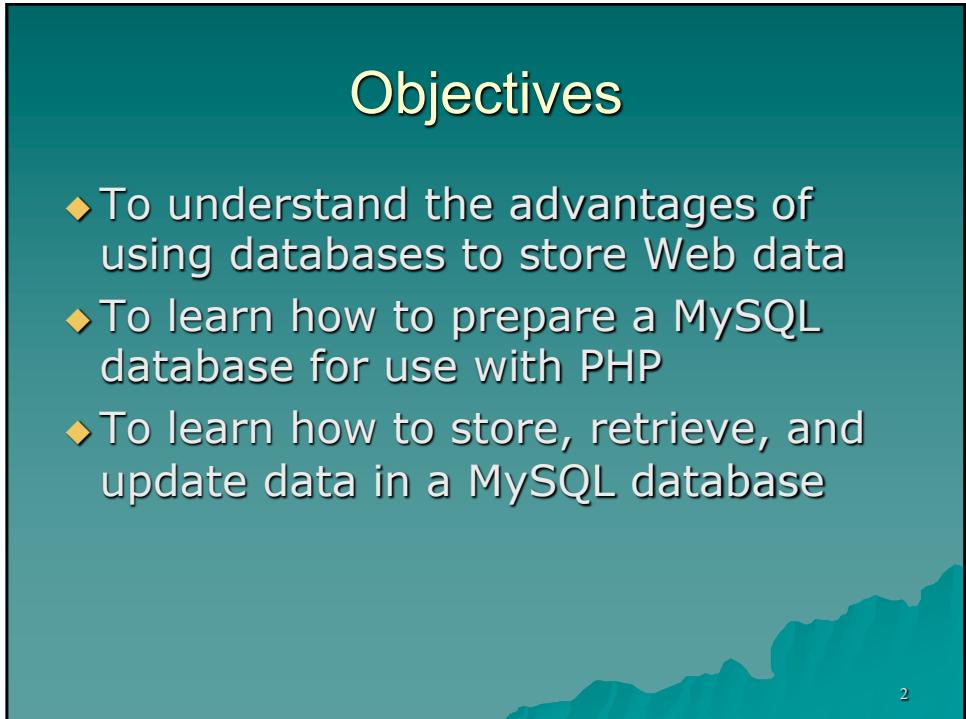


Web Development

Chapter 6.1. Using MySQL with PHP

1



Objectives

- ◆ To understand the advantages of using databases to store Web data
- ◆ To learn how to prepare a MySQL database for use with PHP
- ◆ To learn how to store, retrieve, and update data in a MySQL database

2

2

1

Content

- ⇒ **1. Database and MySQL Overview**
- 2. Basic SQL commands**
- 3. Creating a table**
- 4. Inserting data to a table**
- 5. Retrieving data from a table**
- 6. Updating data for a table**

3

What is a database?

- ◆ A set of data organized into one or more computer files.
- ◆ Using files for product inventory is a type of database
- ◆ Generally the term is reserved for more formal database systems like access, Oracle or MySQL.

4

4

2

Advantages of Databases Over Files

- ◆ Faster access
- ◆ Better concurrent access
- ◆ Easier changes to data and scripts
- ◆ Increased security

5

Relational Database?

- ◆ A database is a collection of tables with defined relationships between them
- ◆ Columns define attributes of the data
 - All data in a column must have the same data type
- ◆ A record is stored in a row

table name

First Name	Last Name	Phone
Nadia	Li	2687
Madhu	Charu	7856
Ajuma	Kinsaka	4489
Wade	Randal	5257
Helen	Clark	2147

row

column

Primary key

Product Number	Product	Cost	Weight	Number Avail
0	Hammer	\$5.00	12	123
1	Screw Driver	\$3.00	2	144
2	Wrench	\$2.50	1.5	244

6

6

3

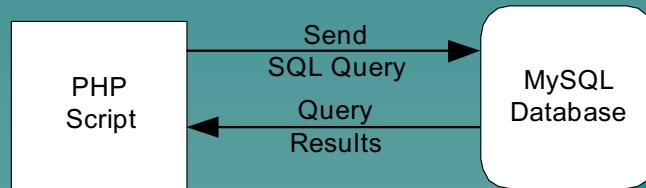
Which Database System

- ◆ PHP works with a variety of databases that include:
 - Oracle
 - Access
 - Ingres
 - SQL Server
 - MySQL
- ◆ Will use MySQL since simple to use, free and very popular.

7

Using A Query Language

- ◆ When using a database, use a separate query language to work with database
- ◆ Within MySQL, use Structured Query Language (SQL), to access database



8

Content

1. Database and MySQL Overview

⇒ **2. Basic SQL commands**

3. Creating a table

4. Inserting data to a table

5. Retrieving data from a table

6. Updating data for a table

9

9

2. Basic SQL commands

- Connecting to MySQL from the Command Line

`mysql -uusername -p`

E.g.:

`>mysql -uroot`

- To EXIT MySQL:

`EXIT;`

10

5

2. Basic SQL Commands (2)

- ◆ SQL statements end with a semicolon

- ◆ View databases

```
SHOW DATABASES;
```

- ◆ Creating a database

```
CREATE DATABASE trii;
```

- ◆ Importing a database:

```
mysql -uusername -ppassword  
        databasename < filename.sql
```

E.g.:

```
mysql -uroot trii < trii.sql
```

11

2. Basic SQL Commands (2)

- ◆ Use database *databasename*

```
USE databasename;
```

- ◆ Display all tables in a database

```
SHOW TABLES;
```

- ◆ View column details for a table

```
DESC tablename;
```

12

Creating a Database Instance

- ◆ Once you have access to a server with MySQL installed, need to get a database instance created for you.
 - Usually created by a database administrator
 - Creates a database instance, userid and password.

13

13

Content

1. Database and MySQL Overview
2. Basic SQL commands
- ⇒ 3. Creating a table
4. Inserting data to a table
5. Retrieving data from a table
6. Updating data for a table

14

14

3. Creating a table

- ◆ Once database instance is created need to create your tables.
 - Use SQL CREATE TABLE command

The diagram shows a screenshot of a computer screen displaying a SQL command:

```
CREATE TABLE Products
  (ProductID INT,
   Product_descr TEXT);
```

Annotations explain the code:

- The name of the table: Points to "Products".
- First table column can hold integer data.: Points to "ProductID INT".
- Second table column can hold character data.: Points to "Product_descr TEXT".
- SQL commands are shown in upper case but either upper or lower case can be used.: Points to the entire command.

15

15

MySQL Data Types

- ◆ TEXT
 - hold a large amount of character data
 - Use space inefficiently since it reserves space for up to 65,535 characters.
- ◆ CHAR(N)
 - hold a fixed length string of up to N characters (N must be less than 256).
- ◆ VARCHAR(N)
 - hold a variable length string of up to N characters
 - removes any unused spaces on the end of the entry.

16

16

MySQL Data Types (2)

◆ INT

- hold an integer with a value from about -2 billion to about 2 billion.

◆ INT UNSIGNED

- hold an integer with a value from 0 to about 4 billion.

◆ SMALLINT

- hold an integer with a value from -32,768 to 32,767.

◆ SMALLINT UNSIGNED

- hold an integer with a value from 0 to 65,535.

◆ DECIMAL(N,D)

- a number that supports N total digits, of which D digits are to the right of the decimal point.

17

17

Some additional CREATE TABLE Options

◆ Can specify some additional options in CREATE TABLE:

```
CREATE TABLE Products
  (ProductID INT UNSIGNED NOT NULL
   Product_desc VARCHAR(50),
   Cost INT,
   Weight INT,
   Numb INT);
```

An INT UNSIGNED means that ProductID must be positive values.
ProductID must be specified for each row.
AUTO_INCREMENT PRIMARY KEY,
Up to 50 characters long
Automatically add one to each new ProductID.
Make this the primary key for table.

18

18

Issuing CREATE TABLE From PHP Script Segment

```
1. $connect = mysql_connect($server, $user, $pass);
2. if ( !$connect ) {
3.     die ("Cannot connect to $server using $user");
4. } else {
5.     mysql_select_db('MyDatabaseName');
6.     $SQLcmd = 'CREATE TABLE Products(
                    ProductID INT UNSIGNED NOT NULL
                                AUTO_INCREMENT PRIMARY KEY,
                    Product_desc VARCHAR(50), Cost INT,
                    Weight INT, Numb INT)';
7.     mysql_query($SQLcmd, $connect);
8.     mysql_close($connect);
9. }
```

Connect to MySQL

Issue the SQL query to the database.

Updated API: https://www.w3schools.com/php/php_mysql_connect.asp

19

Full Script

```
1. <html><head><title>Create Table</title></head><body>
2. <?php
3. $server = 'localhost';
4. $user = 'phppgm';
5. $pass = 'mypasswd';
6. $mydb = 'mydatabase';
7. $table_name = 'Products';
8. $connect = mysql_connect($server, $user, $pass);
9. if (!$connect) {
10.     die ("Cannot connect to $server using $user");
11. } else {
12.     $SQLcmd = "CREATE TABLE $table_name (
                    ProductID INT UNSIGNED NOT NULL
                                AUTO_INCREMENT PRIMARY KEY,
                    Product_desc VARCHAR(50),
                    Cost INT, Weight INT, Numb INT)";
```

20

20

10

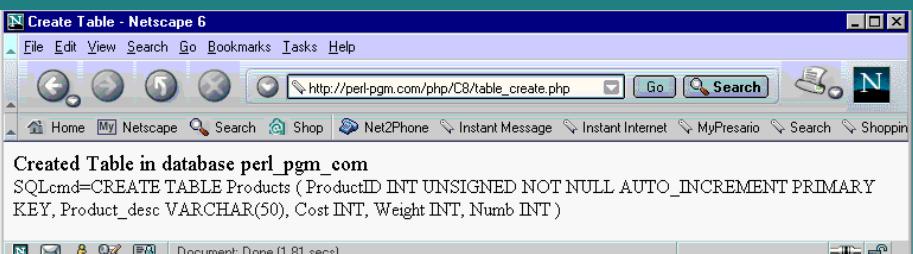
Full Script (2)

```
13. mysql_select_db($mydb);
14. if (mysql_query($SQLcmd, $connect)){
15.     print '<font size="4" color="blue" >Created Table';
16.     print "<i>$table_name</i> in database<i>$mydb</i><br></font>';
17.     print "<br>SQLcmd=$SQLcmd";
18. } else {
19.     die ("Table Create Creation Failed SQLcmd=$SQLcmd");
20. }
21. mysql_close($connect);
22. }
23. ?></body></html>
```

21

21

Script Browser Output



The screenshot shows a Netscape 6 browser window with the title "Create Table - Netscape 6". The address bar displays the URL "http://perl-pgm.com/php/C8/table_create.php". The main content area of the browser shows the output of a MySQL query:

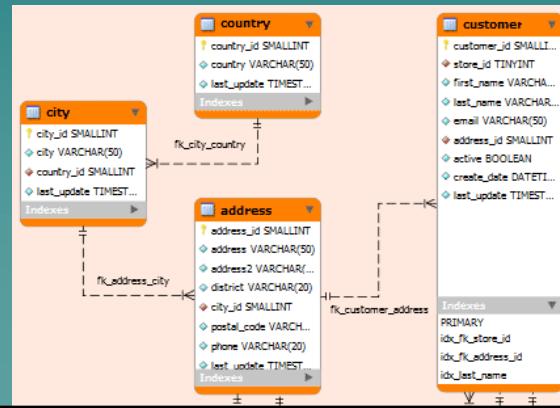
```
Created Table in database perl_pgm_com
SQLcmd=CREATE TABLE Products ( ProductID INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY, Product_desc VARCHAR(50), Cost INT, Weight INT, Numb INT )
```

22

22

MySQL Visual Designer Tools

- ◆ phpMyAdmin (web-app)
- ◆ MySQL Workbench (Win, Linux, Mac)
- ◆ **SQLyog**
- ◆ ...



23

Content

1. Database and MySQL Overview
2. Basic SQL commands
3. Creating a table
4. Inserting data to a table
5. Retrieving data from a table
6. Updating data for a table

24

24

4. Inserting data to a table

- Once database is created will need to insert data
- Use the SQL INSERT command

```
Table Name  
INSERT INTO Products VALUES  
( '0', 'Hammer', 5, 12, 123 );  
Each item goes into a separate table column in a table row.
```

25

25

A Full Example

- Consider an application that allows end-user to enter inventory data:

```
Item Description: <input type="text" size="20"  
                      maxlength="20" name="Item">  
Weight: <input type="text" size="5"  
                      maxlength="20" name="Weight">  
Cost: <input type="text" size="5"  
                      maxlength="20" name="Cost">  
Number Available:<input type="text" size="5"  
                      maxlength="20" name="Quantity">
```

26

26

13

```

1. <html><head><title>Insert Results</title></head><body>
2. <?php
3. $host = 'localhost';
4. $user = 'phppgm';
5. $passwd = 'mypasswd';
6. $database = 'mydatabase';
7. $connect = mysql_connect($host, $user, $passwd);
8. $table_name = 'Products';
9. $query = "INSERT INTO $table_name VALUES
    ('0', '$Item', '$Cost', '$Weight', '$Quantity')";
10. print "The Query is <i>$query</i><br>";
11. mysql_select_db($database);
12. print '<br><font size="4" color="blue">';
13. if (mysql_query($query, $connect)) {
14.     print "Insert into $database was successful!</font>";
15. } else {
16.     print "Insert into $database failed!</font>";
17. } mysql_close ($connect);
18. ?></body></html>

```

Receiving PHP Script

27

27

Script Output

The screenshot shows two stacked browser windows from Netscape 6.

Top Window: Insert Inventory - Netscape 6

- URL: http://perl-pgm.com/php/C8/insert.html
- Form Fields:
 - Item Description: Hammer
 - Weight: 12
 - Cost: 5.50
 - Number Available: 122
- Buttons: Click To Submit, Reset

Bottom Window: Insert Results - Netscape 6

- URL: http://65.108.8.8/php/C8/insert_table.php
- Text Output:


```
The Query is INSERT INTO Products VALUES ('0', 'Hammer', '5.50', '12', '122')
Insert into perl_pgm_com was successful!
```
- Buttons: Back, Forward, Stop, Home, Search, Favorites, Help
- Status Bar: Document: Done (2.91 secs)

28

28

Content

1. Database and MySQL Overview
2. Basic SQL commands
3. Creating a table
4. Inserting data to a table
5. Retrieving data from a table
6. Updating data for a table

29

29

5. Retrieving data from a table

- ◆ Two major ways to retrieve data:
 - Retrieving all elements from a table
 - Searching for specific records in a table
- ◆ To retrieve all data, use following SQL command

SQL SELECT Statement.

SELECT * FROM TableName;

The asterisk ("*") means get all the data

The name of the table to get the data from.

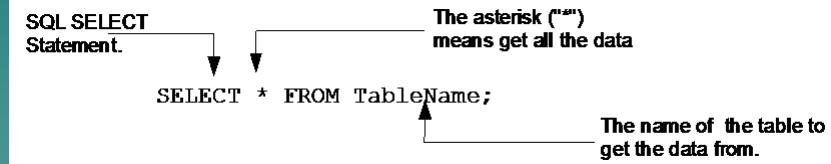
30

30

15

5. Retrieving Data (2)

- ◆ To retrieve all data, use following SQL command



- ◆ For example

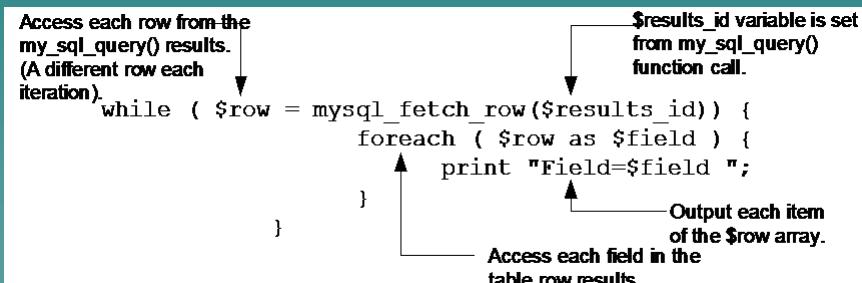
```
1. $connect = mysql_connect('localhost', 'phppgm',
   'mypasswd');
2. $SQLcmd = 'SELECT * FROM Products';
3. mysql_select_db('MyDatabase');
4. $results_id = mysql_query($SQLcmd, $connect);
```

31

31

5.1. Using mysql_fetch_row()

- ◆ Use the `mysql_fetch_row()` function to retrieve data on row at a time



32

32

```
1. <html><head><title>Table Output</title></head><body>
2. <?php
3. $host= 'localhost';
4. $user = 'phppgm';
5. $passwd = 'mypasswd';
6. $database = 'phppgm';
7. $connect = mysql_connect($host, $user, $passwd);
8. $table_name = 'Products';
9. print '<font size="5" color="blue">';
10. print "$table_name Data</font><br>';
11. $query = "SELECT * FROM $table_name";
12. print "The query is <i>$query </i><br>";
13. mysql_select_db($database);
14. $results_id = mysql_query($query, $connect);
15. if ($results_id) {
16.     print '<table border=1>';
17.     print '<th>Num<th>Product<th>Cost<th>Weight<th>Count';
```

A Script Example

33

33

A Script Example (2)

```
18.     while ($row = mysql_fetch_row($results_id)) {
19.         print '<tr>';
20.         foreach ($row as $field) {
21.             print "<td>$field</td> ";
22.         }
23.         print '</tr>';
24.     }
25. } else { die ("Query=$query failed!"); }
26. mysql_close($connect);
27. ?> </table></body></html>
```

34

34

Script Output

The screenshot shows a Netscape 6 browser window with the title "Table Output - Netscape 6". The address bar shows the URL "http://65.108.8.8/php/C8/query.php". The page content is titled "Products Data" and displays the following table:

Num	Product	Cost	Weight	Count
1	Hammer	12	5	123
2	Screw Driver	2	3	144
3	Wrench	1	2	244

35

5.2. Searching For Specific Records

- ◆ Use the SELECT SQL statement with a WHERE clause

– **SELECT * FROM TableName WHERE
(test_expression);**

The asterisk (“*”) means look at all table columns.

Specify the table name to look at.

Specify a test expression to evaluate

36

18

Selected WHERE CLAUSE Test Operators

Operator	SQL Query Example	Meaning
=	<code>SELECT * FROM Products WHERE (Product_desc = 'Hammer');</code>	Retrieve those rows from the Products table that have a Product_desc column with a value equal to Hammer.
>	<code>SELECT * FROM Products WHERE (Cost > '5');</code>	Retrieve those rows from the Products table that have a Cost column with a value greater than 5.
<	<code>SELECT * FROM Products WHERE (Numb < '3');</code>	Retrieve those rows from the Products table that have a Numb column with a value less than 3.
<=	<code>SELECT * FROM Products WHERE (Cost <= '3');</code>	Retrieve those rows from the Products table that have a Cost column with a value less than or equal to 3.
>=	<code>SELECT * FROM Products WHERE (Weight >= '10');</code>	Retrieve those rows from the Products table that have a Weight column with a value greater than or equal to 10.

Slide 8-37

37

Consider the following example ...

- ◆ The following example searches a hardware inventory database for a specific part name entered by the user.
- ◆ The form uses the following key HTML form element definition.
 - `<input type="text" name="Search" size="20">`

38

38

19

PHP Source

```
1. <html><head><title>Search Results</title></head><body>
2. <?php
3. $host= 'localhost';
4. $user = 'phppgm';
5. $passwd = 'mypasswd';
6. $database = 'phppgm';
7. $connect = mysql_connect($host, $user, $passwd);
8. $table_name = 'Products';
9. print '<font size="5" color="blue">';
10. print "$table_name Data</font><br>';
11. $query = "SELECT * FROM $table_name WHERE
                (Product_desc = '$Search')";
12. print "The query is <i>$query</i> <br>";
13. mysql_select_db($database);
14. $results_id = mysql_query($query, $connect);
```

39

39

PHP Source (2)

```
15. if ($results_id) {
16.     print '<br><table border=1>';
17.     print '<th>Num<th>Product<th>Cost<th>Weight <th>Count';
18.     while ($row = mysql_fetch_row($results_id)) {
19.         print '<tr>';
20.         foreach ($row as $field) {
21.             print "<td>$field</td> ";
22.         }
23.         print '</tr>';
24.     }
25. } else { die ("query=$Query Failed");}
26. mysql_close($connect);
27. ?> </body></html>
```

40

40

Would have the following output ...

The screenshot shows two stacked windows from Netscape 6. The top window is titled "Inventory Search - Netscape 6" and contains a form with a search input field containing "Hammer", a "Click To Submit" button, and a "Reset" button. The bottom window is titled "Search Results - Netscape 6" and displays the results of a SQL query. The title bar of this window says "Products Data" and the query is shown as "The query is `SELECT * FROM Products WHERE (Product_desc = 'Hammer')`". A table is displayed with the following data:

Num	Product	Cost	Weight	Count
0	Hammer	5	12	123

41

41

Content

1. Database and MySQL Overview
2. Basic SQL commands
3. Creating a table
4. Inserting data to a table
5. Retrieving data from a table
6. Updating data for a table

42

42

21

6. Updating data for a table

- ◆ Use SQL UPDATE command when needing to update a database record:

```
UPDATE Table_name  
SET col1=chng_express1,col2=chng_express2, ...  
WHERE test_expression
```

Specify the name of the table to update.

Optionally specify a WHERE clause and test expression.

Specify one or more table column to receive the results of an expression. Optionally specify a WHERE

43

43

For Example ...

- ◆ The following searches the Products table for values of Product_desc equal to Hammer.

```
UPDATE Products  
SET Cost=2  
WHERE Product_desc = 'Hammer'
```

44

44

For Example ...

- ◆ The following looks through the Products table for values of Product_desc equal to Hammer.
- ◆ When it finds it, it decrements the Count column value by 1.

```
UPDATE Products  
SET Count=Count-1  
WHERE 'Product_desc=Hammer'
```

45

45

A Full Example ...

- ◆ Consider the following example
 - Displays current inventory
 - Asks end-user to decrement value for 1 item
 - Uses the following HTML
- ```
Hammer: <input type="radio" name="Product"
value="Hammer">

Screwdriver: <input type="radio"
name="Product" value="Screwdriver">

Wrench: <input type="radio" name="Product"
value="Wrench">
```

46

46

## Full Example

```
1. <html><head><title>Product Update
Results</title></head><body>
2. <?php
3. $host= 'localhost';
4. $user = 'phppgm';
5. $passwd = 'mypasswd';
6. $database = 'phppgm';
7. $connect = mysql_connect($host, $user, $passwd);
8. $table_name = 'Products';
9. print ''
10. print "Update Results for Table
$table_name
\n";
11. $query = "UPDATE $table_name
SET Numb = Numb-1
WHERE (Product_desc = '$Product')";
12. print "The query is <i> $query </i>

\n";
13. mysql_select_db($database);
```

47

47

## A Full Example (2)

```
14. $results_id = mysql_query($query, $connect);
15. if ($results_id){
16. Show_all($connect, $database,$table_name);
17. } else {
18. print "Update=$query failed";
19. }
20. mysql_close($connect);
```

48

48

## A Full Example (3)

```
21. function Show_all($connect, $database, $table_name) {
22. $query = "SELECT * from $table_name";
23. $results_id = mysql_query($query, $connect);
24. print '<table border=1><th> Num </th>
 <th>Product</th><th>Cost</th>
 <th>Weight</th><th>Count</th>';
26. while ($row = mysql_fetch_row($results_id)) {
27. print '<tr>';
28. foreach ($row as $field){
29. print "<td>$field</td> ";
30. }
31. print '</tr>';
32. }
33. }
34. ?> </body></html>
```

49

49

Would output the following:

The query is `SELECT * from Products`

| Num | Product     | Cost | Weight | Count |
|-----|-------------|------|--------|-------|
| 2   | Wrench      | 2    | 1      | 150   |
| 1   | Screwdriver | 3    | 2      | 144   |
| 0   | Hammer      | 5    | 12     | 123   |

The query is `UPDATE Products SET Numb = Numb-1 WHERE (Product_desc = 'Hammer')`

| Num | Product     | Cost | Weight | Count |
|-----|-------------|------|--------|-------|
| 2   | Wrench      | 2    | 1      | 150   |
| 1   | Screwdriver | 3    | 2      | 144   |
| 0   | Hammer      | 5    | 12     | 122   |

Slide 8-50

50

# Question?



51

51

26