعدة بسيطة: إذا كنت رمي ​​شيء، لديك للقبض عليه.*

*PHP تصفية*

*وتستخدم مرشحات PHP للتحقق من صحة وتصفية البيانات القادمة من مصادر غير آمنة، مثل إدخال المستخدم.*

*ما هو تصفية PHP؟*

*ويستخدم فلتر PHP للتحقق من صحة وتصفية البيانات القادمة من مصادر غير آمنة.*

*لاختبار والتحقق من صحة إدخال المستخدم وتصفية البيانات أو العرف هو جزء مهم من أي تطبيق ويب.*

*تم تصميم تمديد PHP فلتر لجعل البيانات تصفية أسهل وأسرع.*

*لماذا استخدام تصفية؟*

*تعتمد تقريبا جميع تطبيقات الويب على المدخلات الخارجية. عادة هذا يأتي من مستخدم أو تطبيق آخر (مثل خدمة ويب). باستخدام مرشحات يمكنك التأكد من التطبيق الخاص بك يحصل نوع الإدخال الصحيح.*

*يجب عليك دائما تحديد كل البيانات الخارجية!*

*تصفية المدخلات هي واحدة من القضايا الأمنية التطبيق الأكثر أهمية.*

*ما هو البيانات الخارجية؟*

* *إدخال البيانات من نموذج*
* *بسكويت*
* *خدمات البيانات على شبكة الإنترنت*
* *متغيرات الملقم*
* *نتائج استعلام قاعدة بيانات*

*وظائف وفلاتر*

*لتصفية متغير، استخدم إحدى الوظائف التصفية التالية:*

* *filter\_var () - فلاتر متغير واحد مع مرشح محدد*
* *filter\_var\_array () - تصفية العديد من المتغيرات مع نفس أو مختلفة مرشحات*
* *filter\_input - الحصول على واحد متغير المدخلات وتصفيته*
* *filter\_input\_array - الحصول على العديد من المتغيرات المدخلات وتصفية لهم نفس أو مختلفة مرشحات*

*في المثال أدناه، ونحن تحقق من صحة عدد صحيح باستخدام filter\_var (وظيفة):*

*<?php  
$int = 123;  
  
if(!filter\_var($int, FILTER\_VALIDATE\_INT))  
  {  
  echo("Integer is not valid");  
  }  
else  
  {  
  echo("Integer is valid");  
  }  
?>*

*رمز أعلاه يستخدم "FILTER\_VALIDATE\_INT" فلتر لتصفية المتغير. منذ صحيحا صالحا، فإن الناتج من التعليمات البرمجية أعلاه يكون: "صحيح هو صالح".*

*إذا حاولنا مع المتغير الذي ليس عدد صحيح (مثل "123ABC")، فإن الناتج يكون: "صحيح غير صالح".*

*للحصول على قائمة كاملة من وظائف والمرشحات، وزيارة موقعنا*[*PHP تصفية مرجع.*](http://www.w3schools.com/php/php_ref_filter.asp)

*التحقق من صحة والتعقيم*

*هناك نوعان من الفلاتر:*

*التحقق من المرشحات:*

* *تستخدم للتحقق من صحة إدخال المستخدم*
* *قواعد تنسيق صارمة (مثل URL أو التحقق من صحة البريد الإلكتروني)*
* *بإرجاع نوع المتوقع على النجاح أو FALSE عند الفشل*

*التعقيم المرشحات:*

* *وتستخدم للسماح أو عدم السماح الحروف المحددة في سلسلة*
* *توجد قواعد تنسيق البيانات*
* *العودة دائما السلسلة*

*الخيارات وأعلام*

*وتستخدم الخيارات وأعلام لإضافة خيارات تصفية إضافية للمرشحات المحدد.*

*مرشحات مختلفة لديها خيارات وأعلام مختلفة.*

*في المثال أدناه، ونحن تحقق من صحة عدد صحيح باستخدام filter\_var () و "min\_range" و "max\_range" خيارات:*

*<?php  
$var=300;  
  
$int\_options = array(  
"options"=>array  
  (  
  "min\_range"=>0,  
  "max\_range"=>256  
  )  
);  
  
if(!filter\_var($var, FILTER\_VALIDATE\_INT, $int\_options))  
  {  
  echo("Integer is not valid");  
  }  
else  
  {  
  echo("Integer is valid");  
  }  
?>*

*مثل رمز أعلاه، يجب وضع خيارات في مجموعة النقابي مع اسم "خيارات". إذا تم استخدام العلم أنها لا تحتاج إلى أن تكون في صفيف.*

*منذ صحيحا هو "300" أنها ليست في النطاق المحدد، وسوف الناتج من التعليمات البرمجية أعلاه يكون: "صحيح غير صالح".*

*للحصول على قائمة كاملة من وظائف والمرشحات، وزيارة موقعنا*[*PHP تصفية مرجع.*](http://www.w3schools.com/php/php_ref_filter.asp)*تحقق كل مرشح لنرى ما هي الخيارات والأعلام المتاحة.*

*التحقق من صحة الإدخال*

*دعونا نحاول التحقق من صحة المدخلات من نموذج.*

*أول شيء يتعين علينا القيام به هو التأكد من أن إدخال البيانات ونحن نبحث عن موجودا.*

*ثم أننا تحديد إدخال البيانات باستخدام filter\_input (وظيفة).*

*في المثال التالي، يتم إرسال متغير المدخلات "البريد الإلكتروني" لصفحة PHP:*

*<?php  
if(!filter\_has\_var(INPUT\_GET, "email"))  
  {  
  echo("Input type does not exist");  
  }  
else  
  {  
  if (!filter\_input(INPUT\_GET, "email", FILTER\_VALIDATE\_EMAIL))  
    {  
    echo "E-Mail is not valid";  
    }  
  else  
    {  
    echo "E-Mail is valid";  
    }  
  }  
?>*

*وأوضح مثال*

*المثال أعلاه لديه مدخلا (البريد الإلكتروني) وجهت اليها باستخدام طريقة "GET":*

1. *تحقق ما إذا كان "البريد الالكتروني" متغير مدخلات "GET" نوع موجود*
2. *في حالة وجود متغير المدخلات، ومعرفة ما اذا كان هو عنوان بريد إلكتروني صالح*

*تطهير الإدخال*

*دعونا نحاول تنظيف عنوان URL المرسلة من نموذج.*

*أولا نؤكد أن إدخال البيانات ونحن نبحث عن موجودا.*

*ثم نحن تطهير إدخال البيانات باستخدام filter\_input (وظيفة).*

*في المثال التالي، يتم إرسال الإدخال متغير "رابط" لصفحة PHP:*

*<?php  
if(!filter\_has\_var(INPUT\_POST, "url"))  
  {  
  echo("Input type does not exist");  
  }  
else  
  {  
  $url = filter\_input(INPUT\_POST,   
  "url", FILTER\_SANITIZE\_URL);  
  }  
?>*

*وأوضح مثال*

*المثال أعلاه لديه مدخلا (رابط) وجهت اليها باستخدام "POST" الأسلوب:*

1. *معرفة ما اذا كان "رابط" مدخلات "POST" نوع موجود*
2. *في حالة وجود متغير المدخلات، وتطهير (يسلب أحرف غير صالحة) وتخزينها في المتغير $ رابط*

*إذا كان المتغير المدخلات عبارة عن سلسلة من هذا القبيل "http://www.W3Schools.com/"، ورابط المتغير $ بعد التعقيم تبدو مثل هذا:*

*http://www.W3Schools.com/*

*تحديد المدخلات متعددة*

*نموذج تتكون دائما تقريبا من حقل إدخال أكثر من واحد. لتجنب استدعاء filter\_var أو filter\_input وظائف أكثر وأكثر، يمكننا استخدام filter\_var\_array أو وظائف filter\_input\_array.*

*في هذا المثال نستخدم filter\_input\_array (وظيفة) لتصفية ثلاثة متغيرات GET. متغيرات GET تلقى هو اسم، والعمر، وعنوان البريد الإلكتروني:*

*<?php  
$filters = array  
  (  
  "name" => array  
    (  
    "filter"=>FILTER\_SANITIZE\_STRING  
    ),  
  "age" => array  
    (  
    "filter"=>FILTER\_VALIDATE\_INT,  
    "options"=>array  
      (  
      "min\_range"=>1,  
      "max\_range"=>120  
      )  
    ),  
  "email"=> FILTER\_VALIDATE\_EMAIL  
  );  
  
$result = filter\_input\_array(INPUT\_GET, $filters);  
  
if (!$result["age"])  
  {  
  echo("Age must be a number between 1 and 120.<br>");  
  }  
elseif(!$result["email"])  
  {  
  echo("E-Mail is not valid.<br>");  
  }  
else  
  {  
  echo("User input is valid");  
  }  
?>*

*وأوضح مثال*

*المثال أعلاه من ثلاثة المدخلات (الاسم والعمر والبريد الإلكتروني) وجهت اليها باستخدام طريقة "GET":*

1. *تعيين مجموعة تحتوي على اسم المتغيرات المدخلات والفلاتر المستخدمة في متغيرات الإدخال المحدد*
2. *استدعاء filter\_input\_array () وظيفة مع متغيرات المدخلات GET ومجموعة وضعناها فقط*
3. *تحقق من "العمر" والمتغيرات "البريد الإلكتروني" في المتغير $ نتيجة لمدخلات غير صالحة. (إذا كان أي من المتغيرات المدخلات غير صالحة، وسوف يكون هذا المتغير المدخلات FALSE بعد filter\_input\_array (وظيفة))*

*المعلمة الثانية من الدالة filter\_input\_array () يمكن أن يكون صفيف أو هوية مرشح واحد.*

*إذا كانت المعلمة هي هوية مرشح واحد يتم تصفيتها كافة القيم في صفيف الإدخال بواسطة عامل التصفية المحددة.*

*إذا كانت المعلمة عبارة عن صفيف يجب أن تتبع هذه القواعد:*

* *يجب أن يكون مجموعة النقابي تحتوي على متغير المدخلات كمفتاح مجموعة (مثل "العمر" متغير المدخلات)*
* *يجب أن تكون قيمة مجموعة هوية مرشح أو صفيف تحديد مرشح والأعلام والخيارات*

*باستخدام تصفية الاستدعاء*

*فمن الممكن لاستدعاء دالة المعرفة من قبل المستخدم واستخدامها كعامل تصفية باستخدام فلتر FILTER\_CALLBACK. بهذه الطريقة، لدينا السيطرة الكاملة على تصفية البيانات.*

*يمكنك إنشاء وظيفة المستخدم الخاص بك أو استخدام تعريف وظيفة PHP القائمة*

*الوظيفة التي ترغب في استخدامها لتصفية المحدد بنفس الطريقة كما تم تحديد الخيار. في مجموعة النقابي مع اسم "خيارات"*

*في المثال أدناه، ونحن نستخدم وظيفة مستخدم أنشئت لتحويل جميع "\_" إلى الفراغات:*

*<?php  
function convertSpace($string)  
{  
return str\_replace("\_", " ", $string);  
}  
  
$string = "Peter\_is\_a\_great\_guy!";  
  
echo filter\_var($string, FILTER\_CALLBACK,  
array("options"=>"convertSpace"));  
?>*

*النتيجة من التعليمات البرمجية أعلاه ينبغي أن تبدو هذه:*

*Peter is a great guy!*

*وأوضح مثال*

*المثال أعلاه تحويل كافة "\_" إلى الفراغات:*

1. *إنشاء دالة لتحل محل "\_" إلى الفراغات*
2. *استدعاء filter\_var (وظيفة) مع مرشح FILTER\_CALLBACK ومجموعة تحتوي على وظيفة لدينا*

*PHP صالحة*

*مجموعة بتخزين قيم متعددة في متغير واحد واحد:*

*مثال*

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

*ما هو صفيف؟*

*صفيف متغير الخاصة، والتي يمكن أن تعقد أكثر من قيمة واحدة في كل مرة.*

*إذا كان لديك قائمة من العناصر (قائمة بأسماء السيارات، على سبيل المثال)، وتخزين السيارات في المتغيرات واحد يمكن أن تبدو مثل هذا:*

*$cars1="Volvo";  
$cars2="BMW";  
$cars3="Toyota";*

*ومع ذلك، ما إذا كنت تريد أن يتكرر خلال السيارات والعثور على وظيفة معينة؟ وماذا لو لم يكن لك 3 سيارات، ولكن 300؟*

*الحل هو إنشاء صفيف!*

*مجموعة يمكن أن تعقد العديد من القيم تحت اسم واحد، ويمكنك الوصول إلى القيم من خلال الإشارة إلى رقم الفهرس.*

*إنشاء صفيف في PHP*

*في PHP، يتم استخدام الدالة مجموعة () لإنشاء صفيف:*

*array();*

*في PHP، وهناك ثلاثة أنواع من المصفوفات:*

* *صفائف المفهرسة - صالحة مع مؤشر رقمي*
* *صفائف النقابي - صالحة مع مفاتيح اسمه*
* *المصفوفات متعددة الأبعاد - صالحة تحتوي على واحد أو أكثر صفائف*

*PHP المفهرس صالحة*

*هناك طريقتان لإنشاء صفائف المفهرسة:*

*يمكن تعيين المؤشر تلقائيا (يبدأ المؤشر دائما عند 0):*

*$cars=array("Volvo","BMW","Toyota");*

*أو مؤشر يمكن تعيين يدويا:*

*$cars[0]="Volvo";  
$cars[1]="BMW";  
$cars[2]="Toyota";*

*المثال التالي بإنشاء مجموعة مفهرسة باسم $ السيارات، يعين ثلاثة عناصر لذلك، ومن ثم يطبع النص الذي يحتوي على مجموعة القيم:*

*مثال*

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

*الحصول على طول صفيف - العد () وظيفة*

*يتم استخدام وظيفة العد () لإرجاع طول (عدد العناصر) من مجموعة:*

*مثال*

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

*حلقة من خلال صفيف مفهرسة*

*لحلقة من خلال وطباعة جميع القيم من مجموعة مفهرسة، هل يمكن استخدام للحلقة، مثل هذا:*

*مثال*

*<?php  
$cars=array("Volvo","BMW","Toyota");  
$arrlength=count($cars);  
  
for($x=0;$x<$arrlength;$x++)  
  {  
  echo $cars[$x];  
  echo "<br>";  
  }  
?>*

*PHP النقابي صالحة*

*صفائف النقابي هي المصفوفات التي تستخدم مفاتيح اسمه أن تعيين لهم.*

*هناك طريقتان لإنشاء مجموعة النقابي:*

*$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");*

*أو:*

*$age['Peter']="35";  
$age['Ben']="37";  
$age['Joe']="43";*

*ويمكن بعد ذلك أن تستخدم مفاتيح اسمه في البرنامج النصي:*

*مثال*

*<?php  
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");  
echo "Peter is " . $age['Peter'] . " years old.";  
?>*

*حلقة من خلال صفيف النقابي*

*لحلقة من خلال وطباعة جميع القيم من مجموعة النقابي، هل يمكن استخدام حلقة foreach، مثل هذا:*

*مثال*

*<?php  
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");  
  
foreach($age as $x=>$x\_value)  
  {  
  echo "Key=" . $x . ", Value=" . $x\_value;  
  echo "<br>";  
  }*

*صفائف متعددة الأبعاد*

[*المصفوفات متعددة الأبعاد*](http://www.w3schools.com/php/php_arrays_multi.asp)*وسيتم شرح في المقطع المتقدمة PHP.*

*كاملة PHP صفيف المرجعي*

*للإشارة كاملة من جميع وظائف مجموعة، انتقل إلى لدينا على أكمل*[*مجموعة بي إتش بي المرجعي*](http://www.w3schools.com/php/php_ref_array.asp)*.*

*يحتوي على المرجع وصفا موجزا، وأمثلة على الاستخدام، لكل وظيفة!*

PHP الخلية جملة WHERE

يتم استخدام جملة WHERE لتصفية السجلات.

جملة WHERE

يتم استخدام جملة WHERE لاستخراج تلك السجلات فقط التي تلبي معيار محدد.

بناء الجملة

SELECT column\_name(s)  
FROM table\_name  
WHERE column\_name operator value

لمعرفة المزيد عن SQL، يرجى زيارة موقعنا على [البرنامج التعليمي SQL](http://www.w3schools.com/sql/default.asp) .

للحصول على PHP لتنفيذ العبارة أعلاه يجب علينا أن نستخدم mysqli\_query (وظيفة). يتم استخدام هذه الدالة لإرسال استعلام أو الأمر إلى اتصال الخلية.

مثال

المثال التالي تحديد كافة الصفوف من "الأشخاص" الجدول حيث "الاسم الأول = 'بيتر'":

<?php  
$con=mysqli\_connect("example.com","peter","abc123","my\_db");  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
  
$result = mysqli\_query($con,"SELECT \* FROM Persons  
WHERE FirstName='Peter'");  
  
while($row = mysqli\_fetch\_array($result))  
  {  
  echo $row['FirstName'] . " " . $row['LastName'];  
  echo "<br>";  
  }  
?>

والناتج من التعليمات البرمجية أعلاه يكون:

Peter Griffin

PHP الخلية طلب عن طريق الكلمات الرئيسية

يتم استخدام ORDER BY الكلمة لفرز البيانات في السجلات.

وORDER BY كلمة

يتم استخدام ORDER BY الكلمة لفرز البيانات في السجلات.

وORDER BY الكلمة فرز السجلات بترتيب تصاعدي بشكل افتراضي.

إذا كنت ترغب في فرز السجلات في ترتيب تنازلي، يمكنك استخدام الكلمة الأساسية DESC.

بناء الجملة

SELECT column\_name(s)  
FROM table\_name  
ORDER BY column\_name(s) ASC|DESC

لمعرفة المزيد عن SQL، يرجى زيارة موقعنا على [البرنامج التعليمي SQL](http://www.w3schools.com/sql/default.asp) .

مثال

المثال التالي تحديد كافة البيانات المخزنة في الجدول "الأشخاص"، وفرز النتيجة عن طريق عمود "العمر":

<?php  
$con=mysqli\_connect("example.com","peter","abc123","my\_db");  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
  
$result = mysqli\_query($con,"SELECT \* FROM Persons ORDER BY age");  
  
while($row = mysqli\_fetch\_array($result))  
  {  
  echo $row['FirstName'];  
  echo " " . $row['LastName'];  
  echo " " . $row['Age'];  
  echo "<br>";  
  }  
  
mysqli\_close($con);  
?>

والناتج من التعليمات البرمجية أعلاه يكون:

Glenn Quagmire 33  
Peter Griffin 35

النظام من قبل اثنين من الأعمدة

فمن الممكن أيضا أن تأمر من قبل أكثر من عمود واحد. عند الطلب من قبل أكثر من عمود واحد، ويستخدم العمود الثاني فقط إذا كانت القيم في العمود الأول متساوون:

SELECT column\_name(s)  
FROM table\_name  
ORDER BY column1, column2

PHP الخلية تحديث

يتم استخدام عبارة UPDATE لتعديل البيانات في جدول.

تحديث البيانات في قاعدة بيانات

يتم استخدام عبارة UPDATE لتحديث السجلات الموجودة في الجدول.

بناء الجملة

UPDATE table\_name  
SET column1=value, column2=value2,...  
WHERE some\_column=some\_value

**ملاحظة:** لاحظ جملة WHERE في بناء جملة UPDATE. يحدد جملة WHERE التي سجل أو السجلات التي ينبغي تحديثها. إذا قمت بحذف جملة WHERE، سيتم تحديث كافة السجلات!

لمعرفة المزيد عن SQL، يرجى زيارة موقعنا على [البرنامج التعليمي SQL](http://www.w3schools.com/sql/default.asp) .

للحصول على PHP لتنفيذ العبارة أعلاه يجب علينا أن نستخدم mysqli\_query (وظيفة). يتم استخدام هذه الدالة لإرسال استعلام أو الأمر إلى اتصال الخلية.

مثال

في البرنامج التعليمي في وقت سابق أنشأنا جدول يسمى "الأشخاص". هنا هو كيف يبدو:

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Age** |
| Peter | Griffin | 35 |
| Glenn | Quagmire | 33 |

المثال التالي بتحديث بعض البيانات في الجدول "الأشخاص":

<?php  
$con=mysqli\_connect("example.com","peter","abc123","my\_db");  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
  
mysqli\_query($con,"UPDATE Persons SET Age=36  
WHERE FirstName='Peter' AND LastName='Griffin'");  
  
mysqli\_close($con);  
?>

بعد التحديث، سوف الطاولة "الأشخاص" تبدو مثل هذا:

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Age** |
| Peter | Griffin | 36 |
| Glenn | Quagmire | 33 |

PHP الخلية حذف

يتم استخدام عبارة DELETE لحذف السجلات في الجدول.

حذف البيانات في قاعدة بيانات

وحذف من البيان يستخدم لحذف السجلات من جدول قاعدة البيانات.

بناء الجملة

DELETE FROM table\_name  
WHERE some\_column = some\_value

**ملاحظة:** لاحظ جملة WHERE في بناء جملة DELETE. يحدد جملة WHERE التي سجل أو السجلات التي ينبغي حذفها. إذا قمت بحذف جملة WHERE، سيتم حذف جميع السجلات!

لمعرفة المزيد عن SQL، يرجى زيارة موقعنا على [البرنامج التعليمي SQL](http://www.w3schools.com/sql/default.asp) .

للحصول على PHP لتنفيذ العبارة أعلاه يجب علينا أن نستخدم mysqli\_query (وظيفة). يتم استخدام هذه الدالة لإرسال استعلام أو الأمر إلى اتصال الخلية.

مثال

إلقاء نظرة على الجدول "الأشخاص" التالية:

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Age** |
| Peter | Griffin | 35 |
| Glenn | Quagmire | 33 |

حذف المثال التالي كافة السجلات في "الأشخاص" الجدول حيث اسم العائلة = 'غريفين':

<?php  
$con=mysqli\_connect("example.com","peter","abc123","my\_db");  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
  
mysqli\_query($con,"DELETE FROM Persons WHERE LastName='Griffin'");  
  
mysqli\_close($con);  
?>

بعد الحذف، فإن الجدول تبدو مثل هذا:

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Age** |
| Glenn | Quagmire | 33 |

PHP ODBC قاعدة بيانات

[«السابق](http://www.w3schools.com/php/php_mysql_delete.asp)

[الفصل التالي »](http://www.w3schools.com/php/php_xml_parser_expat.asp)

ODBC هو واجهة برمجة التطبيقات (API) الذي يسمح لك بالاتصال بمصدر بيانات (مثل قاعدة بيانات Access MS).

إنشاء اتصال ODBC

مع اتصال ODBC، يمكنك الاتصال بأي قاعدة البيانات، على أي جهاز كمبيوتر في شركتك، طالما اتصال ODBC هو متاح.

هنا هو كيفية إنشاء اتصال ODBC إلى قاعدة بيانات Access MS:

1. فتح **أدوات إدارية** الرمز في لوحة التحكم الخاصة بك.
2. انقر نقرا مزدوجا فوق على **مصادر البيانات (ODBC)** رمز الداخل.
3. اختيار **نظام DSN** التبويب.
4. انقر على **إضافة** في علامة التبويب نظام DSN.
5. حدد **برنامج تشغيل Microsoft Access** . انقر فوق **إنهاء.**
6. في الشاشة التالية، انقر فوق **تحديد** لتحديد موقع قاعدة البيانات.
7. تعطي قاعدة بيانات **اسم مصدر البيانات (DSN)** .
8. انقر فوق **موافق** .

لاحظ أن هذا التكوين الذي ينبغي القيام به على الكمبيوتر حيث يقع موقع الويب الخاص بك. إذا كنت تقوم بتشغيل ملقم معلومات إنترنت (IIS) على جهاز الكمبيوتر الخاص بك، وسوف تعمل الإرشادات المذكورة أعلاه، ولكن إذا كان موجودا موقع الويب الخاص بك على ملقم بعيد، يكون لديك الوصول الفعلي إلى هذا الخادم، أو اطلب استضافة الويب الخاص بك إلى ل إعداد DSN لتتمكن من استخدامها.

ربط إلى ODBC

يتم استخدام الدالة odbc\_connect () للاتصال بمصدر بيانات ODBC. وظيفة يأخذ المعلمات الأربعة: اسم مصدر البيانات، اسم المستخدم، كلمة السر، ونوع المؤشر اختياري.

يتم استخدام الدالة odbc\_exec () لتنفيذ عبارة SQL.

مثال

المثال التالي بإنشاء اتصال DSN دعا إلى NORTHWIND، مع عدم وجود اسم المستخدم وكلمة المرور لا. بعد ذلك يخلق SQL وتنفيذ ما يلي:

$conn=odbc\_connect('northwind','','');  
$sql="SELECT \* FROM customers";  
$rs=odbc\_exec($conn,$sql);

استرجاع السجلات

يتم استخدام الدالة odbc\_fetch\_row () لإرجاع السجلات من نتيجة المجموعة. هذه الدالة تقوم بإرجاع صحيح إذا كان قادرا على العودة الصفوف، كاذبة خلاف ذلك.

وظيفة معلمتين: معرف نتيجة ODBC ورقم الصف اختياري:

odbc\_fetch\_row($rs)

استرجاع حقول من سجل

يتم استخدام الدالة odbc\_result () لقراءة الحقول من السجل. هذه وظيفة معلمتين: معرف نتيجة ODBC وعدد الحقل أو الاسم.

سطر التعليمات البرمجية أدناه بإرجاع قيمة الحقل الأول من السجل:

$compname=odbc\_result($rs,1);

سطر التعليمات البرمجية أدناه بإرجاع قيمة حقل يسمى "اسم الشركة":

$compname=odbc\_result($rs,"CompanyName");

إغلاق ملف اتصال ODBC

يتم استخدام الدالة odbc\_close () لإغلاق اتصال ODBC.

odbc\_close($conn);

مثال ODBC

يوضح المثال التالي كيفية إنشاء أول اتصال قاعدة البيانات، ثم مجموعة، ونتيجة لذلك، ومن ثم عرض البيانات في جدول HTML.

<html>  
<body>  
  
<?php  
$conn=odbc\_connect('northwind','','');  
if (!$conn)  
  {exit("Connection Failed: " . $conn);}  
$sql="SELECT \* FROM customers";  
$rs=odbc\_exec($conn,$sql);  
if (!$rs)  
  {exit("Error in SQL");}  
echo "<table><tr>";  
echo "<th>Companyname</th>";  
echo "<th>Contactname</th></tr>";  
while (odbc\_fetch\_row($rs))  
  {  
  $compname=odbc\_result($rs,"CompanyName");  
  $conname=odbc\_result($rs,"ContactName");  
  echo "<tr><td>$compname</td>";  
  echo "<td>$conname</td></tr>";  
  }  
odbc\_close($conn);  
echo "</table>";  
?>  
  
</body>  
</html>

PHP 5 صفيف وظائف

PHP 5 وظائف صفيف

وظائف مجموعة بي إتش بي هي جزء من جوهر PHP. لا يلزم تركيب لاستخدام هذه الوظائف.

|  |  |
| --- | --- |
| **Function** | **Description** |
| [array()](http://www.w3schools.com/php/func_array.asp) | Creates an array |
| [array\_change\_key\_case()](http://www.w3schools.com/php/func_array_change_key_case.asp) | Changes all keys in an array to lowercase or uppercase |
| [array\_chunk()](http://www.w3schools.com/php/func_array_chunk.asp) | Splits an array into chunks of arrays |
| array\_column() | Returns the values from a single column in the input array |
| [array\_combine()](http://www.w3schools.com/php/func_array_combine.asp) | Creates an array by using the elements from one "keys" array and one "values" array |
| [array\_count\_values()](http://www.w3schools.com/php/func_array_count_values.asp) | Counts all the values of an array |
| [array\_diff()](http://www.w3schools.com/php/func_array_diff.asp) | Compare arrays, and returns the differences (compare values only) |
| [array\_diff\_assoc()](http://www.w3schools.com/php/func_array_diff_assoc.asp) | Compare arrays, and returns the differences (compare keys and values) |
| [array\_diff\_key()](http://www.w3schools.com/php/func_array_diff_key.asp) | Compare arrays, and returns the differences (compare keys only) |
| [array\_diff\_uassoc()](http://www.w3schools.com/php/func_array_diff_uassoc.asp) | Compare arrays, and returns the differences (compare keys and values, using a user-defined key comparison function) |
| [array\_diff\_ukey()](http://www.w3schools.com/php/func_array_diff_ukey.asp) | Compare arrays, and returns the differences (compare keys only, using a user-defined key comparison function) |
| [array\_fill()](http://www.w3schools.com/php/func_array_fill.asp) | Fills an array with values |
| [array\_fill\_keys()](http://www.w3schools.com/php/func_array_fill_keys.asp) | Fills an array with values, specifying keys |
| [array\_filter()](http://www.w3schools.com/php/func_array_filter.asp) | Filters the values of an array using a callback function |
| [array\_flip()](http://www.w3schools.com/php/func_array_flip.asp) | Flips/Exchanges all keys with their associated values in an array |
| [array\_intersect()](http://www.w3schools.com/php/func_array_intersect.asp) | Compare arrays, and returns the matches (compare values only) |
| [array\_intersect\_assoc()](http://www.w3schools.com/php/func_array_intersect_assoc.asp) | Compare arrays and returns the matches (compare keys and values) |
| [array\_intersect\_key()](http://www.w3schools.com/php/func_array_intersect_key.asp) | Compare arrays, and returns the matches (compare keys only) |
| [array\_intersect\_uassoc()](http://www.w3schools.com/php/func_array_intersect_uassoc.asp) | Compare arrays, and returns the matches (compare keys and values, using a user-defined key comparison function) |
| [array\_intersect\_ukey()](http://www.w3schools.com/php/func_array_intersect_ukey.asp) | Compare arrays, and returns the matches (compare keys only, using a user-defined key comparison function) |
| [array\_key\_exists()](http://www.w3schools.com/php/func_array_key_exists.asp) | Checks if the specified key exists in the array |
| [array\_keys()](http://www.w3schools.com/php/func_array_keys.asp) | Returns all the keys of an array |
| [array\_map()](http://www.w3schools.com/php/func_array_map.asp) | Sends each value of an array to a user-made function, which returns new values |
| [array\_merge()](http://www.w3schools.com/php/func_array_merge.asp) | Merges one or more arrays into one array |
| [array\_merge\_recursive()](http://www.w3schools.com/php/func_array_merge_recursive.asp) | Merges one or more arrays into one array recursively |
| [array\_multisort()](http://www.w3schools.com/php/func_array_multisort.asp) | Sorts multiple or multi-dimensional arrays |
| [array\_pad()](http://www.w3schools.com/php/func_array_pad.asp) | Inserts a specified number of items, with a specified value, to an array |
| [array\_pop()](http://www.w3schools.com/php/func_array_pop.asp) | Deletes the last element of an array |
| [array\_product()](http://www.w3schools.com/php/func_array_product.asp) | Calculates the product of the values in an array |
| [array\_push()](http://www.w3schools.com/php/func_array_push.asp) | Inserts one or more elements to the end of an array |
| [array\_rand()](http://www.w3schools.com/php/func_array_rand.asp) | Returns one or more random keys from an array |
| [array\_reduce()](http://www.w3schools.com/php/func_array_reduce.asp) | Returns an array as a string, using a user-defined function |
| [array\_replace()](http://www.w3schools.com/php/func_array_replace.asp) | Replaces the values of the first array with the values from following arrays |
| [array\_replace\_recursive()](http://www.w3schools.com/php/func_array_replace_recursive.asp) | Replaces the values of the first array with the values from following arrays recursively |
| [array\_reverse()](http://www.w3schools.com/php/func_array_reverse.asp) | Returns an array in the reverse order |
| [array\_search()](http://www.w3schools.com/php/func_array_search.asp) | Searches an array for a given value and returns the key |
| [array\_shift()](http://www.w3schools.com/php/func_array_shift.asp) | Removes the first element from an array, and returns the value of the removed element |
| [array\_slice()](http://www.w3schools.com/php/func_array_slice.asp) | Returns selected parts of an array |
| [array\_splice()](http://www.w3schools.com/php/func_array_splice.asp) | Removes and replaces specified elements of an array |
| [array\_sum()](http://www.w3schools.com/php/func_array_sum.asp) | Returns the sum of the values in an array |
| [array\_udiff()](http://www.w3schools.com/php/func_array_udiff.asp) | Compare arrays, and returns the differences (compare values only, using a user-defined key comparison function) |
| [array\_udiff\_assoc()](http://www.w3schools.com/php/func_array_udiff_assoc.asp) | Compare arrays, and returns the differences (compare keys and values, using a built-in function to compare the keys and a user-defined function to compare the values) |
| [array\_udiff\_uassoc()](http://www.w3schools.com/php/func_array_udiff_uassoc.asp) | Compare arrays, and returns the differences (compare keys and values, using two user-defined key comparison functions) |
| [array\_uintersect()](http://www.w3schools.com/php/func_array_uintersect.asp) | Compare arrays, and returns the matches (compare values only, using a user-defined key comparison function) |
| [array\_uintersect\_assoc()](http://www.w3schools.com/php/func_array_uintersect_assoc.asp) | Compare arrays, and returns the matches (compare keys and values, using a built-in function to compare the keys and a user-defined function to compare the values) |
| [array\_uintersect\_uassoc()](http://www.w3schools.com/php/func_array_uintersect_uassoc.asp) | Compare arrays, and returns the matches (compare keys and values, using two user-defined key comparison functions) |
| [array\_unique()](http://www.w3schools.com/php/func_array_unique.asp) | Removes duplicate values from an array |
| [array\_unshift()](http://www.w3schools.com/php/func_array_unshift.asp) | Adds one or more elements to the beginning of an array |
| [array\_values()](http://www.w3schools.com/php/func_array_values.asp) | Returns all the values of an array |
| [array\_walk()](http://www.w3schools.com/php/func_array_walk.asp) | Applies a user function to every member of an array |
| [array\_walk\_recursive()](http://www.w3schools.com/php/func_array_walk_recursive.asp) | Applies a user function recursively to every member of an array |
| [arsort()](http://www.w3schools.com/php/func_array_arsort.asp) | Sorts an associative array in descending order, according to the value |
| [asort()](http://www.w3schools.com/php/func_array_asort.asp) | Sorts an associative array in ascending order, according to the value |
| [compact()](http://www.w3schools.com/php/func_array_compact.asp) | Create array containing variables and their values |
| [count()](http://www.w3schools.com/php/func_array_count.asp) | Returns the number of elements in an array |
| [current()](http://www.w3schools.com/php/func_array_current.asp) | Returns the current element in an array |
| [each()](http://www.w3schools.com/php/func_array_each.asp) | Returns the current key and value pair from an array |
| [end()](http://www.w3schools.com/php/func_array_end.asp) | Sets the internal pointer of an array to its last element |
| [extract()](http://www.w3schools.com/php/func_array_extract.asp) | Imports variables into the current symbol table from an array |
| [in\_array()](http://www.w3schools.com/php/func_array_in_array.asp) | Checks if a specified value exists in an array |
| [key()](http://www.w3schools.com/php/func_array_key.asp) | Fetches a key from an array |
| [krsort()](http://www.w3schools.com/php/func_array_krsort.asp) | Sorts an associative array in descending order, according to the key |
| [ksort()](http://www.w3schools.com/php/func_array_ksort.asp) | Sorts an associative array in ascending order, according to the key |
| [list()](http://www.w3schools.com/php/func_array_list.asp) | Assigns variables as if they were an array |
| [natcasesort()](http://www.w3schools.com/php/func_array_natcasesort.asp) | Sorts an array using a case insensitive "natural order" algorithm |
| [natsort()](http://www.w3schools.com/php/func_array_natsort.asp) | Sorts an array using a "natural order" algorithm |
| [next()](http://www.w3schools.com/php/func_array_next.asp) | Advance the internal array pointer of an array |
| [pos()](http://www.w3schools.com/php/func_array_pos.asp) | Alias of current() |
| [prev()](http://www.w3schools.com/php/func_array_prev.asp) | Rewinds the internal array pointer |
| [range()](http://www.w3schools.com/php/func_array_range.asp) | Creates an array containing a range of elements |
| [reset()](http://www.w3schools.com/php/func_array_reset.asp) | Sets the internal pointer of an array to its first element |
| [rsort()](http://www.w3schools.com/php/func_array_rsort.asp) | Sorts an indexed array in descending order |
| [shuffle()](http://www.w3schools.com/php/func_array_shuffle.asp) | Shuffles an array |
| [sizeof()](http://www.w3schools.com/php/func_array_sizeof.asp) | Alias of count() |
| [sort()](http://www.w3schools.com/php/func_array_sort.asp) | Sorts an indexed array in ascending order |
| [uasort()](http://www.w3schools.com/php/func_array_uasort.asp) | Sorts an array by values using a user-defined comparison function |
| [uksort()](http://www.w3schools.com/php/func_array_uksort.asp) | Sorts an array by keys using a user-defined comparison function |
| [usort()](http://www.w3schools.com/php/func_array_usort.asp) | Sorts an array using a user-defined comparison function |

PHP 5 التقويم وظائف

PHP التقويم مقدمة

تمديد الجدول الزمني يتضمن الوظائف التي يبسط تحويل بين صيغ التقويم المختلفة.

**لأنه يقوم على عدد يوم جوليان، وهو عدد الأيام بدءا من 1 يناير، 4713 قبل الميلاد**

**ملاحظة:** لتحويل بين صيغ التقويم، يجب أولا تحويل لعدد يوم جوليان، ثم إلى تقويم من اختيارك.

**ملاحظة:** الكونت جوليان اليوم ليست هي نفس جوليان التقويم!

تركيب

لهذه الوظائف للعمل، لديك لتجميع PHP مع - تمكين التقويم.

نسخة من ويندوز PHP وقد بنيت في تقديم الدعم لهذا التمديد.

PHP 5 وظائف التقويم

|  |  |
| --- | --- |
| **Function** | **Description** |
| [cal\_days\_in\_month()](http://www.w3schools.com/php/func_cal_cal_days_in_month.asp) | Returns the number of days in a month for a specified year and calendar |
| [cal\_from\_jd()](http://www.w3schools.com/php/func_cal_cal_from_jd.asp) | Converts a Julian Day Count into a date of a specified calendar |
| [cal\_info()](http://www.w3schools.com/php/func_cal_cal_info.asp) | Returns information about a specified calendar |
| [cal\_to\_jd()](http://www.w3schools.com/php/func_cal_cal_to_jd.asp) | Converts a date in a specified calendar to Julian Day Count |
| [easter\_date()](http://www.w3schools.com/php/func_cal_easter_date.asp) | Returns the Unix timestamp for midnight on Easter of a specified year |
| [easter\_days()](http://www.w3schools.com/php/func_cal_easter_days.asp) | Returns the number of days after March 21, that the Easter Day is in a specified year |
| [frenchtojd()](http://www.w3schools.com/php/func_cal_frenchtojd.asp) | Converts a French Republican date to a Julian Day Count |
| [gregoriantojd()](http://www.w3schools.com/php/func_cal_gregoriantojd.asp) | Converts a Gregorian date to a Julian Day Count |
| [jddayofweek()](http://www.w3schools.com/php/func_cal_jddayofweek.asp) | Returns the day of the week |
| [jdmonthname()](http://www.w3schools.com/php/func_cal_jdmonthname.asp) | Returns a month name |
| [jdtofrench()](http://www.w3schools.com/php/func_cal_jdtofrench.asp) | Converts a Julian Day Count to a French Republican date |
| [jdtogregorian()](http://www.w3schools.com/php/func_cal_jdtogregorian.asp) | Converts a Julian Day Count to a Gregorian date |
| [jdtojewish()](http://www.w3schools.com/php/func_cal_jdtojewish.asp) | Converts a Julian Day Count to a Jewish date |
| [jdtojulian()](http://www.w3schools.com/php/func_cal_jdtojulian.asp) | Converts a Julian Day Count to a Julian date |
| [jdtounix()](http://www.w3schools.com/php/func_cal_jdtounix.asp) | Converts Julian Day Count to Unix timestamp |
| [jewishtojd()](http://www.w3schools.com/php/func_cal_jewishtojd.asp) | Converts a Jewish date to a Julian Day Count |
| [juliantojd()](http://www.w3schools.com/php/func_cal_juliantojd.asp) | Converts a Julian date to a Julian Day Count |
| [unixtojd()](http://www.w3schools.com/php/func_cal_unixtojd.asp) | Converts Unix timestamp to Julian Day Count |

PHP 5 محددة مسبقا التقويم الثوابت

|  |  |  |
| --- | --- | --- |
| **Constant** | **Type** | **PHP Version** |
| CAL\_GREGORIAN | Integer | PHP 4 |
| CAL\_JULIAN | Integer | PHP 4 |
| CAL\_JEWISH | Integer | PHP 4 |
| CAL\_FRENCH | Integer | PHP 4 |
| CAL\_NUM\_CALS | Integer | PHP 4 |
| CAL\_DOW\_DAYNO | Integer | PHP 4 |
| CAL\_DOW\_SHORT | Integer | PHP 4 |
| CAL\_DOW\_LONG | Integer | PHP 4 |
| CAL\_MONTH\_GREGORIAN\_SHORT | Integer | PHP 4 |
| CAL\_MONTH\_GREGORIAN\_LONG | Integer | PHP 4 |
| CAL\_MONTH\_JULIAN\_SHORT | Integer | PHP 4 |
| CAL\_MONTH\_JULIAN\_LONG | Integer | PHP 4 |
| CAL\_MONTH\_JEWISH | Integer | PHP 4 |
| CAL\_MONTH\_FRENCH | Integer | PHP 4 |
| CAL\_EASTER\_DEFAULT | Integer | PHP 4.3 |
| CAL\_EASTER\_ROMAN | Integer | PHP 4.3 |
| CAL\_EASTER\_ALWAYS\_GREGORIAN | Integer | PHP 4.3 |
| CAL\_EASTER\_ALWAYS\_JULIAN | Integer | PHP 4.3 |
| CAL\_JEWISH\_ADD\_ALAFIM\_GERESH | Integer | PHP 5.0 |
| CAL\_JEWISH\_ADD\_ALAFIM | Integer | PHP 5.0 |
| CAL\_JEWISH\_ADD\_GERESHAYIM | Integer | PHP 5.0 |

PHP 5 تاريخ / وقت ظائف

[«السابق](http://www.w3schools.com/php/php_ref_calendar.asp)

[الفصل التالي »](http://www.w3schools.com/php/php_ref_directory.asp)

PHP وقت / زمن مقدمة

وظائف التاريخ / الوقت تسمح لك للحصول على التاريخ والوقت من الخادم حيث يعمل البرنامج النصي PHP الخاصة بك. يمكنك بعد ذلك استخدام وظائف التاريخ / الوقت لتنسيق التاريخ والوقت في عدة طرق.

**ملاحظة:** تعتمد هذه الوظائف على الإعدادات المحلية الملقم الخاص بك. تذكر أن تأخذ التوقيت الادخار وسنة كبيسة في الاعتبار عند التعامل مع هذه المهام.

تركيب

وظائف التاريخ والوقت PHP / هي جزء من جوهر PHP. لا يلزم تركيب لاستخدام هذه الوظائف.

تكوين وقت التشغيل

يتأثر سلوك هذه الوظائف من خلال إعدادات ملف php.ini في:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Description** | **Default** | **PHP Version** |
| date.timezone | The default timezone (used by all date/time functions) | "" | PHP 5.1 |
| date.default\_latitude | The default latitude (used by date\_sunrise() and date\_sunset()) | "31.7667" | PHP 5.0 |
| date.default\_longitude | The default longitude (used by date\_sunrise() and date\_sunset()) | "35.2333" | PHP 5.0 |
| date.sunrise\_zenith | The default sunrise zenith (used by date\_sunrise() and date\_sunset()) | "90.83" | PHP 5.0 |
| date.sunset\_zenith | The default sunset zenith (used by date\_sunrise() and date\_sunset()) | "90.83" | PHP 5.0 |

وظائف PHP 5 تاريخ / وقت

|  |  |
| --- | --- |
| **Function** | **Description** |
| [checkdate()](http://www.w3schools.com/php/func_date_checkdate.asp) | Validates a Gregorian date |
| [date\_add()](http://www.w3schools.com/php/func_date_date_add.asp) | Adds days, months, years, hours, minutes, and seconds to a date |
| [date\_create\_from\_format()](http://www.w3schools.com/php/func_date_create_from_format.asp) | Returns a new DateTime object formatted according to a specified format |
| [date\_create()](http://www.w3schools.com/php/func_date_date_create.asp) | Returns a new DateTime object |
| [date\_date\_set()](http://www.w3schools.com/php/func_date_date_set.asp) | Sets a new date |
| [date\_default\_timezone\_get()](http://www.w3schools.com/php/func_date_default_timezone_get.asp) | Returns the default timezone used by all date/time functions |
| [date\_default\_timezone\_set()](http://www.w3schools.com/php/func_date_default_timezone_set.asp) | Sets the default timezone used by all date/time functions |
| [date\_diff()](http://www.w3schools.com/php/func_date_date_diff.asp) | Returns the difference between two dates |
| [date\_format()](http://www.w3schools.com/php/func_date_date_format.asp) | Returns a date formatted according to a specified format |
| [date\_get\_last\_errors()](http://www.w3schools.com/php/func_date_get_last_errors.asp) | Returns the warnings/errors found in a date string |
| date\_interval\_create\_from\_date\_string() | Sets up a DateInterval from the relative parts of the string |
| [date\_interval\_format()](http://www.w3schools.com/php/func_date_interval_format.asp) | Formats the interval |
| [date\_isodate\_set()](http://www.w3schools.com/php/func_date_isodate_set.asp) | Sets the ISO date |
| [date\_modify()](http://www.w3schools.com/php/func_date_modify.asp) | Modifies the timestamp |
| [date\_offset\_get()](http://www.w3schools.com/php/func_date_offset_get.asp) | Returns the timezone offset |
| [date\_parse\_from\_format()](http://www.w3schools.com/php/func_date_parse_from_format.asp) | Returns an associative array with detailed info about a specified date, according to a specified format |
| [date\_parse()](http://www.w3schools.com/php/func_date_date_parse.asp) | Returns an associative array with detailed info about a specified date |
| [date\_sub()](http://www.w3schools.com/php/func_date_date_sub.asp) | Subtracts days, months, years, hours, minutes, and seconds from a date |
| [date\_sun\_info()](http://www.w3schools.com/php/func_date_sun_info.asp) | Returns an array containing info about sunset/sunrise and twilight begin/end, for a specified day and location |
| [date\_sunrise()](http://www.w3schools.com/php/func_date_sunrise.asp) | Returns the sunrise time for a specified day and location |
| [date\_sunset()](http://www.w3schools.com/php/func_date_sunset.asp) | Returns the sunset time for a specified day and location |
| [date\_time\_set()](http://www.w3schools.com/php/func_date_time_set.asp) | Sets the time |
| [date\_timestamp\_get()](http://www.w3schools.com/php/func_date_timestamp_get.asp) | Returns the Unix timestamp |
| [date\_timestamp\_set()](http://www.w3schools.com/php/func_date_timestamp_set.asp) | Sets the date and time based on a Unix timestamp |
| [date\_timezone\_get()](http://www.w3schools.com/php/func_date_timezone_get.asp) | Returns the time zone of the given DateTime object |
| [date\_timezone\_set()](http://www.w3schools.com/php/func_date_timezone_set.asp) | Sets the time zone for the DateTime object |
| [date()](http://www.w3schools.com/php/func_date_date.asp) | Formats a local date and time |
| [getdate()](http://www.w3schools.com/php/func_date_getdate.asp) | Returns date/time information of a timestamp or the current local date/time |
| [gettimeofday()](http://www.w3schools.com/php/func_date_gettimeofday.asp) | Returns the current time |
| [gmdate()](http://www.w3schools.com/php/func_date_gmdate.asp) | Formats a GMT/UTC date and time |
| [gmmktime()](http://www.w3schools.com/php/func_date_gmmktime.asp) | Returns the Unix timestamp for a GMT date |
| [gmstrftime()](http://www.w3schools.com/php/func_date_gmstrftime.asp) | Formats a GMT/UTC date and time according to locale settings |
| [idate()](http://www.w3schools.com/php/func_date_idate.asp) | Formats a local time/date as integer |
| [localtime()](http://www.w3schools.com/php/func_date_localtime.asp) | Returns the local time |
| [microtime()](http://www.w3schools.com/php/func_date_microtime.asp) | Returns the current Unix timestamp with microseconds |
| [mktime()](http://www.w3schools.com/php/func_date_mktime.asp) | Returns the Unix timestamp for a date |
| [strftime()](http://www.w3schools.com/php/func_date_strftime.asp) | Formats a local time and/or date according to locale settings |
| [strptime()](http://www.w3schools.com/php/func_date_strptime.asp) | Parses a time/date generated with strftime() |
| [strtotime()](http://www.w3schools.com/php/func_date_strtotime.asp) | Parses an English textual datetime into a Unix timestamp |
| [time()](http://www.w3schools.com/php/func_date_time.asp) | Returns the current time as a Unix timestamp |
| [timezone\_abbreviations\_list()](http://www.w3schools.com/php/func_date_timezone_abbreviations_list.asp) | Returns an associative array containing dst, offset, and the timezone name |
| [timezone\_identifiers\_list()](http://www.w3schools.com/php/func_date_timezone_identifiers_list.asp) | Returns an indexed array with all timezone identifiers |
| [timezone\_location\_get()](http://www.w3schools.com/php/func_date_timezone_location_get.asp) | Returns location information for a specified timezone |
| [timezone\_name\_from\_ abbr()](http://www.w3schools.com/php/func_date_timezone_name_from_abbr.asp) | Returns the timezone name from abbreviation |
| [timezone\_name\_get()](http://www.w3schools.com/php/func_date_timezone_name_get.asp) | Returns the name of the timezone |
| [timezone\_offset\_get()](http://www.w3schools.com/php/func_date_timezone_offset_get.asp) | Returns the timezone offset from GMT |
| [timezone\_open()](http://www.w3schools.com/php/func_date_timezone_open.asp) | Creates new DateTimeZone object |
| timezone\_transitions\_get() | Returns all transitions for the timezone |
| [timezone\_version\_get()](http://www.w3schools.com/php/func_date_timezone_version_get.asp) | Returns the version of the timezone db |

PHP 5 محددة مسبقا وقت / زمن الثوابت

|  |  |
| --- | --- |
| **Constant** | **Description** |
| DATE\_ATOM | Atom (example: 2005-08-15T16:13:03+0000) |
| DATE\_COOKIE | HTTP Cookies (example: Sun, 14 Aug 2005 16:13:03 UTC) |
| DATE\_ISO8601 | ISO-8601 (example: 2005-08-14T16:13:03+0000) |
| DATE\_RFC822 | RFC 822 (example: Sun, 14 Aug 2005 16:13:03 UTC) |
| DATE\_RFC850 | RFC 850 (example: Sunday, 14-Aug-05 16:13:03 UTC) |
| DATE\_RFC1036 | RFC 1036 (example: Sunday, 14-Aug-05 16:13:03 UTC) |
| DATE\_RFC1123 | RFC 1123 (example: Sun, 14 Aug 2005 16:13:03 UTC) |
| DATE\_RFC2822 | RFC 2822 (Sun, 14 Aug 2005 16:13:03 +0000) |
| DATE\_RSS | RSS (Sun, 14 Aug 2005 16:13:03 UTC) |
| DATE\_W3C | World Wide Web Consortium (example: 2005-08-14T16:13:03+0000) |

PHP 5 دليل وظائف

PHP دليل مقدمة

وظائف الدليل تسمح لك لاسترداد المعلومات حول الدلائل ومحتوياتها.

تركيب

وظائف دليل PHP هي جزء من جوهر PHP. لا يلزم تركيب لاستخدام هذه الوظائف.

PHP 5 وظائف دليل

|  |  |
| --- | --- |
| **Function** | **Description** |
| [chdir()](http://www.w3schools.com/php/func_directory_chdir.asp) | Changes the current directory |
| [chroot()](http://www.w3schools.com/php/func_directory_chroot.asp) | Changes the root directory |
| [closedir()](http://www.w3schools.com/php/func_directory_closedir.asp) | Closes a directory handle |
| [dir()](http://www.w3schools.com/php/func_directory_dir.asp) | Returns an instance of the Directory class |
| [getcwd()](http://www.w3schools.com/php/func_directory_getcwd.asp) | Returns the current working directory |
| [opendir()](http://www.w3schools.com/php/func_directory_opendir.asp) | Opens a directory handle |
| [readdir()](http://www.w3schools.com/php/func_directory_readdir.asp) | Returns an entry from a directory handle |
| [rewinddir()](http://www.w3schools.com/php/func_directory_rewinddir.asp) | Resets a directory handle |
| [scandir()](http://www.w3schools.com/php/func_directory_scandir.asp) | Returns an array of files and directories of a specified directory |

PHP خطأ وتسجيل وظائف

خطأ PHP وتسجيل مقدمة

الخطأ وظائف تسجيل يسمح معالجة الخطأ وقطع الأشجار.

وظائف الخطأ تسمح للمستخدمين لتحديد قواعد معالجة الخطأ، وتعديل الطريقة يمكن تسجيل الأخطاء.

وظائف تسمح للمستخدمين تسجيل الدخول إلى التطبيقات وإرسال رسائل سجل إلى البريد الإلكتروني وسجلات النظام أو أجهزة أخرى.

تركيب

وظائف الخطأ وقطع الأشجار هي جزء من جوهر PHP. ليس هناك حاجة لتركيب لاستخدام هذه الوظائف.

خطأ PHP وتسجيل وظائف

**PHP** : يشير إلى الإصدار الأقدم من PHP التي تدعم وظيفة.

|  |  |  |
| --- | --- | --- |
| **Function** | **Description** | **PHP** |
| [debug\_backtrace()](http://www.w3schools.com/php/func_error_debug_backtrace.asp) | Generates a backtrace | 4 |
| [debug\_print\_backtrace()](http://www.w3schools.com/php/func_error_debug_print_backtrace.asp) | Prints a backtrace | 5 |
| [error\_get\_last()](http://www.w3schools.com/php/func_error_get_last.asp) | Gets the last error occurred | 5 |
| [error\_log()](http://www.w3schools.com/php/func_error_log.asp) | Sends an error to the server error-log, to a file or to a remote destination | 4 |
| [error\_reporting()](http://www.w3schools.com/php/func_error_reporting.asp) | Specifies which errors are reported | 4 |
| [restore\_error\_handler()](http://www.w3schools.com/php/func_error_restore_error_handler.asp) | Restores the previous error handler | 4 |
| [restore\_exception\_handler()](http://www.w3schools.com/php/func_error_restore_exception_handler.asp) | Restores the previous exception handler | 5 |
| [set\_error\_handler()](http://www.w3schools.com/php/func_error_set_error_handler.asp) | Sets a user-defined function to handle errors | 4 |
| [set\_exception\_handler()](http://www.w3schools.com/php/func_error_set_exception_handler.asp) | Sets a user-defined function to handle exceptions | 5 |
| [trigger\_error()](http://www.w3schools.com/php/func_error_trigger_error.asp) | Creates a user-defined error message | 4 |
| [user\_error()](http://www.w3schools.com/php/func_error_user_error.asp) | Alias of trigger\_error() | 4 |

خطأ PHP والثوابت تسجيل

**PHP** : يشير إلى الإصدار الأقدم من PHP التي تدعم ثابت.

|  |  |  |  |
| --- | --- | --- | --- |
| **Value** | **Constant** | **Description** | **PHP** |
| 1 | E\_ERROR | Fatal run-time errors. Errors that cannot be recovered from. Execution of the script is halted |  |
| 2 | E\_WARNING | Non-fatal run-time errors. Execution of the script is not halted |  |
| 4 | E\_PARSE | Compile-time parse errors. Parse errors should only be generated by the parser |  |
| 8 | E\_NOTICE | Run-time notices. The script found something that might be an error, but could also happen when running a script normally |  |
| 16 | E\_CORE\_ERROR | Fatal errors at PHP startup. This is like an E\_ERROR in the PHP core | 4 |
| 32 | E\_CORE\_WARNING | Non-fatal errors at PHP startup. This is like an E\_WARNING in the PHP core | 4 |
| 64 | E\_COMPILE\_ERROR | Fatal compile-time errors. This is like an E\_ERROR generated by the Zend Scripting Engine | 4 |
| 128 | E\_COMPILE\_WARNING | Non-fatal compile-time errors. This is like an E\_WARNING generated by the Zend Scripting Engine | 4 |
| 256 | E\_USER\_ERROR | Fatal user-generated error. This is like an E\_ERROR set by the programmer using the PHP function trigger\_error() | 4 |
| 512 | E\_USER\_WARNING | Non-fatal user-generated warning. This is like an E\_WARNING set by the programmer using the PHP function trigger\_error() | 4 |
| 1024 | E\_USER\_NOTICE | User-generated notice. This is like an E\_NOTICE set by the programmer using the PHP function trigger\_error() | 4 |
| 2048 | E\_STRICT | Run-time notices. PHP suggest changes to your code to help interoperability and compatibility of the code | 5 |
| 4096 | E\_RECOVERABLE\_ERROR | Catchable fatal error. This is like an E\_ERROR but can be caught by a user defined handle (see also set\_error\_handler()) | 5 |
| 6143 | E\_ALL | All errors and warnings, except of level E\_STRICT | 5 |

PHP 5 نظام الملفات وظائف

PHP نظام الملفات مقدمة

وظائف نظام الملفات تسمح لك للوصول والتعامل مع الملفات.

تركيب

وظائف نظام الملفات هي جزء من جوهر PHP. ليس هناك حاجة لتركيب لاستخدام هذه الوظائف.

تكوين وقت التشغيل

يتأثر سلوك ظائف نظام الملفات من خلال إعدادات ملف php.ini في.

خيارات التكوين الملفات:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Default** | **Description** | **Changeable** |
| allow\_url\_fopen | "1" | Allows fopen()-type functions to work with URLs (available since PHP 4.0.4) | PHP\_INI\_SYSTEM |
| user\_agent | NULL | Defines the user agent for PHP to send (available since PHP 4.3) | PHP\_INI\_ALL |
| default\_socket\_timeout | "60" | Sets the default timeout, in seconds, for socket based streams (available since PHP 4.3) | PHP\_INI\_ALL |
| from | "" | Defines the anonymous FTP password (your email address) | PHP\_INI\_ALL |
| auto\_detect\_line\_endings | "0" | When set to "1", PHP will examine the data read by fgets() and file() to see if it is using Unix, MS-Dos or Mac line-ending characters (available since PHP 4.3) | PHP\_INI\_ALL |

يونيكس / ويندوز التوافق

عند تحديد مسار على منصات يونيكس، يتم استخدام خط مائل (/) كفاصل الدليل. ومع ذلك، على منصات ويندوز، سواء خط مائل (/) ومائل (\) يمكن استخدامها.

PHP 5 وظائف نظام الملفات

|  |  |
| --- | --- |
| **Function** | **Description** |
| [basename()](http://www.w3schools.com/php/func_filesystem_basename.asp) | Returns the filename component of a path |
| [chgrp()](http://www.w3schools.com/php/func_filesystem_chgrp.asp) | Changes the file group |
| [chmod()](http://www.w3schools.com/php/func_filesystem_chmod.asp) | Changes the file mode |
| [chown()](http://www.w3schools.com/php/func_filesystem_chown.asp) | Changes the file owner |
| [clearstatcache()](http://www.w3schools.com/php/func_filesystem_clearstatcache.asp) | Clears the file status cache |
| [copy()](http://www.w3schools.com/php/func_filesystem_copy.asp) | Copies a file |
| delete() | See unlink() or unset() |
| [dirname()](http://www.w3schools.com/php/func_filesystem_dirname.asp) | Returns the directory name component of a path |
| [disk\_free\_space()](http://www.w3schools.com/php/func_filesystem_disk_free_space.asp) | Returns the free space of a directory |
| [disk\_total\_space()](http://www.w3schools.com/php/func_filesystem_disk_total_space.asp) | Returns the total size of a directory |
| [diskfreespace()](http://www.w3schools.com/php/func_filesystem_diskfreespace.asp) | Alias of disk\_free\_space() |
| [fclose()](http://www.w3schools.com/php/func_filesystem_fclose.asp) | Closes an open file |
| [feof()](http://www.w3schools.com/php/func_filesystem_feof.asp) | Tests for end-of-file on an open file |
| [fflush()](http://www.w3schools.com/php/func_filesystem_fflush.asp) | Flushes buffered output to an open file |
| [fgetc()](http://www.w3schools.com/php/func_filesystem_fgetc.asp) | Returns a character from an open file |
| [fgetcsv()](http://www.w3schools.com/php/func_filesystem_fgetcsv.asp) | Parses a line from an open file, checking for CSV fields |
| [fgets()](http://www.w3schools.com/php/func_filesystem_fgets.asp) | Returns a line from an open file |
| [fgetss()](http://www.w3schools.com/php/func_filesystem_fgetss.asp) | Returns a line, with HTML and PHP tags removed, from an open file |
| [file()](http://www.w3schools.com/php/func_filesystem_file.asp) | Reads a file into an array |
| [file\_exists()](http://www.w3schools.com/php/func_filesystem_file_exists.asp) | Checks whether or not a file or directory exists |
| [file\_get\_contents()](http://www.w3schools.com/php/func_filesystem_file_get_contents.asp) | Reads a file into a string |
| [file\_put\_contents](http://www.w3schools.com/php/func_filesystem_file_put_contents.asp) | Writes a string to a file |
| [fileatime()](http://www.w3schools.com/php/func_filesystem_fileatime.asp) | Returns the last access time of a file |
| [filectime()](http://www.w3schools.com/php/func_filesystem_filectime.asp) | Returns the last change time of a file |
| [filegroup()](http://www.w3schools.com/php/func_filesystem_filegroup.asp) | Returns the group ID of a file |
| [fileinode()](http://www.w3schools.com/php/func_filesystem_fileinode.asp) | Returns the inode number of a file |
| [filemtime()](http://www.w3schools.com/php/func_filesystem_filemtime.asp) | Returns the last modification time of a file |
| [fileowner()](http://www.w3schools.com/php/func_filesystem_fileowner.asp) | Returns the user ID (owner) of a file |
| [fileperms()](http://www.w3schools.com/php/func_filesystem_fileperms.asp) | Returns the permissions of a file |
| [filesize()](http://www.w3schools.com/php/func_filesystem_filesize.asp) | Returns the file size |
| [filetype()](http://www.w3schools.com/php/func_filesystem_filetype.asp) | Returns the file type |
| [flock()](http://www.w3schools.com/php/func_filesystem_flock.asp) | Locks or releases a file |
| [fnmatch()](http://www.w3schools.com/php/func_filesystem_fnmatch.asp) | Matches a filename or string against a specified pattern |
| [fopen()](http://www.w3schools.com/php/func_filesystem_fopen.asp) | Opens a file or URL |
| [fpassthru()](http://www.w3schools.com/php/func_filesystem_fpassthru.asp) | Reads from an open file, until EOF, and writes the result to the output buffer |
| [fputcsv()](http://www.w3schools.com/php/func_filesystem_fputcsv.asp) | Formats a line as CSV and writes it to an open file |
| [fputs()](http://www.w3schools.com/php/func_filesystem_fputs.asp) | Alias of fwrite() |
| [fread()](http://www.w3schools.com/php/func_filesystem_fread.asp) | Reads from an open file |
| [fscanf()](http://www.w3schools.com/php/func_filesystem_fscanf.asp) | Parses input from an open file according to a specified format |
| [fseek()](http://www.w3schools.com/php/func_filesystem_fseek.asp) | Seeks in an open file |
| [fstat()](http://www.w3schools.com/php/func_filesystem_fstat.asp) | Returns information about an open file |
| [ftell()](http://www.w3schools.com/php/func_filesystem_ftell.asp) | Returns the current position in an open file |
| [ftruncate()](http://www.w3schools.com/php/func_filesystem_ftruncate.asp) | Truncates an open file to a specified length |
| [fwrite()](http://www.w3schools.com/php/func_filesystem_fwrite.asp) | Writes to an open file |
| [glob()](http://www.w3schools.com/php/func_filesystem_glob.asp) | Returns an array of filenames / directories matching a specified pattern |
| [is\_dir()](http://www.w3schools.com/php/func_filesystem_is_dir.asp) | Checks whether a file is a directory |
| [is\_executable()](http://www.w3schools.com/php/func_filesystem_is_executable.asp) | Checks whether a file is executable |
| [is\_file()](http://www.w3schools.com/php/func_filesystem_is_file.asp) | Checks whether a file is a regular file |
| [is\_link()](http://www.w3schools.com/php/func_filesystem_is_link.asp) | Checks whether a file is a link |
| [is\_readable()](http://www.w3schools.com/php/func_filesystem_is_readable.asp) | Checks whether a file is readable |
| [is\_uploaded\_file()](http://www.w3schools.com/php/func_filesystem_is_uploaded_file.asp) | Checks whether a file was uploaded via HTTP POST |
| [is\_writable()](http://www.w3schools.com/php/func_filesystem_is_writable.asp) | Checks whether a file is writeable |
| [is\_writeable()](http://www.w3schools.com/php/func_filesystem_is_writeable.asp) | Alias of is\_writable() |
| lchgrp() | Changes group ownership of symlink |
| lchown() | Changes user ownership of symlink |
| [link()](http://www.w3schools.com/php/func_filesystem_link.asp) | Creates a hard link |
| [linkinfo()](http://www.w3schools.com/php/func_filesystem_linkinfo.asp) | Returns information about a hard link |
| [lstat()](http://www.w3schools.com/php/func_filesystem_lstat.asp) | Returns information about a file or symbolic link |
| [mkdir()](http://www.w3schools.com/php/func_filesystem_mkdir.asp) | Creates a directory |
| [move\_uploaded\_file()](http://www.w3schools.com/php/func_filesystem_move_uploaded_file.asp) | Moves an uploaded file to a new location |
| [parse\_ini\_file()](http://www.w3schools.com/php/func_filesystem_parse_ini_file.asp) | Parses a configuration file |
| parse\_ini\_string() | Parses a configuration string |
| [pathinfo()](http://www.w3schools.com/php/func_filesystem_pathinfo.asp) | Returns information about a file path |
| [pclose()](http://www.w3schools.com/php/func_filesystem_pclose.asp) | Closes a pipe opened by popen() |
| [popen()](http://www.w3schools.com/php/func_filesystem_popen.asp) | Opens a pipe |
| [readfile()](http://www.w3schools.com/php/func_filesystem_readfile.asp) | Reads a file and writes it to the output buffer |
| [readlink()](http://www.w3schools.com/php/func_filesystem_readlink.asp) | Returns the target of a symbolic link |
| [realpath()](http://www.w3schools.com/php/func_filesystem_realpath.asp) | Returns the absolute pathname |
| realpath\_cache\_get() | Returns realpath cache entries |
| realpath\_cache\_size() | Returns realpath cache size |
| [rename()](http://www.w3schools.com/php/func_filesystem_rename.asp) | Renames a file or directory |
| [rewind()](http://www.w3schools.com/php/func_filesystem_rewind.asp) | Rewinds a file pointer |
| [rmdir()](http://www.w3schools.com/php/func_filesystem_rmdir.asp) | Removes an empty directory |
| [set\_file\_buffer()](http://www.w3schools.com/php/func_filesystem_set_file_buffer.asp) | Sets the buffer size of an open file |
| [stat()](http://www.w3schools.com/php/func_filesystem_stat.asp) | Returns information about a file |
| [symlink()](http://www.w3schools.com/php/func_filesystem_symlink.asp) | Creates a symbolic link |
| [tempnam()](http://www.w3schools.com/php/func_filesystem_tempnam.asp) | Creates a unique temporary file |
| [tmpfile()](http://www.w3schools.com/php/func_filesystem_tmpfile.asp) | Creates a unique temporary file |
| [touch()](http://www.w3schools.com/php/func_filesystem_touch.asp) | Sets access and modification time of a file |
| [umask()](http://www.w3schools.com/php/func_filesystem_umask.asp) | Changes file permissions for files |
| [unlink()](http://www.w3schools.com/php/func_filesystem_unlink.asp) | Deletes a file |

PHP تصفية وظائف

PHP تصفية مقدمة

يتم استخدام هذه الفلاتر PHP للتحقق من صحة وتصفية البيانات القادمة من مصادر غير آمنة، مثل إدخال المستخدم.

تركيب

وظائف التصفية هي جزء من جوهر PHP. ليس هناك حاجة لتركيب لاستخدام هذه الوظائف.

PHP تصفية وظائف

**PHP** : يشير إلى الإصدار الأقدم من PHP التي تدعم وظيفة.

|  |  |  |
| --- | --- | --- |
| **Function** | **Description** | **PHP** |
| [filter\_has\_var()](http://www.w3schools.com/php/func_filter_has_var.asp) | Checks if a variable of a specified input type exist | 5 |
| [filter\_id()](http://www.w3schools.com/php/func_filter_id.asp) | Returns the ID number of a specified filter | 5 |
| [filter\_input()](http://www.w3schools.com/php/func_filter_input.asp) | Get input from outside the script and filter it | 5 |
| [filter\_input\_array()](http://www.w3schools.com/php/func_filter_input_array.asp) | Get multiple inputs from outside the script and filters them | 5 |
| [filter\_list()](http://www.w3schools.com/php/func_filter_list.asp) | Returns an array of all supported filters | 5 |
| [filter\_var\_array()](http://www.w3schools.com/php/func_filter_var_array.asp) | Get multiple variables and filter them | 5 |
| [filter\_var()](http://www.w3schools.com/php/func_filter_var.asp) | Get a variable and filter it | 5 |

PHP الفلاتر

|  |  |
| --- | --- |
| **ID Name** | **Description** |
| [FILTER\_CALLBACK](http://www.w3schools.com/php/filter_callback.asp) | Call a user-defined function to filter data |
| [FILTER\_SANITIZE\_STRING](http://www.w3schools.com/php/filter_sanitize_string.asp) | Strip tags, optionally strip or encode special characters |
| [FILTER\_SANITIZE\_STRIPPED](http://www.w3schools.com/php/filter_sanitize_stripped.asp) | Alias of "string" filter |
| [FILTER\_SANITIZE\_ENCODED](http://www.w3schools.com/php/filter_sanitize_encoded.asp) | URL-encode string, optionally strip or encode special characters |
| [FILTER\_SANITIZE\_SPECIAL\_CHARS](http://www.w3schools.com/php/filter_sanitize_special_chars.asp) | HTML-escape '"<>& and characters with ASCII value less than 32 |
| [FILTER\_SANITIZE\_EMAIL](http://www.w3schools.com/php/filter_sanitize_email.asp) | Remove all characters, except letters, digits and !#$%&'\*+-/=?^\_`{|}~@.[] |
| [FILTER\_SANITIZE\_URL](http://www.w3schools.com/php/filter_sanitize_url.asp) | Remove all characters, except letters, digits and $-\_.+!\*'(),{}|\\^~[]`<>#%";/?:@&= |
| [FILTER\_SANITIZE\_NUMBER\_INT](http://www.w3schools.com/php/filter_sanitize_number_int.asp) | Remove all characters, except digits and +- |
| [FILTER\_SANITIZE\_NUMBER\_FLOAT](http://www.w3schools.com/php/filter_sanitize_number_float.asp) | Remove all characters, except digits, +- and optionally .,eE |
| [FILTER\_SANITIZE\_MAGIC\_QUOTES](http://www.w3schools.com/php/filter_sanitize_magic_quotes.asp) | Apply addslashes() |
| [FILTER\_UNSAFE\_RAW](http://www.w3schools.com/php/filter_unsafe_raw.asp) | Do nothing, optionally strip or encode special characters |
| [FILTER\_VALIDATE\_INT](http://www.w3schools.com/php/filter_validate_int.asp) | Validate value as integer, optionally from the specified range |
| [FILTER\_VALIDATE\_BOOLEAN](http://www.w3schools.com/php/filter_validate_boolean.asp) | Return TRUE for "1", "true", "on" and "yes", FALSE for "0", "false", "off", "no", and "", NULL otherwise |
| [FILTER\_VALIDATE\_FLOAT](http://www.w3schools.com/php/filter_validate_float.asp) | Validate value as float |
| [FILTER\_VALIDATE\_REGEXP](http://www.w3schools.com/php/filter_validate_regexp.asp) | Validate value against regexp, a Perl-compatible regular expression |
| [FILTER\_VALIDATE\_URL](http://www.w3schools.com/php/filter_validate_url.asp) | Validate value as URL, optionally with required components |
| [FILTER\_VALIDATE\_EMAIL](http://www.w3schools.com/php/filter_validate_email.asp) | Validate value as e-mail |
| [FILTER\_VALIDATE\_IP](http://www.w3schools.com/php/filter_validate_ip.asp) | Validate value as IP address, optionally only IPv4 or IPv6 or not from private or reserved ranges |

PHP FTP وظائف

PHP FTP مقدمة

وظائف FTP تعطي العميل إمكانية الوصول إلى خوادم الملفات من خلال بروتوكول نقل الملفات (FTP).

وتستخدم وظائف FTP لفتح، وتسجيل الدخول وثيقة وصلات، وكذلك تحميل وتنزيل، وإعادة تسمية، حذف، والحصول على معلومات حول الملفات من خوادم الملفات. ليس كل من وظائف FTP ستعمل مع كل خادم أو العودة على نفس النتائج. أصبح ظائف FTP المتاحة مع PHP 3.

وتهدف هذه الوظائف للوصول مفصلة إلى ملقم FTP. إذا كنت ترغب فقط في القراءة من أو الكتابة إلى ملف على ملقم بروتوكول نقل الملفات، والنظر في استخدام المجمع ftp:// مع وظائف نظام الملفات.

تركيب

نسخة ويندوز من PHP وقد بنيت في تقديم الدعم لتمديد بروتوكول نقل الملفات. لذلك، فإن وظائف FTP تعمل تلقائيا.

ومع ذلك، إذا كنت تقوم بتشغيل نسخة لينكس من PHP، سيكون لديك لتجميع PHP مع *- تمكين بروتوكول نقل الملفات* (PHP 4 +) أو *- مع بروتوكول نقل الملفات* (PHP 3) للحصول على وظائف FTP للعمل.

وظائف PHP FTP

**PHP** : يشير إلى الإصدار الأقدم من PHP التي تدعم وظيفة.

|  |  |  |
| --- | --- | --- |
| **Function** | **Description** | **PHP** |
| [ftp\_alloc()](http://www.w3schools.com/php/func_ftp_alloc.asp) | Allocates space for a file to be uploaded to the FTP server | 5 |
| [ftp\_cdup()](http://www.w3schools.com/php/func_ftp_cdup.asp) | Changes the current directory to the parent directory on the FTP server | 3 |
| [ftp\_chdir()](http://www.w3schools.com/php/func_ftp_chdir.asp) | Changes the current directory on the FTP server | 3 |
| [ftp\_chmod()](http://www.w3schools.com/php/func_ftp_chmod.asp) | Sets permissions on a file via FTP | 5 |
| [ftp\_close()](http://www.w3schools.com/php/func_ftp_close.asp) | Closes an FTP connection | 4 |
| [ftp\_connect()](http://www.w3schools.com/php/func_ftp_connect.asp) | Opens an FTP connection | 3 |
| [ftp\_delete()](http://www.w3schools.com/php/func_ftp_delete.asp) | Deletes a file on the FTP server | 3 |
| [ftp\_exec()](http://www.w3schools.com/php/func_ftp_exec.asp) | Executes a program/command on the FTP server | 4 |
| [ftp\_fget()](http://www.w3schools.com/php/func_ftp_fget.asp) | Downloads a file from the FTP server and saves it to an open file | 3 |
| [ftp\_fput()](http://www.w3schools.com/php/func_ftp_fput.asp) | Uploads from an open file and saves it to a file on the FTP server | 3 |
| [ftp\_get\_option()](http://www.w3schools.com/php/func_ftp_get_option.asp) | Returns runtime behaviors of the FTP connection | 4 |
| [ftp\_get()](http://www.w3schools.com/php/func_ftp_get.asp) | Downloads a file from the FTP server | 3 |
| [ftp\_login()](http://www.w3schools.com/php/func_ftp_login.asp) | Logs on to an FTP connection | 3 |
| [ftp\_mdtm()](http://www.w3schools.com/php/func_ftp_mdtm.asp) | Returns the last modified time of a specified file | 3 |
| [ftp\_mkdir()](http://www.w3schools.com/php/func_ftp_mkdir.asp) | Creates a new directory on the FTP server | 3 |
| [ftp\_nb\_continue()](http://www.w3schools.com/php/func_ftp_nb_continue.asp) | Continues retrieving/sending a file (non-blocking) | 4 |
| [ftp\_nb\_fget()](http://www.w3schools.com/php/func_ftp_nb_fget.asp) | Downloads a file from the FTP server and saves it to an open file (non-blocking) | 4 |
| [ftp\_nb\_fput()](http://www.w3schools.com/php/func_ftp_nb_fput.asp) | Uploads from an open file and saves it to a file on the FTP server (non-blocking) | 4 |
| [ftp\_nb\_get()](http://www.w3schools.com/php/func_ftp_nb_get.asp) | Downloads a file from the FTP server (non-blocking) | 4 |
| [ftp\_nb\_put()](http://www.w3schools.com/php/func_ftp_nb_put.asp) | Uploads a file to the FTP server (non-blocking) | 4 |
| [ftp\_nlist()](http://www.w3schools.com/php/func_ftp_nlist.asp) | Lists the files in a specified directory on the FTP server | 3 |
| [ftp\_pasv()](http://www.w3schools.com/php/func_ftp_pasv.asp) | Turns passive mode on or off | 3 |
| [ftp\_put()](http://www.w3schools.com/php/func_ftp_put.asp) | Uploads a file to the FTP server | 3 |
| [ftp\_pwd()](http://www.w3schools.com/php/func_ftp_pwd.asp) | Returns the current directory name | 3 |
| [ftp\_quit()](http://www.w3schools.com/php/func_ftp_quit.asp) | Alias of ftp\_close() | 3 |
| [ftp\_raw()](http://www.w3schools.com/php/func_ftp_raw.asp) | Sends a raw command to the FTP server | 5 |
| [ftp\_rawlist()](http://www.w3schools.com/php/func_ftp_rawlist.asp) | Returns a detailed list of files in the specified directory | 3 |
| [ftp\_rename()](http://www.w3schools.com/php/func_ftp_rename.asp) | Renames a file or directory on the FTP server | 3 |
| [ftp\_rmdir()](http://www.w3schools.com/php/func_ftp_rmdir.asp) | Removes a directory on the FTP server | 3 |
| [ftp\_set\_option()](http://www.w3schools.com/php/func_ftp_set_option.asp) | Sets runtime options for the FTP connection | 4 |
| [ftp\_site()](http://www.w3schools.com/php/func_ftp_site.asp) | Sends a SITE command to the server | 3 |
| [ftp\_size()](http://www.w3schools.com/php/func_ftp_size.asp) | Returns the size of the specified file | 3 |
| [ftp\_ssl\_connect()](http://www.w3schools.com/php/func_ftp_ssl_connect.asp) | Opens a secure SSL-FTP connection | 4 |
| [ftp\_systype()](http://www.w3schools.com/php/func_ftp_systype.asp) | Returns the system type identifier of the FTP server | 3 |

PHP FTP الثوابت

**PHP** : يشير إلى الإصدار الأقدم من PHP التي تدعم ثابت.

|  |  |  |
| --- | --- | --- |
| **Constant** | **Description** | **PHP** |
| FTP\_ASCII |  | 3 |
| FTP\_TEXT |  | 3 |
| FTP\_BINARY |  | 3 |
| FTP\_IMAGE |  | 3 |
| FTP\_TIMEOUT\_SEC |  | 3 |
| FTP\_AUTOSEEK |  | 4 |
| FTP\_AUTORESUME | Determine resume position and start position for get and put requests automatically | 4 |
| FTP\_FAILED | Asynchronous transfer has failed | 4 |
| FTP\_FINISHED | Asynchronous transfer has finished | 4 |
| FTP\_MOREDATA | Asynchronous transfer is still active | 4 |

PHP HTTP وظائف

PHP HTTP مقدمة

وظائف HTTP تمكنك من التعامل مع المعلومات التي يتم إرسالها إلى مستعرض من قبل ملقم ويب، قبل أن يتم إرسال أي الإخراج الأخرى.

تركيب

وظائف الدليل هي جزء من جوهر PHP. ليس هناك حاجة لتركيب لاستخدام هذه الوظائف.

PHP HTTP وظائف

**PHP** : يشير إلى الإصدار الأقدم من PHP التي تدعم وظيفة.

|  |  |  |
| --- | --- | --- |
| **Function** | **Description** | **PHP** |
| [header()](http://www.w3schools.com/php/func_http_header.asp) | Sends a raw HTTP header to a client | 3 |
| [headers\_list()](http://www.w3schools.com/php/func_http_headers_list.asp) | Returns a list of response headers sent (or ready to send) | 5 |
| [headers\_sent()](http://www.w3schools.com/php/func_http_headers_sent.asp) | Checks if / where the HTTP headers have been sent | 3 |
| [setcookie()](http://www.w3schools.com/php/func_http_setcookie.asp) | Sends an HTTP cookie to a client | 3 |
| [setrawcookie()](http://www.w3schools.com/php/func_http_setrawcookie.asp) | Sends an HTTP cookie without URL encoding the cookie value | 5 |

PHP HTTP الثوابت

**لا شيء.**

PHP libxml وظائف

PHP libxml مقدمة

وتستخدم وظائف libxml والثوابت معا مع SimpleXML، XSLT وظائف DOM.

تركيب

هذه الوظائف تتطلب حزمة libxml. [تحميل في xmlsoft.org](http://www.xmlsoft.org/downloads.html)

PHP libxml وظائف

**PHP** : يشير إلى الإصدار الأقدم من PHP التي تدعم وظيفة.

|  |  |  |
| --- | --- | --- |
| **Function** | **Description** | **PHP** |
| [libxml\_clear\_errors()](http://www.w3schools.com/php/func_libxml_clear_errors.asp) | Clear libxml error buffer | 5 |
| [libxml\_get\_errors()](http://www.w3schools.com/php/func_libxml_get_errors.asp) | Retrieve array of errors | 5 |
| [libxml\_get\_last\_error()](http://www.w3schools.com/php/func_libxml_get_last_error.asp) | Retrieve last error from libxml | 5 |
| libxml\_set\_streams\_context() | Set the streams context for the next libxml document load or write | 5 |
| [libxml\_use\_internal\_errors()](http://www.w3schools.com/php/func_libxml_use_internal_errors.asp) | Disable libxml errors and allow user to fetch error information as needed | 5 |

الثوابت PHP libxml

|  |  |  |
| --- | --- | --- |
| **Function** | **Description** | **PHP** |
| LIBXML\_COMPACT | Set small nodes allocation optimization. This may improve the application performance | 5 |
| LIBXML\_DTDATTR | Set default DTD attributes | 5 |
| LIBXML\_DTDLOAD | Load external subset | 5 |
| LIBXML\_DTDVALID | Validate with the DTD | 5 |
| LIBXML\_NOBLANKS | Remove blank nodes | 5 |
| LIBXML\_NOCDATA | Set CDATA as text nodes | 5 |
| LIBXML\_NOEMPTYTAG | Change empty tags (e.g. <br/> to <br></br>), only available in the DOMDocument->save() and DOMDocument->saveXML() functions | 5 |
| LIBXML\_NOENT | Substitute entities | 5 |
| LIBXML\_NOERROR | Do not show error reports | 5 |
| LIBXML\_NONET | Stop network access while loading documents | 5 |
| LIBXML\_NOWARNING | Do not show warning reports | 5 |
| LIBXML\_NOXMLDECL | Drop the XML declaration when saving a document | 5 |
| LIBXML\_NSCLEAN | Remove excess namespace declarations | 5 |
| LIBXML\_XINCLUDE | Use XInclude substitution | 5 |
| LIBXML\_ERR\_ERROR | Get recoverable errors | 5 |
| LIBXML\_ERR\_FATAL | Get fatal errors | 5 |
| LIBXML\_ERR\_NONE | Get no errors | 5 |
| LIBXML\_ERR\_WARNING | Get simple warnings | 5 |
| LIBXML\_VERSION | Get libxml version (e.g. 20605 or 20617) | 5 |
| LIBXML\_DOTTED\_VERSION | Get dotted libxml version (e.g. 2.6.5 or 2.6.17) | 5 |

PHP الخلية مقدمة

مع PHP، يمكنك الاتصال والتعامل مع قواعد البيانات.

MySQL هو نظام قاعدة البيانات الأكثر شعبية المستخدمة مع PHP.

ما هو الخلية؟

* MySQL هو نظام قاعدة البيانات المستخدمة على شبكة الإنترنت
* MySQL هو نظام قاعدة البيانات التي تعمل على الخادم
* MySQL هو مثالي للتطبيقات الصغيرة والكبيرة
* MySQL هو سريع جدا وموثوق بها، وسهلة الاستخدام
* الخلية يدعم SQL القياسية
* الخلية يجمع على عدد من منصات
* MySQL هو مجاني لتحميل واستخدام
* تم تطوير الخلية، وزعت، وبدعم من شركة أوراكل
* يدعى الخلية بعد ابنة المؤسس المشارك في مونتي Widenius: بلدي

يتم تخزين البيانات في الخلية في الجداول. جدول عبارة عن مجموعة من البيانات ذات الصلة، وأنها تتكون من الأعمدة والصفوف.

قواعد البيانات هي مفيدة عند تخزين المعلومات بشكل قاطع. قد يكون لدى الشركة قاعدة بيانات مع الجداول التالية:

* موظف
* المنتجات
* الزبائن
* أوامر

PHP + ماي

* PHP جنبا إلى جنب مع الخلية هي عبر منصة (يمكنك وضع في ويندوز وتخدم على منصة يونكس)

الاستفسارات

الاستعلام هو سؤال أو طلب.

يمكننا الاستعلام عن قاعدة بيانات لمعلومات محددة ويكون إرجاع السجلات.

ننظر في الاستعلام التالي (باستخدام SQL القياسية):

SELECT LastName FROM Employees

الاستعلام أعلاه تحديد كافة البيانات في العمود "اسم العائلة" من جدول "الموظفين".

لمعرفة المزيد عن SQL، يرجى زيارة موقعنا على [البرنامج التعليمي SQL](http://www.w3schools.com/sql/default.asp) .

تحميل قاعدة بيانات الخلية

إذا لم يكن لديك ملقم PHP مع قاعدة بيانات الخلية، يمكنك تحميل الخلية مجانا هنا: [http://www.mysql.com](http://www.mysql.com/)

حقائق حول قاعدة بيانات الخلية

شيء واحد كبير حول الخلية هو أنه يمكن تقليصها لدعم تطبيقات قواعد البيانات جزءا لا يتجزأ. ربما هو بسبب هذا كثير من الناس يعتقدون أن الخلية يمكن التعامل مع النظم الصغيرة والمتوسطة الحجم فقط.

الحقيقة هي أن MySQL هو نظام قاعدة البيانات المعايير الفعلية للمواقع على شبكة الإنترنت مع كميات ضخمة من البيانات والمستخدمين النهائيين (مثل فريندستر، ياهو، وجوجل).

ننظر <http://www.mysql.com/customers/> لمحة عامة عن الشركات التي تستخدم الخلية.

PHP اتصال إلى خادم الخلية

PHP إنشاء قاعدة بيانات وجداول

يحمل قاعدة بيانات جدول واحد أو أكثر.

إنشاء قاعدة بيانات

يتم استخدام عبارة CREATE DATABASE لإنشاء جدول قاعدة البيانات في الخلية.

يجب أن نضيف عبارة CREATE DATABASE إلى mysqli\_query () وظيفة لتنفيذ الأمر.

المثال التالي بإنشاء قاعدة بيانات باسم "my\_db":

<?php  
$con=mysqli\_connect("example.com","peter","abc123");  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
  
// Create database  
$sql="CREATE DATABASE my\_db";  
if (mysqli\_query($con,$sql))  
  {  
  echo "Database my\_db created successfully";  
  }  
else  
  {  
  echo "Error creating database: " . mysqli\_error($con);  
  }  
?>

إنشاء الجدول

يتم استخدام عبارة CREATE TABLE لإنشاء الجدول في الخلية.

يجب أن نضيف عبارة CREATE TABLE لmysqli\_query () وظيفة لتنفيذ الأمر.

المثال التالي بإنشاء جدول يسمى "الأشخاص"، مع ثلاثة أعمدة: "الاسم الأول"، "اسم العائلة" و "العمر":

<?php  
$con=mysqli\_connect("example.com","peter","abc123","my\_db");  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
  
// Create table  
$sql="CREATE TABLE Persons(FirstName CHAR(30),LastName CHAR(30),Age INT)";  
  
// Execute query  
if (mysqli\_query($con,$sql))  
  {  
  echo "Table persons created successfully";  
  }  
else  
  {  
  echo "Error creating table: " . mysqli\_error($con);  
  }  
?>

**ملاحظة:** عند إنشاء حقل من نوع CHAR، يجب تحديد الحد الأقصى لطول الحقل، مثل CHAR (50).

يحدد نوع البيانات نوع البيانات يمكن أن تعقد العمود. للإشارة كاملة لجميع أنواع البيانات المتوفرة في الخلية، انتقل إلى لدينا على أكمل [أنواع البيانات المرجعية](http://www.w3schools.com/sql/sql_datatypes.asp) .

مفاتيح الأولية والسيارات حقول الاضافة

يجب أن يكون كل جدول في قاعدة بيانات حقل المفتاح الأساسي.

يستخدم المفتاح الأساسي لتحديد فريد الصفوف في الجدول. يجب أن تكون كل قيمة المفتاح الأساسي فريدة من نوعها داخل الجدول. وعلاوة على ذلك، يمكن للحقل المفتاح الأساسي لا يكون باطلا لأن محرك قاعدة البيانات يتطلب قيمة لتحديد موقع السجل.

المثال التالي بتعيين حقل PID كحقل مفتاح أساسي. حقل المفتاح الأساسي غالبا ما يكون رقم الهوية، وغالبا ما يستخدم مع إعداد AUTO\_INCREMENT. AUTO\_INCREMENT تلقائيا يزيد من قيمة الحقل بمقدار 1 كل مرة يتم فيها إضافة سجل جديد. لضمان أن حقل المفتاح الأساسي لا يمكن أن تكون فارغة، يجب أن نضيف الإعداد NOT NULL إلى الميدان:

$sql = "CREATE TABLE Persons   
(  
PID INT NOT NULL AUTO\_INCREMENT,   
PRIMARY KEY(PID),  
FirstName CHAR(15),  
LastName CHAR(15),  
Age INT  
)";

PHP الخلية تضاف الى

يتم استخدام INSERT INTO بيان لإدراج سجلات جديدة في جدول.

إدراج البيانات في جدول قاعدة البيانات

يتم استخدام INSERT INTO بيان لإضافة سجلات جديدة إلى جدول قاعدة البيانات.

بناء الجملة

فمن الممكن لكتابة INSERT INTO بيان في شكلين.

لا يحدد النموذج الأول أسماء الأعمدة حيث سيتم إدراج البيانات، فقط قيمها:

INSERT INTO table\_nameVALUES (value1, value2, value3,...)

يحدد النموذج الثاني كل من أسماء الأعمدة والقيم التي ستدرج لاحقا:

INSERT INTO table\_name (column1, column2, column3,...)  
VALUES (value1, value2, value3,...)

لمعرفة المزيد عن SQL، يرجى زيارة موقعنا على [البرنامج التعليمي SQL](http://www.w3schools.com/sql/default.asp) .

للحصول على PHP لتنفيذ البيانات أعلاه يجب علينا أن نستخدم mysqli\_query (وظيفة). يتم استخدام هذه الدالة لإرسال استعلام أو الأمر إلى اتصال الخلية.

مثال

في الفصل السابق أنشأنا جدول يسمى "الأشخاص"، مع ثلاثة أعمدة؛ "الاسم الأول"، "اسم العائلة" و "العمر". سوف نستخدم نفس الجدول في هذا المثال. يضيف المثال التالي سجلين جديدة إلى جدول "الأشخاص":

<?php  
$con=mysqli\_connect("example.com","peter","abc123","my\_db");  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
  
mysqli\_query($con,"INSERT INTO Persons (FirstName, LastName, Age)  
VALUES ('Peter', 'Griffin',35)");  
  
mysqli\_query($con,"INSERT INTO Persons (FirstName, LastName, Age)   
VALUES ('Glenn', 'Quagmire',33)");  
  
mysqli\_close($con);  
?>

إدراج البيانات من نموذج إلى قاعدة بيانات

الآن سوف نقوم إنشاء نموذج HTML والتي يمكن استخدامها لإضافة سجلات جديدة إلى الجدول "الأشخاص".

هنا هو شكل HTML:

<html>  
<body>  
  
<form action="insert.php" method="post">  
Firstname: <input type="text" name="firstname">  
Lastname: <input type="text" name="lastname">  
Age: <input type="text" name="age">  
<input type="submit">  
</form>  
  
</body>  
</html>

عندما ينقر المستخدم على زر الإرسال في شكل HTML، في المثال أعلاه، يتم إرسال بيانات النموذج إلى "insert.php".

يربط "insert.php" ملف إلى قاعدة البيانات، ويسترد القيم من النموذج مع PHP $ \_POST المتغيرات.

ثم، وظيفة mysqli\_query () ينفذ INSERT INTO بيان، وسيتم إضافة سجل جديد إلى الجدول "الأشخاص".

هنا هي الصفحة "insert.php":

<?php  
$con=mysqli\_connect("example.com","peter","abc123","my\_db");  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
  
$sql="INSERT INTO Persons (FirstName, LastName, Age)  
VALUES  
('$\_POST[firstname]','$\_POST[lastname]','$\_POST[age]')";  
  
if (!mysqli\_query($con,$sql))  
  {  
  die('Error: ' . mysqli\_error($con));  
  }  
echo "1 record added";  
  
mysqli\_close($con);  
?>

استخدام وظيفة PHP mysqli\_connect () لفتح اتصال جديد إلى خادم MySQL.

فتح اتصال إلى خادم الخلية

قبل أن نتمكن من الوصول إلى البيانات في قاعدة بيانات، يجب علينا فتح اتصال إلى خادم MySQL.

في PHP، ويتم ذلك مع mysqli\_connect (وظيفة).

بناء الجملة

mysqli\_connect(host,username,password,dbname);

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| host | Optional. Either a host name or an IP address |
| username | Optional. The MySQL user name |
| password | Optional. The password to log in with |
| dbname | Optional. The default database to be used when performing queries |

**ملاحظة:** هناك أكثر المعلمات المتاحة، ولكن تلك المذكورة أعلاه هي الأكثر أهمية.

في المثال التالي نقوم بتخزين الاتصال في متغير ($ يخدع) لاستخدامها لاحقا في البرنامج النصي:

<?php  
// Create connection  
$con=mysqli\_connect("example.com","peter","abc123","my\_db");  
  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
?>

إغلاق اتصال

سيتم إغلاق الاتصال تلقائيا عند انتهاء البرنامج النصي. لإغلاق الاتصال من قبل، استخدام mysqli\_close (وظيفة):

<?php  
$con=mysqli\_connect("example.com","peter","abc123","my\_db");  
  
// Check connection  
if (mysqli\_connect\_errno())  
  {  
  echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  
  }  
  
mysqli\_close($con);  
?>