

نبذة عن لغة PHP

إن لغة php هي اختصار لـ (Personal Home Page) أي الصفحات الشخصية نتجها في البداية شخص يدعى (راسموس ليدروف) وجعلها مجانية ومفتوحة المصدر وهذا جعل الكثيرون يشاركون في تطويرها إلى أن أصبحت كبيرة ومهمة جداً خصوصاً في عالم الانترنت .
وتعرف لغة الـ php :- عبارة عن لغة برمجية تمكنك من انشاء صفحات ومواقع الويب الديناميكية التي تتغير حسب اختيار المستخدم .

وتتميز لغة php بالكثير من المميزات التي جعلتها الخيار الامثل لمبرمجي الويب في العالم .

• مجانية وحررة المصدر :

بمعنى آخر يمكننا استخدام php مجاناً وبدون أن نضطر أن ندفع ريال واحد للحصول على مترجم اللغة ولأنها تعمل أيضاً على أنظمة اللينكس المجانية بحيث يمكن أن نبني ويب سيرفر كامل بدون أي تكلفة تذكر . هذا بعكس بعض التقنيات الأخرى والتي تتطلب الكثير سواء من ناحية شراء حقوق تشغيل تلك التقنيات وشراء الأنظمة والبرامج التي تحتاجها هذا بالنسبة للمجانبة أما بالنسبة لمصطلح حررة المصدر فيعني أن كود اللغة نفسها مفتوح ومتوفر ويمكن لأي شخص أن يحصل على الكود ويعدل في بناء اللغة نفسها بالشكل الذي يريده .

• السهولة :

تعتبر لغة PHP من أسهل لغات البرمجة تعلمها، فهي تريحك من جميع تعقيدات إدارة الذاكرة وتعقيدات معالجة النصوص الموجودة في C من جهة، والكثير من الضعف الموجود في بنية وتصميم لغة البرمجة Perl من جهة أخرى. تمتلك لغة PHP بنية وقواعد ثابتة وواضحة جداً، معظم قواعد اللغة مأخوذة من كل من C و Java و Perl لصنع لغة برمجة عالية السهولة والسلاسة دون فقدان أي من القوة في اللغة، يفيدك ذلك إذا كنت تعلم أي شيء عن لغات البرمجة الأخرى مثل Visual Basic أو C أو Java حيث ستجد دائماً بأنك تفهم مواد الدورة بسرعة، وستكتشف كيف تقوم PHP بتسهيل أصعب الأمور وإذلال العقبات التي تواجه المبرمج حتى يتفرغ تماماً للإبداع فقط، كل ما تفكر به تستطيع تنفيذه بلغة PHP.

• السرعة :

لغة PHP من اللغات المعروفة بسرعتها العالية في تنفيذ البرامج، وخاصة في الإصدار الرابعة من المترجم، حيث تمت كتابة مترجم PHP من الصفر ليعطي أداء في منتهى الروعة . وتوجد نسخة أخرى منه تتركب على مزود Apache أيضاً في صورة وحدة خارجية، وتوجد أيضاً نسخة مخصصة للدمج مع شفرة مزود Apache بحيث تصبح جزءاً من برنامج Apache نفسه، وهي الطريقة الأكثر استخداماً الآن في مزودات الويب التي تعمل على أنظمة UNIX وهي الطريقة التي تعطي أفضل أداء لمترجم PHP، حيث يصبح المترجم جزءاً من المزود، وبالتالي فإنه سيكون محملاً في الذاكرة بانتظار صفحات PHP ليقوم بترجمتها وعرضها للزوار مباشرة دون التأخير الإضافي الذي تتطلبه برامج Perl/CGI مثلًا حيث يجب أن يتم تشغيل مترجم Perl مع كل زيارة للصفحة لترجمة الصفحة، ثم يتم إغلاق المترجم، ثم استدعاه مجدداً عند الزيارة الثانية وهكذا، وهذا يشكل فارقاً كبيراً في المواقع ذات الضغط العالي بالذات، ويكون استخدام PHP حلاً أفضل بكثير.

• الإمكانيات

يأتي مترجم PHP لوحده محملاً بعدد هائل من الدوال الجاهزة الاستخدام في جميع المجالات، من دوال المعالجة الرياضية والحسابية إلى دوال الوصول إلى قواعد البيانات ومزودات FTP، توفر لك دوال PHP مثلاً وصولاً إلى مزودات البيانات MySQL و PostgreSQL و MS SQL و Oracle وغيرها من مزودات قواعد البيانات، وهناك أيضاً مجموعة من الدوال لمعالجة ملفات XML، ودوال أخرى لإرسال واستقبال الملفات عن بعد باستخدام بروتوكول FTP، وهناك مجموعة من الدوال لمعالجة وإنتاج الصور ديناميكياً وملفات Flash ديناميكياً، ناهيك عن جميع الدوال الخاصة بمعالجة النصوص والمصفوفات.

متطلبات العمل بلغة php

• محرر نصوص

وهو ليس برنامجاً خاصاً بل أي محرر نصوص كبرنامج المفكرة (Notepad) الموجود في نظام التشغيل .

• متصفح للإنترنت

وذلك ترجمة الاوامر ومعاينة الصفحات التي نقوم بتصميمها مثل (Internet Explorer) المنزل مع نظام التشغيل .

• خادم ويب (Web Server)

سنقوم باستخدام خادم ويب محلي لكي نقوم بفحص جميع صفحات الموقع قبل ارسالها على خادم ويب عالمي وعندما نقوم بتجريب جميع صفحات الموقع على السيرفر المحلي لا نحتاج الى الاتصال بالإنترنت وبالتالي تكون عملية معالجة الاخطاء أسهل وأسرع.

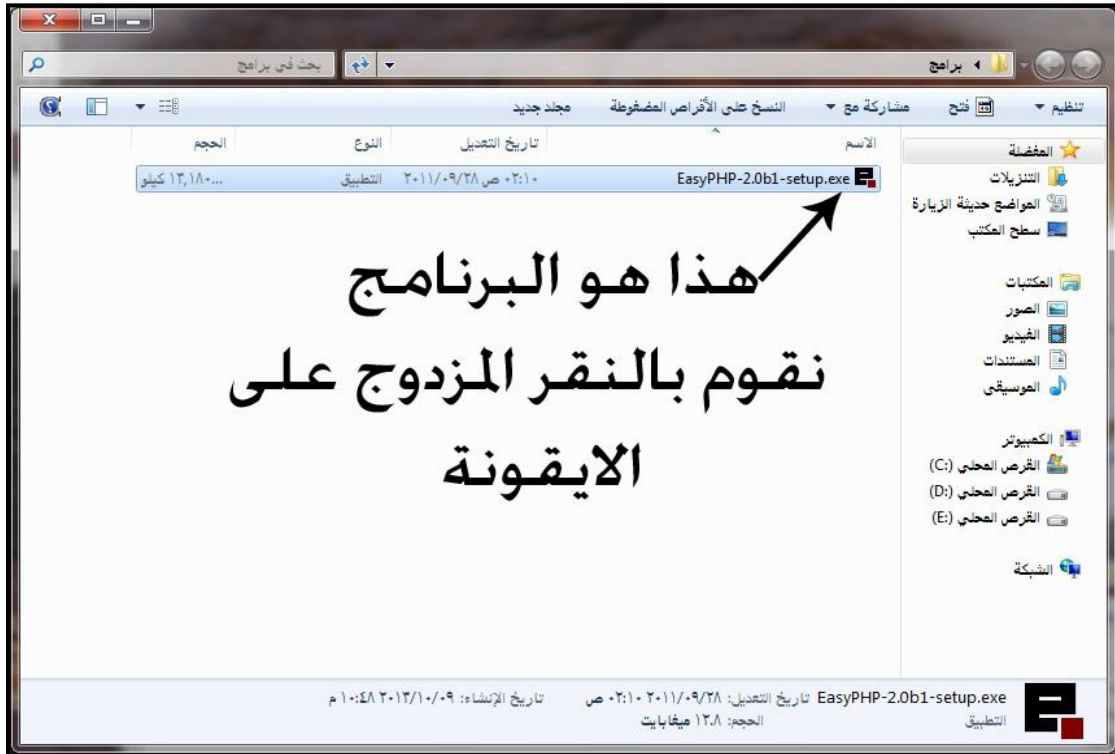
وبعد الانتهاء من جميع عمليات الفحص سوف نقوم بإرسال الموقع بجميع صفحاته الى احد شركات استضافة مواقع الويب التي تمتلك العديد من الاجهزة التي تعمل كـ web server أو خادم مواقع ويب يمكن دفع مبلغ لها بوضع موقعك على هذه السيرفرات الخاصة بهذه الشركة ومن مميزات الشركات عندما تضع موقعك على أحد خادماتها أن موقعك يصبح متاحاً لجميع الأشخاص الموجودين على شبكة الانترنت وكذلك يكون لك عنوان خاص.

تركيب سيرفر محلي على جهازك الشخصي

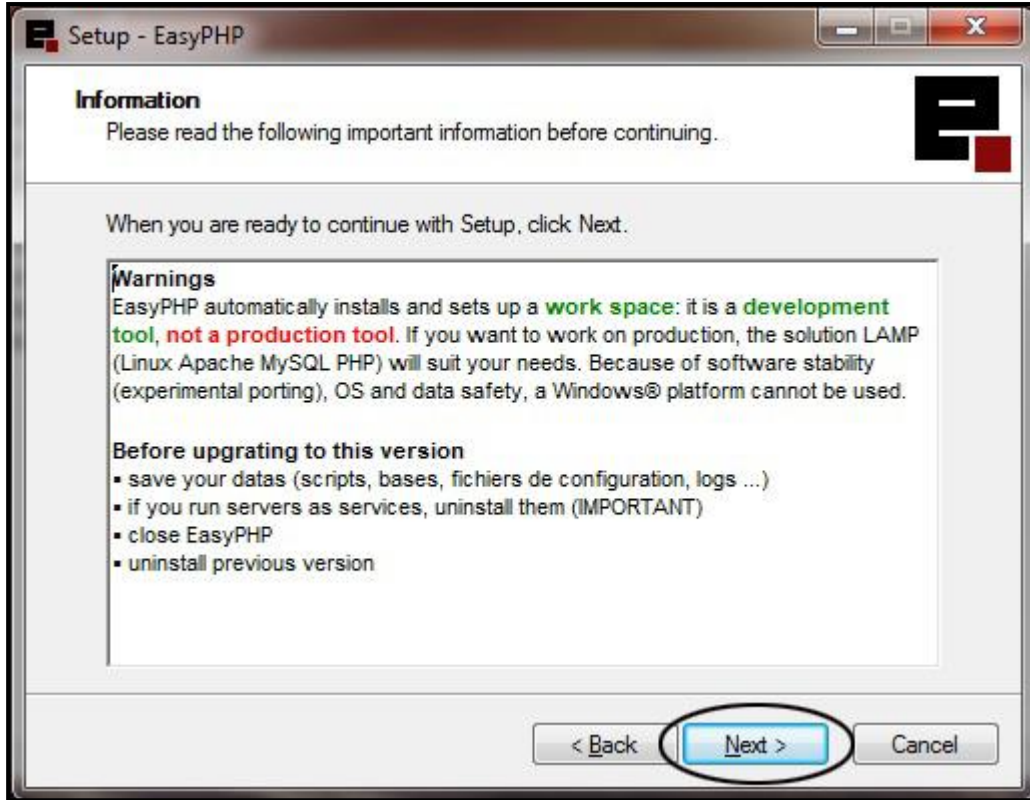
جميع من يكتبون برامج php يكون لديهم سيرفر محلي ونقصد بسيرفر محلي هو أن تحول جهازك الشخصي الى موقع استضافة من خلاله نقوم بتجريب برامجك قبل ارسالها الى الإنترنت وهناك العديد من البرامج التي تجعل

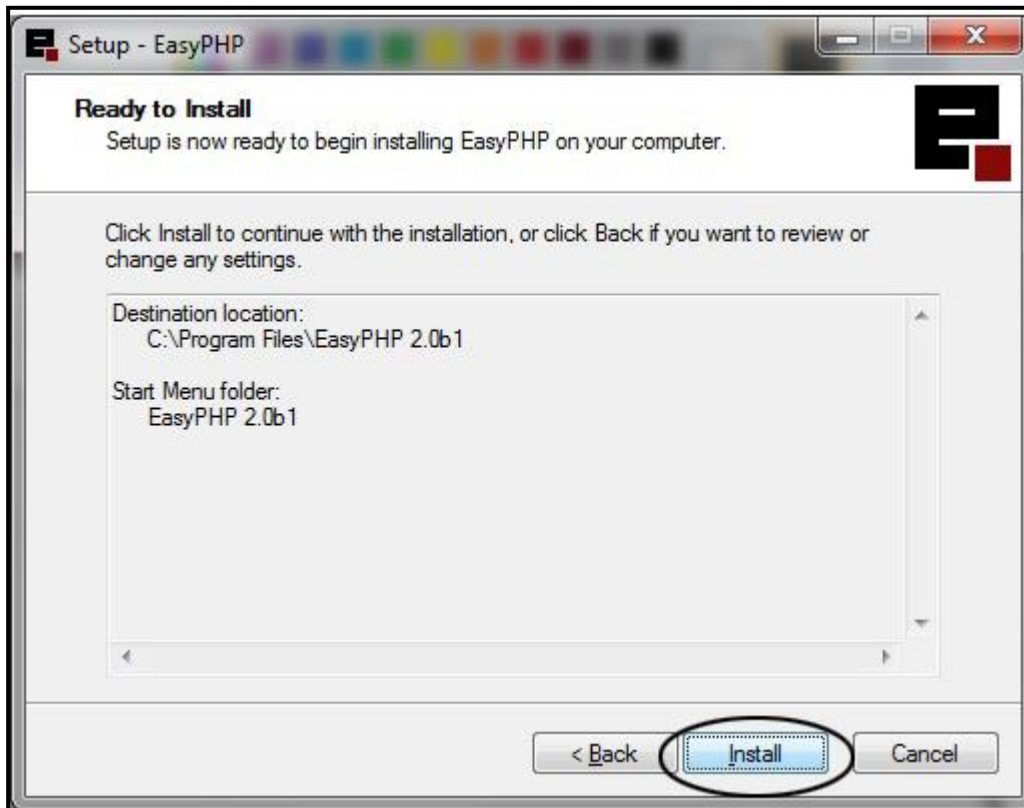
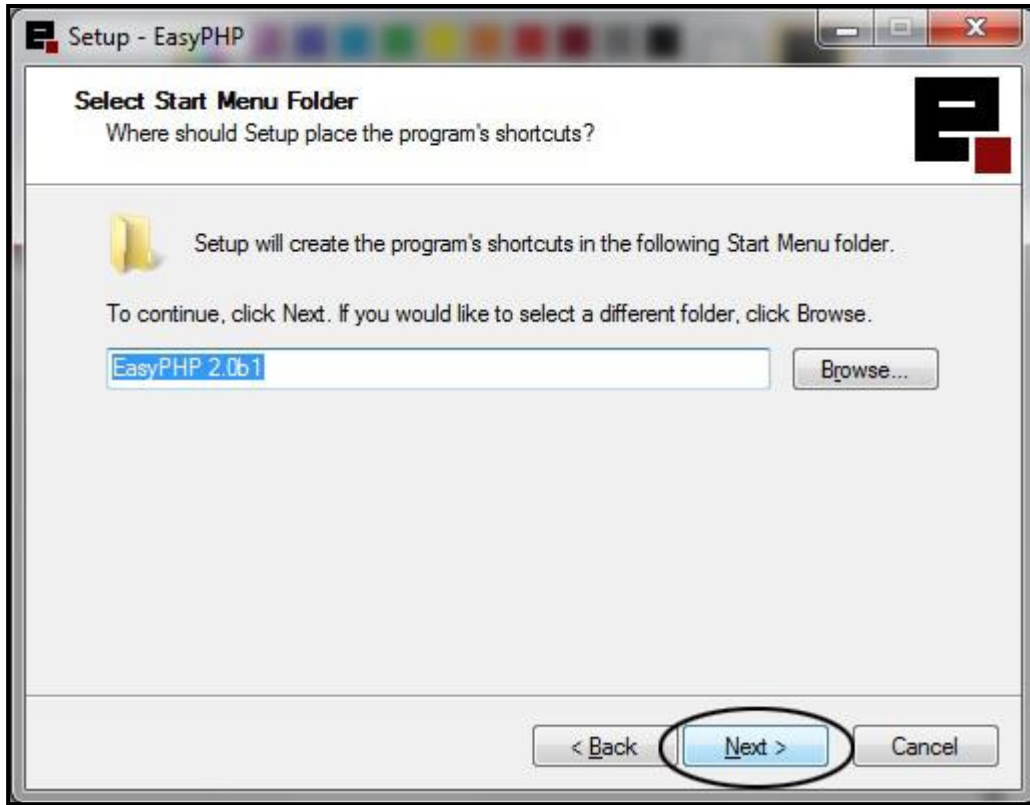
جهازك سيرفر ومنها Apache HTTP Server - EasyPHP Server

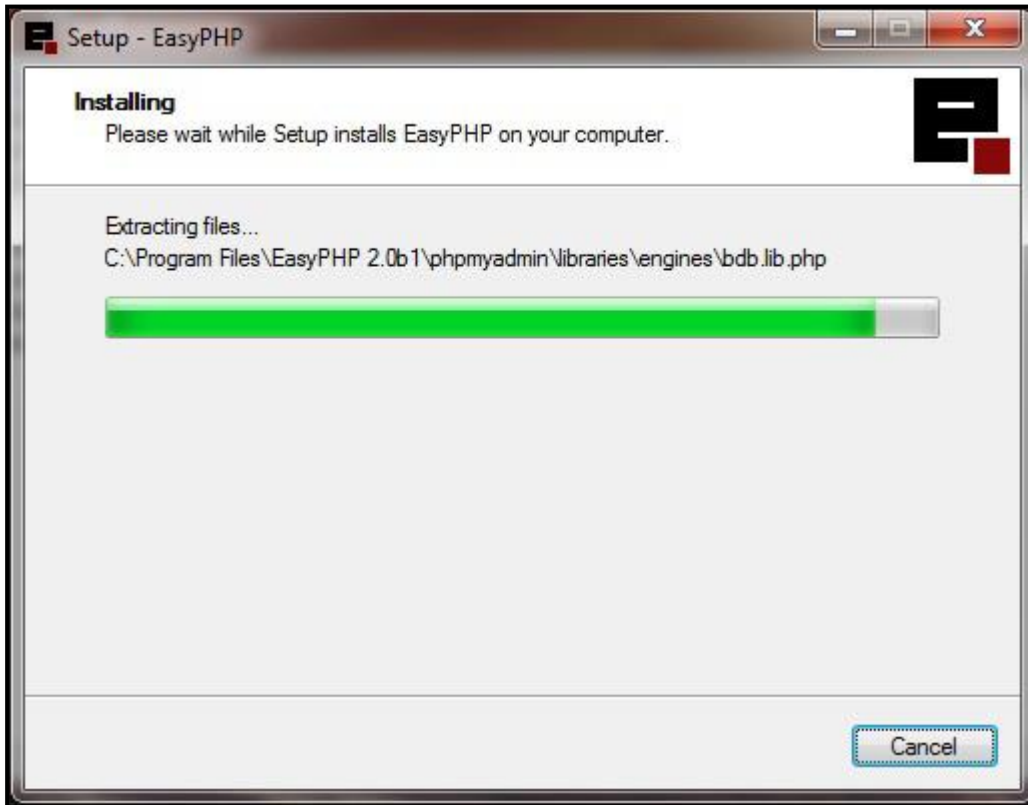
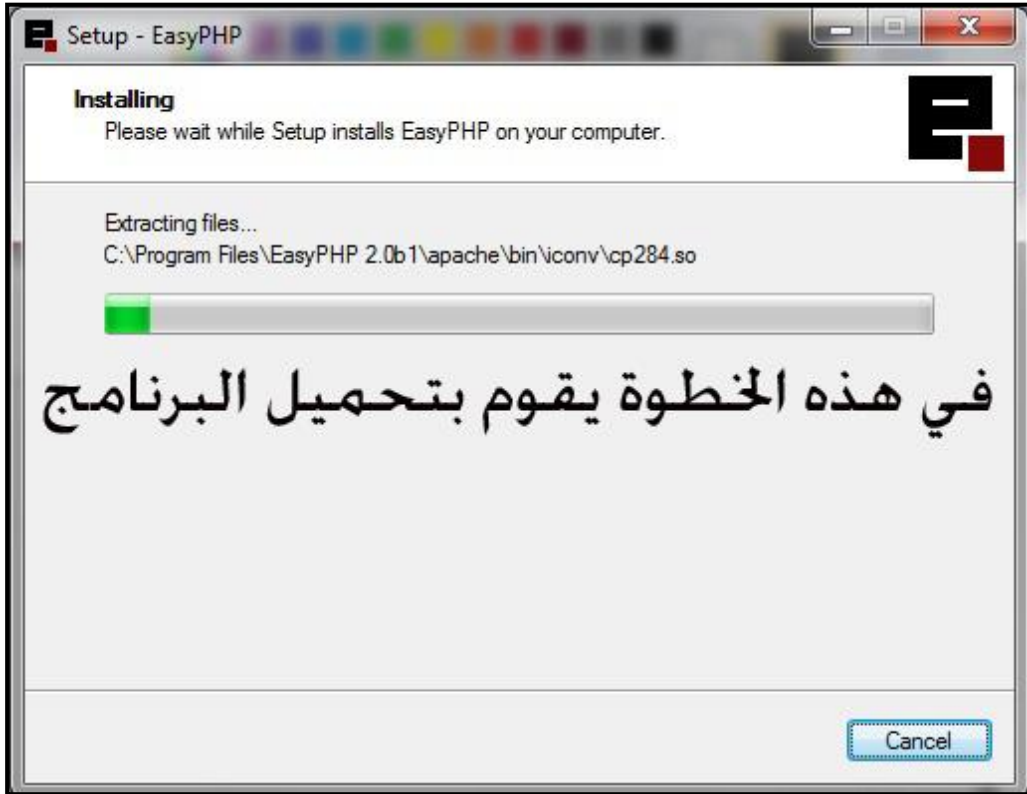
سنقوم بتحميل برنامج (EasyPHP-Server) خطوات تنزيل :-











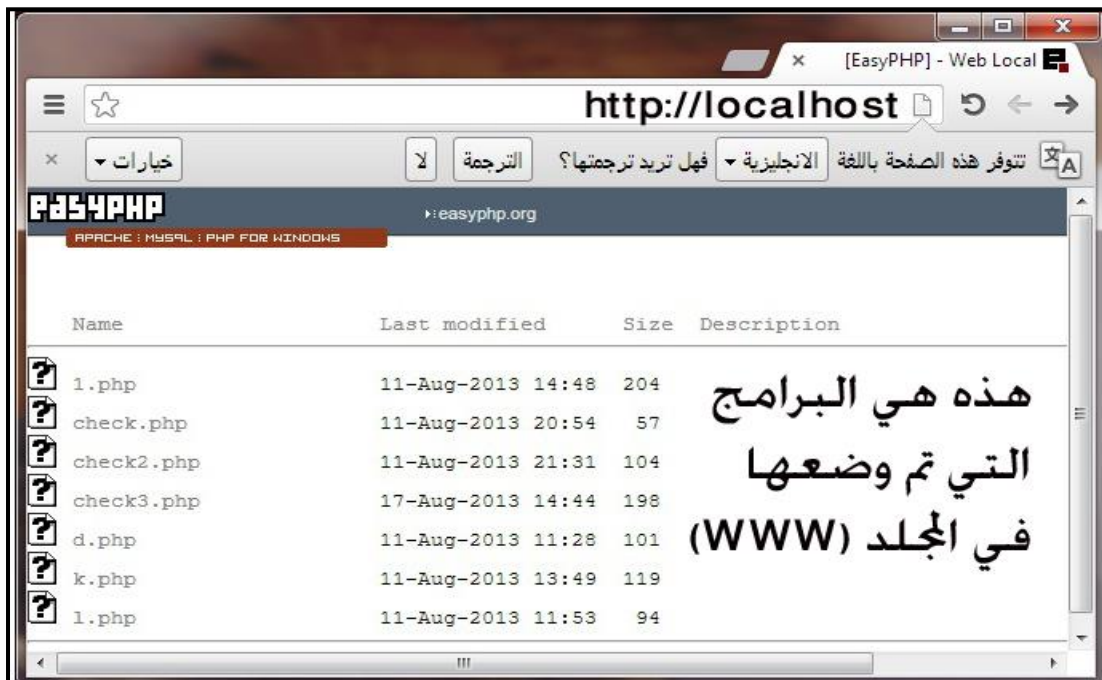


أين تضع الصفحات والبرامج التي تقوم بكتابتها

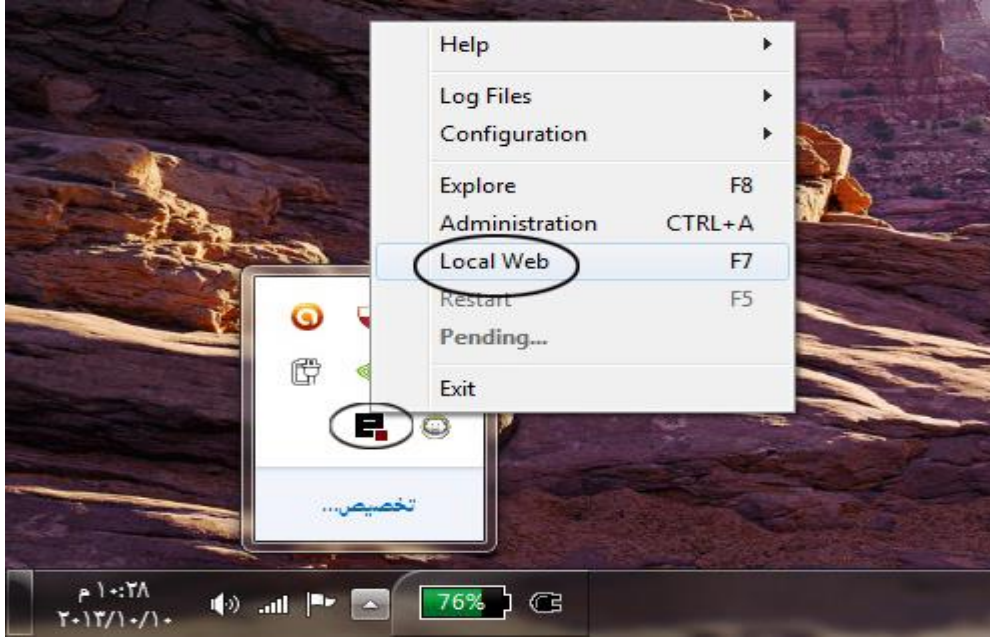
إن السيرفر الشخصي لا يخدم الجهاز كاملاً أي أن الصفحات أو البرامج الموجودة خارج حدود السيرفر الشخصي لا يمكن التعامل معها كصفحات انترنت نشطة أو متفاعلة .
إذن أين اضع الصفحات والسكربتات حتى أستطيع تصفحها وتنفيذها كصفحة انترنت فاعلة
الجواب :

C:\Program Files\EasyPHP 2.0b1\www

فبرنامج السيرفر الذي قمنا بتحميله يحتوي على مجلد اسمه (www) ويعتبره العنوان الرئيسي لموقعك وللوصول الى هذا المجلد نكتب في المتصفح ([HTTP://localhost](http://localhost)) ينقلك مباشرة الى هذا المجلد ومشاهدة جميع البرامج التي قمت بعملها ومنها سنقوم باختيار البرامج الذي نريد تنفيذه



أو ممكن الوصول الى هذا المجلد من شريط المهام (منطقة الاعلام)



بنية ملفات PHP

بما أن صفحات الويب يمكن أن تتضمن أكواد غير أكواد PHP والسيرفر سوف يعتبر أي ملف كـ HTML ولكي نجعل السيرفر يتعرف اننا سنبدأ في ادخال أكواد PHP فعلينا استخدام احدى الطرق الاربعة التالية :

الطريقة الاولى :

```
<?  
هنا يتم كتابة الكود  

```

الطريقة الثانية :

```
<?php  
هنا يتم كتابة الكود
```

الطريقة الثالثة :

```
<script language="php">  
هنا يتم كتابة الكود
```

الطريقة الرابعة :

```
<%  
هنا يتم كتابة الكود
```


البرنامج الاول :

ملاحظة:

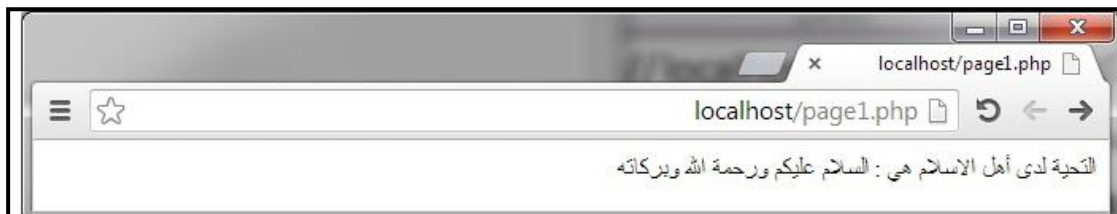
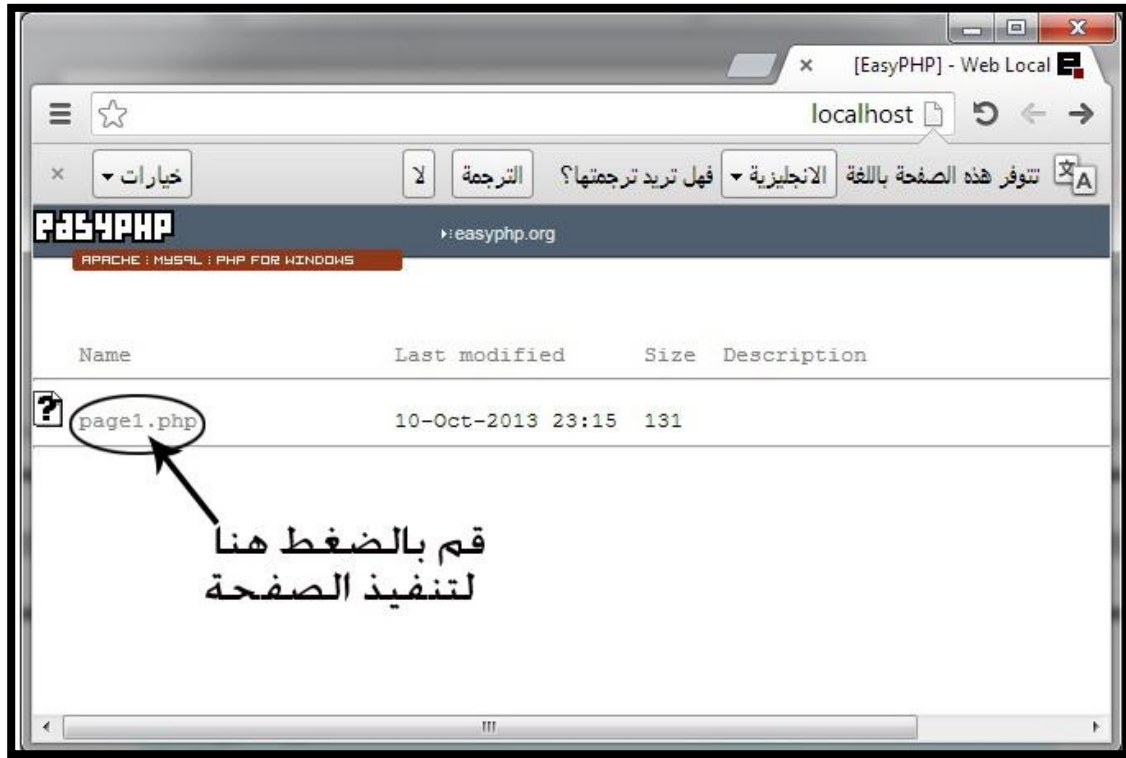
Echo دالة تستخدم للطباعة على المتصفح .

```
<html dir =" rtl">
<body>
التحية لدى أهل الإسلام هي
<?php
Echo ("السلام عليكم ورحمة الله وبركاته")
?>
</body></html>
```

قم بحفظ الملف باسم page1.php في المسار التالي :

C:\Program Files\EasyPHP 2.0b1\www

اكتب في المتصفح (http:// localhost) سنقوم بمشاهدة الملف كما في الصورة :



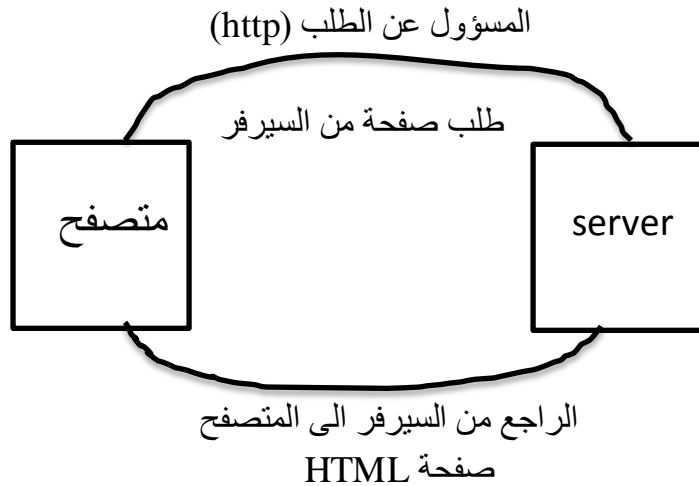
قم بعمل للصفحة عرض المصدر لمشاهدة الكود الذي تم بناء به هذه الصفحة سنلاحظ الكود الذي بنا هذه الصفحة كما يلي

```
1 <html dir ="rtl">
2 <body>
3 : التحية لدى أهل الاسلام هي
4 السلام عليكم ورحمة الله وبركاته
5 </body>
6 </html>
7
```

نلاحظ كود HTML هو الذي ظهر فقط ولم يظهر كود PHP فما الذي حصل ؟

إذا اردت استعراض اي صفحة من الصفحات الموجودة على خادم الويب عليك طلب هذه الصفحة باستخدام مستعرض الانترنت (Internet Explorer) وكتابة العنوان التالي مثلاً [http:// localhost/page1.php](http://localhost/page1.php) حيث يقوم المستعرض بإرسال الطلب للخادم المحلي المسمى بـ (localhost) ويطلب منه الصفحة المحددة في العنوان (page1.php) يتم ارسال الطلب عن طريق البروتوكول (http) يستقبل الويب سيرفر الطلب ويبحث عن الملف المسمى (page1.php) في المجلد الخاص بالسيرفر الذي يعمل فيه

يقوم الويب سيرفر بقراءة الملف (page1.php) ويميز أن هذا الملف ليس ملف HTML عادي بل يحتوي أيضاً على شفرة PHP لذلك فانه يقوم باستدعاء المفسر الخاص بتنفيذ شفرة PHP الموجود في السيرفر بعد أن ينتهي المترجم الخاص بـ PHP من ترجمة الكود يقوم الويب السيرفر بإعادة النتيجة الى المتصفح بصورة HTML تسمى هذه العملية (rendered) يقرأ المتصفح شفرة الـ HTML المعادة ويقوم بتحويلها الى واجهة مرئية للمستخدم .



وبذلك قد استطاع السيرفر توليد صفحة HTML بشكل آلي وبدون تدخل يدوي وبهذا السبب يطلق على لغة PHP تقنية برمجية تمكنك من انشاء صفحات ومواقع الويب الديناميكية ويقصد بصفحة ويب ديناميكية يعني ان محتوى الصفحة يعاد توليده ديناميكياً في كل مرة يتم طلب الصفحة من المستخدم .

البرنامج الثاني:-

```
<html >
<body>
<?php
echo "<font color=blue size=4>";
echo "welcome in hodeidah university";
echo "</font>";
?>
</body></html>
```

قم بحفظ الملف باسم page2.php .
قم بأخذ الملف ووضعه في مجلد السيرفر لديك .
اكتب في مستعرض الانترنت لديك [http:// localhost/page2.php](http://localhost/page2.php) وسوف تشاهد التالي :



قم بعمل (rendered) للبرنامج ؟

```
1 <html>
2 <body>
3 <font color=blue size=4>welcom in hodeidah university</font></body>
4 </html>
```

أو ممكن كتابة البرنامج الثاني بهذه الطريقة :

```
<html >
<body>
<font color="blue" size=4>
<?php
echo "welcome in hodeidah university";
?>
</font>
</body></html>
```

المتغيرات Variables :-

في أي لغة برمجية تعتبر المتغيرات وطرق كتابتها من أهم الاساليب البرمجية لانه بواسطة المتغيرات نستطيع التحكم ببرامجنا ببسر وسهولة وأبسط تعريف يمكن أن نقوله على المتغير هو أنه مساحة من الذاكرة تستخدم لخرن المعلومات

ومن الميزات في لغة php أنها ألياً تعرف نوع المتغير بل تقوم بالتحويل بين المتغيرات بكل سهولة فيمكن أن تجعل متغير نص ثم تقوم بتحويله إلى رقمي بمجرد كتابة قيمة به .
فالمتغيرات في لغة php تبدأ بعلامة (\$) .

```
$ x= "How Are You Every Body?";
```

\$ قيمه = اسم_المتغير ;

لاحظ أن السطر السابق يتكون من خمسة أشياء :

١ / المتغير وهو x

٢ / وقبله علامة الـ \$ لكي يعرف مترجم الـ PHP أنه متغير

٣ / المعامل (=)

٤ / الفاصلة المنقوطة (;)

٥ / القيمة وهي How Are You Every Body? وهي القيمة الموجودة في المتغير أو التي اقترحناها للمتغير أو التي وضعناها فيه (لأن الذي اقترح القيمة هو أنت (مبرمج الـ php)) .

أنواع المتغيرات المدعومة بلغة php

أولاً: - الأعداد الصحيحة (Integer)

هذا النوع من المتغيرات يأخذ القيم الرقمية الصحيحة الموجبة والسالبة مع العلم أن php تعتبر أي رقم بهذه الخصائص وغير محاط بعلامة تنصيب (" ") عدد صحيح
أمثلة :

```
$a=1234;
```

```
$a=-1234;
```

ثانياً: - الأعداد ذات العلامات العشرية (Double)

هذا النوع من المتغيرات يأخذ القيم الرقمية التي تحتوي على علامات عشرية مع العلم أن لغة php تعتبر أي رقم بهذه الخصائص وغير محاط بعلامة تنصيب عدد ذو علامة عشرية

```
$a=1.234;
```

```
$a=1.2e4;
```

ثالثاً- المتغيرات النصية (String)

المتغير النصي يأخذ حروف وارقام وعلامات مختلفة ويعتبر لغة php أي قيمة بين علامتي تنصيب على انها متغير نصي

```
$str= "How Are You Every Body? ";
```

رابعاً: - المصفوفات (Arrays)

هي سلسلة من القيم في متغير واحد يعبر عنه ككائن يحتوي على العديد من القيم ويمكن التعبير عن قيم المصفوفة بالارقام أو الاسماء حسب الحاجة ومن مهام المصفوفة تسهيل البرمجة وتوفير الوقت والجهد فتستطيع قراءة ملف

بجملته واحدة لتجد أن كل سطر في الملف وضع داخل مصفوفة وسوف نتحدث عن ذلك عندما نتحدث عن كيفية القراءة من الملفات وكذلك قواعد البيانات وغيرها
أمثلة :

```
$a[0]="abc";  
$a[1]="def";  
$b["foo"]=13;
```

ملاحظات

١- أسماء المتغيرات حساسة لحالة الأحرف إذا كانت صغيرة أو كبيرة .

```
<?php  
$A = "waleed";  
$a = "yahya";  
echo $a;  
echo $A;  
?>
```

٢- يمكنك استخدام المعامل (_)

```
$First_name;
```

٣- أسماء المتغيرات في الـ php لا يمكن أن تبدأ برقم .

٤- يمكن الجمع بين أكثر قيم المتغيرات في متغير واحد عن طريقة الـ (.)

```
<?php  
$first = "منندي" ;  
$last = "المطور العربي";  
$fullname = $first. $last  
Echo $fullname ;  
ولكننا نريد وضع فراغ بين الكلمتين//  
$fullname= $first . ' '. $last ;  
Echo $fullname ;  
?>
```

الفقر بين أنواع المتغيرات

تقوم لغة php بتغيير نوع المتغير إذا غيرت القيمة التي يحتويها فإذا كانت القيمة رقمية فإنه سوف يجعله عدد صحيح وإذا كتبت بعد ذلك في نفس المتغير قيمة نصية فسوف يحوله آلياً إلى متغير نصي واليك الأمثلة :

```
$a=1; // هذا عدد صحيح  
$a="sager"; // الان نص  
$a=1; // عاد رقمي مرة اخرى
```

كيفية كتابة المتغيرات

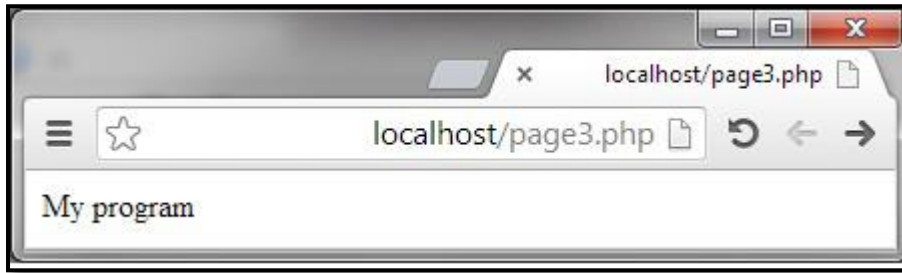
تعتبر لغة الـ php أي كلمة مسبوقة بعلامة (\$) على أنها متغير اليك المثال التالي :

```
<?php
$var1 = "My";
$var2 = "program";
echo "$var2 $var2";
?>
```

قم بحفظ الملف باسم page3.php .

قم بأخذ الملف ووضعه في مجلد السيرفر لديك .

اكتب في مستعرض الانترنت لديك [http:// localhost/page3.php](http://localhost/page3.php) وسوف تشاهد التالي :



قم بعمل (rendered) للبرنامج ؟ على الطالب

العمليات الحسابية على المتغيرات الرقمية

الصيغة	العملية
$\$a + \b	جمع متغيرين
$\$a - \b	طرح متغيرين
$\$a * \b	ضرب متغيرين
$\$a / \b	قسمة متغيرين
$\$a \% \b	نسبة المتغير الاول من الثاني
$++\$a$	اضافة واحد الى قيمة المتغير
$--\$a$	طرح واحد من قيمة المتغير

الآن لنأخذ بعض الامثلة على المتغيرات

مثال (١)

```
$a=3;  
$a+=5;
```

المتغير كان يحمل القيمة (٣) ثم اضيف اليه (٥) فالنتائج يكون (٨) .
نتيجة :

إذا كنا نريد أن نجمع أي قيمة رقمية على متغير نكتب (+=) ثم القيمة .

مثال (٢)

```
$b="My";  
$b.="program";
```

المتغير كان يحمل النص My ثم طلبنا اضافة النص program ليصبح النص الموجود في المتغير هو
Myprogram

نتيجة

إذا كنا نريد أن نضيف إلى متغير نصي نكتب (.=) ثم النص المراد اضافته .

مثال (٣)

```
$a="sager";  
$$a="My name";
```

وضعنا في المتغير الاول النص sager وفي السطر الثاني طلبنا من البرنامج أن ينشأ لنا متغير اسمه النص الموجود في المتغير a ويضع فيه القيمة My name أي أصبح لدينا متغيرين هما \$a و \$sager كما لو اننا كتبنا الجملة التالية :

```
$a="sager";  
$sager="My name";
```

نستخدم هذه الطريقة عندما نريد أن نربط متغير معين بمتغيرات أخرى مثل الاسم وكلمة السر .

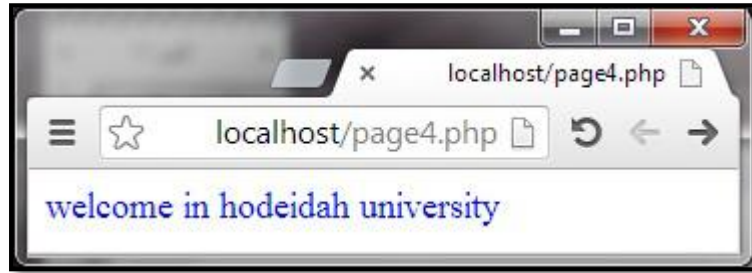
مثال على المتغيرات

```
<html >  
<body>  
<?php  
$a="blue";  
>  
<font color= <?php echo $a; ?> size=4>  
welcome in hodeidah university  
</font>  
</body></html>
```

قم بحفظ الملف باسم page4.php .

قم بأخذ الملف ووضعه في مجلد السيرفر لديك .

اكتب في مستعرض الانترنت لديك [http:// localhost/page4.php](http://localhost/page4.php) وسوف تشاهد التالي :



قم بعمل (rendered) للبرنامج ؟

```
1 <html>
2 <body>
3 <font color=blue size=4>
4 welcome in hodeidah university
5 </font>
6 </body>
7 </html>
```

التعليقات : Comments

عندما نكتب برنامج ونعود اليه بعد مدة نجد اننا قد نسينا الكثير من الخطوات التي قمنا بها أو لماذا كتبنا هذا الامر أو ذاك أو عمل هذه الدالة أو تلك ، بالإضافة إلى أن هذا البرنامج قد يقوم أشخاص آخريين بالتعديل عليه تستخدم التعليقات في الإفادة عن شرح الأكواد أو إضافة معلومات لا تستعمل إلا كتوضيح أو أي شيء آخر . ولذلك تعتبر التعليقات من الامور التي تبين أن المبرمج محترف وعمله منظم ولكتابة التعليقات نتبع التالي: يمكنك عمل تعليق من سطر واحد كالتالي :

```
<?php
// هذا تعليق لا فائدة له له اي معني
?>
```

مثال آخر :

```
<?php
// هذه الدالة تقوم بطباعه الكلمة تعليق
Echo "تعليق";
?>
```

وأیضا يمكنك استخدام تعليق من أكثر من سطر كالتالي :

```
<?php
/* تعليق يتكون من
اكثر من سطر بعلامة السلاش والنجمه
*/
?>
```


دوال التعامل مع المتغيرات

▪ الدالة gettype

تستخدم هذه الدالة من أجل أن تعرف نوع متغير ما .
مثال :

```
<?php
$n=5;
$l="hi";
echo "The n Is " . gettype ($n) . "<br>";
echo "The l is " . gettype ($l);
?>
```

▪ الدالة settype

تستخدم هذه الدالة من أجل تحويل نوع متغير ما .
مثال :

```
<?php
$n = 10 ;
echo "Before is " . gettype ($n) . "<br>";
settype ($n,"string");
echo "After That is go " . gettype ($n);
?>
```

▪ الدالة isset

لمعرفة إذا كان المتغير منشأ مسبقاً أم لم يتم انشاؤه وهي لا تتطلب غير اسم المتغير الذي تريد فحص وجوده وتقوم بإرجاع القيمة (1) إذا كان المتغير تم انشاؤه ولا ترجع أي قيمة إذا كان المتغير غير منشأ أو موجود

مثال :

```
<?php
$n="n";
Echo isset ($n);
?>
```

▪ الدالة unset

تقوم بحذف المتغير إذا كان موجوداً وتحرير الذاكرة منه (لذلك تأكد جيداً قبل استخدام هذه الدالة من اعطاء دمهعة الوداع للمتغير المسكين)

```
<?php
$n = "n";
unset ($n);
Echo isset ($n);
?>
```

▪ الدالة empty

تقوم بإرجاع القيمة (١) إذا كان المتغير غير منشأ أو أن القيمة التي فيه صفر (٠) أو نص فارغ ("") ولا تقوم بإرجاع أي شيء إذا كان المتغير منشأ وفيه قيم غير المذكورة .

```
<?php
$var=0;
if(empty($var)){
Echo "المتغير غير موجود أو فارغ";
}
?>
```

▪ الدالة doubleval

بواسطة هذه الدالة تحول المتغير الى نوع الرقمي المضاعف من أي نوع آخر .

```
<?php
<?php
$var='200.234 Arab programs';
$double_var=doubleval("13.5cm");
Echo $double_var;
?>
```

▪ الدالة intval

تحول هذه الدالة المتغير الى integer

```
<?php
<?php
$var="13.5cm";
$int_var=intval("13.5cm");
Echo $int_var;
?>
```

▪ الدالة var_dump

تقوم هذه الدالة بتحديد نوع المتغير والبيانات الموجودة فيه.

```
<?php
$n = 100;
Echo var_dump($n);
?>
```

في أثناء كتابتك للبرنامج سوف تلاحظ أنك بحاجة الى اتخاذ قرار معين ، مثلاً هل اسمح للزائر بالدخول الى صفحة معينة أو الاطلاع على معلومات محددة ؟ وبالطبع فإن الاجابة على هذا السؤال تعتمد على تحقق شروط معينة مثل هل يملك هذا الشخص صلاحيات لدخول الموقع وهل كلمة السر التي ادخلها صحيحة ؟ وهكذا نحن بحاجة الى اتخاذ مثل هذه القرارات دائماً ولكي نتخذ القرار يجب أن نتأكد من أن الشرط يتحقق مثلاً هل كلمة السر صحيحة ، فإذا كانت كلمة السر صحيحة يعيد لنا البرنامج القيمة True بينما اذا لم تكن صحيحة فسيعيد لنا القيمة false

اذن الشرط و عبارة عن مقارنة بين القيمة التي يدخلها المستخدم والقيمة الموجودة لدينا فاذا كانت القيمتان متساويتان فيعني انه ادخل كلمة السر الصحيحة وهذا يعني انه يمكنه الدخول الى الموقع .

الدالة الشرطية IF () (اذا كان فـ)

في هذه الدالة الشرطية نقوم بالتأكد من الشرط وسوف يكون لدينا هنا حالتان إما أن يتحقق الشرط وهنا سوف يتم الدخول الى الدالة وتنفيذ العمليات الموجودة بداخلها ثم الانتقال وتنفيذ الاوامر التي تقع خارجها ، أو أن لا يتحقق الشرط وهنا سوف يتم الانتقال مباشرة الى نهاية الدالة الشرطية وتنفيذ الاوامر التي تقع خارجها .

الصيغة العامة للدالة :

```
If ( الشرط ) {
    نفذ هذا اذا تحقق الشرط
}
```

مثال :

```
<?php
$a="Ahmed";
if($a=="Ahmed")
echo "you welcome";
?>
```

الدالة الشرطية If else (اذا كان فـ وذا لم يكن فـ)

في هذه الدالة الشرطية نقوم بالتأكد من الشرط وسوف يكون لدينا هنا حالتان إما أن يتحقق الشرط وهنا وسوف يتم الدخول الى الدالة وتنفيذ العمليات الموجودة بداخلها ثم الانتقال وتنفيذ اوامر التي تقع خارجها ، أو أن لا يتحقق الشرط وهنا سوف يتم تنفيذ الاوامر بعد كلمة else ثم الانتقال مباشرة الى نهاية الدالة الشرطية وتنفيذ الاوامر التي تقع خارجها .

الصيغة العامة :

```
If ( الشرط ) {
    نفذ هذا اذا تحقق الشرط
}
else {
    نفذ هذا اذا لم يتحقق الشرط
}
```

مثال :-

```

<?php
$a="Ahmed";
$b="123";
if($a=="Ahmed" and $b=="123")
echo "you welcome";
else
echo "check your Id";
?>

```

الدالة الشرطية elseif (إذا كان ف أو إذا كان ف أو إذا لم يكن ف)

في هذه الدالة الشرطية نقوم بالتأكد من الشرط وسوف يكون لدينا هنا حالتان إما أن يتحقق الشرط وهنا سوف يتم الدخول الى الدالة وتنفيذ العمليات الموجودة بداخلها ثم الانتقال وتنفيذ الاوامر التي تقع خارجها ، أو ان لا يتحقق الشرط وهنا سوف يتم التأكد من الشرط الثاني وهكذا الثالث واذا لم تتحقق الشروط جميعاً يتم تنفيذ الاوامر الموجودة بعد كلمة else ثم الانتقال مباشرة الى نهاية الدالة الشرطية وتنفيذ الاوامر التي تقعد خارجها

الصيغة العامة :

```

If( الشرط الاول ){
نفذ هذا اذا تحقق الشرط الاول
}
elseif( الشرط الثاني ){
نفذ هذا اذا تحقق الشرط الثاني
}
else{
نفذ هذا اذا لم يتحقق اي من الشروط
}

```

مثال :-

```

<?php
$name="Ahmed";
$password="123";
if($name=="Ahmed" and $password=="123")
echo "أهلا وسهلا بك ";
elseif($name=="Ahmed" or $password=="123")
echo "أهلا وسهلا بك ";
}
else
echo " عفوا ";
?>

```

الدالة الشرطية (Switch - case)

عندما يكون لدينا متغير مثلا اسم المستخدم ونريد أن نتأكد هل هو محمد أم خالد أم سعد أم فهد فعلينا استخدام الدالة if و elseif وتكرارها بشكل ممل ويجعل متابعة البرنامج فيما بعد متعبة جداً ولذلك من الأفضل هنا استخدام الدالة Switch التي سوف تسهل العمل كبير كما سنرى .
صيغتها :-

<pre>switch(Expression) { case 1 : statement 1; break; case 2 : statement 2; break; . . . case x : statement x; break; default : statement; }</pre>	<p>Expression التعبير الذي يتم اختباره</p>
	<p>case 1 تعني الحالة المحتملة الأولى</p> <p>case 2 تعني الحالة المحتملة الثانية</p> <p>وهكذا</p> <p>Default الحالة الافتراضية التي تنفذ إذا لم يتحقق أي من الحالات السابقة</p> <p>break جملة للتوقف والخروج من جملة switch</p>

مثال ١ :-

```
<?php
$g= "ahmed";
Switch ($g) {
    Case "ahmed":
        Echo "مسموح";
        Break ;
    Case "khaled " :
        Echo " ممنوع " ;
        Break ;
    Case "salem":
        Echo " ممنوع " ;
        Break ;
    Case "Mohmed " :
        Echo " مسموح " ;
        Break ;
    Default ;
        Echo "لقد ادخلت اسم غير صالح";
}
?>
```

مثال آخر :

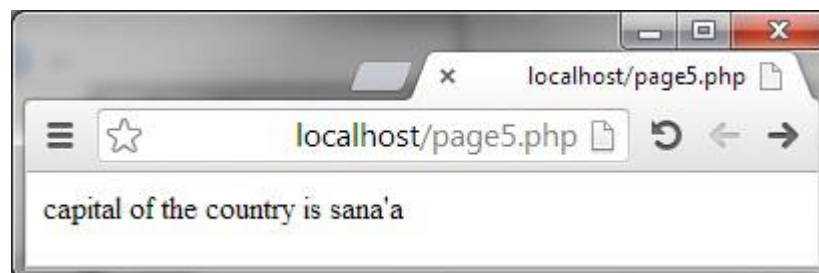
```
<?php
$g=4;
Switch ($g) {
    Case $g>50:
        Echo " كبير ";
        Break ;
    Case 40 :
        Echo " لا باس " ;
        Break ;
    Case ($g<15) :
        Echo " أطفال ممنوع " ;
        Break ;
    Case 30 :
        Echo " مسموح " ;
        Break ;
}
?>
```

لاحظ أننا عند اختبارنا لنصوص نحتاج الى علامتي تنصيب مزدوجة وعند الارقام فأننا لا نحتاج الى ذلك .

مثال:

```
<html><body>
<?php
$n="Yemen";
$c="capital of the country is ";
switch($n){
case"Yemen": echo $c."sana'a";
                break;
case"KSA": echo $c."Riyadh";
                break;
default: echo "this is not a name of the country";
}
?>
</body></html>
```

قم بحفظ الملف باسم page5.php ووضعه في مجلد السيرفر لديك .
اكتب في مستعرض الانترنت لديك [http:// localhost/page5.php](http://localhost/page5.php) وسوف تشاهد التالي :



قم بعمل (rendered) للبرنامج ؟ على الطالب

المعاملات المنطقية

وهي المعاملات المنطقية التي تساعدنا في صناعة الشروط والتقييدات على شي معين وتعطينا تحكما أكبر في الكود .

الصيغة	العملية
$\$a == \b	المتغير ان متساويان
$\$a === \b	المتغير ان متساويان ومن نفس النوع
$\$a != \b	المتغير ان غير متساويان
$\$a !== \b	المتغير ان غير متساويان او ليسا من النوع نفسه
$\$a < \b	المتغير الاول اقل من الثاني
$\$a > \b	المتغير الاول اكبر من الثاني
$\$a <= \b	المتغير الاول اقل او يساوي الثاني
$\$a >= \b	المتغير الاول اكبر او يساوي الثاني

■ المعاملات : < و >

من المفترض أن تكون متألفاً مع علامتي الأكبر من والأصغر من في الرياضيات التي تتعلمها في المدرسة مما يجعل فهم هذا الأمثلة بسيطاً .

```
<?php
If (6>5)
{
echo "الرقم ستة أكبر من الرقم خمسة " ;
}
Echo "end";
?>
```

سيقوم الـ PHP في مثالنا هذا بفحص الشرط (6>5) فإذا كان صحيحاً (true) سيقوم بطباعة السطر (الرقم ستة أكبر من الرقم خمسة) ثم يقوم بطباعة end ، وإذا لم يكن صحيحاً فسيقوم بتجاهل الكود وطباعة (end) فقط .

يمكننا أيضاً استعمالها في المقارنة بين متغير ورقم أو بين متغير وثابت (constant) أو العكس أو المقارنة بين متغيرين .

مثال (١)

```
<html dir="rtl">
<?php
$a = 5;
If ($a<6){
echo ("رقم الحظ أصغر من الرقم ستة ");
}
?>
```

مثال (٢)

```
<html dir="rtl">
<?php
$f=5;
$r=10;
If ($f>$r)
{
echo "$r أكبر من المتغير $f المتغير ";
}
?>
```

■ معاملات المساواة : == و ===

لقد قمنا باستخدام علامة المساواة الفردية سابقاً في تخزين قيمة في متغير وهانحن نأخذ نوعاً من علامات المساواة وهو علامة المساواة المزدوجة (==) و علامة المساواة المضاعفة (===) . لقد كنا نستخدم علامة المساواة الفردية او العادية في تخزين القيم في المتغيرات .
مثال :

```
<?php
$m=12;
?>
```

ولكن العلامات التي نتكلم عنها الآن تستخدم في تحديد إذا ما كانت قيمة معينة تساوي قيمة اخرى.
مثال :

```
<?php
$m="11";
$u=11;
If ($m==$u){
Echo "القيم متساوية ";
}
?>
```

لاحظ أن \$m متغير حرفي وان \$u متغير رقمي . إذا كنا نريد ارجاع قيمة إلى متغير نستخدم علامة المساواة العادية (=) وإذا أردنا اختبار متغيرين أو قيمة معينة من أنها متساوية نقوم باختبار القيم بواسطة علامة المساواة المزدوجة (==) . في الـ php4.01 تم إصدار علامة مساواة جديدة تقوم باختبار القيم ولا تعطي القيمة (true) إلا إذا كانت أنواع القيم متساوية وأنواع البيانات في المتغيرات ايضاً متساوية .

مثال (١) :

```
<?php
$m="11";
$u=11;
If ($m==$u){
Echo "القيم متساوية ";
}
?>
```

مثال (٢) :

```
<?php
$m="11";
$u=11;
If ($m=== $u)
{
Echo "القيم متساوية ";
}
?>
```

التوضيح

لاحظ أننا في المثال الأول استخدمنا علامة المساواة المزدوجة لاختبار القيم وكانت القيم متساوية في المتغيرين فتم طباعة أن القيم متساوية (مع أن نوع البيانات مختلف) ولكن في المثال الثاني عندما استخدمنا علامة المساواة المضاعفة لم يتم طباعة أي شيء وذلك لأن القيم متساوية ولكن نوع البيانات مختلف فالمتغير \$m حرفي بينما المتغير \$u رقمي .

■ المعاملات : != و <>

إن عكس علامة المساواة هي علامة عدم المساواة (!=) مثال :

```
<?php
If (5!=99)
echo "القيم غير متساوية ";
?>
```

لاحظ أن ٥ لا تساوي ٩٩ لذلك فإن الشرط صحيح (true) لذلك قام بطباعة أن القيم غير متساوية . إن الضد من علامة أكبر من وأصغر من هو علامة الـ (<>) وهو يقوم بإرجاع قيمة (true) إذا كانت القيمتين مختلفتين عن بعضهما أي أنه مثل علامة != تقريباً . مثال :

```
<?php
If (5<>99)
echo "القيم غير متساوية ";
?>
```

المعاملات المنطقية (AND,OR,NOT)

إن هذه المعاملات المنطقية تتيح لك بتنفيذ الكود بعد التحقق من مجموعة شروط وأيضاً تنفيذ الكود إذا تحقق أكثر من شرط : (AND) أو تحقق شيء معين من بين عدة أشياء : (OR) ويمكنك مثلاً التحقق من عدم صحة شيء لكي تقوم بتنفيذ شيء آخر : (NOT) .
 فيمكنك مثلاً أن تقول : إذا كان الجو ممطراً والعاصفة شديده فلن أخرج من البيت .
 ويمكنك أن تقول : إذا كان الجو هادئاً أو لا يوجد أمطار فسأقوم بالخروج الى المنتزه .
 ويمكنك أيضاً أن تقول : إذا لم يكن الجو ممطراً سأقوم بالخروج إلى نزهة .
ملاحظة : عند استخدامك لهذه الدوال عليك مراعاة أن تقوم بجعل هذه الشروط بين قوسين .

المعامل (AND) ونظيره (&&)

يمكننا استعمال المعامل (AND) والمعامل (&&) للتحقق من صحة عدة شروط لتنفيذ شيء معين
 مثال (١)

```
<?php
$w=10;
$g=12;
IF ($w=10 and $g=12)
echo ("لقد تحققت جميع الشروط");
?>
```

مثال (٢)

```
<?php
$w=10;
$g=12;
IF ($w=10 && $g=15)
echo ("لقد تحققت جميع الشروط");
?>
```

في المثالين السابقين قمنا بعملية التحقق من أكثر من شرط باستخدام المعاملين (&& و and) فعندما تحققت جميع الشروط تم تنفيذ الأمر وعندما لم تكن جميع الشروط صحيحة تم تجاهل الأمر .
 لاحظ أننا قمنا بجعل الشروط بين قوسين () لكي يعمل الكود بشكل صحيح :

```
($w=10 && $g=15)
($w=10 and $g=12)
```

المعامل (OR) ونظيره (||)

المعامل OR يقوم بالتحقق من عدة شروط وإذا تحقق أي واحد منها فإنه يقوم بتنفيذ الكود ونظيره (||) الذي يقوم بنفس العملية .

مثال (١)

```
<?php
$E=100;
$T=8;
IF ($E=14 OR $E=55 OR $E = 10 OR $T=8 ) echo ("لقد تحقق أحد هذه الشروط");
?>
```

```
<?php
$E=100;
$T=458;
IF ($E=14 || $E=55 || $E = 10 || $T=8 )
echo ("لقد تحقق أحد هذه الشروط");
?>
```

إذن عندما تحقق واحد من هذه الشروط تم طباعة السطر (لقد تحقق أحد هذه الشروط) .

■ المعامل NOT ونظيره (!)

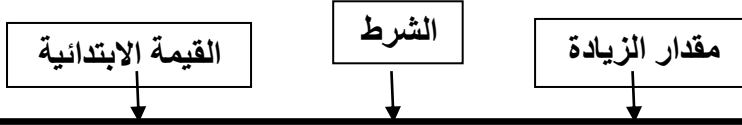
في الواقع لا يمكنك استخدام NOT أبدا لأنها ليست أصلاً موجودة في لغة PHP لكن يمكنك استخدام المعامل (!) كبديل لها فهو يؤدي نفس وظيفتها وهي القيام بالتأكد من أن هناك قيمة غير صحيحة (FALSE) لكي يتم تنفيذ شيء معين .

```
<?php
$F="الفارس";
IF !($F=="نعمان")
echo ("أهلاً بك");
?>
```

في المثال السابق يقوم الـ PHP بالتأكد من أن المتغير \$F لا يحتوي على القيمة الحرفية (نعمان) ويتم ذلك باستخدام المعامل (!) وعندما يتم التأكد من ذلك يقوم بطباعة السطر (أهلاً بك) ونشير إلى أننا نقوم باختبار متغير بواسطة المعامل (!) فإن الـ PHP إذا وجد المتغير فارغاً أو لم يتم انشاؤه يعطيه القيمة صفر وهي FALSE .

أولاً : - حلقة الدوران For :

شكلها العام :



```
For ( initial value ; condition ; increment ){
Statements; نفذ مادام الشرط متحقق
}
```

مثال

```
<html>
<body>
<?php
for($i=1;$i<10;$i++){
print $i;
}
?>
</body></html>
```

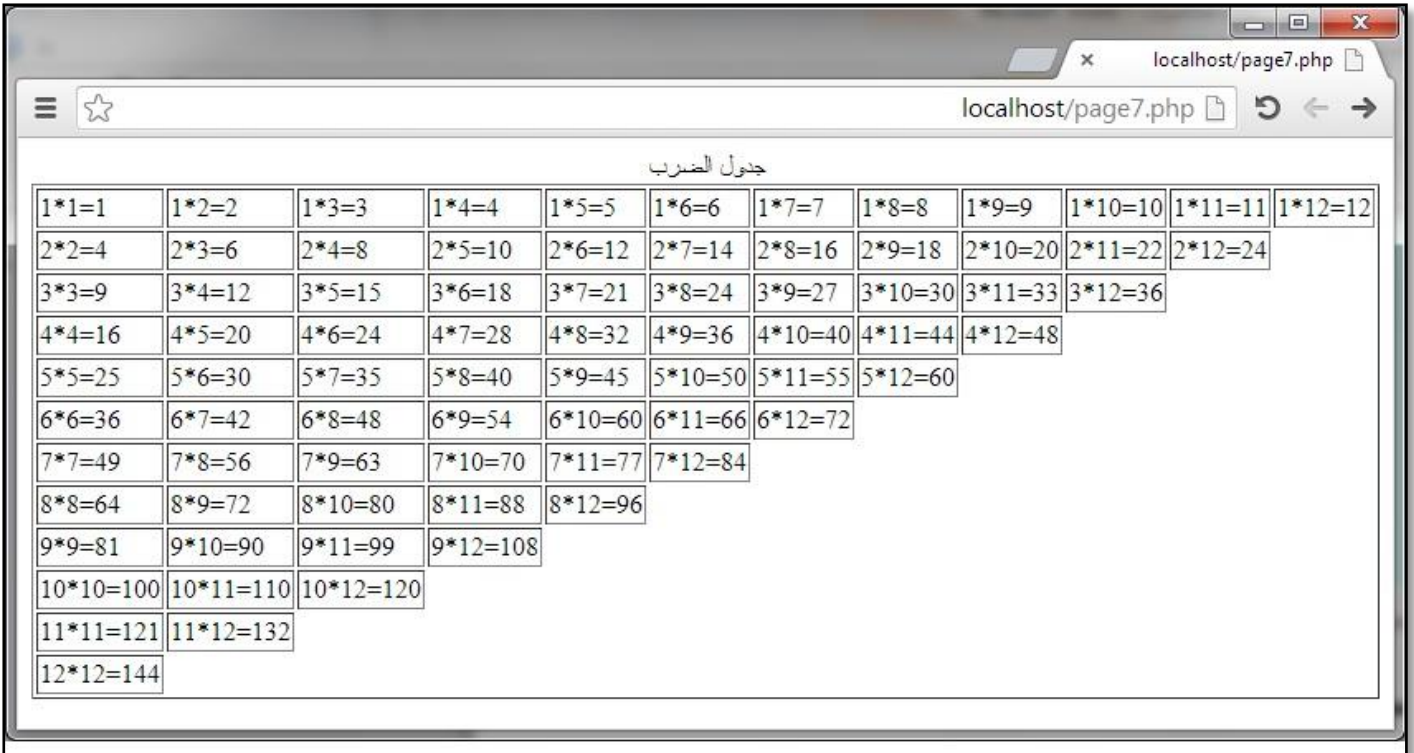
قم بحفظ الملف باسم page6.php ووضعه في مجلد السيرفر لديك.
اكتب في مستعرض الانترنت لديك [http:// localhost/page6.php](http://localhost/page6.php) وسوف تشاهد طباعة الاعداد التالية (123456789) قم بعمل (rendered) للبرنامج ؟
مثال آخر : برنامج يقوم بإيجاد جدول الضرب ووضعه داخل جدول من جدول (1) الى جدول (12) .

```
<html> <body>
<table border=1>
<caption>جدول الضرب</caption>
<?php
for($i=1;$i<=12;$i++)
{
echo "<tr>";
for($j=$i;$j<=12;$j++)
{
echo "<td>";
echo $i."*".$j."=".$i*$j;
echo "</td>";
}
echo "</tr>";
}
?>
</body></html>
```

قم بحفظ الملف باسم page7.php ووضعه في مجلد السيرفر لديك.

مقدمة في لغة PHP

اكتب في مستعرض الانترنت لديك [http:// localhost/page7.php](http://localhost/page7.php) وسوف تشاهد التالي :



جدول الضرب

1*1=1	1*2=2	1*3=3	1*4=4	1*5=5	1*6=6	1*7=7	1*8=8	1*9=9	1*10=10	1*11=11	1*12=12
2*2=4	2*3=6	2*4=8	2*5=10	2*6=12	2*7=14	2*8=16	2*9=18	2*10=20	2*11=22	2*12=24	
3*3=9	3*4=12	3*5=15	3*6=18	3*7=21	3*8=24	3*9=27	3*10=30	3*11=33	3*12=36		
4*4=16	4*5=20	4*6=24	4*7=28	4*8=32	4*9=36	4*10=40	4*11=44	4*12=48			
5*5=25	5*6=30	5*7=35	5*8=40	5*9=45	5*10=50	5*11=55	5*12=60				
6*6=36	6*7=42	6*8=48	6*9=54	6*10=60	6*11=66	6*12=72					
7*7=49	7*8=56	7*9=63	7*10=70	7*11=77	7*12=84						
8*8=64	8*9=72	8*10=80	8*11=88	8*12=96							
9*9=81	9*10=90	9*11=99	9*12=108								
10*10=100	10*11=110	10*12=120									
11*11=121	11*12=132										
12*12=144											

قم بعمل (rendered) للبرنامج ؟

```
<html>
<body>
<table border=1>
<caption>جدول الضرب</caption>
<tr><td>1*1=1</td><td>1*2=2</td><td>1*3=3</td><td>1*4=4</td><td>1*5=5</td><td>1*6=6</td><td>1*7=7</td><td>1*8=8</td><td>1*9=9</td><td>1*10=10</td><td>1*11=11</td>
<td>1*12=12</td></tr><tr><td>2*2=4</td><td>2*3=6</td><td>2*4=8</td><td>2*5=10</td><td>2*6=12</td><td>2*7=14</td><td>2*8=16</td><td>2*9=18</td><td>2*10=20</td>
<td>2*11=22</td><td>2*12=24</td></tr><tr><td>3*3=9</td><td>3*4=12</td><td>3*5=15</td><td>3*6=18</td><td>3*7=21</td><td>3*8=24</td><td>3*9=27</td><td>3*10=30</td>
<td>3*11=33</td><td>3*12=36</td></tr><tr><td>4*4=16</td><td>4*5=20</td><td>4*6=24</td><td>4*7=28</td><td>4*8=32</td><td>4*9=36</td><td>4*10=40</td><td>4*11=44</td>
<td>4*12=48</td></tr><tr><td>5*5=25</td><td>5*6=30</td><td>5*7=35</td><td>5*8=40</td><td>5*9=45</td><td>5*10=50</td><td>5*11=55</td><td>5*12=60</td></tr><tr>
<td>6*6=36</td><td>6*7=42</td><td>6*8=48</td><td>6*9=54</td><td>6*10=60</td><td>6*11=66</td><td>6*12=72</td></tr><tr><td>7*7=49</td><td>7*8=56</td><td>7*9=63</td>
<td>7*10=70</td><td>7*11=77</td><td>7*12=84</td></tr><tr><td>8*8=64</td><td>8*9=72</td><td>8*10=80</td><td>8*11=88</td><td>8*12=96</td></tr><tr><td>9*9=81</td>
<td>9*10=90</td><td>9*11=99</td><td>9*12=108</td></tr><tr><td>10*10=100</td><td>10*11=110</td><td>10*12=120</td></tr><tr><td>11*11=121</td><td>11*12=132</td></tr>
<tr><td>12*12=144</td></tr>
</table>
</body>
</html>
```

ثانياً : - حلقة الدوران While :

استمر في حلقة مادام الشرط متحققا ويتم التحقق من الشرط قبل الدخول الى الحلقة في كل مرة ، فعند بداية الحلقة نتأكد من الشرط فإذا تحقق نقوم بالدخول الى الحلقة وتنفيذ الاوامر مرة واحدة ثم نعود لرأس الحلقة مرة أخرى ونتأكد من الشرط وهكذا يستمر البرنامج حتى يصبح الشرط غير متحقق فيقوم بالخروج من الحلقة وتنفيذ الاوامر الموجودة خارجها .

وتساعد الحلقات كثيراً في اختصار الاوامر التي تكتب فلو اردنا أن نحصل على مائة سجل من قاعدة من قاعدة البيانات وطباعتها فاذا لم نستخدم الحلقة فستحتاج لمئات الاوامر ولكن بواسطة الحلقة فلن نكتب أكثر من عشرة أسطر .

الصيغة العامة :

```
While ( الشرط )
{
    نفذ دائما مادام الشرط متحققاً
}
```

مثال :

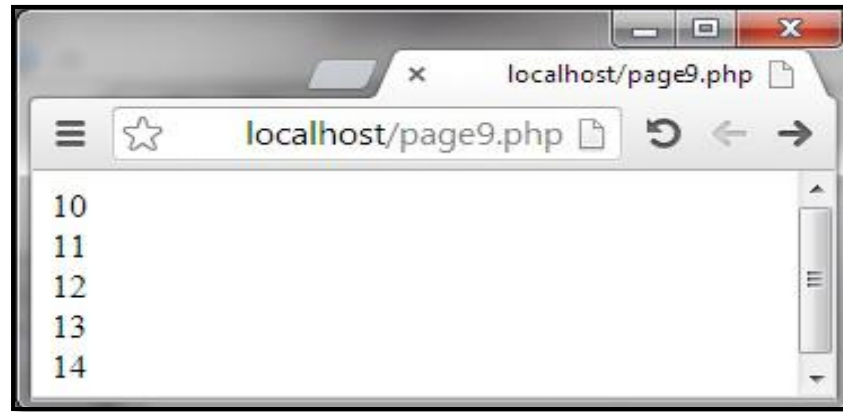
```
<html> <body>
<?php
$a=0;
While($a<18)
{
    ++$a;
    echo $a;
}
?>
</body></html>
```

قم بحفظ الملف باسم page8.php ووضعه في مجلد السيرفر لديك.
اكتب في مستعرض الانترنت لديك [http:// localhost/page8.php](http://localhost/page8.php) وسوف تشاهد طباعة الاعداد التالية(123456789101112131415161718) قم بعمل (rendered) للبرنامج ؟

مثال آخر :-

```
<html> <body>
<?php
$d =10 ;
while ($d<15)
{
    echo "$d <br>"; //echo $d."<br>";
    $d++;
}
?>
</body></html>
```

قم بحفظ الملف باسم page9.php ووضعه في مجلد السيرفر لديك.
اكتب في مستعرض الانترنت لديك [http:// localhost/page9.php](http://localhost/page9.php) وسوف تشاهد التالي:



قم بعمل (rendered) للبرنامج ؟

```
1 <html>
2 <body>
3 10<br>11<br>12<br>13<br>14<br>
4 </body>
5 </html>
```

ثالثاً: - حلقة الدوران do While :

هي نفس حلقة الدوران While ولكن الفرق بينهم أن هو أن البرنامج في الحلقة do while ينفذ مرة واحدة على الأقل حتى إذا لم يتحقق الشرط. في الحلقة while (طالما الشرط محقق نفذ) يتم اختبار الشرط أولاً ثم التنفيذ في الحلقة do while (نفذ طالما الشرط محقق) يتم التنفيذ ثم اختبار الشرط الصيغة العامة :

```
do
{
نفذ دائماً مادام الشرط متحققاً
}
While ( الشرط )
```

مثال :

```
<html> <body>
<?php
$a=0;
do{
++$a;
echo $a;
}
While ($a<18)
?>
</body></html>
```

قم بحفظ الملف باسم page10.php ووضعه في مجلد السيرفر لديك. اكتب في مستعرض الانترنت لديك [http:// localhost/page10.php](http://localhost/page10.php) وسوف تشاهد طباعة الاعداد التالية(123456789101112131415161718) قم بعمل (rendered) للبرنامج ؟

المصفوفات (Arrays)

كما مر معنا في الدروس السابقة ، يمكن للمتغيرات أن تحوي قيمة واحدة فقط ، فجاءت المصفوفات لتحل هذا القصور وتمكن المبرمج من تخزين عدة قيم في متغير واحد يسمى بالمصفوفة .
وإذا كنت قد تعاملت مع المصفوفات بلغات برمجة غير PHP ستجد أن PHP لها طريقة خاصة ومرونة كبيرة جداً في التعامل مع المصفوفات
المصفوفات تتكون من ما يعرف بمفتاح أو معرف العنصر داخل المصفوفة وهو الـ (index)

تعريف مصفوفة



تعريف مصفوفة رقمية الفهرس :-

تبدأ الفهرسة فيها من الصفر إلى أقل من عدد عناصر المصفوفة بمقدار واحد (لان العد يبدأ من الصفر) وكل عنصر من عناصر يحتوي على قيمة مرتبطة بهذا المفتاح لتخزين قيم ما على شكل مصفوفة عليك فقط أن تضع الاقواس المربعة [] بعد اسم المتغير ونقوم بأسناد القيم للمصفوفة
مثال :

```
<? php
$a[ ] = " waleed";
$a[ ] = 13;
$a[ ] = "yahya";
echo $a[0].$a[1].$a[2];
?>
```

في هذا المثال سيقوم الـ PHP بإعطاء الفهرسة تلقائياً فسيقوم بوضع الرقم فتصبح المتغير فهرسته كالتالي :

```
$a[0] = "waleed";
$a[1] = 13;
$a[2] = "yahya";
```

إننا لم نقوم بإدخال هذه الأرقام من تلقاء أنفسنا ولكن الـ PHP قام بوضعها مع أنه يمكننا أن ندخلها بشكل عادي فمثلاً لو كتبنا :

```
<?php
$a[0]= " waleed";
$a[1] = 13;
$a[2] = "yahya";
for($i=0;$i<=2;$i++)
echo $a[$i];
?>
```


سيقوم الـ PHP بأخذ الفهرسة المعتمدة ولن يضع أي فهرسة أخرى يمكننا أيضا أن نكتب أي فهرسة ولا نعتمد على الترتيب في الأرقام .
مثال :

```
<? php
$a[10 ] = " waleed";
$a[ 25] = 13;
$a[ 30] = "yahya";
?>
```

هل لاحظت أيضا أننا لم نقم بتعريف نوع متغيرات المصفوفة وقام الـ PHP بتعريفها تلقائياً بدلاً منا فمرة استخدمنا قيمة حرفية ومرة استخدمنا رقماً ورغم ذلك فلم يقيم الـ PHP بعمل أي اعتراض إضافة إلى ذلك فإن الـ PHP يقوم بتحديد عدد عناصر المصفوفة تلقائياً فهو يعرف مثلاً من المثال السابق أن عدد عناصر المصفوفة الكلي هو ثلاثة عناصر .

ولإعطاء قيم للمصفوفة عند تعريفها دفعة واحدة نستخدم الدالة (array) وتوضع العناصر داخل القوسين ويفصل بينهما بالفاصلة العادية كالتالي :

```
<?php      0      1      2      3
$t=array ("waleed", 13, "yahya", "yousif");
for($i=0;$i<=3;$i++)
echo $a[$i];
?>
```

لاحظ أن الـ PHP قام بإعطاء رقم الفهرسة وقام بالبداية من الصفر ولكن يمكننا جعل الـ PHP يبدأ الفهرسة من الرقم الذي نريده كالتالي :

```
<?php
$r = array (1=>"ahmed",2=> "ali",3=>"salem",4=> "alfarsi");
//$r = array (1=>"ahmed", "ali", "salem", "alfarsi");
?>
```

تعريف مصفوفة نصية الفهرس :-

تبدأ الفهرسة فيها من الصفر إلى أقل من عدد عناصر المصفوفة بمقدار واحد (لان العد يبدأ من الصفر) وكل عنصر من عناصر يحتوي على قيمة مرتبطة بهذا المفتاح لتخزين قيم ما على شكل مصفوفة عليك فقط أن تضع الاقواس المربعة [] بعد اسم المتغير ونقوم بأسناد القيم للمصفوفة
مثال :

```
<?php
$a["no"] =1;
$a["name"] ="waleed";
$a["email"] ="waleed_ahmed@yahoo.com";
echo $a["no"]. $a["name"].$a["email"];
?>
```

دوال التعامل مع المصفوفات

الدالة () foreach

دالة تستخدم للدوران على عناصر المصفوفة وتعتبر من أفضل الطرق للدوران على عناصر المصفوفة وخاصة المصفوفات نصية الفهرس .

تعتمد هذه الحلقة على وجود كائن نبحت في عناصره مثل المصفوفات نستمر في الحلقة مادام هناك عناصر موجودة من الكائن المحدد وتستخدم هذه الحلقة لتسهيل العمل مع المصفوفات نصية الفهرس .

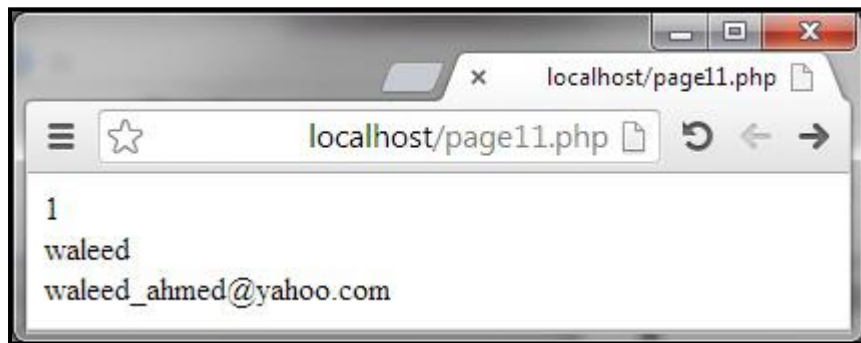
اسم المصفوفة
المحتوى
Foreach(\$a as \$v)

الصيغة العامة لهذه الدالة :

اسم المصفوفة
الفهرس (index)
المحتوى
Foreach(\$a as \$k=>\$v)

```
<?php
$a["no"] =1;
$a["name"] ="waleed";
$a["email"] ="waleed_ahmed@yahoo.com";
foreach($a as $v)
echo $v."<br>";
?>
```

قم بحفظ الملف باسم page11.php ووضعه في مجلد السيرفر لديك .
اكتب في مستعرض الانترنت لديك [http:// localhost/page11.php](http://localhost/page11.php) وسوف تشاهد التالي :



قم بعمل (rendered) للبرنامج ؟

```
1 1<br>waleed<br>waleed_ahmed@yahoo.com<br>
2
```


▪ الدالة array_reverse()

تقوم هذه الدالة بعكس ترتيب عناصر المصفوفة من النهاية الى البداية بحيث يصبح العنصر الاول هو الاخير والعنصر الاخير هو الاول .

مثال :

```
<?php
$a = array(1,2,3,4);
$b=array_reverse($a);
print_r($b);
?>
```

سوف يكون الناتج:

```
Array ( [0] => 4 [1] => 3 [2] => 2 [3] => 1 )
```

▪ الدالة Count()

تقوم هذه الدالة بارجاع عدد العناصر في المصفوفة .

```
<?php
$a = array("a","b","c");
Print count($a);
?>
```

سوف يكون الناتج ٣

▪ الدالة array_pad

تقوم هذه الدالة بجعل قيم المصفوفة تساوي عددا معينا وتضع بدل القيم الناقصة الافتراضية التي ترسلها لها .

```
<?php
$test1=array("a","b","f");
$test=array_pad($test1,5,"zzz");
print("<PRE>");
print_r($test);
print("</PRE>\n");
?>
```

▪ الدالة array_pop

تقوم هذه الدالة بحذف آخر عنصر في المصفوفة .

```
<?php
$test=array("a","b","f");
$lastcolor=array_pop($test);
print("<PRE>");
print_r($test);
print("</PRE>\n");
?>
```

تلاحظ أنه في المصفوفة test ثلاث عناصر و عند تنفيذ هذه الدالة عليها نحصل على التالي :

كما تلاحظ أصبح في المصفوفة عنصرين .

```
Array
(
  [0]=>a
  [1]=>b
)
```

الدالة array_push

تقوم هذه الدالة بإضافة قيمة أو اكثر إلى نهاية الدالة .

```
<?php
$test1=array("a","b","f");
$lastcolor=array_push($test,"d","y");
print("<PRE>");
print_r($test);
print("</PRE>\n");
?>
```

و عند تنفيذ هذا المثال نحصل على التالي :

```
Array
(
  [0] => a
  [1] => b
  [2] => f
  [3] => d
  [4] => y
)
```

الدالة array_count_values

عندما نريد أن نتعرف على تكرار قيم في مصفوفة معينة فأنا نستخدم هذه الدالة وسوف تعود لنا بمصفوفة جديدة تحتوي على اسم العنصر وعدد مرات تكراره في المصفوفة .

لنفترض أن لدينا المصفوفة التالية :

```
$test=array("a","h","a","b","b","a","d");
```

تلاحظ انه في هذه المصفوفة أن الحرف a يتكرر 3 مرات والحرف b يتكرر مرتين بينما الحرف d و h لا يتكرران اذن لمعرفة ذلك بواسطة هذه الدالة نكتب المثال التالي :

```
<?php
$test= array("a","h","a","b","b","a","d");
$count= array_count_values ($test);
Foreach($count as $key=>$value){
print("$key:$value <BR>\n");
}
?>
```

وهنا سوف يتم طباعة القيمة وعدد مرات تكرارها في المصفوفة

```
a:3
h:1
b:2
d:1
```

▪ الدالة array_shift

تقوم هذه الدالة بحذف أول قيمة من المصفوفة

```
<?php
$test=array("a","b","f");
$shift=array_shift($test);
print("<PRE>");
print_r($test);
print("</PRE>\n");
?>
```

Array

```
(
[0]=>b
[1]=>f
)
```

تلاحظ أن المصفوفة تحتوي على ثلاث عناصر وعند تنفيذ البرنامج :

نلاحظ أن قد تم حذف العنصر الأول من المصفوفة .

▪ الدالة array_keys

عندما نتعامل مع المصفوفة من المهم جداً أن نعرف رقم أو اسم العنصر لان هذا الرقم أو الاسم هو من نستطيع من خلاله معرفة قيمة العنصر أو الخاصية بالتحديد ، وحيث أن هناك الكثير من المصفوفات التي لاتعرف كم عنصر تعود به وماهي الارقام لهذه العناصر لذلك نستخدم هذه الدالة لتعود لنا فقط بأسم أو رقم جميع العناصر الموجودة في المصفوفة .

```
<?php
$test=array(1=>"a","ss"=>"h",4=>"arab");
foreach(array_keys($test) as $key){
print("$key<br>\n");
}
?>
```

في هذا المثال لدينا ثلاث عناصر هي :

الأول هو الرقم ١ وقيمه a

الثاني هو الاسم ss وقيمه h

الثالث هو الرقم ٤ وقيمه arab

ونحن هنا يهنا المفاتيح الاساسية (٤ ، ss ، ١) ولذلك استخدمنا الدالة array_keys لتعيد لنا اسماء هذه المفاتيح

```
1
ss
4
```

▪ الدالة array_merge

عندما يكون لدينا مجموعة من المصفوفات ونرى أنه من الأفضل دمجها فلن نحتاج الى اعادة كتابة هذه المصفوفات من جديد ، بل نحتاج فقط الى ارسال اسمائها الى هذه الدالة لتقوم بكل العمليات بكل سهولة .

```
<?php
$test1=array("a","b","c");
$test2=array("d","e","f");
$test=array_merge($test1,$test2);
print("<pre>");
print_r($test);
print("</pre>\n");
?>
```

في هذا المثال لدينا مصفوفتان الاولى test1 والثانية test2 وقمنا بدمجها في مصفوفة واحدة هي test وطلبنا من البرنامج طباعة عناصر المصفوفة الجديدة فيكون الناتج لدينا :

```
Array
(
    [0] => a
    [1] => b
    [2] => c
    [3] => d
    [4] => e
    [5] => f
)
```

▪ الدالة array_reverse

تقوم هذه الدالة بعكس ترتيب عناصر المصفوفة من النهاية الى البداية بحيث يصبح العنصر الاول هو الاخير والعنصر الاخير هو الاول

```
<?php
$test1=array("a","b","f");
$test=array_reverse($test1);
print("<pre>");
print_r($test);
print("</pre>\n");
?>
```

تلاحظ ترتيب عناصر المصفوفة اولاً a ثم b واخيراً f وعند تنفيذ البرنامج :

```
Array
(
    [0] => f
    [1] => b
    [2] => a
)
```

▪ الدالة in_array

تعود بقيمة true اذا كانت القيمة موجودة داخل المصفوفة

```
<?php
$test=array("a","b","c");
If(in_array("b",$test))
Print "found b";
?>
```

في هذا المثال اذا كانت القيمة موجودة سوف يتم طباعة الرسالة .

▪ الدالة array_slice

تقوم هذه الدالة بنسخ مجموعة قيم من مصفوفة على شكل مصفوفة جديدة ، ويتم الحصول على القيم عن طريق تحديد بداية النسخ وعدد القيم التي تنسخ .

```
<?php
$test1=array("a","b","f","d","g","p");
$test=array_slice($test1,2,3);
print("<pre>");
print_r($test);
print("</pre>\n");
?>
```

في هذا المثال نطلب من البرنامج نسخ القيم من المصفوفة ابتداء من القيمة رقم ٢ ويعود بعدد ٣ قيم وسوف تكون النتيجة

```
Array
(
    [0] => f
    [1] => d
    [2] => g
)
```

ملاحظة : عندما نريد الحصول على جميع القيم ابتداء من قيمة معينة فقط نرسل للدالة قيمة البداية

▪ الدالة array_unique

تقوم هذه الدالة بارجاع القيم من مصفوفة بدون تكرار

```
<?php
$test1=array("a","b","a","a","p","p");
$test=array_unique ($test1);
print("<pre>");
print_r($test);
print("</pre>\n");
?>
```

وبعد تنفيذ البرنامج نلاحظ انها اعادتنا لنا القيم بدون تكرار

```
Array
(
    [0] => a
    [1] => b
    [4] => p
)
```


دوال فرز المصفوفات

هناك العديد من الدوال التي يوفرها لنا الـ PHP لفرز المصفوفات . ومن هذه الدوال :

الدالة (Sort)

تقوم هذه الدالة بترتيب عناصر مصفوفة رقمية الفهرس تصاعدياً فإذا كانت عناصر المصفوفة ارقام يتم الترتيب من القيمة الاقل الى القيمة الاكبر واذا كانت عناصر المصفوفة حروف يتم الترتيب اعتماداً على الاحرف الكبيرة أولاً ثم الصغيرة .. تتطلب هذه الدالة اسم المصفوفة التي سيتم عليها الفرز

Sort (ArrayName);

إذا قمنا بإنشاء مصفوفة بالشكل التالي :

```
$a=array ("ali","salem","hythem","Khaled","Ammar","Hesham");
```

فإذا أردنا فرزها عن طريق الدالة sort() فإننا نقوم باستخدامها كالتالي :

```
<?php  
$a=array ("ali","salem","hythem","Khaled","Ammar","Hesham");  
sort($a);  
for($i=0;$i<=5;$i++)  
echo $a[$i]."&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;";  
?>
```

لاحظ أنه عند تنفيذك للمثال ستجد أن الـ PHP قام بالفرز اعتماداً على الأحرف الكبيرة أولاً ثم قام بالفرز بعدها اعتماداً على الأحرف الصغيرة .

سيكون الناتج

```
Ammar Hesham Khaled ali hythem salem
```

وإذا قمنا بتغيير المصفوفة بالشكل التالي :

```
$a=array(6,96,0,1,50,13);
```

فسيكون الناتج :

```
0 1 6 13 50 96
```

وإذا قمنا بتغيير المصفوفة بالشكل التالي :

```
$a=array("ali","salem","hythem",8,"Ammar",0);
```

فسيكون الناتج :

```
ali 0 Ammar hythem salem 8
```

الدالة (rsort)

تقوم هذه الدالة بترتيب عناصر مصفوفة رقمية الفهرس تنازلياً هي عكس الدالة السابقة حيث تتطلب هذه الدالة اسم المصفوفة التي سيتم عليها الفرز

RSort (ArrayName);

الدالة () asort

تقوم هذه الدالة بترتيب عناصر مصفوفة نصية الفهرس تصاعدياً حسب المحتوى (value) فإذا كانت عناصر المصفوفة ارقام يتم الترتيب من القيمة الاقل الى القيمة الاكبر وإذا كانت عناصر المصفوفة حروف يتم الترتيب اعتماداً على الاحرف الكبيرة أولاً ثم الصغيرة .. تتطلب هذه الدالة اسم المصفوفة التي سيتم عليها الفرز RSort (ArrayName);

مثال

```
<?php
$a=array("ad"=>"ahmed","kh"=>"Khaled");
asort($a);
foreach($a as $x=>$y)
echo "$x,$y <br>";
?>
```

فسيكون الناتج:

```
kh,Khaled
ad,ahmed
```

الدالة () arsort

تقوم هذه الدالة بترتيب عناصر مصفوفة نصية الفهرس تنازلياً حسب المحتوى (value) فإذا كانت عناصر المصفوفة ارقام يتم الترتيب من القيمة الاكبر الى القيمة الاقل وإذا كانت عناصر المصفوفة حروف يتم الترتيب اعتماداً على الاحرف الصغيرة أولاً ثم الكبيرة .
مثال :

```
<?php
$a=array("ad"=>"ahmed","kh"=>"Khaled");
arsort($a);
foreach($a as $x=>$y)
echo "$x,$y <br>";
?>
```

فسيكون الناتج:

```
ad,ahmed
kh,Khaled
```

الدالة () ksort

تقوم هذه الدالة بترتيب عناصر مصفوفة نصية الفهرس تصاعدياً حسب الفهرسة (index) .
مثال :

```
<?php
$a=array("ad"=>"ahmed","kh"=>"Khaled");
asort($a);
foreach($a as $x=>$y)
echo "$x,$y <br>";
?>
```

ad,ahmed
kh,Khaled

الدالة () krsort

تقوم هذه الدالة بترتيب عناصر مصفوفة نصية الفهرس تنازلياً حسب الفهرسة (index).
مثال:

```
<?php
$a=array("ad"=>"ahmed","kh"=>"Khaled");
krsort($a);
foreach($a as $x=>$y)
echo "$x,$y <br>";
?>
```

فسيكون الناتج:-

kh,Khaled
ad,ahmed

دوال الحركة لمؤشر قراءة المصفوفة

عندما نقوم بقراءة المصفوفة يقوم البرنامج بوضع مؤشر وهمي عند المتغير الذي نقرأه الان ومن خلال المؤشر نستطيع التوجه للأمام أو الخلف داخل المصفوفة واليك في الجدول التالي دوال الحركة وعمل كل منها:

عملها	الدالة
تصفير المؤشر ووضعه في بداية المصفوفة	Rest
التوجه للعنصر التالي	Next
التوجه للعنصر السابق	Prev
قيمة المتغير الحالي	Current
وضع المؤشر عند اخر عنصر	End

```
<?php
$test=array("a","b","c");
for(reset($test);$value=current($test);next($test))
{
print (" $value <br>\n");
}
?>
```

وعند تنفيذ البرنامج يتم طباعة القيم الموجودة في المصفوفة.

a
b
c

في هذا الجزء سوف نتحدث عن كيفية انشاء دوال خاصة بك داخل البرنامج ولكن في البداية دعنا نتعلم ماهي الدوال ولماذا نحتاجها .
 الدالة : هي مجموعة من الاوامر والمتغيرات تكون بشكل مستقل داخل البرنامج ولكن البرنامج يتجاهلها ولا ينفذها إلا اذا تم استدعائها عن طريق كتابة اسم الدالة وارسال المتغيرات المطلوبة لها .
 وقد تتساءل هنا ما اهمية هذه الدوال أو لماذا لا اكتب الاوامر بدون الدخول إلى هذه المتاهة؟ والجواب انك سوف تجد ان هذه الدوال سوف توفر لك الكثير من الوقت بالإضافة الى جعل برنامج مرتب وسهل التطوير فمثلا عندما نحتاج ان نتأكد من اسم المستخدم وكلمة المرور في برنامجنا وهذه سوف نحتاج اليها عشرات المرات في برامج قواعد البيانات فكل ما علينا هو كتابة دالة نرسل لها الاسم وكلمة السر لتتأكد من ان هذا المستخدم لديه الصلاحيات في الدخول واجراء التغييرات .

ما هي الصيغة التي نكتب بها الدوال

كل دالة يجب ان تبدأ بالكلمة Function ثم اسم الدالة وهو اسم تختاره انت ثم تكتب المتغيرات التي تود ان ترسلها للدالة وبعد هذا تكتب العمليات التي تريد من الدالة القيام بها وفي النهاية سوف تعيد لك الدالة متغير نتيجة هذه العمليات ويجب ان تسبق المتغير المعاد الكلمة return حسب ما شرحنا فإن الصيغة العامة للدالة هي :

```
Function اسم الدالة ( ) {
```

```
هنا الاوامر والعمليات
```

```
}
```

مثال :

```
<?php
Function AddNo($no1,$no2){
$no3=$no1 + $no2;
Return $no3;
}
Print AddNo(5,3);
Print AddNo(4,2);
?>
```

في هذا المثال انشئنا دالة اسمها AddNo وتستقبل متغيرين وتقوم بعملية جمع المتغير الاول مع المتغير الثاني وتعيد لنا النتيجة ، وتلاحظ اننا استدعينا الدالة مرتين الاولى ارسلنا لها رقمين هما ٥ و ٣ وسوف يعيد لنا الرقم ٨ وفي المرة الثانية ارسلنا لها رقمين ٤ و ٢ وسوف يعيد لنا الرقم ٦ ، وايضاً تلاحظ اننا كتبنا الدالة قبل امر الاستدعاء لكي تتأكد فقط أن البرنامج لا ينفذها إلا اذا تم استدعائها فلو مسحت سطر الاستدعاء فلن يحدث شيء عند تنفيذ البرنامج .

ملاحظات حول الدوال :-

في المثال السابق يحدث خطأ اذا قام المستخدم باستدعاء الدالة دون أن يرسل لها قيم أو ارسال لها قيمة واحدة فقط فهل يمكن أن اجعل الدالة تضع قيمة افتراضية اذا لم يتم المستخدم بارسال قيم فارغة ؟

نعم يمكن يمكن وضع قيم افتراضية للمتغيرات التي تستقبلها الدالة وهذه القيم تستخدمها الدالة فقط في حالة لم يتم المستخدم بارسال أي قيمة واليك المثال التالي :

```
<?php
Function AddNo($no1,$no2){
$no3=$no1 + $no2;
Return $no3;
}
Print AddNo();
Print AddNo(4);
Print AddNo(4,2);
?>
```

في هذا المثال وضعنا قيم افتراضية للمتغير الاول والثاني وهي القيمة صفر ثم استدعينا الدالة ثلاث مرات الاولى بدون ارسال أي قيم فكانت النتيجة صفر ، والثاني بارسال القيمة 4 فكانت النتيجة 4 واخيرا ارسلنا قيمتين فكانت النتيجة حاصل جمعهما .

- في المثال السابق اريد ان ارسل قيمتين و أود أن تعيد لي حاصل الجمع والضرب فهل اعمل دالة جديدة أم يمكن أن تعيد لي الدالة أكثر من قيمة ؟

في الحقيقة الدالة تعيد قيمة واحدة فقط ولكن نستطيع التغلب على هذا بواسطة عدة طرق من اهمها أن نجعل الدالة تعيد لنا مصفوفة من القيم فلكي نحقق الهدف نحتاج أن نجعل الدالة تعيد لنا مصفوفة تتكون من رقمين الاول حاصل الجمع والثاني حاصل الضرب والعملية في غاية السهولة واليك المثال :

```
<?php
Function AddNo($no1,$no2){
$no3=$no1 + $no2;
$no4=$no1 * $no2;
Return array($no3,$no4);
}
List($back1,$back2)=AddNo(2,6);
Print $back1;
Print $back2;
?>
```

في هذا المثال جعلنا البرنامج يعيد لنا مصفوفة تحتوي على حاصل الجمع والضرب وعند استدعائنا للمصفوفة طلبنا وضع القيم القادمة في متغيرين الاول \$back والثاني \$back2 وبعد ذلك يمكن التعامل مع هذه المتغيرات كما نشاء وهنا اخترنا طباعتها .

- في المثال السابق أود ان احصل على المتغيرات بدون استخدام return ولا المصفوفات فقد تعلمت في المتغيرات أن بإمكانني أن اجعل المتغير عام فكيف افعل ذلك هنا ؟

مع انني افضل دائما استخدام المصفوفات لما فيها من المرونة والقوة ولكني الاحظ ان الكثير من المبرجين يستخدمون المتغيرات العامة بكثرة داخل الدوال ولكل رأيته الخاصة في هذا المجال ولذلك رأيت أن اجعل الطريقتين معروفتين لديك وتستخدم ما تشاء وإليك نفس المثال السابق ولكن بالمتغيرات العامة :

```
<?php
Function AddNo($no1=0,$no2=0){
$GLOBALS ["no3"]=$no1 + $no2;
$GLOBALS ["no4"]=$no1 * $no2;
}
AddNo(2,3);
Print $no3;
Print $no4;
?>
```

مثال :

```
<?php
Function r($y){
If($y>=50){
$m="old";
}
else{
$m="young";
}
return $m;
}
Echo r(50);
?>
```

في هذا البرنامج يتم اختبار القيمة المرسله للدالة اذا كانت اكبر من أو يساوي ٥٠ يطبع كلمة old أما اذا كانت اقل من ٥٠ يطبع كلمة young

التعامل مع الوقت والتاريخ

تستطيع بواسطة PHP الحصول على تاريخ اليوم بل أكثر دقة تستطيع أن تطلبه منه أن يعطي لك اسم اليوم والساعة والدقيقة والثانية وهل الوقت صباحاً أو مساءً وهل تريد أن تحصل على اسم الشهر مختصراً (sep) أو كاملاً (September) .

ولكي تحصل على التاريخ والوقت هناك دالتان لهذا الغرض وهما :
الدالة date والدالة gmdate وكلاهما يقومان بنفس المهمة ويأخذان نفس المتغيرات ولكن الاختلاف الوحيد بينهما أن الدالة gmdate تحاول الرجوع بالوقت على حسب التوقيت العالمي (توقيت غرينش) .
لكي تحصل على الوقت من الدالة date عليك كتابة الصيغة التالية :

Date(" ماذا تريد ")

فلو اردنا أن نعرف السنة الان فإننا سوف نكتب :

Date("Y")

لاحظ اننا كتبنا حرف Y بشكل كبير اذن سوف تعيد لنا الدالة السنة كاملة .
عموماً الدالة data تأخذ المتغيرات التالية:

المتغير	يعود بـ
Y	يعود بالسنة من اربعة ارقام مثل ١٩٩٩
y	يعود بالسنة من رقمين مثل ٩٩
w	يعود برقم اليوم من الاسبوع الاحد=٠ الاثنين=١ وهكذا حتى نصل الى السبت=٦
t	يعود بعدد ايام الشهر الحالي
s	يعود بالثواني
S	يعود بامتداد ارقام اليوم th أو nd
U	يعود بالثواني بالجزء من الثانية
Z	تعود برقم اليوم من السنة
F	تعود باسم الشهر كاملاً مثل September
M	تعود باسم الشهر مختصراً مثل Sep
m	تعود برقم الشهر من ٠١-١٢
n	تعود برقم الشهر من ١-١٢
j	تعود برقم اليوم من الشهر من ١ - ٣١
l	(حرف L صغير) تعود باسم اليوم كاملاً مثل Thursday

i	تعود بالدقائق من ٠٠ - ٥٩
G	تعود بالساعة بنظام ٢٤ ساعة من ١ - ١٢
g	تعود بالساعة بنظام ١٢ ساعة من ١ - ٢٤
D	تعود باسم اليوم مختصراً مثل Wed
d	تعود برقم اليوم من الشهر من ٠١ - ٣١
a	تعود بـ am أو pm
A	تعود بـ AM أو PM

من دمج القيم الموجودة في الجدول السابق نستطيع الحصول على عدة اشكال من التاريخ والوقت حسب حاجتنا واليك بعض الامثلة :

- للحصول على اليوم والشهر والسنة بالأرقام نكتب التالي :

```
print date("j/n/Y");
```

وسوف تكون النتيجة مشابهة لـ (17/10/2013)

- للحصول على الساعة والدقائق والثواني بالإضافة الى AM أو PM نكتب التالي :

```
print date("h:i:s A");
```

وسوف تكون النتيجة مشابهة لـ (10:32:52 PM)

وهكذا نستطيع الحصول على التشكيلة التي تريد مهما كان طبعها .

مثال :-

لو اردنا طباعة اسم اليوم بالعربي أي السبت أو الاحد وهكذا كيف نستطيع ذلك ؟

لحل هذه المسألة نعود للجدول السابق لنجد أن المتغير w يعيد لنا رقم اليوم في الاسبوع وهنا نرى هذا المتغير كم

قيمه فإذا كانت = ٠ فهو يوم الاحد أما اذا كانت = ١ فهو يوم الاثنين وهكذا بقية الايام وسوف نستخدم الدالة

الشرطية Switch التي شرحناها من قبل

```
<?php
$a= date("w");
Switch ($a) {
    Case 0:
        Echo "الاحد";
        Break ;
    Case 1 :
        Echo " الاثنين " ;
        Break ;
    Case 2:
        Echo " الثلاثاء " ;
        Break ;
```



```

Case 3 :
    Echo " الاربعاء " ;
    Break ;
Case 4 :
    Echo " الخميس " ;
    Break ;
Case 5 :
    Echo " الجمعة " ;
    Break ;
Case 6 :
    Echo " السبت " ;
    Break ;
}
?>

```

التعامل مع الارقام والدوال الرياضية

الان سوف نتعرف على الدوال التي تعطينا القدرة على التعامل مع الارقام والحصول منها على مخرجات محددة حسب متطلبات البرنامج .

▪ الدالة bcadd

تتيح هذه الدالة بجمع رقمين مهما كان نوعهما كما تتيح لك تحديد عدد الارقام بعد الفاصلة العشرية التي تدخل ضمن هذه العملية ونرسل لهذه الدالة ثلاث متغيرات الاول والثاني هما الارقام المطلوب جمعها بينما المتغير الثالث هو عدد الارقام بعد الفاصلة العشرية التي تدخل ضمن نطاق الجمع فلو ارسلنا لها صفر فسوف تهمل اي رقم خلف الفاصلة العشرية وتعود لنا بحاصل جمع الرقمين الصحيحين فقط .
مثال :

```

<?php
print(bcadd("1.59","9.87654",0));
print "<br>";
print(bcadd("1.59","9.87",1));
print "<br>";
print(bcadd("1.59","9.87",6));
?>

```

في هذ المثال وضعنا الحالات المحتملة لجمع رقمين وسوف تكون النتيجة كمايلي :

```

11
11.4
11.460000

```

تلاحظ أن النتيجة اختلفت فقط بتغيير قيمة المتغير الثالث الذي يتحكم بشكل المخرجات وطبيعتها .

▪ الدالة `bccomp`

تقوم هذه الدالة بالمقارنة بين رقمين لتعود لك بنتيجة المقارنة فاذا كان الرقمان متساويان تعود بالقيمة صفر ، أما اذا كان الرقم الموجود باليسار أكبر من الرقم الموجود باليمين فتعود بالقيمة ١ ، واخيراً اذا كان الرقم الموجود باليمين أكبر من الرقم الموجود باليسار فسوف تعود بالقيمة -١ .
وهذه الدالة تتجاهل الأرقام بعد الفاصلة العشرية الا اذا ارسلت لها قيمة ثالثة تحدد فيها عدد الأرقام التي تضمن في عملية المقارنة .

```
<?php
print(bccomp("1","1.11111"));
print "<br>";
print(bcadd("1","1.111111111111",1));
print "<br>";
print(bcadd("2","1.1111"));
?>
```

وسوف تكون النتيجة كما يلي :

كما تلاحظ اعتبرت الدالة الرقمين الأولين متساويين لأننا طلبنا منها تجاهل الأرقام بعد الفاصلة العشرية ، بينما الرقمين التاليين كانت الجهة اليمنى أكبر بعد أن طلبنا من الدالة ادخال رقم واحد بعد الفاصلة العشرية .
واخيراً كان الجانب الأيسر أكبر لذلك عاد بالقيمة ١ .

▪ الدالة `bcsqrt`

تقوم هذه الدالة بإعادة الجذر التربيعي للرقم الذي سوف نرسله لها ، وايضاً نستطيع أن نحدد عدد الأرقام بعد الفاصلة العشرية التي نعرضها
مثال

```
<?php
print(bcsqrt("16"));
print "<br>";
print(bcsqrt("17"));
print "<br>";
print(bcsqrt("17",3));
?>
```

وسوف تكون النتيجة التالية :

4

4

4.123

تلاحظ أن جذر ١٦ و ١٧ كانت نتيجتهما متساوية عندما لم نطلب من الدالة ارجاع أرقام عشرية .
بينما العملية الاخيرة كان ناتجها مختلفاً بحيث اشتمل على الأرقام العشرية

▪ الدالة abs

تعود بالقيمة الحقيقية للرقم
مثال :

```
Print(abs(-13));
```

تعود بالرقم ١٣

▪ الدالة max

تعود بأكبر قيمة من بين القيم المرسله لها سواء كانت أرقام أو نصوص
مثال

```
<?php  
print(max(2,5,16));  
print "<br>";  
print(max("arab",15,33));  
print "<br>";  
print(max("arab","sager"));  
?>
```

وسوف تعود بالقيم التالية :

16

33

Sager

▪ الدالة Min

نفس طريقة عمل الدالة السابقة ولكنها تعود بأصغر قيمة من القيم المرسله لها .

▪ الدالة ceil

مهمة هذه الدالة تقريب الرقم الى أكبر رقم صحيح لاحق .
مثال :

```
Print(ceil(13.01));
```

سوف نحصل على الرقم ١٤

▪ الدالة log

الحصول على لوغارتم العدد
مثال :

```
Print(log(20.13));
```

▪ الدالة sqrt

تعود بالجذر التربيعي للرقم المرسل لها
مثال :

```
Print(sqrt(9.9));
```

النماذج Forms

النماذج في الويب أو صفحات الانترنت عبارة عن استمارات تقوم بتعبئتها ثم عند إرسالها لخدم الويب (السيرفر) يتلقاها برنامج يقوم بإجراء العمليات عليها مثل JavaScript أو ASP أو php (في حالتنا) .

فائدة النماذج

لنقل أنك مثلاً أردت شراء كتاب من الانترنت فإنك في الواقع تحتاج إلى تعبئة استمارة ببياناتك ورقم بطاقة الائتمان وغير ذلك من المعلومات ويتم ذلك عن طريق نموذج (فورم) . في الواقع أنت تقوم باختيار الكتاب الذي تريد وتكتب اسمك ورقم هاتفك وصندوق بريدك (ربما) في فراغات أو عن طريق الإشارة الى مجموعة من الخيارات .

يتم تخزين هذه القيم في المتغيرات التي يتم كتابتها في الخاصية name (نتكلم عنها في هذا الدرس) ويتم إرسالها عند ضغط زر - ارسال البيانات - (submit) الى (البرنامج) الصفحة التي سوف تقوم بمعالجة هذه البيانات (والتي يتم تحديدها في الخاصية ACTION) وإجراء العمليات عليها مثل تخزينها مثلاً في قاعدة البيانات أو إرسالها إلى البريد الإلكتروني وذلك عن طريق الـ php .

ماذا يعمل العميل في النماذج ؟

إنه باختصار يقوم بتعبئة مربعات نصوص (textBox) ويقوم بوضع علامة صح في مربعات الاختيار (check boxes) أو يقوم بالتصويت أحياناً لشي معين فيختر زر اختيار (ازرار الراديو) .

هذه الأشياء كلها يتم انشاءها بواسطة الـhtml ودرسنا لهذا اليوم يناقش كيفية انشاءها وكيفية التعامل والحصول على البيانات منها ، بقي علينا كبداية أن نعرف أن هذه الأدوات تنشأ في الواقع بين وسمين من وسوم لغة الـhtml وهي الـوسمين

<form> </form>

خصائص النماذج

يجمع النموذج جميع خصائص المضيف لكننا هنا سنتطرق الى اثنين منهما وهما ACTION و METHOD التي تستخدم بكثرة و مهمة لنا في دروسنا القادمة

ACTION

وظيفة هذه الخاصية أن تخبر السيرفر مكان الصفحة التي يقوم بإرسال معلومات النموذج إليها أو عنوانها أي كان نوعها ، وطبعاً في حالتنا ستكون الصفحة الثانية هي الصفحة التي تحتوي على سكربت الـ php . ليس مهماً أن تكون الصفحة php فقد تكون html ولكنها تحتوي على كود يختص بالتعامل مع برنامج تفاعلي لصفحات الويب مثل الجافا . ولا نريد أن نخرج عن نطاق الموضوع فدعنا نعطي مثلاً على هذه الخاصية :

```
<FORM ACTION ="TEST.PHP">
```

```
.....
```

```
</FORM>
```

METHOD

هذه الخاصية تقوم بإخبار النموذج طريقة ارسال المعلومات الى الصفحة الهدف وفي الواقع هناك طريقتين مشهورتين ومعروفتين لارسال المعلومات هما GET و POST .

```
<FORM ACTION ="test.php" METHOD = "GET">
```

أو

```
<FORM ACTIN = "test.php" METHOD ="POST">
```

أيهما تستخدم GET أم POST ؟

تقوم هذه الخاصية بإخبار مستعرض الانترنت لديك بأن يقوم بإضافة المعلومات التي تمت كتابتها في النموذج إلى متصفح الانترنت لديك وقد يكون العيب في الخاصية GET عدم سرية المعلومات التي تقوم بكتابتها ومن الممكن أن تظهر للشخص الذي يجلس الى جوارك ... خاصة عندما تريد الحفاظ على سرية معلوماتك .
أضف إلى ذلك أنها غير مفيدة في النصوص الكبيرة الحجم .

ولكنها مفيدة في أشياء كثيرة فمثلاً محركات البحث يجب أن تستخدم هذه الخاصية لكي يستطيع المستخدم أن يستخدم عنوان البحث ويحتفظ به لوقت آخر ولا يقوم من جديد بكتابة الكلمة التي يبحث عنها .

أيضاً POST مفيدة في إخفاء المعلومات وإحتواء كميات كبيرة من البيانات ولكن لا يمكن الاحتفاظ بعنوان الصفحة مع ذلك فإنها أيضاً ليست جيدة في الحماية بحيث أن أي هكر خبير يمكنه الحصول على المعلومات إذا لم يكن لها تشفير معين في نقلها .

أدوات التحكم في النماذج :

في الواقع أن أدوات التحكم عبارة عن مربعات النصوص العادية (التي يدخل فيها المستخدم اسمه وعنوانه) وازرار الراديو (والتي يقوم المستخدم فيها باختيار شيء معين (مثل الوجبة المفضلة لديه او المشروب المفضل اليه) ومربعات الاختيار (التي تتيح للمستخدم أن يختار ما يشتهي ويحب من الخيارات المعروضة) وأيضاً القوائم التي تساعدك على اختيار أكثر من شيء أو شيء واحد .
في أغلب هذه الأشياء يتم استعمال الوسم

<INPUT>

وتلخيص تفصيله كالتالي :

<INPUT TYPE= type NAME= name VALUE= value other attribute>

الشرح :

1 - TYPE= type

نحدد نوع الكائن إذا كان زر راديو أو مربع نص عادي أو مربعات الاختيار .

2 - NAME= name

تقوم فيها بإعطاء اسم لمتغير يتم حفظ القيمة فيه .

3 - VALUE= value

سيوضح وظيفته أكثر عندما ندرج عليه أمثله إذ أن عمله يختلف من أداة إلى أخرى .

تطبيقات عمليه

سنقوم في هذه التطبيقات بصنع برامج بسيطة تتكون من ملفين ، الملف الاول يحتوي على كود HTML يقوم بتكوين النموذج والملف الثاني يقوم بإستقبال النتائج وطباعتها .

✓ مربعات النصوص (TEXT Box) :

مثال :

```
<html dir="rtl">
<FORM METHOD="GET" ACTION="textbox.php">
ماهي وجبتك المفضلة في الصباح ؟
<br>
<INPUT TYPE=text name=food>
<br>
<INPUT TYPE=submit value="إرسال">
<INPUT TYPE=reset value="مسح">
</form>
</html>
```

- ١ - قم بحفظ الملف كصفحة HTML . وقم بتسميته (textbox.html) .
- ٢ - افتح محرر النصوص إذا كنت أغلقته .
- ٥ - اكتب الكود التالي :

```
<?php
$t1=$_GET['food'];
Echo "$t1 " : وجبتك المفضلة والتي تموت في حبها هي "
?>
```

- ٦ - قم بحفظ الملف ك php . وقم بتسميته textbox.php .
- ٧ - الآن قم بأخذ الملفين وضعهما في مجلد السيرفر لديك .
- ٨ - اكتب في مستعرض الانترنت لديك

<http://localhost/textbox.html>

- ٩ - قم بكتابة وجبتك المفضلة واضغط زر إرسال.
- ١٠ - ستظهر النتيجة .

لاحظ كيف ظهر العنوان :

<http://localhost/textbox.php?food=القيمة التي تمت كتابتها في مربع النص>

```
<html>
<form action="textbox2.php" method="post">
<table border="1" width="400">
<tr>
<td>Name </td>
<td><input type="text" name="f1"> </td>
</tr>
<tr>
<td>Age </td>
<td><input type="text" name="f2"> </td>
</tr>
</tr>
<td><input type="submit" value="Check"> </td>
<td><input type="reset" value="Rest"> </td>
</tr>
</table>
</form>
</html>
```

- قم بحفظ الملف كصفحة HTML . وقم بتسميته (textbox2.html) .
- ٢ - افتح محرر النصوص إذا كنت أغلقته .
 - ٥ - اكتب الكود التالي :

```
<?php
$name=$_POST['f1'];
$age=$_POST['f2'];
If($name=="mohamed" && $age==25){
echo "Name: $name and Age: $age" ;
}
else{
echo "error";
}
?>
```

- ٦ - قم بحفظ الملف ك php . وقم بتسميته textbox2.php .
- ٧ - الآن قم بأخذ الملفين وضعهما في مجلد السيرفر لديك .
- ٨ - اكتب في مستعرض الانترنت لديك

<http://localhost/textbox2.html>

✓ مربعات النصوص الكبيرة (text area)

إذا كنت تريد أن تكتب رسالة متعددة الأسطر فإنك تحتاج إلى أداة تحكم تختلف تماماً عن مربع النص العادي وهي مربعات النصوص الكبيرة التي يمكنك فيها من إدخال نصوص كبيرة الحجم ومتعددة الأسطر . تستخدم هذه الأداة وسم فتح ووسم إغلاق

<TEXTAREA>

</TEXTAREA>

ويمكنك تحديد حجمها بواسطة تحديد الصفوف بالخاصية rows والأعمدة بالخاصية cols .
مثال :

```
<html><body>
ما هي وجبتك المفضلة ؟
<br><FORM METHOD="GET" ACTION="tarea.php">
<TEXTAREA NAME =food ROWS="10" COLS ="50" >
جبنة
مربي
مكرونة
همبرجر
سمبوسة
معصوب
مطبق
بيض
</TEXTAREA>
<br>
<INPUT TYPE =SUBMIT VALUE ="قم بإرسال الطلبات إلى الجرسون">
</FORM></body>
</html>
```

- 1 - قم بحفظ الملف باسم TAREA.html .
- 2 - الآن قم بفتح ملف جديد في محرر النصوص .
- 3 - قم بكتابة الكود التالي :

```
<html dir="rtl">
<?php
$t1=$_GET['food'];
Echo " ".$t1 ." : وجبتك المفضلة والتي تموت في حبها هي " ;
?>
```

- 4 - قم بحفظ الملف باسم tarea.php
- 5 - قم بوضعهما في مجلد السيرفر لديك .
- 6 - قم بتشغيل البرنامج .

<http://localhost/tarea.html>

- 7 - قم بضغط الزر لإرسال البيانات .

✓ مربعات الاختيار (CheckBox) اكثر من خيار في وقت واحد !

في الواقع قد نرى مربعات الاختيار في صفحات الويب عندما نريد الاشتراك في موقع معين لرؤيه محتوياته أو عندما نريد تسجيل بريد إلكتروني أو حجز مساحة عند موقع . وفائدتها هي إتاحة فرصة للمستخدم لتحديد أنواع الأشياء التي يريد أن يشترك فيها مثلاً أو إتاحة فرصة له لقبول إتفاقية أو غير ذلك أو رفض الجميع أو قبول الجميع .

يمكننا صنع مربع العلامة بواسطة الوسم INPUT

```
<INPUT TYPE = "CHECKBOX" NAME = "swalif " value= "سوالف" >
```

نقوم بتحديد نوع الأداة بأنها مربع علامة في هذا الجزء

```
TYPE = "CHECKBOX"
```

نقوم بتحديد اسم المتغير في هذا الجزء

```
NAME = "swalif "
```

ونقوم بتحديد القيمة التي يتم وضعها في المتغير اذا قام المستخدم باختيار مربع العلامة في هذا الجزء :

```
value= "سوالف "
```

اذا لم تقم بوضع الخيار value فستكون القيمة الافتراضية هي on عند اختيار المستخدم مربع العلامة وستكون فراغ اذا لم يقم المستخدم باختيار المربع.

ونقوم بوضع القيمة الافتراضية بإضافة الكلمة checked فإذا تم وضع هذه الكلمة يكون مربع العلامة مختار تلقائياً أما إذا لم نكتبها فسيكون بدون علامة الاختيار .

مثال :

١ - قم بفتح المفكرة وقم بكتابة الكود التالي :

```
<html dir="rtl">
<form action="check.php" method="get">
ما الذي تريد أن تفعله في الحياة ؟ (يمكنك اختيار أكثر من إجابة)
<br>
<INPUT type=checkbox name=t>
<br>
الذي أريد أن أفعله في الحياة هو أنني أتزوج وأخلص وافتك من الزهق.
<input type=submit value ="إرسال">
</FORM>
</html>
```

٢ - قم بحفظ الملف باسم check.html .

٣ - قم بفتح ملف جديد في المفكرة وقم بكتابة التالي :

```
<html dir="rtl">
<?php
$t1=$_GET[' t '];
Echo $t1;
?>
```

٤ - قم بحفظ الملف باسم check.php .

٥ - قم بنقل الملفين الى مجلد السيرفر لديك .

٦ - اكتب في المتصفح العنوان التالي :

<http://localhost/check.html>

مثال (٢) :-

١ - افتح المفكرة واكتب الكود التالي وقم بحفظه في ملف جديد باسم check2.html

```
<html dir="rtl">
<FORM ACTION="check2.php" METHOD ="GET">
ما الذي تريد أن تفعله في الحياة ؟ (يمكنك اختيار أكثر من إجابة)
<br>
<INPUT TYPE=CHECKBOX name="e" value="زوجة">
الذي اريد افعله في الحياة هو ادنى أتزوج واخلص واقتك من الزهق
<br>
<INPUT TYPE=CHECKBOX name="b" value="جهاد">
ابغي اروح الجهاد واخمع رؤوس الكفرة والمشركين
<br>
<INPUT TYPE=CHECKBOX name="c" value="قران">
والله لو ألتحق بتحفيظ قرآن واحفظ القران كامل وأطبقه في عملي وحياتي حرتاح في حياتي كثير
<br>
<input type=submit value ="ارسال">
</FORM>
</html>
```

٢ - قم بفتح ملف جديد وقم بوضع الكود التالي فيه :

```
<html dir ="rtl">
<?php
$t1=$_GET['e'];
$t2=$_GET['b'];
$t3=$_GET['c'];
Echo $t1.$t2.$t3;
?>
</html>
```

٣ - قم بحفظه باسم check2.php

٤ - قم بنقل الملفين الى مجلد السيرفر لديك .

٥ - اكتب في المتصفح العنوان التالي :

<http://localhost/check2.html>

✓ ازرار الراديو (RADIO BUTTONS) (اختر المشروب المفضل !)

ما هو اختيارك المفضل؟ علماً بأنه لا يمكنك اختيار أكثر من خيار واحد!!
في الواقع إن زر الراديو يتيح لك أن تختار شيء واحد من بين عدة اختيارات ونراه كثيراً عند اتفاقيات البرامج حيث يعطيك فرصه إما بقبول الاتفاقية أو رفضها ويكون واحد من الاختيارين محددًا (وهو خيار الرفض!).

يتم استخدام ازرار الراديو باستخدام العبارة <INPUT> كالتالي :

```
<INPUT TYPE ="radio" NAME = "name" value= "value" checked>
```

نقوم بتحديد نوع الكائن بأنه زر راديو في هذا الجزء :

```
TYPE ="radio"
```

نقوم بتحديد اسم المتغير في هذا الجزء :

```
NAME ="name"
```

نقوم بتحديد القيمة التي ستكون في المتغير هنا :

```
value="value"
```

في الواقع مع ازار الراديو نقوم بجعل اسم المتغير name هو نفسه والقيم مختلفة value لكل سؤال . وإذا لم نقم بوضع قيمة فسيقوم PHP بوضع القيمة on للمتغير .

مثال (١) :-

١- قم بتشغيل محرر النصوص لديك واكتب الكود التالي وقم بحفظه في ملف اسمه radio.html .

```
<html dir="rtl">
<form action ="radio.php" method ="GET">
<br> ما هو مشروبك المفضل ؟
<INPUT TYPE =radio NAME =e value="شاي">
شاي
<br>
<INPUT TYPE =radio NAME =e value="قهوة">
قهوة
<br>
<INPUT TYPE =submit value="ارسال">
</form>
</html></html>
```

٢- قم بفتح محرر النصوص واكتب الكود التالي وقم بحفظه باسم radio.php

```
<html dir="rtl">
<?php
$t1=$_GET['e'];
echo " $t1 : مشروبك المفضل هو "
?>
```

٣- قم بنقل الملفين الى مجلد السيرفر لديك .

٤- اكتب في المتصفح العنوان التالي :

<http://localhost/radio.html>

✓ القوائم (Lists Or drop down menus) اختر مواصفات زوجتك للمستقبل واسمها :

تستخدم القوائم في الـ html بشكل مختلف قليلاً عن الأدوات السابقة إذ أننا نستخدم وسمين من وسوم لغة html وهما :

<select> لنقوم بإنشاء القائمة و <OPTION> ونستخدم الخاصية MULTIPLE إذا كنا نريد إتاحة الفرصة للمستخدم أن يختار أكثر من قيمة ونقوم بوضع القيمة التي يختارها المستخدم في متغير بواسطة الخاصية NAME أو في مصفوفة متغيرات .

تطبيق عملي :

١- قم بفتح محرر النصوص لديك واكتب الكود التالي واحفظه في ملف باسم lists.html :

```
<html dir="rtl">
<form action="lists.php" method="GET">
ماذا تريد ان يكون اسم زوجة المستقبل(لغير المتزوجين) ؟
<br>
<select name="wife">
<option> هناء </option>
<option> جمانة </option>
<option> رزان </option>
<option> سحر </option>
<option> سارة </option>
<option> سمية </option>
<option> روان </option>
<option> دلال </option>
</select>
<INPUT TYPE=SUBMIT VALUE="ارسال">
</html>
```

٢- قم بفتح ملف جديد واكتب فيه الكود التالي وقم بحفظه باسم lists.php :

```
<html dir="rtl">
<?php
$t1=$_GET['wife'];
Echo " . $t1 . " : لقد أردت أن يكون اسم زوجتك "
?>
</html>
```

٣- قم بتنشغيل البرنامج

<http://localhost/lists.html>

٤- اختر ما تريد ثم اضغط زر ارسال

✓ الاداة الخفية (والمعلومات السرية!) (hidden control)

هناك بعض الأوقات تحتاج فيها إلى إرسال بعض المعلومات من صفحة ويب إلى صفحة ويب أخرى عن طريق النماذج وفي نفس الوقت أنت لا تريد المستخدم أن يقوم برؤية هذه المعلومات . في الواقع هناك أداة تساعدك على إخفاء هذه المعلومات على المستخدم يسمونها بحقل النموذج المخفي أو الاداة الخفية (hidden form field or hidden control) . هذه الاداة تلعب دوراً مختلفاً ومتميزاً عن بقية الأدوات وهي إخفاء المعلومات التي تم ادخالها كما شرحنا في السابق وهي مفيدة جداً مع النماذج المصنوع بواسطة الـ PHP إذ أنها تسمح لنا أيضاً بان تكون المعلومات المخفية هي متغيرات PHP . يتم صنع هذه الحقول المخفية كالتالي :

```
<INPUT TYPE=HIDDEN NAME =hidden1 VALUE="الرسالة السرية">
```

نقوم بوضع HIDDEN لكي يعرف المتصفح أن هذه المعلومات خفية (لا تظهر للمستخدم) ونضع اسماً للمتغير الذي يقوم بالاحتفاظ بالمعلومات والذي يتخزن اسمه في الـ NAME ونقوم بوضع المعلومات التي نريد إخفاءها في الـ VALUE .

نستطيع الاستفادة أيضاً منها عن طريق الـ php وذلك عن طريق كتابة كود الـ HTML بواسطة الأمر (echo) في الـ PHP كما في المثال التالي :

```
<?php  
$msg1 = "هذه العبارة لن تظهر";  
Echo "<form>";  
Echo "<input type=hidden name =secret value= '$msg1'>";  
Echo "<input type=submit>";  
Echo "</form>";  
?>
```

هذا الكود الذي تراه عبارة عن كود HTML تم كتابته بالـ PHP عن طريق الأمر (echo) ولقد استطعنا تخزين قيمة متغير php (\$msg) في متغير html (secret) .

✓ استخدام حقول كلمات السر (Password fields)

لكي تجعل المعلومات أكثر حماية من التعرض إلى السرقة أو غير ذلك يمكنك استخدام حقول كلمات السر الذي هو عبارة عن مربع نص بسيط يقوم بإظهار النص على شكل نجوم **** في حال كان الجهاز يستخدم على يد أكثر من شخص فان هذه الطريقة جيدة قليلاً في أن لايري شخص معلومات الآخر السرية .
في الواقع مع ذلك فإنك لا تكون قد اديت حماية إذا كان الاسلوب المستخدم في ارسال بيانات المستخدم هو الاسلوب get إلا إذا كنت تستخدم تشفير البيانات ويكون أكثر جودة اذا استخدمت الاسلوب post وايضا لن يكون محمياً من الهاكر إذا لم تكن تستخدم SSL (Secure Socket Layer) لكي تقوم بتنشيط تشفير البيانات

مثال :

قم بفتح محرر النصوص لديك واكتب الكود التالي واحفظه باسم pass.php

```
<html dir="rtl">
<body>
<form method="GET" action="pass.php">
<table border=1>
<tr>
<td>اسم المستخدم</td>
<td><input type=text name =user></td>
</tr>
<tr>
<td>كلمة المرور</td>
<td><input type=password name =pass></td>
</tr>
<tr>
<td><input type =submit value="ارسال"></td>
<td><input type=reset value="حذف"></td>
</tr>
</table>
</form>
</body></html>
```

قم بفتح محرر النصوص لديك واكتب الكود التالي واحفظه باسم pass1.php

```
<html dir="rtl">
<?php
$user=$_GET['user'];
$pass=$_GET['pass'];
Echo " اسم المستخدم هو : ";
Echo "$user<br>";
Echo " كلمة المرور هي : ";
Echo "$pass <br>"
```

ماهي الـ MySQL

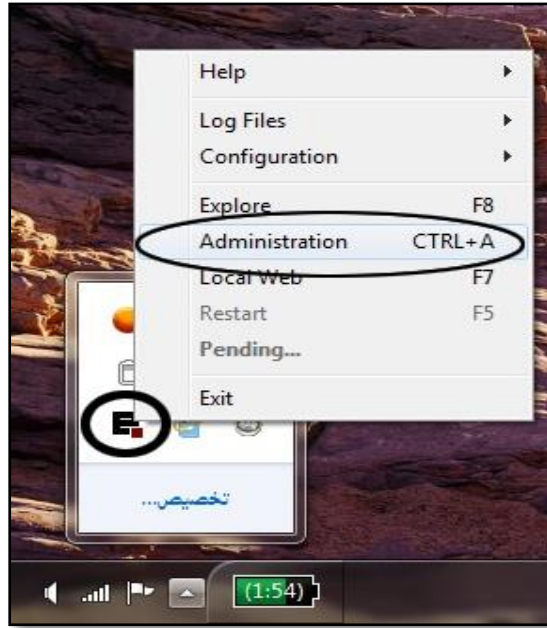
عندما نتحدث عن البرامج الكبيرة والقوية لا بد أن نتحدث عن قواعد البيانات وبالرغم من أن لغة الـ php تدعم تقريباً كل محركات قواعد البيانات إلا أن ما يهمنا هنا هو دعمها لقواعد البيانات MySQL المجانية والمتوفرة في أغلب مواقع الاستضافة المجانية والمدفوعة الاجر ، وبدأت الشركات تفضل MySQL على قواعد البيانات المشهورة مثل أوراكل وغيرها بسبب أنها مستقرة ومجانية .

التعامل مع قواعد البيانات My SQL

عندما تود التحكم بقواعد البيانات My SQL بشكل قوي وسريع سواء على الانترنت أو على جهازك الشخصي فانت بحاجة الى البرنامج phpMyadmin الذي يتيح لك تحكم متميز بقواعد البيانات وسوف أقوم بشرح طريقة استخدام هذا البرنامج بكل تفصيل فيما يلي :

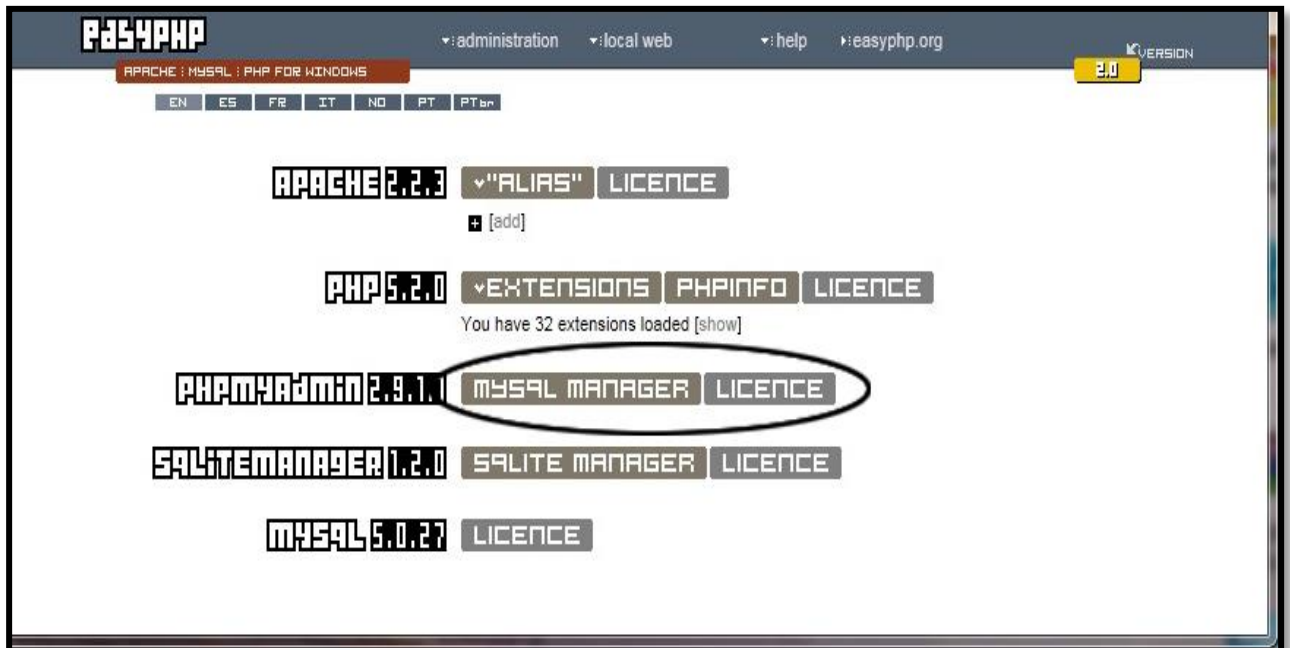
البرنامج يكون موجود مسبقاً في أي مستضيف يدعم MySQL كما انه ينزل ألياً مع السيرفر الشخصي لذلك كل ما عليك لشغيله كتابة العنوان التالي في متصفحك
(http://127.0.0.1/home/mysql/) لتشاهد التالي:

The screenshot displays the phpMyAdmin 2.9.1.1 interface. On the left, there is a sidebar with navigation icons and a 'Database' section with a dropdown menu showing '(Databases) ...' and the text 'Please select a database'. The main content area is divided into two columns. The left column, titled 'localhost', lists various server settings and options: Server version: 5.0.27-community-log, Protocol version: 10, Server: localhost via TCP/IP, User: root@localhost, MySQL charset: UTF-8 Unicode (utf8), and MySQL connection collation: utf8_general_ci. Below these are links for 'Create new database', 'Show MySQL runtime information', 'Show MySQL system variables', 'Processes', 'Character Sets and Collations', 'Storage Engines', 'Reload privileges', 'Privileges', 'Binary log', 'Databases', 'Export', and 'Import'. The right column, titled 'phpMyAdmin - 2.9.1.1', shows client version: 5.0.22, used PHP extensions: mysql, language: English, theme/style: Original, and font size: 100%. It also includes links for 'phpMyAdmin documentation', 'Official phpMyAdmin Homepage', and '[ChangeLog] [CVS] [Lists]'. At the bottom, a yellow warning box states: 'Your configuration file contains settings (root with no password) that correspond to the default MySQL privileged account. Your MySQL server is running with this default, is open to intrusion, and you really should fix this security hole.' A link 'Open new phpMyAdmin window' is visible at the bottom right.



أو ممكن الوصول اليها من مكان اخر وهو :

تظهر لك هذه الصورة :



طريقة انشاء قاعدة بيانات داخل MySQL

نقوم أولاً بكتابة اسم القاعدة المراد انشائها والضغط على الزر Create والصورة توضح ذلك :

هنا نكتب اسم القاعدة ونحن قمنا بكتابة basic
 بعد كتابة اسم القاعدة نقوم بالضغط هنا

هنا قمنا باختيار عدد الحقول داخل الجدول
 بعد الانتهاء من اختيار اسم الجدول وعدد الحقول نقوم بالضغط هنا

اسم الحقل	نوع الحقل	طول الحقل	نوع الترميز	نسبة البيانات الى صفة معينة	الحقل فارغ أم لا
Field	Type	Length/Values	Collation	Attributes	Null
	VARCHAR				not null
	VARCHAR				not null
	VARCHAR				not null

Table comments: Storage Engine: MyISAM Collation:

بعد الانتهاء تتم عملية الحفظ Or Add 1

القيمة الافتراضية	أخرى	مفهرس مفتاح أساسي	وحيد	---	Comments
Default2	Extra	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>

بعد الانتهاء من عملية الحفظ يتم الانتقال
من خلال الضغط على الزر GO

الآن لتتعرف على ماذا تعني هذه المصطلحات

- Field : يطلب هنا اسم الحقل .
- Type : نوع الحقل هل هة رقمي أم نص أم تأريخ (اختر من القائمة) .
- Length/values : طول البيانات الممكن ادخالها في الحقل .
- Attributes : هل تود نسبة البيانات الى صفة معينة .
- Null : هل يمكن يكون هذا الحقل فارغاً (لا يحتوي على قيم) .
- Default : القيمة الافتراضية لهذا الحقل .
- Extra : لعمل زيادة تلقائية للحقل (auto_ increment) أم لا وتفضل الزيادة التلقائية عندما يكون الحقل مفتاح اساسي .
- Primary : هل هو مفتاح أساسي .
- Index : هل هو مفهرس هذا لسرعة البحث .
- Unique : هل هو وحيد أي قيمته لا تتكرر .

الآن سنقوم بإنشاء الجدول التالي :

اسم الحقل	نوع البيانات	طول البيانات	مفتاح أساسي
St_no	Int	5	✓
St_name	varchar	20	—
age	Int	5	—

بعد كتابة البيانات المدرجة في الجدول سنلاحظ ظهور هذه الشاشة التالية :

phpMyAdmin

Database: basic (1)

اسم القاعدة (1) basic

اسم الجدول student

Table student has been created.

نقوم بالضغط على الامر insert عندما نريد تعبئة البيانات داخل الجدول

SQL query: أوامر SQL التي تم انشائها تلقائياً أثناء انشاء الجدول

```
CREATE TABLE `student` (
  `st_no` INT(5) NOT NULL,
  `st_name` VARCHAR(20) NOT NULL,
  `age` INT(5) NOT NULL,
  PRIMARY KEY (`st_no`)
) ENGINE = MYISAM
```

الجدول الذي تم انشائه

Field	Type	Collation	Attributes	Null	Default	Extra	Action
st_no	int(5)			No			
st_name	varchar(20)	latin1_swedish_ci		No			
age	int(5)			No			

الآن أصبح لدينا في الجدول (student) ثلاثة حقول ولكن هذه الحقول فارغة ولكي نضيف البيانات الى هذه الحقول نضغط على الكلمة insert الموجودة أعلى بيانات الجدول كما في الصورة السابقة . وعندما نقوم بالضغط على الكلمة insert تظهر لنا الصورة التالية :

مشاهدة البيانات داخل جدول نضغط

Database: basic (1)

اسم الجدول student

Field Type Function Null Value

st_no int(5) نقوم بتعبئة رقم الطالب هنا

st_name varchar(20) اسم الطالب هنا

age int(5) العمر هنا

Insert as new row and then Go back to previous page











بعد تعبئة البيانات نقوم بالضغط هنا

سنقوم بإدخال البيانات التالية الى الجدول :

St_no	St_name	age
1	Waleed	25
2	Essam	30
3	Hassen	22
4	Abdu	28
5	yahya	25

لقد تم اضافة البيانات الى الجدول ولكي نشاهد البيانات التي اضعناها نضغط على الكلمة (Browse) كما في الصورة السابقة .

وعند الضغط على الكلمة Browse تظهر لنا البيانات داخل جدول كما في الصورة :

للحذف للتعديل			st_no	st_name	ago
<input type="checkbox"/>			1	waleed	25
<input type="checkbox"/>			2	Essam	30
<input type="checkbox"/>			3	Hassen	22
<input type="checkbox"/>			4	Abdu	28
<input type="checkbox"/>			5	yahya	25

كما تلاحظ أيضاً لديك أمام كل سجل الكلمتين :

Edit : لتغيير وتعديل البيانات في السجل .

Delete : لحذف بيانات هذا السجل .

دوال التعامل مع قواعد البيانات

▪ الدالة mysql_connect

هذه الدالة تستخدم لإجراء عملية الاتصال بقاعدة البيانات وهي تحتاج الى ثلاث متغيرات
المتغير الاول : اسم السيرفر (Localhost)
المتغير الثاني : اسم المستخدم قاعدة البيانات (root)
المتغير الثالث : كلمة السر (يترك فراغ)

مقبض الاتصال

```
$con=mysql_connect("localhost","root"," ");
```

وتعود هذه الدالة بمقبض الاتصال في حالة نجاح الاتصال أو رسال رسالة تعبر عن الخطأ في حالة فشل الاتصال

▪ الدالة mysql_select_db

هذه الدالة تحدد قاعدة البيانات التي نتعامل معها حالياً ، وتحتاج هذه الدالة الى متغيرين هما :
المتغير الاول : اسم قاعدة البيانات التي تود التعامل معها .
المتغير الثاني : مقبض الاتصال الذي اجريناه .

مقبض الاتصال
اسم القاعدة

```
$db=mysql_select_db("basic",$con);
```

في حالة حدوث أي خطأ . false في حالة نجاحها أو True وتعود هذه الدالة بالقيمة

الان بعد نجاح الاتصال وتحديد قاعدة البيانات التي سوف نتعامل معها تستطيع الحصول أو التحكم بجميع الجداول الموجودة بقاعدة البيانات ولكي نفعل ذلك يوفر php لنا عدة دوال تستخدم لادارة البيانات سوف نشرحها فيما يلي:

▪ الدالة mysql_affected_rows

تعيد لنا هذه الدالة بعدد السجلات التي تأثرت من احدى العمليات (insert – update – delete) وتحتاج هذه الدالة الى متغير واحد وهو مقبض الاتصال .

```
$c=mysql_affected_rows ( );
```

▪ الدالة mysql_num_rows

تعيد لنا هذه الدالة بعدد السجلات التي حصلنا عليها بعد استخدام التعليمة Select فقط .
وتحتاج هذه الدالة الى متغير واحد وهو مقبض الاتصال .
ملاحظة : لا تستخدم هذه الدالة مع التعليمات (insert – update – delete) .

▪ الدالة mysql_query

تعتبر من أهم الدوال الشائعة استخدمها حيث تقوم هذه الدالة بتنفيذ جمل الاستعلام على قواعد البيانات وتعيد بالقيم الناتجة عن عملية الاستعلام على شكل مقبض لعملية الاستعلام .

```
$sql=mysql_query("select *from student ");
```

```
$ sql =mysql_query("delete from student where id=$t1");
```

```
$ sql =mysql_query("update student set name='$t1',city='$t2' where id=$t");
```

```
$ sql =mysql_query("insert into student(id,name,city) values ($t,$t1)");
```

تلاحظ أن القيم المستخرجة من الجدول (student) وضعت في المتغير sql ولكي نستفيد من هذه القيم نستخدم الدالة (mysql_fetch_row) .

▪ الدالة mysql_fetch_row

تضع هذه الدالة القيم المسترجعة من العملية Select في مصفوفة حتى يتم التعامل معها بسهولة (True هنا قيم false لا يوجد قيم) .

```
$row=mysql_fetch_row($sql) ;
```

في هذه الحالة القيم موجودة في مصفوفة يمكن التعامل معها بسهولة .

▪ الدالة mysql_create_db

تستخدم هذه الدالة لإنشاء قاعدة بيانات جديدة ، وتحتاج فقط الى اسم القاعدة وتعيد القيمة True في حالة نجاح العملية و false في حالة فشلها .

▪ الدالة mysql_drop_db

تستخدم هذه الدالة لحذف قاعدة البيانات وجميع الجداول فيها وتحتاج فقط الى اسم قاعدة البيانات المطلوب حذفها . وتعيد القيمة True في حالة نجاح العملية و false في حالة فشلها .

▪ الدالة mysql_list_dbs

بواسطة هذه الدالة تستطيع معرفة قواعد البيانات المتوفرة لدينا وتحتاج فقط الى مقبض الاتصال .

مقبض تم الحصول عليه بعد اجراء عملية الاتصال



```
$result=mysql_list_dbs($con);
```

تلاحظ أن القيم المستخرجة وضعت في المتغير result ولكي نستفيد من هذه القيم نستخدم الدالة mysql_fetch_row التي تعرفنا عليها مسبقاً كالتالي .

```
$row=mysql_fetch_row($sql) ;
```

▪ الدالة mysql_list_tables

بواسطة هذه الدالة نستطيع معرفة الجداول الموجودة في قاعدة البيانات وتحتاج هذه الدالة الى متغير واحد فقط وهو اسم قاعدة البيانات .

```
$result=mysql_list_tables("mydatabase");
```

▪ الدالة mysql_list_fields

بواسطة هذه الدالة نستطيع معرفة الحقول الموجودة في جدول معين ، وتحتاج هذه الدالة متغيرين : المتغير الاول : اسم قاعدة البيانات .

المتغير الثاني : اسم الجدول
وسوف تعود هذه الدالة بمقبض للعملية لنستخدمه في معرفة معلومات حقول الجدول سواء الاسم أو النوع أو
الطول أو الشرط بواسطة الدوال التالية :

- الدالة `mysql_field_name` : لمعرفة اسم الحقل .
- الدالة `mysql_field_type` : لمعرفة نوع الحقل .
- الدالة `mysql_field_len` : لمعرفة طول الحقل .
- الدالة `mysql_field_flags` : الشروط على الحقل .

التعامل مع MySQL داخل PHP











الآن سوف نتعرف على كيفية الاتصال والتحكم بـ MySQL بواسطة php لأن المستخدم الذي يزور موقعك لن يكون لديه صلاحية استخدام البرامج أو الدخول واستخدام قاعدة البيانات الموجودة في موقعك مباشرة ولكنه سوف يجد الشاشات التي تحددها له بحسب الصلاحيات المحددة أيضاً .

لكي نتعامل مع قاعدة البيانات داخل php نفذ الخطوات التالية :

- قم أولاً بإنشاء قاعدة بيانات باسم (basic) .
- وقم بإنشاء جدول داخل هذه القاعدة باسم (Student) يحتوي على 4 حقول :

الخصائص	St_no	St_name	city	phone
نوع البيانات	int	Varchar	Varchar	Varchar
حجم الحقل	_____	50	50	15
مفتاح أساسي	Y	_____	_____	_____
أخرى	Auto_increment	_____	_____	_____

- قم بإدخال هذه البيانات داخل الجدول كما في الشكل :

←T→	st_no	st_name	city	phone
<input type="checkbox"/>  	1	waleed ahmed	Hodeidah	734588935
<input type="checkbox"/>  	2	yahya mahdi	haja	736103694
<input type="checkbox"/>  	3	mohammed salah	Hodeidah	712967194
<input type="checkbox"/>  	4	mahmmed soleman	Bajil	736181043
<input type="checkbox"/>  	5	bazel adham	mahyout	700265761

- قم بكتابة الكود التالي داخل المفكرة :

```
<?php
$con=mysql_connect("localhost","root","");
$db=mysql_select_db("basic",$con);
$sql= mysql_query("select *from student");
while($row=mysql_fetch_array($sql)){
    $st_no=$row[0];
    $st_name=$row[1];
    $city=$row[2];
    $phone=$row[3];
    echo"st_no: $st_no <br>st_name: $st_name <br>city: $city<br>phone :$phone<br>";
    echo " _____<br>";
}
?>
```

قم بحفظ الملف باسم index.php في المجلد الخاص بالسيرفر و اكتب في المتصفح العنوان التالي :

<http://localhost/index.php>

سنلاحظ ظهور جميع البيانات الموجودة داخل الجدول

```
st_no: 1
st_name: waleed ahmed
city: Hodeidah
phone :734588935

st_no: 2
st_name: yahya mahdi
city: haja
phone :736103694

st_no: 3
st_name: mohammed salah
city: Hodeidah
phone :712967194

st_no: 4
st_name: mahmmed soleman
city: Bajil
phone :736181043

st_no: 5
st_name: bazel adham
city: mahyout
phone :700265761
```

عملية الاضافة (insert):

سنقوم باضافة بيانات الى الجدول من داخل صفحة الويب ولتنفيذ ذلك نفذ ما يلي :

قم بتشغيل محرر النصوص لديك واكتب الكود التالي وقم بحفظه في ملف اسمه index.html .

```
<html>
<body>
<form action="insert.php" method="post">
name : <input type="text" name="name"> <br>
city : <input type="text" name="city"><br>
phone : <input type="text" name="phone"><br>
<input type="submit" value="insert">
</body>
</html>
```

مقدمة في لغة PHP

قم بفتح محرر النصوص واكتب الكود التالي وقم بحفظه باسم insert.php

```
<?php
$name=$_POST['name'];
$city=$_POST['city'];
$phone=$_POST['phone'];
$con=mysql_connect("localhost","root","");
$db=mysql_select_db("basic",$con);
$sql=mysql_query("insert into student values(null,$name,$city,$phone)");
if($sql){
echo "تمت عملية الاضافة";
}
else{
echo "لم تتم عملية الاضافة";
}
?>
```

قم بنقل الملفين الى مجلد السيرفر لديك .













اكتب في المتصفح العنوان التالي : <http://localhost/index.html>

name :	<input type="text" value="hamdi abdu"/>
city :	<input type="text" value="Taiz"/>
phone :	<input type="text" value="7344 198 12"/>
<input type="button" value="insert"/>	بعد تعبئة البيانات سنقوم بالضغط على الزر insert

تظهر الصفحة التي باسم (insert.php) مكتوب فيها :

تمت عملية الاضافة

لاحظ في الجدول أنه قد تم اضافة البيانات :

←T→	st_no	st_name	city	phone
<input type="checkbox"/>  	1	waleed ahmed	Hodeidah	734588935
<input type="checkbox"/>  	2	yahya mahdi	haja	736103694
<input type="checkbox"/>  	3	mohammed salah	Hodeidah	712967194
<input type="checkbox"/>  	4	mohammed soleman	Bajil	736181043
<input type="checkbox"/>  	5	bazel adham	mahyout	700265761
<input type="checkbox"/>  	6	hamdi abdu	Taiz	734419812

عملية التعديل (update):

سنقوم بالتعديل على البيانات الموجودة في الجدول من داخل صفحة الويب ولتنفيذ ذلك نفذ ما يلي :
سنقوم هنا بإنشاء ثلاث صفحات :

الصفحة الاولى سنقوم بكتابة رقم الطالب المراد التعديل على بياناته

قم بتشغيل محرر النصوص لديك واكتب الكود التالي وقم بحفظه في ملف اسمه update1.html .

```
<html dir="rtl">
<body>
<form method=get action=update2.php>
<table border=1>
<tr>
<td> رقم الطالب </td>
<td><input type=text name=t></td>
</tr>
<tr>
<td><input type=submit value=send></td>
</tr>
</table>
</form>
</body>
</html>
```

الصفحة الثانية سنقوم باجراء التعديل على البيانات المطلوبة

قم بفتح محرر النصوص واكتب الكود التالي وقم بحفظه باسم update2.php

```
<html dir="rtl">
<?php
$t=$_GET['t'];
$con=mysql_connect("localhost","root","");
$db=mysql_select_db("basic",$con);
$sql=mysql_query("select *from student where st_no=$t");
$row=mysql_fetch_row($sql);
?>
<form method=get action=update3.php >
<input type=hidden name=t value=<?php echo $row[0];?>><br>
<input type=text name=t1 value=<?php echo $row[1];?>><br>
<input type=text name=t2 value=<?php echo $row[2];?>><br>
<input type=text name=t3 value=<?php echo $row[3];?>><br>
<input type=submit value=update>
</body>
</html>
```

الصفحة الثالثة: سنقوم بإرسال التعديلات التي أجريناها الى الجدول :-
قم بفتح محرر النصوص واكتب الكود التالي وقم بحفظه باسم update3.php

```
<html dir="rtl">
<?php
$t=$_GET['t'];
$t1=$_GET['t1'];
$t2=$_GET['t2'];
$t3=$_GET['t3'];
$con=mysql_connect("localhost","root","");
$db=mysql_select_db("basic",$con);
$sql=mysql_query("update student set st_name='$t1',city='$t2', phone='$t3' where
st_no=$t");
if($sql)
echo "تم التعديل";
else
echo "لم يتم التعديل";
?>
</body>
</html>
```

اكتب في المتصفح العنوان التالي : <http://localhost/update1.html>

رقم الطالب

send

سنقوم بتسجيل
رقم الطالب وليكن (1)

بعد ادخال الرقم سنقوم بالضغط على الزر (Send) ستظهر الصفحة الثانية :

الرقم لم يظهر

waleed ahmed
Hodeidah
734588935

الاسم (ALi yousif) سنقوم بالتعديل في

update

بعد اجراء التعديل سنقوم بالضغط على الزر (update) ستظهر الصفحة الثالثة :

تم التعديل

لاحظ التعديلات التي تمت على الجدول :

←T→	st_no	st_name	city	phone
<input type="checkbox"/>	1	ALi yousif	Hodeidah	734588935
<input type="checkbox"/>	2	yahya mahdi	haja	736103694
<input type="checkbox"/>	3	mohammed salah	Hodeidah	712967194
<input type="checkbox"/>	4	mohammed soleman	Bajil	736181043
<input type="checkbox"/>	5	bazel adham	mahyout	700265761

عملية الحذف (delete) :

سنقوم بحذف البيانات من داخل صفحة الويب ولتنفيذ ذلك نفذ ما يلي :
قم بتشغيل محرر النصوص لديك واكتب الكود التالي وقم بحفظه في ملف اسمه delete1.html .

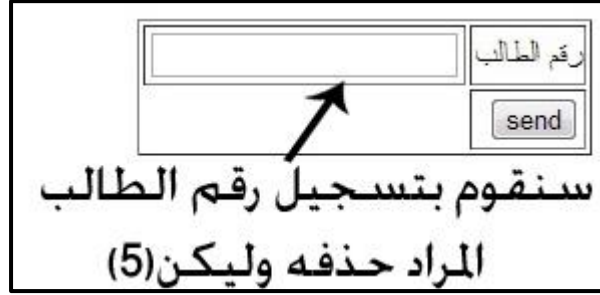
```
<html dir="rtl">
<body>
<form method=get action=delete2.php>
<table border=1>
<tr>
<td> رقم الطالب </td>
<td><input type=text name=t></td>
</tr>
<tr>
<td><input type=submit value=send></td>
</tr>
</table>
</form>
</body>
</html>
```

قم بفتح محرر النصوص واكتب الكود التالي وقم بحفظه باسم delete2.php

```
<html dir="rtl">
<?php
$t1=$_GET['t'];
$con=mysql_connect("localhost","root","");
$db=mysql_select_db("basic",$con);
$sql=mysql_query("delete from student where st_no=$t1");
if($sql)
echo "تم الحذف";
else
echo "لم يتم الحذف";
?>
</body>
</html>
```

قم بنقل الملفين الى مجلد السيرفر لديك .
اكتب في المتصفح العنوان التالي :

<http://localhost/delete1.html>



سنقوم بتسجيل رقم الطالب
المراد حذفه وليكن (5)

بعد كتابة الرقم سنقوم بالضغط على الزر (Send) ستظهر الصفحة التالية :

تم الحذف

لاحظ من خلال الجدول ان الطالب رقم (5) قد تم حذفه :

←T→	st_no	st_name	city	phone
<input type="checkbox"/> ✎ ✖	1	ALi yousif	Hodeidah	734588935
<input type="checkbox"/> ✎ ✖	2	yahya mahdi	haja	736103694
<input type="checkbox"/> ✎ ✖	3	mohammed salah	Hodeidah	712967194
<input type="checkbox"/> ✎ ✖	4	mohammed soleman	Bajil	736181043

تم بحمد لله