الفصل السابع

مراحل بناء قاعدة بيانات على الويب

فى هذا الفصل سوف نسير معا خطوة بخطوة نحو بناء تطبيق بسيط على الشبكة العنكبوتية Web فى منظومة الإنترنت، وذلك عن طريق لغة البرمجة PHP ومنظومة إدارة قواعد البيانات MySQL.

المحتويات:

- ما هى تطبيقات الويب؟
- بناء جدول قاعدة البيانات.
- طباعة بيانات الجدول على شاشة المتصفح.
- عرض تسجيلة واحدة على شاشة المتصفح.
- استخدام نموذج لإدخال وتحرير البيانات.
- حفظ بيانات النموذج داخل قاعدة البيانات.
- حذف تسجيلات من قاعدة البيانات.

الأهداف التعليمية

١. القدرة على استخدام لغة PHP لإدخال بيانات ديناميكية فى ملف HTML.
٢. التمكن استخدام لغة PHP لاسترجاع بيانات من جدول قاعدة بيانات MySQL.
٣. القدرة على إنشاء نموذج form معيارى بلغة HTML لإدخال وتعديل بيانات الجدول.
٤. التمكن من حفظ بيانات النموذج داخل قاعدة البيانات.
٥. القدرة على حذف تسجيلة من داخل قاعدة البيانات.

ما هى تطبيقات الويب؟

"تطبيقات الويب Web application" عبارة عن مصطلح لوصف موقع إنترنت يقوم بوظيفة تطبيق ما؛ حيث يتم من خلاله الحصول على مدخلات المستخدمين، وتجهيزها، وغالبا تخزينها للاستخدام لاحقا.

فهارس المكتبات، عربات التسوق، مواقع البريد الإلكترونى، ومواقع الأغانى... كل هذه الأمثلة عبارة عن تطبيقات على شبكة الويب، نستخدمها جميعا فى حياتنا اليومية على الإنترنت.

أحد أكثر استخدامات لغة PHP هو تخزين واسترجاع وعرض معلومات نصية من قاعدة بيانات. وفى هذا الفصل سوف نقوم ببناء قاعدة بيانات الكتب،

والذى يقوم بتخزين عنوان الكتاب واسم المؤلف وتاريخ النشر، ورقم الطلب لكل كتاب موجود بالمكتبة، بالإضافة إلى رقم معرف id فريد لا يتكرر لكل تسجيلة.

ومن خلال هذا النموذج البسيط سوف نستكشف معا أساسيات استخدام لغة PHP لبناء قاعدة بيانات MySQL والتفاعل معها. وفى النهاية سوف نمتلك جميع المهارات والأدوات التى نحتاجها لبناء تطبيقات PHP/MySQL أكثر تقدما وتعقيدا.

للمضى قدما فى هذا النموذج التطبيقى يجب أن يكون لديك المعرفة الأساسية ببعض المفاهيم الضرورية، مثل المتغيرات variables، الكائنات/ الأشياء objects، وأسلوب كتابة كود البرمجة programming syntax. ولذلك يجب عليك دراسة الفصل الخاص بأساسيات البرمجة بلغة PHP. كما يجب عليك دراسة الفصل الخاص بأساسيات MySQL وكذلك الفصل الخاص بلغة HTML.

الآن كل ما تحتاج إليه هو محرر نصوص text editor، وبرنامج FTP، وخادم server مزود بلغة PHP و MySQL. ويمكنك الحصول على جميع هذه الأدوات مجانا من على شبكة الإنترنت. أنظر الفصل الرابع: أدوات PHP/MySQL.

المرحلة الأولى: بناء جدول قاعدة البيانات

فى هذه الخطوة يجب أن تعرف اسم قاعدة البيانات، واسم المستخدم، وكلمة السر، ومكان قاعدة البيانات.

سوف نقوم بكتابة كود PHP قصير لا يفعل شيئا إلا إنشاء جدول قاعدة البيانات. يمكن لكل قاعدة بيانات فى MySQL أن تحتوى جداول كثيرة ومختلفة، أو مجموعات من البيانات.

أولا قم بفتح ملف نصى جديد، وأعطه الاسم create.php. ابدأ بكتابة بالنص التالى:

```
<html>
<body>

<?php
  // creation.php: installation script
  // connect to the server
  if( mysql_connect( 'localhost', 'username', 'password' ) )
    echo( "Connected to database.<br>\n" );
  else
    die( "Error! Could not connect to server: ". mysql_error() );

  // select the database
  if( mysql_select_db( 'database_name' ) )
    echo( "Selected the database.<br>" );
  else
    die( "Error! Could not select the database: ". mysql_error() );

  // create the table
  $create_query = "CREATE TABLE `books` ( 'title' VARCHAR(100), `author` VARCHAR(50), `year` VARCHAR(4), 'callNo' VARCHAR(15), `id` INT AUTO_INCREMENT, UNIQUE (`id`) )";

  if( mysql_query( $create_query ) )
    echo( "Table created successfully.<br>" );
  else
    die( "Error! Could not create table: ". mysql_error() );

  // put in dummy values
  $insert_query = "INSERT INTO `books`
    ( `title` , `author`, `year` ) VALUES
    ( 'Descriptive cataloging' , ' Mohammed Fathy ', '2000', '025.5' ) ,
    ( 'Information services' , 'Hishmat Kasim', '2002', '025,4' ),
```

```
    ( 'Database design', 'Ali Shaker', '2005', '025,6' ) ";

  if( mysql_query( $insert_query ) )
    echo( "Values inserted successfully." );
  else
    die( "Error! Could not insert values: ". mysql_error() );

?>

</body>
</html>
```

دعنا ننظر الآن إلى مكونات الكود السابق:

```
// connect to the server
if( mysql_connect( 'localhost', 'username', 'password' ) )
  echo( "Connected to database.<br>" );
else
  die( "Error! Could not connect to server: ". mysql_error() );
```

هنا استخدمنا أحد دوال PHP وهى الدالة ( ) mysql_connect تتطلب هنا تحديد اسم الخادم، اسم المستخدم، كلمة السر. وتقوم هذه الدالة بالربط مع خادم قاعدة البيانات. ونتيجة هذه الدالة إما أن تكون حقيقية true إذا تم الربط بنجاح، أو خاطئة false إذا لم يتم الربط بنجاح. ولذلك قمنا بوضع الدالة داخل العبارة if/else حتى نعرف النتيجة على شاشة المتصفح. ووضع حقل HTML معين مثل
 فى نهاية المخرجات يجعل من السهل قراءتها فى المتصفح.

كما توجد دالة PHP أخرى، وهى ( ) mysql_error والتى تطبع آخر خطأ نتج عن تنفيذ الكود. والملاحظ هنا أن طباعة الجملة التى تتضمنها دالة الخطأ، سوف يساعد على حل أية مشاكل فى الكود.

```
// select the database
if( mysql_select_db( 'database_name' ) )
  echo( "Selected the database.<br>" );
```

```
else
  die( "Error! Could not select the database: ". mysql_error() );
```

تقوم وظيفة mysql_select_db('database_name') وهى إحدى دوال PHP باختيار قاعدة البيانات من خادم MySQL كقاعدة البيانات النشطة. وهذا الكود سوف يسترجع القيمة خطأ false إذا كانت قاعدة البيانات غير موجودة.

```
// create the table
  $create_query = "CREATE TABLE `books` ( 'title' VARCHAR(100), `author` VARCHAR(50), `year` VARCHAR(4),
    `id` INT AUTO_INCREMENT, UNIQUE (`id`) )";
  if( mysql_query( $create_query ) )
    echo( "Table created successfully.<br>" );
  else
    die( "Error! Could not create table: ". mysql_error() );
```

الآن وصلنا إلى أهم جزء فى الكود. تقوم الوظيفة mysql_query(string query) بإرسال أمر استعلام إلى قاعدة بيانات MySQL. ويلاحظ هنا مدى سهولة كود MySQL فى القراءة والفهم:

فى اللغة الإنجليزية، نريد أن نقول للكود:

```
" Select rows from contacts where first name is Susan ".
```

والكود الفعلى هو:

```
" SELECT * FROM `books` WHERE `title` = 'descriptive cataloging' ".
```

وبنفس البساطة، يكون الكود الخاص بإنشاء جدول جديد على النحو التالى:

```
CREATE TABLE `table_name` ( `column1` data_type, `column2` data_type, `column3` data_type )
```

أيا كان عدد الأعمدة التى تحتاجها.

أنواع البيانات: أكثر أنواع البيانات استخداما فى قواعد بيانات MySQL هو النوع VARCHAR أو حروف متغيرة، على سبيل المثال عبارة نصية string. وفى

الكود السابق title VARCHAR(100) نقوم بإنشاء عمود تكون قيمته عبارة نصية، بطول أقصى 100 حرف. ومن الأنواع الأخرى الشهيرة النوع INT (integer) للأعداد الصحيحة، والنوع FLOAT (decimal vales) للقيم العشرية، والنوع BINARY (binary data) للبيانات الثنائية.

ومن المهم هنا تحديد طول مناسب للنوع VARCHAR يسمح باستضافة البيانات التى تريد تخزينها فى هذه النوعية من الحقول. وبإمكانك دائما تعديل الجدول، وتقليص طول الحقل (تحقيقا لكفاءة قاعدة البيانات)، لكن من المهم فى البداية زيادة الحد الأقصى لطول البيانات.

ومن هنا يقوم كود MySQL فى المثال السابق (والذى قمنا بتخزينه فى المتغير $create_query) بأمر قاعدة بيانات MySQL بإنشاء جدول باسم 'books' ويحتوى على الأعمدة title، author، year، وcallNo. أعمدة title، author، year، وcallNo تشرح نفسها، وجميعهم يحتوون على بيانات نصية متغيرة الطول. وقد قمنا أيضا بإضافة عمود id وحددنا نوع بياناته كأرقام صحيحة INT (integer number). سوف يستخدم هذا الحقل للتمييز بين كل مدخل فى قاعدة البيانات.

تعتبر الخاصية AUTO_INCREMENT أحد الملامح العظيمة فى قواعد بيانات MySQL. أى عمود يحتوى على أرقام صحيحة، وله خاصية AUTO_INCREMENT أو الترقيم التلقائى، سوف يتم ملؤه أوتوماتيكيا بقيمة رقمية فريدة، بمجرد إدخال تسجيلة جديدة إلى القاعدة. وهنا يجب الإشارة إلى أهمية وضع الخاصية UNIQUE('id') لتحديد احتواء العمود فقط على قيم فريدة (غير متكررة).

وبالتالى عندما يتم تمرير استعلام MySQL إلى الوظيفة mysql_query ( ) سوف يتم إنشاء جدول على النحو التالى:

callNo	year	author	title	id

أما عن عملية إدراج البيانات فهى بنفس البساطة التى اتسمت بها عملية إنشاء قاعدة البيانات. فقط قم بتمرير كود الكتابة فى MySQL إلى الوظيفة mysql_query ( ). وباستخدام أمر INSERT يمكنك إدراج صفوف متعددة من البيانات. أما الآن سوف نقوم بإدراج بيانات الكتب: descriptive cataloging، information services، وdatabase design إلى جدول books، وذلك لكى يكون لدينا بعض المعلومات فى قاعدة البيانات حتى نتمكن من العمل معها فى هذا النموذج التطبيقى.

```
// put in dummy values
$insert_query = "INSERT INTO `books`
  ( `title` , `author`, `year`, 'callNo' ) VALUES
  ( Descriptive cataloging , 'Mohammed Fathy', '2000', '025,5' ) ,
  ( 'Information services' , 'Hishmat Kasim', '2002', '025,4' ),
  ( 'Database design', 'Ali Shaker', '2005', '025,6' ) ";
if( mysql_query( $insert_query ) )
  echo( "Values inserted successfully." );
else
  die( "Error! Could not insert values: ". mysql_error() );
```

كود MySQL لإدراج بيانات إلى الجدول هو:

```
" INSERT INTO `table_name` ( column, column ) VALUES
( value, value ) "
```

ويمكنك تعريف أعمدة كثيرة – وبأى ترتيب – كما تريد. فالكود التالى:

```
( columnA, columnB ) VALUES ( valueA, valueB )
```

سوف يعطى النتيجة نفسها التى يعطيها الكود التالى:

```
( columnB, columnA ) VALUES ( valueB, valueA )
```

كما يمكنك أيضا إدراج أى عدد من صفوف البيانات كما تريد، على أن تضع بيانات كل صف بين هلاليتين، وتفصل بينهما بفاصلة. ومن ثم عند تمرير الأمر الذى قمنا بتخزينه فى المتغير إلى الوظيفة mysql_query ( ) سوف يتم إدراج بيانات الأصدقاء الثلاثة، ويصير الجدول على النحو التالى:

callNo	year	author	title
025,5	2000	Mohammed Fathy	Descriptive cataloging
025,4	2002	Hishmat Kasim	Information servics
025,6	2005	Ali Shaker	Database design

لاحظ أننا لم نقم بتخصيص أية قيم للعمود id. وهذا يرجع إلى تصميم الجدول. وهنا تقوم MySQL بوضع القيم أوتوماتيكيا فى هذا العمود.

وإلى هنا نكون قد وصلنا إلى نهاية هذا الجزء. ومن خلال أسطر قليلة من الكود قمنا باستخدام لغة PHP ودخلنا على خادم MySQL وقمنا باختيار قاعدة البيانات، وأنشأنا جدولا جديدا فى القاعدة، وكذلك قمنا بإدراج ثلاثة صفوف من البيانات فى الجدول.

لاحظ أنه لكى تقوم ببناء قاعدة البيانات، سوف تقوم بالدخول على هذا الملف مرة واحدة فقط. وإذا تم إعادة تنفيذ كود PHP مرة أخرى، سوف تحصل على رسالة خطأ من MySQL تشير إلى أن الجدول الذى تحاول إنشاءه موجود بالفعل.

فى الخطوة التالية سوف نتعلم معا كيفية طباعة البيانات الموجودة فى الجدول على شاشة المتصفح.

المرحلة الثانية: طباعة بيانات الجدول على شاشة المتصفح

يجب على كل قاعدة بيانات أن توفر طريقة للمستخدم لكى يستعرض من خلالها محتويات القاعدة من البيانات المختلفة. وتوفر لغة PHP مع MySQL ذلك بسهولة. لإعداد قائمة بمداخل قاعدة البيانات، يجب أن نقوم بإنشاء وثيقة PHP جديدة تحتوى على الكود التالى:

```
<html>
<body>
```

```
<?php
  // listing script

  // connect to the server
  mysql_connect( 'localhost', 'username', 'password' )
    or die( "Error! Could not connect to database: ". mysql_error() );

  // select the database
  mysql_select_db( 'database_name' )
    or die("Error! Could not select the database:". mysql_error();

  // retrieve all the rows from the database
  $query = "SELECT * FROM `books` ORDER BY `title`, `author`";

  $results = mysql_query( $query );

  // print out the results
  if( $results )
  {
    while( $books = mysql_fetch_object( $results ) )
    {
      // print out the info
      $id = $books -> id;
      $title = $books -> title;
      $author = $books -> author;
      echo( "$title, $author<br>" );
    }
  }
  else
  {
    die( "Trouble getting contacts from database: ". mysql_error() );
```

```
    }

  ?>

  </body>
  </html>
```

سوف يقوم الكود السابق بعرض قائمة مبسطة جدا بالكتب الموجودة فى قاعدة البيانات، مرتبة بالعنوان ثم المؤلف:

```
Database design, Ali Shaker
Descriptive cataloging, Mohammed Fathy
Information services, Hishmat Kasim
```

لقد شاهدنا من قبل الكثير من الكود السابق، فالكود الخاص بالربط مع الخادم، واختيار قاعدة البيانات هو ذاته الذى استخدمناه فى الجزء الأول، مع استثناء وحيد، وهو أننا استخدمنا جملة or بدلا من if else.

```
  // connect to the server
    mysql_connect( 'localhost', 'username', 'password' )
      or die( "Error! Could not connect to database: ".
mysql_error() );
```

فى حالة استرجاع خطأ false كنتيجة للدالة ( ) mysql_connect يقوم هذا الكود بتنفيذ جملة die. وهذه طريقة سريعة لطباعة رسالة على شاشة المتصفح فقط فى حالة وجود خطأ ما.

```
  // retrieve all the rows from the database
    $query = "SELECT * FROM `books` ORDER BY `title`,
`author`";
```

فى استفسار MySQL السابق، نقوم باختيار جميع الصفوف rows من جدول books، ونرتبها أولا بالعنوان، ثم بالمؤلف بعد ذلك، باستخدام أمر ORDER BY.

```
  // print out the results
    if( $results )
    {
```

```
        while( $books = mysql_fetch_object( $results ) )
        {
          // print out the info
          $id = $contact -> id;
          $title = $books -> title;
          $author = $books -> author;
          echo( "$title, $author<br>" );
        }
      }
      else
      {
        die( "Trouble getting contacts from database: ". mysql_error()
);
      }
```

الآن إذا كانت هناك نتائج للاستفسار، ندخل فى الحلقة التكرارية While. وتقوم الدالة mysql_fetch_object (resource result) باسترجاع كل صف ضمن النتائج، وعندما لا يكون هناك مزيد من الصفوف، تقم باسترجاع غير صحيح false.

يتم تخزين بيانات الصف فى المتغير $books. وللوصول إلى خصائص كائن معين object properties فى لغة PHP نكتب الكود بالشكل التالى:

$object - > property

ومن ثم نقوم بالحصول على القيم المختلفة للمتغيرات $id و $title و $author من المتغير $books. ثم نستخدم أمر ( ) echo لطباعة (على الشاشة) الاسم الأخير متبوعا بالاسم الأول.

والآن قد يرى البعض أن قائمة بسيطة بالكتب تعتبر غير ذات فائدة كبيرة دون الحصول على تفاصيل إضافية مثل رأس الموضوع. ومن ثم سوف نقوم فى الجزء التالى بإنشاء وثيقة PHP جديدة باسم details.php والتى تعرض تفاصيل كتاب معين فى الجدول.

وحتى تتمكن هذه الصفحة من العمل، يجب أن نربط الأسماء الموجودة فى قاعدة البيانات بهذه الصفحة (details.php). إذا قمنا بتغيير السطر:

```
echo( "$title, $author<br>" );
```

لإدراج رابطة html بها المتغير "id" لكل كتاب مع صفحة details.php

```
echo( "<a href='details.php?id=$id'>$title, $author<br></a>" );
```

سوف تكتمل قائمة التسجيلات المطلوبة. سوف يؤدى تغيير الكود عاليه إلى إعادة توجيه الصفحات إلى صفحة details.php، وسوف يظهر عنوان الويب بالشكل التالى:

```
/details.php?id=x
```

حيث أن x هو رقم فريد يقوم بتعريف كل تسجيلة ضمن قاعدة بيانات MySQL. هذه الطريقة لتمرير المتغير $id بقيمة x يطلق عليها جملة استدعاء/ طلب عنوان URL request string.

فى الجزء التالى سوف نقوم بإنشاء صفحة details.php لعرض التفاصيل الخاصة بكل شخص فى قاعدة البيانات.

المرحلة الثالثة: عرض تسجيلة واحدة على شاشة المتصفح

فى هذه الخطوة سوف نقوم بالحصول على متغير من عنوان URL ونستخدمه فى اختيار تسجيلة واحدة من الجدول؛ ومن ثم عرض محتوياتها.

```
<html>
<body>
<?php
  // listing script

  // connect to the server
  mysql_connect( 'localhost', 'low_test', 'test' )
    or die( "Error! Could not connect to database: ".
mysql_error() );
```

```
    // select the database
    mysql_select_db( 'low_test' )
      or die( "Error! Could not select the database: ". mysql_error()
);

    // get the id from the URL request
    $id = $_REQUEST['id'];

    // retrieve the row from the database
    $query = "SELECT * FROM `books` WHERE `id`='$id'";
    $result = mysql_query( $query );
    // print out the results
    if( $result && $books = mysql_fetch_object( $result ) )
    {
        // print out the info
      $title = $books -> title;
      $author = $books -> author;
      $year = $books -> year;

      ?>
      <table border="1" cellpadding="3" cellspacing="3">
        <tr>
          <th align="left">Author</th>
          <td align="left"><?php echo($author) ?></td>
        </tr>
        <tr>
          <th align="left">Title</th>
          <td align="left"><?php echo($title) ?></td>
        </tr>
        <tr>
          <th align="left">Year</th>
          <td align="left"><?php echo($year) ?></td>
        </tr>
```

```
    </table>
    <?php
  }
  else
  {
    die( "Error: Could not get contact from database. ".mysql_error() );
  }

?>
</body>
</html>
```

يقوم هذا الكود بإنشاء ملخص للمعلومات فى جدول بسيط:

Author	Mohammed Fathy
Title	Descriptive cataloging
Year	2000

ولكى نعرف التسجيلة التى يتم عرضها، نقوم بالحصول على المعلومات من جملة طلب/ استدعاء عنوان URL Request String، باستخدام أحد المتغيرات الخاصة بلغة PHP وهو المتغير $_REQUEST. وهذا المتغير عبارة عن مصفوفة تحتوى على جميع المتغيرات والقيم التى توجد فى جملة طلب عنوان URL.

```
// get the id from the URL request query string
  $id = $_REQUEST['id'];
```

وبالتالى إذا كان لدينا عنوان URL مثل details.php?id=3 فإن المتغير $_REQUEST[id] سوف يعطينا القيمة ٣. وهذه هى الطريقة التى نحصل بها على id فريد للتسجيلة التى نريد عرضها.

```
// retrieve the row from the database
  $query = "SELECT * FROM `books` WHERE `id`='$id'";
  $result = mysql_query( $query );
```

وبعد ذلك نوجه استفسارا إلى قاعدة بيانات MySQL للتسجيلة التى يساوى id فيها القيمة التى حصل عليها المتغير $_REQUEST[id].

```
// print out the results
  if( $result && $books = mysql_fetch_object( $result ) )
  {
    // print out the info
    $title = $books -> title;
    $author = $books -> author;
    $year = $books -> year;
```

فإذا كان لدينا نتيجة، نحصل على التسجيلة من قاعدة البيانات. وبعد ذلك نقوم بتحديد المتغيرات $title و $author و $year باستخدام المعلومات من الكائن $books.

```
?>
<table border="1" cellpadding="3" cellspacing="3">
    <tr>
    <th align="left">Author</th>
    <td align="left"><?php echo($author) ?></td>
    </tr>
    <tr>
    <th align="left">Title</th>
    <td align="left"><?php echo($title) ?></td>
    </tr>
    <tr>
    <th align="left">Year</th>
    <td align="left"><?php echo($year) ?></td>
    </tr>
  </table>
  <?php
```

والآن نقوم بتصميم جدول HTM لعرض معلومات التسجيلة. وقد استخدمنا هنا حقل جدول table tag بسيط، إطاره واحد بيكسيل one pixel border. سوف نقوم بطباعة صف فى الجدول لكل من البيانات الثلاثة التى لدينا (المؤلف، العنوان، تاريخ النشر).

لكى نبدأ هذا الجزء من الكود، قمنا بوضع حقل نهاية PHP (<?). وأنهيناه بحقل بداية PHP (php?>). حقل النهاية يجعل الخادم يعرف أين يبدأ كود PHP. أما حقل البداية فيجعل الخادم يعرف أين يبدأ كود PHP مرة أخرى. وأى شيء فيما بين حقلى البداية والنهاية <?... php?> يعتبر كود، وأى شيء آخر يعرض مباشرة على شاشة المتصفح.

بمعنى أنه للخروج مؤقتا من كود PHP يمكننا كتابة كود HTML كما نفعل بشكل طبيعى. وبدون الحاجة لاستخدام وظيفتى ( ) echo و ( ) print.

وبذلك نكون قد أنشأنا جدولا بسيطا، تكون الخلية الأولى فى كل صف عبارة عن رأس الصف header، مثل المؤلف. وتعرض الخلية الثانية فى كل صف المعلومات الخاصة بقاعدة البيانات. وعن طريق كتابة كود PHP التالى:

```
<?php echo($author) ?>
```

نقوم بإدراج كود PHP مباشرة ضمن جدول HTML. وببساطة نستخدم الوظيفة ( ) echo لطباعة المتغير أو النص المطلوب، فى هذا المثال: المتغير $author.

وهذا كل ما فى الأمر.. الآن نستطيع عرض التفاصيل الكاملة لتسجيلة معينة فى قاعدة البيانات.. فى شكل HTML.. على شاشة المتصفح. فى الجزء التالى سوف نتعرف معا على كيفية إضافة تسجيلات إلى قاعدة البيانات عن طريق نموذج HTML.

المرحلة الرابعة: استخدام نموذج لإدخال وتحرير البيانات

تسهل لغة PHP عملية الحصول على المعلومات من خلال نموذج HTML form،

ثم إضافتها على قاعدة بيانات MySQL.

بداية يجب أن نقوم بإنشاء نموذج HTML form. ويمكن إعادة استخدام النموذج الذى قمنا بتنفيذه فى الجزء السابق لعرض بيانات التسجيلة.

وإذا كنت تتذكر.. لقد كنا نعرض تفاصيل التسجيلة فى جدول بسيط:

Author	Mohammed Fathy
Title	Descriptive cataloging
Year	2000

وهذه تعتبر أفضل طريقة لتقديم المعلومات بغرض تعديلها. إن استخدام ذات الشكل يسمح لنا بإعادة استخدام كود HTML كما يحافظ على ثبات شكل الصفحات. وهذا ما يسهل استخدام قاعدة البيانات.

لإنشاء نموذج نستطيع من خلاله تعديل تفاصيل التسجيلات، يجب أن ننشئ ملف جديد باسم edititem.php. وسوف ترى أن كود PHP فى الملف edititem.php سوف يشبه إلى حد كبير ذلك الموجود فى ملف details.php مع بعض التغييرات الطفيفة.

```
<html>
<body>
<?php
  // editing/adding script
  // get the id from the URL request
  $id = $_REQUEST['id'];
    if( $id )
  {
    // connect to the server
    mysql_connect( 'localhost', 'low_test', 'test' )
      or die( "Error! Could not connect to database: ".
```

```
mysql_error() );

        // select the database
        mysql_select_db( 'low_test' )
         or die( "Error! Could not select the database: ".
mysql_error() );
              // retrieve the row from the database
        $query = "SELECT * FROM `books` WHERE `id`='$id'";

        $result = mysql_query( $query );

        if( $result && $books = mysql_fetch_object( $result ) )
        {
          // set our variables
          $title = $books -> title;
          $author = $books -> author;
          $year = $books -> year;
        }
            }
     // print out the form
     ?>
  <form action="saveitem.php" method="get">
   <input type="hidden" name="id" value="<?php echo($id) ?>">

  <table border="1" cellpadding="3" cellspacing="3">
   <tr>
   <th align="left">المؤلف</th>
   <td align="left">
   <input name="author" type="text" value="<?php echo($author)
   ?>" />
   </td>
    </tr>
    <tr>
   <th align="left">العنوان</th>
```

```
    <td align="left">
   <input name="title" type="text" value="<?php echo($title) ?>"
/>
    </td>
    </tr>
    <tr>
      <th align="left">تاريخ النشر</th>
   <td align="left">
    <input name="year" type="text" value="<?php echo($year) ?>"
/>
      </td>
    </tr>
  </table>
  <br>
    <input type="Submit" value="حفظ التسجيلة">
   <br>
     <a href="delete.php?id=<?php echo($id) ?>">حذف التسجيلة</a>
      </form>
  </body>
   </html>
```

الكود السابق سوف ينتج عنه جدول HTML يشبه كثيرا الجدول الذى صادفناه فى الجزء السابق:

أول الأشياء التى قمنا بتغييرها هنا هى عند استرجاع البيانات فى ملف details.php كنا نسترجع المتغير id من جملة طلب/ استدعاء عنوان URL، ثم نعرض بيانات التسجيلة التى تقابل هذا الرقم (id). وفى حالة عدم وجود متغير id فى جملة طلب عنوان URL، لن نحصل على نتائج من قاعدة البيانات؛ ومن ثم سوف تعرض رسالة الخطأ التالية: "Error: Could not get contact from database" على شاشة المتصفح.

أما فى هذا الكود فإنه فى حالة وجود متغير id فى جملة طلب عنوان URL فإننا سوف نفترض أننا نقوم بإضافة تسجيلة جديدة إلى قاعدة البيانات. وبالتالى إذا كان عنوان URL الصفحة هو edititem.php?id=2 سوف نقوم بتحرير التسجيلة رقم ٢. أما إذا لم يكن لدينا جملة طلب عنوان URL – على سبيل المثال: edititem.php – سوف نفترض أننا نقوم بإدخال تسجيلة جديدة.

```
// get the id from the URL request
$id = $_REQUEST['id'];

if( $id )
{
  // connect to the server
  ...
     $title = $books -> title;
  $author = $books -> author;
  $year = $books -> year;
```

إذا كانت هناك قيمة للمتغير $id سوف نحصل على المتغيرات $title و $author و $year. وخلاف ذلك سوف تترك هذه المتغيرات غير معرفة.

الآن يجب أن نقوم ببعض التغييرات الطفيفة على مخرجات HTML. أولا نضيف حقل نموذج form tag:

```
<form action="saveitem.php" method="get">
```

```
...
</form>
```

هذا نموذج html بسيط، يستخدم طريقة GET (جملة طلب عنوان URL) لإرسال معلومات. وهو يشير إلى saveitem.php وهو عبارة عن ملف سوف نقوم بإنشائه فى الجزء التالى.

كما قمنا أيضا بإدراج حقل مدخلات مخفى hidden للمتغير $id؛ وذلك لأن حقل id فى التسجيلة يجب ألا يكون قابلا للتحرير بواسطة مستخدم القاعدة. والغرض الوحيد من المتغير id هنا هو أن تتمكن قاعدة بيانات MySQL من تمييز كل مدخل (تسجيلة) بشكل فريد.

```
$id = $contact -> id;
...
<input type="hidden" name="id" value="<?php echo($id) ?>">
```

وبدلا من مجرد عرض المتغيرات $author و $title و $year فى خلايا جدول HTML، قمنا بتحديدهم كقيمة افتراضية default value فى عناصر النموذج.

وبعد ذلك قمنا بإضافة زر button لإرسال المعلومات، وآخر الأشياء التى قمنا بها هو إنشاء رابطة لحذف البيانات/ التسجيلة:

```
<input type="submit" value="حفظ التسجيلة">
<br>
<a href="delete.php?id=<?php echo($id) ?>">حذف التسجيلة</a>
```

تقوم هذه الرابطة بتمرير رقم id فى التسجيلة إلى ملف delete.php والذى سوف نقوم بإنشائه لاحقا.

فى الجزء التالى سوف نقوم بإنشاء الملف saveitem.php الذى أشرنا إليه فى أثناء هذا الجزء. وظيفة هذا الملف هى إضافة بيانات النموذج إلى جدول قاعدة البيانات.

المرحلة الخامسة: حفظ بيانات النموذج داخل قاعدة البيانات

بعد أن انتهينا من إنشاء النموذج الخاص بإدخال وإرسال البيانات، يجب البدء

فى كتابة كود PHP الذى سيتولى استلام وحفظ المعلومات داخل جدول قاعدة بيانات MySQL.

```
<html>
<body>


<?php
  // saving script

  // connect to the server
  mysql_connect( 'localhost', 'username', 'password' )
    or die( "Error! Could not connect to database: ". mysql_error() );
   // select the database
  mysql_select_db( 'database_name' )
    or die( "Error! Could not select the database: ". mysql_error() );

  // get the variables from the URL request string
  $id = $_REQUEST['id'];
  $author = $_REQUEST['author'];
  $title = $_REQUEST['title'];
  $year = $_REQUEST['year'];

  // if $id is not defined, we have a new entry, otherwise update the old entry
  if( $id )
  {
    $query = "UPDATE `books` SET `author`='$author', `title`='$title',
      `year`='$year' WHERE `id`='$id'";
  }
```

```
    else
    {
      $query = "INSERT INTO `books` ( `author`,`title`,`year` )
        VALUES ( '$author','$title','$year' )";
    }

    // save the info to the database
    $results = mysql_query( $query );

    // print out the results
    if( $results )
    {
      echo( "تم حفظ التسجيلة بنجاح" );
    }
    else
    {
      die( "حفظ المعلومات بقاعدة البيانات: هناك مشكلة فى". mysql_error()
);
    }

  ?>

  </body>
  </html>
```

بعد الربط مع قاعدة البيانات والحصول على المتغيرات من استعلام URL، نريد أن نرى ما إذا كان للمتغير $id قيمة معينة. فإن كان الأمر كذلك، نقوم بتحديث التسجيلة. وإذا لم يكن للمتغير $id قيمة معينة، نقوم بإدراج البيانات فى تسجيلة جديدة.

```
  // if $id is not defined, we have a new entry, otherwise update the
old entry
  if( $id )
  {
```

```
    $query = "UPDATE `books` SET `author`='$author',
`title`='$title',
      `year`='$year' WHERE `id`='$id'";
  }
  else
  {
    $query = "INSERT INTO `books` ( `author`,`title`,`year` )
      VALUES ( '$author','$title','$year' )";
  }
```

نستخدم الأمر UPDATE لتحديث تسجيلة موجودة بقاعدة البيانات. ويكون شكل هذا الأمر على النحو التالى:

```
"UPDATE `table_name` SET `columnA`='valueA',
`columnB`='valueB' ".
```

وإذا كنا نقوم بإدخال تسجيلة جديدة، نستخدم الأمر INSERT بنفس الشكل الذى صادفناه فى الجزء الأول عندما كنا نقوم بعملية إنشاء جدول قاعدة البيانات.

المرحلة السادسة: حذف تسجيلات من قاعدة البيانات

آخر ملف نقوم بإنشائه هو delete.php والذى يقوم ببساطة بحذف التسجيلة المعروضة على الشاشة من داخل قاعدة البيانات.

سوف نحصل على id من جملة طلب عنوان URL ثم نقوم بحذف هذه التسجيلة من جدول قاعدة البيانات.

```
<html>
<body>

<?php
  // deleting script

  // connect to the server
  mysql_connect( 'localhost', 'low_test', 'test' )
    or die( "Error! Could not connect to database: ".
```

```
mysql_error() );

    // select the database
    mysql_select_db( 'low_test' )
      or die( "Error! Could not select the database: ". mysql_error()
);

    // get the id from the URL request
    $id = $_REQUEST['id'];

    // delete the row from the database
    $query = "DELETE FROM `books` WHERE `id`='$id'";

        $result = mysql_query( $query );

    // print out the results
    if( $result  )
    {
      echo( "تم حذف التسجيلة بنجاح" );
    }
    else
    {
      die( "خطأ: لم تحذف التسجيلة:". mysql_error() );
    }

  ?>

  </body>
  </html>
```

يعتبر هذا الكود الخاص بالحذف أبسط الأكواد الذى استخدمناه فى هذا النموذج التطبيقى، وأقصرها أيضا.

نقوم بالربط مع قاعدة البيانات، ثم نحصل على المتغير $id من عنوان URL

وبعد ذلك استخدمنا أمر DELETE لحذف التسجيلة التى يتساوى فيها القيمة الموجودة داخل عمود id مع قيمة المتغير $id.

الخاتمة: ما الذى تعلمناه؟ وما هى الخطوة التالية؟

قمنا فى هذا الفصل بتقديم أساسيات لغة البرمجة PHP وكذلك مبادئ قاعدة بيانات MySQL، وكيفية استخدامهما معا. وفى معرض إنشاء قاعدة بيانات بسيطة، تعلمنا معا:

- استخدام لغة PHP لإدخال بيانات ديناميكية فى ملف HTML.
- استخدام لغة PHP لاسترجاع بيانات من جدول قاعدة بيانات MySQL.
- استخدام نموذج form معيارى بلغة HTML لإدخال وتعديل بيانات الجدول.
- حفظ بيانات النموذج داخل قاعدة البيانات.
- حذف تسجيلة من داخل قاعدة البيانات.

قاعدة البيانات الحالية قد تكون مفيدة، لكنها سوف تصبح أكثر فائدة بإضافة المزيد من الإمكانات إليها. على سبيل المثال: حقول بيانات أكثر (مثل مكان النشر، اسم الناشر، التبصرات، الخ)، كل ذلك يمكن بسهولة إضافته، وجعل قاعدة البيانات أكثر فائدة. فقط، كل ما علينا هو تعديل أكواد PHP لإدراج أعمدة جديدة عند تصميم الجدول، وإضافة المزيد من العناصر إلى نموذج HTML form.

ومع زيادة حجم قاعدة البيانات، تصبح إمكانية البحث search ضرورية. وهذا ليس صعبا فى لغة PHP من خلال إجراء بعض التعديلات والإضافات على الأكواد، وكذلك على هيكل قاعدة بيانات MySQL.

فى المستقبل القريب إن شاء الله، سوف أسعى إلى إضافة المزيد حول استخدام PHP و MySQL. أما الآن فيمكنك الاستعانة ببعض المصادر الموجودة فى الملحق المرفق لتعلم المزيد.

تدريبات عملية

١. عرف بإيجاز تطبيقات الويب، مع ذكر أمثلة؟

٢. تستخدم الدالة..... لتحديد اسم الخادم، واسم المستخدم، وكلمة المرور فى تطبيقات قواعد البيانات على الويب.

٣. ما هى وظيفة الدالة ( ) mysql_error؟

٤. اشرح الكود التالى:

```
$insert_query = "INSERT INTO `books` ( `title` , `author`, `year`, 'callNo' ) VALUES ( Descriptive cataloging , 'Mohammed Fathy', '2000', '025,5' ) , ( 'Information services' , 'Hishmat Kasim', '2002', '025,4' ), ( 'Database design', 'Ali Shaker', '2005', '025,6' )";
```

٥. وتقوم الدالة..... باسترجاع كل تسجيلة ضمن نتائج البحث فى قاعدة البيانات، وعندما لا يكون هناك مزيد من التسجيلات، تعطى قيمة "غير صحيح false".

٦. ما الدور الذى يلعبه المتغير $_REQUEST فى لغة PHP؟

٧. فسر الكود التالى:

```
<input type="submit" value="بحث">
<br>
<a href="delete.php?id=<?php echo($id) ?>">مسح التسجيلة</a>
```

٨. نستخدم الأمر..... لتحديث تسجيلة موجودة بقاعدة البيانات.

٩. ما هى وظيفة ملف delete.php فى مشروع قاعدة البيانات الذى عرضناه فى هذا الفصل؟

١٠. كيف يمكنك إضافة أعمدة جديدة إلى جدول قاعدة بيانات MySQL؟

المصادر

1. Barry Eaglestone and Mick Ridley. ***Web database systems.*** London : McGraw-Hill, c2001.
2. David Taniar. ***Web-powered databases [electronic resource].*** Hershey, PA : Idea Group Inc., c2002.