PHP與MySQL的運作

mysql\_connect():建立資料連結

mysql\_connect(MYSQL 伺服器位址,帳號,密碼)

連線到MYSQL後會回傳一個連接識別碼(link identifier),否則會回傳FALSE

mysql\_pconnect():建立永久性資料連結

mysql\_connect(MYSQL 伺服器位址,帳號,密碼)

連線到MYSQL後會回傳一個連接識別碼(link identifier),否則會回傳FALSE

1. 使用mysql\_pconnect()函式建立連線之前，會先檢查相同設定的連線是否存在。如果已經有的話則會直接使用而不重覆建立。
2. 使用mysql\_connect()函式建立連線，在程式碼執行完畢後就會自動關閉，若有需要也可以使用mysql\_close()函式將連線關閉。
3. 使用mysql\_pconnect()函式建立連線，在程式碼執行完畢後不會自動關閉，會保持以供後續程式使用，也不能使用mysql\_close()關閉。

例：

<?php

header("content-type:text/html;charset=utf-8");

$db\_link=mysql\_connect("localhost","root","1234");

if (!$db\_link){

echo "資料連結失敗辣";

}else{

echo "資料連結成功辣";

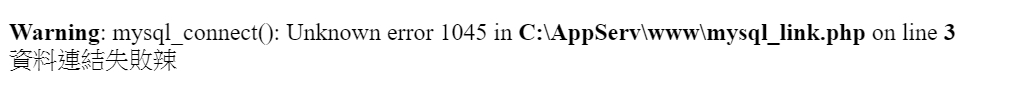
}

?>

控制無法連線的錯誤訊息

提供連線的資料(連線位置,帳密)有誤時，會出現PHP的錯誤訊息

例：



而這些訊息會顯示出一些可能會造成安全疑慮的資訊

只要在mysql\_connect前加上一個『@』就可以抑制錯誤訊息的產生，使用die函式可以馬上中止程式的執行，避免往下執行不必要的動作

例：

<?php

header("content-type:text/html;charset=utf-8");

$db\_link=@mysql\_connect("localhost","root","1qq740528");

if (!$db\_link){

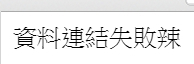
die ("資料連結失敗辣");

}else{

echo "資料連結成功辣";

}

?>



設定字元集與連線校對

mysql\_query()函式可以執行指定的SQL指令，設定字元集與連線校對的指令是SET NAMES ‘編碼方式’

例：

mysql\_query(“SET NAMES ‘utf-8’”);

使用引入檔-提高程式開發效率,方便維護

<?php

//資料庫主機設定

$db\_host="localhost";

$db\_username="root";

$db\_password="qq740528";

//連線伺服器

$db\_link=@mysql\_connect($db\_host, $db\_username, $db\_password);

if (!db\_link) die ("資料庫連線失敗辣");

//設定字元集與連線校對

mysql\_query("SET NAMES 'utf-8'");

?>

在程式頁面上使用連線引入檔

include(connMysql.php);或require(connMysql.php);

require和include的不同

require適合用來引入靜態的內容，而include則適合用來引入動態的程式碼。

include在執行時，如果include進來的檔案發生錯誤的話，會顯示警告，不會立刻停止

而require 則是會顯示錯誤，立刻終止程式，不再往下執行。include可以用在迴圈；require不行。

若您無論如何都要引用某個檔案，則使用 require 或 include 皆可。若您需經條件判斷之後，才能決定是否引用該檔案的話，則只能使用 include。

選擇資料庫:mysql\_select\_db()

mysql\_select\_db(資料庫名稱[,連接識別碼])

如果沒有設定識別碼，會直接使用最近的資料庫連線所產生的識別碼。

例：

<?php

header ("Content-Type:text/hmtl;charset=utf-8");

include ("connMysql.php");

$seldb=@mysql\_select\_db("class");

if (!seldb){

die ("資料庫連結成功辣");

}else {

echo "資料庫連接失敗辣";

}

?>

執行資料表查詢:mysql\_query()

mysql\_query(SQL指定字串[,連接識別碼])

取得資料表查詢結果:mysql\_fetch\_row()

取得以整數為索引鍵的陣列結果

讀取查詢結果的一筆資料並將指標往下移動，到達底部會回傳FALSE

例:

<?php

header ("Content-Type:text/html;charset=utf-8");

include ("connMysql.php");

$seldb=@mysql\_select\_db("class");

if (!$seldb) die("資料庫選擇成功辣");

$sql\_query = "SELECT \* FROM students";

$result = mysql\_query($sql\_query);

while($row\_result=mysql\_fetch\_row($result)){

foreach($row\_result as $key=>$value){

echo $key."=".$value."<br/>";

}

echo "<hr/>";

}

?>

使用mysql\_query()函式執行$sql\_query中的SQL指令取得資料並放入$result

結果:



取得資料表查詢結果:mysql\_fetch\_assoc()

取得以欄位為索引鍵的陣列結果

讀取查詢結果的一筆資料並將指標往下移動，到達底部會回傳FALSE

例：

<?php

header ("Content-Type:text/html;charset=utf-8");

include ("connMysql.php");

$seldb=@mysql\_select\_db("class");

if (!$seldb) die("資料庫選擇成功辣");

$sql\_query = "SELECT \* FROM students";

$result = mysql\_query($sql\_query);

while($row\_result=mysql\_fetch\_assoc($result)){

foreach($row\_result as $key=>$value){

echo $key."=".$value."<br/>";

}

echo "<hr/>";

}

?>

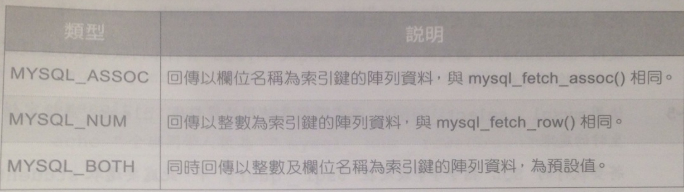
結果:



取得資料表查詢結果:mysql\_fetch\_array()

取得以陣列儲存的查詢結果

mysql\_fetch\_array(資源識別碼[,類型])



mysql\_num\_rows():取得查詢結果筆數

my\_num\_rows(資源識別碼)

mysql\_data\_seek():移動記錄指標

mysql\_data\_seek(資源識別碼,記錄指標位置)

記錄指標位置是以數字表示，0表示第一筆資料,1代表第二筆資料，依此類推 如果移動失敗會回傳FALSE

例：

<?php

header ("Content-Type:text/html;charset=utf-8");

include ("connMysql.php");

$seldb=@mysql\_select\_db("class");

if (!$seldb) die("資料庫選擇成功辣");

$sql\_query = "SELECT \* FROM students";

$result = mysql\_query($sql\_query);

mysql\_data\_seek($result,4);

$row\_result=mysql\_fetch\_assoc($result);

foreach($row\_result as $key=>$value){

echo $key."=".$value."<br/>";

}

?>

結果:

