

سوف نبدأ بأخذ فكرة عن البرنامج الذي سوف أشرح عنه وهو عبارة عن دليل مواقع بسيط جداً جداً يعتمد علي قواعد البيانات.

بالطبع لا يوجد فيه تقسيمات والسبب أننا سوف نضطر إلي وضع أو خلق أكثر من جدول (Table) في قواعد البيانات وسوف نضطر إلي وضع علاقات بينهم وهذا حالياً يعقد الموضوع فسأكتفي الآن بجدول واحد (one table) وسوف اشرح إذا اتسع لدي الوقت الكثير عن (sql) عامة وعن العلاقات (Relationships) لأنها مثيرة أيضاً جداً

نبدأ في الخطوة الأولى وهي قاعدة البيانات الخاصة بدليل الموقع الذي لدينا , فالبيانات والمعلومات التي غالباً ما يهتم بها دليل الموقع هي عنوان الموقع واسم الموقع والبريد الخاص بالموقع

### هذا هو الكود الخاص بإنشاء قاعدة البيانات الخاصة بدليل الموقع

```
CREATE TABLE dalal (
  id int not null auto_increment,
  sitename varchar(100),
  add_date date,
  email varchar(100),
  site_url varchar(100),
  description text,
  primary key (id));
```

هنا سوف يخزن في قواعد البيانات المعلومات التالية :  
id وهي المشرح الأساسي والفريد الذي يتعرف علي الجدول بواسطته .  
sitename وهي اسم الموقع وهي من نوع char  
add\_date تاريخ إضافة الموقع وهو من نوع date  
email البريد الإلكتروني وهو من نوع char وطولها نفس طول اسم الموقع (100)  
site\_url وهو عنوان الموقع أيضاً من نوع char

سوف اكتفي بشرح id لأنه ربما يكون جديد علي البعض وسوف اشرح عمله وليس المهم أن تفهم علمه في هذه الخطوة إنما سوف تفهمها بالتفصيل في الخطوات القادمة بعد قليل إن شاء الله  
id هو عبارة عن المشرح الأساسي لهذا الجدول (primary key)

وتلخص في آخر سطر من الجدول وجود:

primary key (id)

ربما ترجمة المصطلحات العلمية وخاصةً في الكمبيوتر تضر أكثر مما تنفع وهذا سبب وجيه لي في كتابتي السكرت مهمما كان بلغة إنجليزية ومن ثم تعريبها  
المهم نحن الآن نريد أن نعرف ماذا يعني id سوف أعطي مثال لكي يتضح فلو أردنا مشرح أساسي أو كود فريد نميز به هذا الجدول فلن نستطيع أن نضع هذا الكود الفريد اسم الموقع لأن ربما يكون لدي موقعاً بنفس الاسم والاختلاف بينهما في الوصلة وربما يكون هناك موقع بعنوانين مثل سواف وياهوو إذاً اسم الموقع لا يكمن أن نضعه كود أساسي وأيضاً نفس الكلام ينطبق علي عنوان الموقع وأيضاً علي التاريخ وحتى البريد لا نستطيع أن نضع البريد هو المشرح الوحيد والأساسي للجدول ربما يأتي هذا لكن منطقياً لا يجوز

ربما ترى id من نوع عدد صحيح ( int ) لكن ربما يصعب عليك الذي بجانبه وهو ( not null ) وهي تعني بعدم السماح لهذا الحقل أن يكون بدون قيمة  
not null تعني بكل وضوح ربما يسمح بقيمة فارغة ربما تفكر في الصفر فهذا غير صحيح .  
وترى أيضاً هذه العبارة بجانب not null وهي ( auto\_increment ) وهي تعني بكل وضوح الإضافة التلقائية أو الأتوماتيكية ودائماً ما يبدأ بالقيمة ( 1 )  
وهذا يعني مع كل إضافة موقع سوف تزيد قيمة ( id ) وهي تبدأ من ( 1 )

config.php

وهو عبارة عن ملف التوجيه الذي يحتوي معلومات الموقع وهي المعلومات التالية:

```
<?
//this model config
$dbname = "links";
$dbuname = "root";
$dbpws = "";
$dbhost = "localhost";
?>
```

لا يحتاج إلي شرح لأنه واضح فالذي جرب أن يركب أي برنامج php يعتمد علي قواعد البيانات فأكيد راح يعدل في ملف مشابه له أو نفسه ومع ذلك سنشرحه :

**الآن نبدأ في add.php وهذا هو :**

[illegible]

من رؤيتك له تلاحظ أنه عبارة عن html و php لكن كود php أكثر

بالنسبة html فهي بسيطة بإمكانك فهمها وهي عبارة عن حقول تطلب من المستخدم إدخال معلومات الموقع سوف أبدأ في شرح كود php فتلاحظ بدايته ونهايته بهذين الوسمين <? و >?

وبعدها يأتي شرط التحقق وهو

[illegible]

هذا هو كود الفورم الخاص بإدخال معلومات الموقع فإذا لاحظته في بدايته هنا

`<form method="POST" action="add.php?action=addsite">`  
 وضعت الإرسال بال ( post ) وقمت بوضع قيمة ( action ) الصفحة ( add.php?addsite )  
 هذا يعني أن قيمة (action) في حال ضغط submit ستكون ما هو مكتوب بعد علامة الاستفهام ( ؟ ) بعد  
 اسم الصفحة وهي (add.php?addsite)  
 أتمنى أنك فهمت المقصود

بعد ذلك إذا تحقق الشرط ووجد من أرسل سوف ينفذ الأوامر والأولى منها

```
include("config.php");
```

وأمر include يعني استجلاب هذا الملف وقيم المتغيرات الموجودة به بعد هذا الأمر طلبت الاتصال بقواعد البيانات لكي يتم تجهيز الجدول للإضافة وهو هنا

```
mysql_pconnect($dbhost, $dbuname, $dbpw);
    @mysql_select_db("$dbname") or die ("Unable to select database");
```

الدالة `mysql_pconnect` تطلب الاتصال بخادم قاعدة البيانات علي أساس المعلومات الموجودة بها وهنا نحن استعنا بعد الله بالمتغيرات الموجودة في ملف `config.php` وربما هذا يوضح لنا أكثر فائدة الدالة `include`

ربما لاحظت في طلب الاتصال في الدالة `mysql_pconnect` في السطر الثاني بعدها هذا

```
@mysql_select_db("$dbname") or die ("Unable to select database");
```

هنا في الدالة mysql\_select\_db وضعنا قيمة المتغير dbname وهو اسم قاعدة البيانات فهذه الدالة تقوم بتحديد اسم قاعدة البيانات المراد الاتصال بها وهذا ربما يوضح استخدام علامة @ لكي يعمل كل شيء بنفس الوقت  
بعد هذا نلاحظ السطر التالي :

`mysql_query("insert into dalal values ('','$name','$email','$siteurl','$des')");`  
 قمنا في هذا السطر بتنفيذ الدالة `mysql_query` وهذه الدالة تقوم باختصار بتنفيذ استعلامات (sql) وفي هذا السطر استخدمنا أمر الاستعلام المراد تنفيذه بواسطة الدالة `mysql_query` هو (Insert) هذا الأمر أو الدالة في sql تقوم بإضافة البيانات في قاعدة البيانات والصيغة أو القاعدة الثابتة له هي

## INSERT INTO tablename VALUES (var and values )

### INSERT INTO

هذا ثابت لدينا بعده يأتي اسم الجدول الذي نريد إضافة البيانات بداخله وبعد ذلك تأتي كلمة VALUES وتفتح قوس وتخطط المتغيرات والقيم المراد إضافتها بأقواس اقتباس (" وتنهيه بفاصلة منقوطة ;  
تلاحظ حين استخدمت mysql\_query و insert لم أضع القيم بل وضعت المتغيرات التي تحمل القيم المراد تخزينها في جدول dalal ربما هذا يسهل علينا الكثير  
بعد ذلك في تأتي الدالة mysql\_close وهي تعني طلب إغلاق الاتصال بخادم قاعدة البيانات بهذا انتهينا من ملف add.php و config.php

هنا سوف نعرف كيف نستخرج البيانات الموجودة داخل قاعدة البيانات وهي في غاية السهولة

**بداية سوف أضع كود show.php كاملاً وأشرحه مثل سابقه**

```
<html>
<head>
<style type="text/css">
body {font-family:verdana,verdana; font-size:10pt}
TD {font-family:verdana,verdana; font-size:10pt}
#header {color:black; font-weight:bold; font-family:verdana,verdana}
A {color:navy; text-decoration:underline}
A:hover {color:red}
A:visit {color:navy}
</style>
</head><titel><center><b>view The sites</b></center><br><center><a href='add.php'>
Add Your Site</a></center> <hr color="#000080"></titel>

<body>
<?php
include("config.php");
mysql_pconnect($dbhost, $dbuname, $dbpw);
    @mysql_select_db("$dbname") or die ("Unable to select database");

$query="select * from dalal ";

$result=mysql_query($query);

mysql_close();

/*Display Results*/

$num=mysql_numrows($result);

$i=0;
while ($i < $num) {

    $sitename=mysql_result($result,$i,"sitename");
    $add_date=mysql_result($result,$i,"add_date");
    $email=mysql_result($result,$i,"email");
    $siteurl=mysql_result($result,$i,"site_url");
    $description=mysql_result($result,$i,"description");
```

```

$id=mysql_result($result,$i,"id");
?>
<table border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="100%"><b>SiteName :</b><a href='<?echo $siteurl;?>'><i><? echo $sitena
me;?></i></a>
<p><b>Date Add :</b> <i> <?echo $add_date;?></i></p>
<p><b>description :</b><?echo $description;?></p>
<p><b>URL :</b><i><?echo $siteurl;?></i></p>
<p><b>Email :</b><i> <a href="mailto:<?echo $email;?>"><?echo $email;?></a></i>
</p>
</td>
</tr>
</table>
<a href='edit.php?id=<?echo $id;?>'><b><b>[Edit This Site]</b></b></a>
<hr color="#000080">
<?
++$i;
}
?>
</body>
</html>

```

كما تلاحظ تقريباً شبيهه بالملف add.php في عملية الاتصال لكن الجديد لدينا هنا السطر الذي يلي عملية الاتصال وهو

```
$query="select * from dalal ";
```

في هذا السطر قمنا بتعريف متغير يدعى query وقمنا بوضع قيمة هذا المتغير عملية الاستعلام عن المعلومات الموجودة في الجدول dalal  
أظن هذا واضح لكن دعني أشرح عمل select فهي تقوم بتحديد الحقول بداخل الجداول أو الجدول وهي دائماً تستخدم في إخراج البيانات من الجداول فتلاحظ وضعنا قيمة الحقل أو الحقول في استعلامنا هذا علامة الضرب ( \* ) وهي تعني تحديد جميع الحقول الموجودة وبعدها استعملنا from وهذا يعني من وداًئماً يأتي بعده أسماء الجدول أو اسم الجدول الذي قمت بتحديد الحقول بداخله  
ببساطة واختصاراً لهذا السطر select \* from dalal  
حدد جميع الحقول من جدول الدليل

هنا سوف يحدد جميع الحقول الموجودة في جدول الدليل ويقوم بتخزينها في المتغير query بسيطة  
لا تنس أن تضعها بين أقواس اقتباس مزدوجة ( " " )  
بعد ذلك يأتي هذا السطر

```
$result=mysql_query($query);
```

قمنا هنا أيضاً بتعريف المتغير result وقمنا بتخزين قيمة الدالة mysql\_query وقيمة المتغير الذي استعملت به الدالة mysql\_query في المتغير result  
بوضوح هنا قمنا بالاستعانة بالدالة mysql\_query وانسبنا اليها قيمة المتغير query الموجود به مسبقاً  
قيمة الاستعلام مسبقاً في السطر الذي يسبق هذا السطر وقمنا بهذا لكي يتم فرز جميع القيم الموجودة في الجدول dalal بواسطة الدالة mysql\_query  
( هذا أبسط من اللي قبل )  
بعد ذلك قمنا بإغلاق قاعدة البيانات بالدالة mysql\_close  
وبعد ذلك قمنا بتنويه بسيط وتعليق يوضح أننا سوف نعرض ونطبع القيم الموجودة في الجدول في هذا السطر  
/\*Display Results\*/

بعد هذا السطر قمنا بتعريف متغير يدعى num وأضافنا القيمة إليه باستخدام الدالة mysql\_numrows مسندين إليه قيمة المتغير result  
 الدالة mysql\_numrows تقوم باسترجاع ومعالجة نتيجتنا المخزنة في المتغير result وتقوم بتخزينها في المتغير num علي شكل مصفوفة  
 بعد ذلك قمنا بتعريف عداد يدعى i ووضعنا قيمته صفر لكي نستخدمه مع أمر التكرار while وقمنا بوضع الشرط في أمر التكرار while

```
i < $num$
```

هذا يعني إذا كان قيمة العداد i اصغر من قيمة المتغير num المتواجد بداخله جميع المعلومات فاستمر بطباعة التالي وهي هذه السطور

```
$sitename=mysql_result($result,$i,"sitename");
$add_date=mysql_result($result,$i,"add_date");
$email=mysql_result($result,$i,"email");
$siteurl=mysql_result($result,$i,"site_url");
$description=mysql_result($result,$i,"description");
$id=mysql_result($result,$i,"id");
?>
<table border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="100%"><b>SiteName :</b><a href='<?echo $siteurl;?>'><i><? echo $sitename;?></i></a>
<p><b>Date Add :</b> <i> <?echo $add_date;?></i></p>
<p><b>description :</b><?echo$description;?></p>
<p><b>URL :</b><i><?echo$siteurl;?></i></p>
<p><b>Email :</b><i> <a href="mailto:<?echo$email;?>"><?echo $email;?></a></i>
</p>
</td>
</tr>
</table>
<a href='edit.php?id=<?echo $id;?>'><b><b>[Edit This Site]</b></b></a>
<hr color="#000080">
<?
++$i;
}
```

هنا قمنا بالكثير في أمر التكرار WHILE فسوف يطبع جميع القيم الموجودة في الجدول كما رمزت إليها في هذا السطر

```
$sitename=mysql_result($result,$i,"sitename");
```

قمنا في هذا السطر والسطور الخمسة التي تليه بتعريف العديد من المتغيرات وقمنا بتخزين قيم المعلومات بداخله باستخدام الدالة mysql\_result مسندين إليه المتغيرات result و i و الحقل الذي نريد إخراج القيمة منه  
 واستخدمنا في هذا السطر الحقل sitename

كما استخدمنا كود html لكي ننسق ونرتب شكل ظهور المعلومات  
 وفي آخر سطر قبل أن ننهي عملية التكرار while بالقوس } جعلنا قيمة العداد i تزيد بمقدار واحد مع كل لفة أو بمعنى طباعة قيم من قاعدة البيانات لكي يتم توقفها أرجو أن تنتبه لهذه الأمور الصغيرة لأنها من مشاكل البرمجة المشهورة

هنا انتهينا من ملف show.php يتبقى لنا ملف edit.php وهو الملف الخاص بتعديل الموقع فربما يخطئ أحدهم في إدخال اسم أو عنوان موقعه فهنا يمكنك تعديلها حتى بعد تخزينها في قاعدة البيانات  
 هذا آخر ملف سوف اشرحه اليوم من ضمن مشروع دليل المواقع وربما اشرح عملية الحذف والبحث في وقت آخر  
 فقط ادعوا لي بالستر والهداية والوقت الكافي

```

<html>
<head>
<style type="text/css">
body {font-family:verdana,verdana; font-size:10pt}
TD {font-family:verdana,verdana; font-size:10pt}
TH {font-size:10pt; color:white; font-weight:bold; font-family:verdana,verdana}
A {color:navy; text-decoration:underline}
A:hover {color:red}
A:visit {color:navy}
#mySelect {font-family:verdana,arial; font-size:9pt}
</style>
</head>
<titel><center><b>Edit Site</b></center><br><center><a href='add.php'><b>Add Your Si
te </b></a> |
<a href='show.php'><b>View Edit database</b></center></a> <hr color="#000080"></tite
l>
<body>

<table border=0 cellpadding=4 cellspacing=0 width=600>
<? echo "<form method='post' action='$PHP_SELF'>"; ?>
<table border="1" width="100%" bordercolor="#000080" cellspacing="1" cellpadding>
<tr>
<td width="100%">
<p align="center">Edit The Info </td>
<?
include("config.php");
mysql_pconnect($dbhost, $dbuname, $dbpw);
    @mysql_select_db("$dbname") or die ("Unable to select database");

$query1 = "select * from dalal where id = '$id'";

$result1 = mysql_db_query('links',$query1);

while($row = mysql_fetch_object($result1)) {
    echo "<td align=middle>SiteNmae<br><input type=text name='name' value='$row-
>sitename'></td>";
    echo "<td align=middle>Add Date<br><input type=text name='add_date' value='$row-
>add_date'></td>";
    echo "<td align=middle>Email<br><input type=text name='email' value='$row-
>email'></td>";
    echo "<td align=middle>SiteUrl<br><input type=text name='siteurl' value='$row-
>site_url'></td></tr>";
    echo "<td align=middle>Description<br><input type=text name='desc' value='$row-
>description'></td></tr>";
$id = $row->id;
}
echo "</table><p>";
echo "<br><input type=hidden name='id' value='$id'>";
?>

<input type=submit name="update" value="Edit The Info">

```



```

</form>
</tr>
</table>
<?
if ($update) {
    if($name == "" || $add_date == "" || $email == "" || $siteurl == "" || $desc == "" ) {
        die("<b>You left one or more fields blank.</b>");
    }

    $query2 = "update dalal set sitename='$name', add_date='$add_date', email='$email', site_
url='$siteurl', description='$desc'
        where id='$id'";
    mysql_db_query('links',$query2);
    echo "<b>Your record has been updated</b></p>";
}
mysql_close();
?>
</body>
</html>

```

هنا سوف يسهل علي شرح هذا الملف أكثر من سابقه لأنه يكرر ما يفعله إخوانه لكن مع بعض  
التجديدات التي لا تذكر فالجديد هنا  
وفي هذا السطر

```
$query1 = "select * from dalal where id = '$id'";
```

هنا تلاحظ عرفنا متغير يدعى query1 وقمنا بتخزين ناتج الاستعلام بداخله فهذا ليس بجديد الجديد ما  
يوجد في الاستعلام فقد استخدمنا where وهي تعني بالترجمة إلي العربية "بحيث" وتعني في البرمجة  
وخاصة مع أوامر sql "بشرط" أي هنا اشترطنا في استعلامنا أن تكون قيمة المتغير id مساوية لنفس  
القيمة الموجودة في الجدول وهي id وربما هنا يسعني أن أوضح أكثر ما هو id ولماذا عرفته وجعلته  
primary key  
هنا إذا ضغط أحد الذين أضافوا مواقعهم بعد أن أخطأ في إضافة موقعه علي edit فسوف يعتمد علي الرقم  
المخزن في id وهو كما سبق وأن عرفتها في قاعدة البيانات عدد صحيح ويزداد تلقائياً فلو كان لدينا مثلاً  
في قاعدة البيانات أربعة مواقع فلن يجري التعرف علي الأربعة مواقع ومعلوماتهم إلا عن طريق id لأن كل  
موقع منهم رقم بمعنى الأول منهم 1 والأخير 4 أرجو أنك فهمته وأرجو أن تحاول تشغيل السكريبت لكي  
تفهمه أكثر وأكثر  
بعد ذلك يأتي هذا السطر

```
$result1 = mysql_db_query('links',$query1);
```

هنا أيضاً عرفنا المتغير result1 وخرنا بداخله قيمة استعلام الدالة  
mysql\_db\_query وأسندنا إليها اسم قاعدة البيانات والمتغير query

الدالة mysql\_db\_query وهي مشابهة في عملها مع mysql\_query ما عدا الفرق بينهم أنك تستطيع  
تحديد اسم قاعدة البيانات الذي تريد الاستعلام عنه واستعملتها هنا لكي يتم إعطائنا النتائج بوضوح أكثر  
بعد هذا السطر يأتي أمر التكرار while مرة أخرى في هذا الملف

```
while($row = mysql_fetch_object($result1))
```

هنا قمنا بخلق المتغير row ولا تنسى في php ليس من الضروري تعريف المتغيرات قبل استخدامها وقمنا  
بمساواته ( وهو الشرط الخاص بأمر التكرار while ) بالدالة mysql\_fetch\_object مسندين إليه قيمة  
المتغير result1  
الدالة mysql\_fetch\_object وظيفته أو من ضمن الوظائف التي تقوم به أنها تقوم بانتزاع وعرض نتائج  
الاستعلام

فإذا تحقق شرط أمر التكرار while فستنفذ هذا الكود الطويل قليلاً

```

while($row = mysql_fetch_object($result1)) {
    echo "<td align=middle>SiteNmae<br><input type=text name='name' value='$row-
>sitename'></td>";
    echo "<td align=middle>Add Date<br><input type=text name='add_date' value='$row-

```



```

>add_date'></td>";
    echo "<td align=middle>Email<br><input type=text name='email' value='$row-
>email'></td>";
    echo "<td align=middle>SiteUrl<br><input type=text name='siteurl' value='$row-
>site_url'></td></tr>";
    echo "<td align=middle>Description<br><input type=text name='desc' value='$row-
>description'></td></tr>";
$id = $row->id;
}
echo "</table><p>";
echo "<br><input type=hidden name='id' value='$id'>";
?>

<input type=submit name="update" value="Edit The Info">
</form>
</tr>
</table>
<?
if ($update) {
    if($name == "" || $add_date == "" || $email == "" || $siteurl == "" || $desc == "" ) {
        die("<b>You left one or more fields blank.</b>");
    }

    $query2 = "update dalal set sitename='$name', add_date='$add_date', email='$email', site_
url='$siteurl', description='$desc'
        where id='$id'";
    mysql_db_query('links',$query2);
    echo "<b>Your record has been updated</b></p>";
}
mysql_close();
?>

```

هنا قمنا بطباعة المعلومات والتحقق بواسطة شرط IF دعنا نأخذ عملية طباعة المعلومات :

```

echo "<td align=middle>SiteNmae<br><input type=text name='name' value='$row-
>sitename'></td>";
    echo "<td align=middle>Add Date<br><input type=text name='add_date' value='$row-
>add_date'></td>";
    echo "<td align=middle>Email<br><input type=text name='email' value='$row-
>email'></td>";
    echo "<td align=middle>SiteUrl<br><input type=text name='siteurl' value='$row-
>site_url'></td></tr>";
    echo "<td align=middle>Description<br><input type=text name='desc' value='$row-
>description'></td></tr>";
$id = $row->id;

```

هنا في أول سطر قام بطباعة كود HTML وبداخله أيضاً بعض الأشياء من PHP لا تخف فالمقصود هنا أنه قام بتخزين جميع ما يطبعه بداخل المتغير الخاص بلغة HTML فهنا مثلاً

```

echo "<td align=middle>SiteNmae<br><input type=text
name='name' value='$row->sitename'></td>";

```

قام بطباعة كود HTML وهو عبارة عن حقل الإدخال (INPUT) من نوع (TEXT) وقام بترمز إلي الحقل الموجود بقاعدة البيانات من المتغير row وتخزينه في قيمة المتغير في (html name) والأكثر وضوحاً ما قمنا به في السطر السابق قمنا بوضع المتغير row ثم يليه اسم الحقل في الجدول وهنا اسم الحقل sitename أظن مع تشغيلك للسكربت سوف تتضح أكثر وأكثر وهذا الذي استخدمته مع الجميع

بعد ذلك قمت بطباعة أكواد html وهي عبارة عن فورم للإدخال ويأتي بعده شرط if وهو هنا

```
<input type=submit name="update" value="Edit The Info">
</form>
</tr>
</table>
<?
if ($update) {
    if($name == "" || $add_date == "" || $email == "" || $siteurl == "" || $desc == "" ) {
        die("<b>You left one or more fields blank.</b>");
    }

    $query2 = "update dalal set sitename='$name', add_date='$add_date', email='$email', site_
url='$siteurl', description='$desc'
        where id='$id'";
    mysql_db_query('links',$query2);
    echo "<b>Your record has been updated</b></p>";
}
mysql_close();
?>
```

في الشرط نريد التأكد هل تم تعديل أحد البيانات أم لا وهو شبيه بالشرط الذي استخدمته مع ملف add.php بعده إذا كان هناك إدخال فسيقوم بعدة أوامر وأولها شرط آخر if يتأكد هل هناك من الحقول التي جعلت فارغة أم لا وهي هنا

```
if($name == "" || $add_date == "" || $email == "" || $siteurl == "" || $desc == "" ) {
    die("<b>You left one or more fields blank.</b>");
}
```

إذا كان هناك حقول فارغة سوف يطبع له  
You left one or more fields blank  
أي أنك تركت حقل أو أكثر فارغ  
بعد ذلك يأتي هذا

```
$query2 = "update dalal set sitename='$name', add_date='$add_date', email='$email', site_url
='$siteurl', description='$desc'
    where id='$id'";
mysql_db_query('links',$query2);
echo "<b>Your record has been updated</b></p>";
}
mysql_close();
```

قمنا هنا في أول سطر بتعريف المتغير query2 بعد ذلك خزنا بداخله قيمة أمر SQL التحديث update أمر يقوم بتحديث المعلومات في قواعد البيانات من تجديد وحذف ويكتب هكذا

Update tablename Set (values

الكلمات التي تبدأ بحروف كبيتل هي الثوابت بمعنى لا يتم تغييرها والتي تبدأ بحروف سمول مثل sitename و values فهي التي يتم تغييرها  
كما وضحت في الكود السابق فقد قمت بمساوات المتغيرات name,add\_date,email,site\_url,description  
بما يقابلها من الحقول في الجدول dalal ووضعت شرط where  
أن يكون المتغير

```
id=id$
```

وفي السطر الذي يليه هذا قمت باستخدام الدالة mysql\_db\_query

مسنداً إليه اسم قاعدة البيانات والمتغير الذي خزن قيمة التحديث وهو query2 لكي يتم تحديث قاعدة البيانات وفي السطر الذي يليهما استعملت الدالة mysql\_colse لكي يتم إغلاق الاتصال بقاعدة البيانات وهو آخر سطر اليوم

السكربت يتكون من ثلاثة ملفات، الأول view.php لعرض المدخلات في سجل الزوار، والثاني add.php لإضافة مدخل جديد إلى سجل الزوار، والثالث هو config.php ويحتوي على بيانات قاعدة البيانات. في الملف view.php، يتم جلب البيانات المدخلة من قاعدة البيانات وعرضها واحدة تلو الأخرى في صورة جدول HTML، وأسفل هذا الجدول يوجد نموذج لإضافة تعليق جديد في سجل الزوار. الملف add.php يقوم بأخذ البيانات المرسله من النموذج الموجود في الملف view.php، ثم يقوم بمراجعة هذه البيانات والتحقق منها، ثم إضافتها إلى قاعدة البيانات وإعادة المستخدم إلى الملف view.php. يجب أن نحدد الآن البيانات التي نريد تخزينها في قاعدة البيانات:

- الاسم.
- البريد الإلكتروني.
- الصفحة الشخصية.
- التعليق على الموقع.

### \* قاعدة البيانات

من خلال المعلومات السابقة نلاحظ أننا سنحتاج إلى جدول واحد فقط في قاعدة البيانات، وليكن اسمه guestbook، هذه هي الشفرة التي يجب تنفيذها للحصول على البنية الأساسية للجدول:

```
CREATE TABLE guestbook (
  id INTEGER UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,
  name CHAR(100),
  email CHAR(128),
  homepage CHAR(200),
  date DATETIME,
  ip CHAR(15),
  message TEXT
);
```

شفرة SQL السابقة تعرف جدولاً اسمه guestbook يحتوي على الحقول التالية:

- id - هذا الحقل يخزن رقماً تسلسلياً يزداد بمقدار واحد مع كل حقل جديد يضاف إلى الجدول، ومواصفاته:
  - رقم INTEGER
  - بلا إشارة (موجب دائماً) UNSIGNED
  - إلزامي (لا يمكن أن يترك خالياً) NOT NULL
  - يتم تحديثه تلقائياً إلى رقم أكبر من السابق بواحد AUTO\_INCREMENT
  - وهو المفتاح الأساسي للوصول إلى بيانات الجدول PRIMARY KEY

- name - هذا الحقل يخزن اسم الشخص الموقع في سجل الزوار، ومواصفاته كالتالي:

يخزن نصاً بطول 100 حرف كحد أقصى ( 100 ) CHAR

- email - لتخزين البريد الإلكتروني، 128 حرفاً بحد أقصى (هذا الرقم قياسي تقريباً في أغلب البرامج القائمة على الويب لتخزين عناوين البريد الإلكتروني).

- homepage - لتخزين عنوان الصفحة الشخصية للموقع، 200 حرفاً بحد أقصى -- قيمة معقولة.

- date - لتخزين تاريخ المدخل في سجل الزوار، وهو يخزن قيماً من النوع: DATETIME

- message - لتخزين نص الرسالة (تعليقك على الموقع)، وهو حقل يسع لتخزين عدد كبير جداً من الحروف TEXT

### \* الملف config.php

يحتوي على الملف على شفرة برمجية تقوم بإعداد المتغيرات اللازمة للوصول إلى قاعدة البيانات، وسنقوم بالمرّة بإضافة شفرة الاتصال بقاعدة البيانات إلى هذا الملف بدلا من تكرارها في كل من الملفين view.php و add.php.  
أولا سنقوم بإعداد مجموعة من المتغيرات:

```
/* store the hostname of the MySQL server */
$dbhost = "localhost";
/* store the username to login to MySQL */
$dbuser = "root";
/* store the password to login to MySQL */
$dbpass = "";
/* store the name of the MySQL database */
$dbname = "";
```

كما تلاحظ، قمنا بتعريف أربعة متغيرات مختلفة هي:  
\$dbhost -- وتقوم بتخزين عنوان مزود قاعدة البيانات.  
\$dbuser -- وتقوم بتخزين اسم المستخدم الذي سيستخدم للدخول إلى قاعدة البيانات.  
\$dbpass -- وتقوم بتخزين كلمة المرور التي ستستخدم مع اسم المستخدم للدخول إلى قاعدة البيانات.  
\$dbname -- تخزن اسم قاعدة البيانات التي ستحتوي الجدول guestbook.

يمكن للمستخدم الآن تعديل معلومات هذه المتغيرة عن طريق فتح الملف في المفكرة مثلا وتعديل القيم الموجودة بين علامات الاقتباس مقابل أسماء المتغيرات.  
بعد تعريف المتغيرات السابقة، سنقوم كما اتفقنا بوضع الشفرة اللازمة للاتصال بقاعدة البيانات، وهي عبارة عن استدعاء لدالتين:

```
/* connect to the MySQL server */
mysql_connect($dbhost,$dbuser,$dbpass);
/* set $dbname as the database to be used */
mysql_select_db($dbname);
```

## \* شرح الدوال

mysql\_connect(hostname, username, password)

تقوم الدالة السابقة بالاتصال بمزود قاعدة البيانات الموجود على العنوان hostname، وتحاول الدخول إلى المزود قاعدة البيانات عن طريق إرسال اسم المستخدم username وكلمة المرور password، في حال نجاح العملية تعيد موردا ( resource ) نستطيع استخدامه مع دوال أخرى لتنفيذ أوامر قاعدة البيانات على هذا الإتصال المفتوح (يمكن فتح أكثر من اتصال بأكثر من قاعدة بيانات في نفس الوقت).

جميع المتغيرات الممرّة إلى الدالة اختبارية، إذا لم تدخل password فإن الدالة ستحاول إجراء الاتصال دون تمرير كلمة مرور، وإذا لم تدخل username فستحاول الدالة إجراء الاتصال دون اسم مستخدم، وإذا لم تحدد hostname فإن الدالة ستحاول الإتصال بقاعدة البيانات على العنوان المحلي ( localhost ).  
إذا كنت قد ركبت MySQL مباشرة دون تغيير الإعدادات فإن مزود قواعد البيانات سيسمح مباشرة بالاتصال من الجهاز المحلي باسم المستخدم root والذي لن تكون له كلمة مرور افتراضية، وبالتالي يمكنك الاتصال مباشرة إلى مزود MySQL المحلي عن طريق القيم الموضحة في المتغيرات التي في الأعلى.

mysql\_select\_db(database\_name, link\_identifier)

تقوم هذه الدالة بتحديد قاعدة البيانات التي سيتم العمل عليها، حيث أن مزود قاعدة البيانات يمكن أن يحتوي على أكثر من قاعدة بيانات واحدة.

تحدد اسم قاعدة البيانات عن طريق المتغيرة database\_name، أما المتغيرة الثانية فهو الإتصال الذي تريد أن تحدد قاعدة البيانات له، فإذا كنت قد خزنت القيمة التي أعادتها الدالة mysql\_connect في متغيرة، وأرت الآن تعيين قاعدة البيانات الذي يجب استخدامها في ذلك الإتصال، فيمكنك تمرير المتغيرة إلى الدالة mysql\_select\_db في الوسيطة الثانية، ولكن هذه الوسيطة الثانية اختيارية، فإذا لم تقم بتمريرها فإن الدالة ستعيد قاعدة البيانات التي يجب العمل عليها لآخر اتصال تم فتحه بقاعدة البيانات، وإذا لم يكن هنالك اتصال مفتوح من قبل ستحاول الدالة فتح إتصال بقاعدة البيانات كما لو كنت قد شغلت mysql\_connect() بدون تحديد ال hostname و ال username و ال password و ثم تعيين قاعدة البيانات التي يجب العمل عليها لهذا الإتصال الجديد.

في ملفنا المثال قمنا بتنفيذ هذه الدالة مباشرة بعد فتح اتصال بقاعدة البيانات باستخدام الدالة mysql\_connect, وبالتالي عيننا قاعدة البيانات \$dbname كقاعدة البيانات التي يجب استخدامها مع آخر اتصال فتح بقاعدة البيانات. هذه الدالة تغير true ( صح ) في حالة نجاحها و false ( خطأ ) في حالة فشلها. يمكننا الآن النظر إلى نسخة نهائية كاملة من الملف config.php :

```
<?php
/* store the hostname of the MySQL server */
$dbhost = "localhost";
/* store the username to login to MySQL */
$dbuser = "root";
/* store the password to login to MySQL */
$dbpass = "";
/* store the name of the MySQL database */
$dbname = "";
/* connect to the MySQL server */
mysql_connect($dbhost,$dbuser,$dbpass);
/* set $dbname as the database to be used */
mysql_select_db($dbname);
?>
```

#### \* الملف view.php :

في البداية المقدمات العادية ورأس المستند بلغة HTML :

```
<html dir="rtl">
<head>
</title>سجل الزوار<title>
<style>
body,td,input,select,textarea {
font-family: Tahoma;
font-size: x-small;
}
h1,h2,h3,h4,h5,h6 {
font-family: Arial;
}
</style>
</head>
<body>
</h1>سجل الزوار<h1>
```

بعد أن تتم طباعة رأس المستند بلغة HTML نأتي إلى الجزء التالي وهو جلب البيانات من قاعدة البيانات وعرضها للمستخدم.

في البداية نحتاج إلى اتصال بقاعدة البيانات، وكما قلنا من قبل، الملف config.php يحتوي على الشفرة الكاملة لتجهيز اتصال بقاعدة البيانات، ما علينا القيام به الآن هو إعطاء أمر بتنفيذ البيانات الموجودة في الملف config.php، وهذا الأمر هو include، كالتالي:

```
/* get a database connection */
include 'config.php';
```

النتيجة التي يعطيها الأمر include السابق هي نفسها النتيجة التي كنا سنحصل عليها لو كنا وضعنا الشفرة الموجودة في الملف view.php مباشرة، بمعنى أننا بدلا من أن ننسخ شفرة قاعدة البيانات الموجودة في الملف config.php يدويا ونضعها في أعلى الملف view.php للحصول على وصول إلى قواعد البيانات، فإننا نطلب من مترجم PHP أن يقوم بالعملية آليا، حيث نطلب منه تنفيذ الشفرة الموجودة في الملف config.php كما لو كانت جزءا من الملف الذي استدعى الأمر ( view.php ).

#### \* الأمر include :

يستخدم الأمر include كالتالي :

```
include filename;
```

أي الكلمة include وبعدها اسم الملف الذي نريد إدراجه، وفي مثالنا كان الملف هو config.php.

\* طلب البيانات من قاعدة البيانات

المفترض الآن بأن جميع الإضافات التي تمت إلى سجل الزوار مخزنة في الجدول guestbook في قاعدة البيانات، وأن لدينا اتصالاً الآن بقاعدة البيانات من خلال الملف config.php، نستطيع البدء بعمل استعلام لقاعدة البيانات نطلب فيه البيانات التي نريدها، ويكون ذلك بالشفرة التالية:

```
$res = mysql_query("SELECT  
name,email,homepage,  
UNIX_TIMESTAMP(date) AS date,  
message  
FROM guestbook  
ORDER BY date DESC");
```

### الدالة mysql\_query:

```
mysql_query(query,link_identifier);
```

تقوم الدالة بإرسال الإستعلام (query) إلى قاعدة البيانات عبر الإتصال الذي يمرر إليها (link\_identifier)، وإذا لم يكن هنالك اتصال ممرر، فإن الدالة تستخدم آخر اتصال فتح بقاعدة البيانات، وإذا لم يكن هنالك اتصال قد فتح من قبل ستحاول الدالة إجراء اتصال بقاعدة البيانات عن طريق تنفيذ الدالة mysql\_connect وتمرر إليها قيم hostname و username و password فارغة.

في مثالنا السابق، قمنا بتمرير قيمة واحدة فقط إلى الدالة هي الاستعلام الطويل المحاط بعلامتي الاقتباس " و "، وستقوم الدالة باستخدام آخر اتصال فتح بقاعدة البيانات وهو الاتصال الذي فتح في الملف config.php الذي أدرجناه من قبل في ملفنا (view.php).

نأتي الآن إلى الإستعلام، وهو عبارة عن استعلام بسيط من النوع SELECT كتب بلغة SQL . وسأقدم شرحاً سريعاً هنا،  
:SELECT  
يعني اختيار أو جلب.

```
name,email,homepage,UNIX_TIMESTAMP(date) AS date,message
```

هذه هي أسماء الحقول التي طلبناها (جلبناها) من الجدول، وهي عبارة عن قائمة بسيطة من أسماء الحقول، ما عدى التالي:

```
UNIX_TIMESTAMP(date) AS date
```

تقوم الدالة UNIX\_TIMESTAMP بإعادة التاريخ المخزن في قاعدة البيانات في صورة TIMESTAMP الخاص بنظام unix، وهو عبارة عن نظام للتاريخ يحسب عدد الثواني منذ منتصف الليل في 1970/1/1، وبالتالي فإن هذه القيمة تتغير في كل ثانية، تعتبر ال Timestamp الوحدة القياسية لتخزين المعلومات عن التاريخ والوقت في PHP وأغلب لغات برمجة ونظم Unix، وبالتالي فإننا طلبنا من قاعدة البيانات أن تحول لنا قيمة التاريخ ( date ) إلى ال Timestamp المقابل حتى يسهل التعامل معها في PHP وعرضها بالطريقة التي نريدها ..

بعد ذلك أضفنا المقطع AS date، هذا المقطع يعني بأن الحقل هذا يجب أن يسمى بالاسم date، السبب في إضافتنا لهذا المقطع هي أن الحقول تسمى عادة بأسماء التعبيرات التي تتكون منها، فهذا الحقل مثلاً: UNIX\_TIMESTAMP(date)

سيكون اسمه هو :

```
UNIX_TIMESTAMP(date)
```

وهذا سيجعل التعامل معه في PHP صعباً نوعاً ما، لذا قلنا لقاعدة البيانات أن تسمى البيانات هذه بالاسم date بدلاً من الاسم السابق.

بعد الانتهاء من قائمة الحقول نجد المقطع التالي:

```
FROM guestbook
```

FROM

الكلمة FROM تعني (من) ونحدد بعدها اسم الجدول الذي نريد طلب (جلب) البيانات منه، وهو guestbook في مثالنا السابق.

ORDER BY date DESC

المقطع السابق يحدد الطريقة التي ترتب بها البيانات ..

ORDER BY

تعني (ترتيب حسب) وهي تحدد الحقل الذي نريد ترتيب تبعاً له، وفي مثال الحقل date، أي أننا نريد ترتيب المدخلات حسب تاريخ إضافتها، ولكن استخدام ORDER BY وبعده اسم الحقل يعني فرز البيانات حسب ذلك الحقل تصاعدياً، أي من الأقل إلى الأكبر، ولكننا في الواقع نريد فرزها في برنامجنا تنازلياً (من الأحدث أو الأعلى تاريخاً إلى الأقدم) ولهذا السبب أضفنا المقطع DESC وهو اختصار لكلمة descending أي تنازلياً.