Advanced PHP

Database

- Collection of data
- Advantages
  - Maintainability and scalability
  - Portability
  - Avoiding awkward programming
  - Searching

Structured Query Language (SQL)

SELECT field1, field2, field3 FROM table

SELECT * FROM mytable

SELECT * FROM mytable WHERE ID < 100;

INSERT INTO table (col1, col2, col3) VALUES(val1, val2, val3);

INSERT INTO customer(birthmonth, birthflower, birthstone) SELECT * FROM birthday_info WHERE birthmonth = $birthmonth;

UPDATE table SET field1='val1', field2='val2', field3='val3' WHERE condition;

DELETE datapoint FROM table WHERE condition;

Create a Connection to a MySQL Database

Syntax

```
mysql_connect(servername,username,password);
```

Parameter	Description
servername	Optional. Specifies the server to connect to. Default value is "localhost:3306"
username	Optional. Specifies the username to log in with. Default value is the name of the user that owns the server process
password	Optional. Specifies the password to log in with. Default is ""

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }

// some code
?>
```

Closing a Connection

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }
// some code
mysql_close($con);
?>
```

Create a Database

The CREATE DATABASE statement is used to create a database in MySQL.

Syntax

```
CREATE DATABASE database_name
```

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }

if (mysql_query("CREATE DATABASE my_db",$con))
  {
  echo "Database created";
  }
else
  {
  echo "Error creating database: " . mysql_error();
  }
mysql_close($con);
?>
```

Create a Table

The CREATE TABLE statement is used to create a table in MySQL.

Syntax

```
CREATE TABLE table_name
(
column_name1 data_type,
column_name2 data_type,
column_name3 data_type,
....
)
```

```php
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }
// Create database
if (mysql_query("CREATE DATABASE my_db",$con))
  {
  echo "Database created";
  }
else
  {
  echo "Error creating database: " . mysql_error();
  }
// Create table
mysql_select_db("my_db", $con);
$sql = "CREATE TABLE Persons
(
FirstName varchar(15),
LastName varchar(15),
Age int
)";
// Execute query
mysql_query($sql,$con);
mysql_close($con);
?>
```

Primary Keys and Auto Increment Fields

```
$sql = "CREATE TABLE Persons
(
personID int NOT NULL AUTO_INCREMENT,
PRIMARY KEY(personID),
FirstName varchar(15),
LastName varchar(15),
Age int
)";

mysql_query($sql,$con);
```

```
INSERT INTO table_name
VALUES (value1, value2, value3,....)
```

The second form specifies both the column names and the values to be inserted:

```
INSERT INTO table_name (column1, column2, column3,....)
VALUES (value1, value2, value3,...)
```

```php
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }
mysql_select_db("my_db", $con);
mysql_query("INSERT INTO Persons (FirstName, LastName, Age)
VALUES ('Peter', 'Griffin', '35')");
mysql_query("INSERT INTO Persons (FirstName, LastName, Age)
VALUES ('Glenn', 'Quagmire', '33')");
mysql_close($con);
?>
```

Insert Data From a Form Into a Database

```
<html>
<body>

<form action="insert.php" method="post">
Firstname: <input type="text" name="firstname" />
Lastname: <input type="text" name="lastname" />
Age: <input type="text" name="age" />
<input type="submit" />
</form>

</body>
</html>
```

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }
mysql_select_db("my_db", $con);

$sql="INSERT INTO Persons (FirstName, LastName, Age)
VALUES
('$_POST[firstname]','$_POST[lastname]','$_POST[age]')";

if (!mysql_query($sql,$con))
  {
  die('Error: ' . mysql_error());
  }
echo "1 record added";

mysql_close($con)
```

Select Data From a Database Table

```php
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }
mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons");

while($row = mysql_fetch_array($result))
  {
  echo $row['FirstName'] . " " . $row['LastName'];
  echo "<br />";
  }
mysql_close($con);
?>
```

Display the Result in an HTML Table

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons");

echo "<table border='1'>
<tr>
<th>Firstname</th>
<th>Lastname</th>
</tr>";

while($row = mysql_fetch_array($result))
  {
  echo "<tr>";
  echo "<td>" . $row['FirstName'] . "</td>";
  echo "<td>" . $row['LastName'] . "</td>";
  echo "</tr>";
  }
echo "</table>";

mysql_close($con);
?>
```

The WHERE clause

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons
WHERE FirstName='Peter'");

while($row = mysql_fetch_array($result))
  {
  echo $row['FirstName'] . " " . $row['LastName'];
  echo "<br />";
  }
?>
```

The ORDER BY Keyword

```php
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons ORDER BY age");

while($row = mysql_fetch_array($result))
  {
  echo $row['FirstName'];
  echo " " . $row['LastName'];
  echo " " . $row['Age'];
  echo "<br />";
  }

mysql_close($con);
?>
```

Update Data In a Database

The UPDATE statement is used to update existing records in a table.

Syntax

```
UPDATE table_name
SET column1=value, column2=value2,....
WHERE some_column=some_value
```

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }

mysql_select_db("my_db", $con);

mysql_query("UPDATE Persons SET Age = '36'
WHERE FirstName = 'Peter' AND LastName = 'Griffin'");

mysql_close($con);
?>
```

Delete Data In a Database

The DELETE FROM statement is used to delete records from a database table.

Syntax

```
DELETE FROM table_name
WHERE some_column = some_value
```

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }
mysql_select_db("my_db", $con);
mysql_query("DELETE FROM Persons WHERE LastName='Griffin'");
mysql_close($con);
?>
```

The fetching functions are as follows:

- `mysql_fetch_row`: Returns row as an enumerated array
- `mysql_fetch_object`: Returns row as an object
- `mysql_fetch_array`: Returns row as an associative array
- `mysql_result`: Returns one cell of data

```
$query = "SELECT ID, LastName, FirstName
          FROM users WHERE Status = 1";
$result = mysql_query($query);
while ($name_row = mysql_fetch_row($result)) {
  print("{$name_row[0]} {$name_row[1]} {$name_row[2]}<BR>\n");
}
```

```
$query = "SELECT ID, LastName, FirstName
          FROM users WHERE Status = 1";
$result = mysql_query($query);
while ($row = mysql_fetch_object($result)) {
  echo "{$row->ID}, {$row->LastName}, {$row->FirstName}<BR>\n";
}
```

```
$query = "SELECT ID, LastName, FirstName
          FROM users WHERE Status = 1";
$result = mysql_query($query);
while ($row = mysql_fetch_array($result)) {
  echo "{$row['ID']}, {$row['LastName']}, {$row['FirstName']}<BR>\n";
}
```

```
$query = "SELECT count(*) FROM personal_info";
$db_result = mysql_query($query);
$datapoint = mysql_result($db_result, 0, 0);
```

The `mysql_result` function takes three arguments: *result identifier*, *row identifier*, and (optionally) *field*. Field can take the value of the field offset as above or its name as in an associative array ("Surname") or its MySQL field-dot-table name ("`personal_info.Surname`").

MySQL Functions

PHP-MySQL Functions

Function Name	Usage
`mysql_affected_rows([link_id])`	Use after a nonzero `INSERT`, `UPDATE`, or `DELETE` query to check number of rows
`mysql_change_user(user, password[, database] [, link_id])`	Changes MySQL user on an open link.
`mysql_close([link_id])`	Closes the identified link (usually unnecessary).
`mysql_connect([host][:port][:socket][, username][, password])`	Opens a link on the specified host, port, socket; as specified user with password. All arguments are optional.
`mysql_create_db(db_name[, link_id])`	Creates a new MySQL database on the host associated with the nearest open link.
`mysql_data_seek(result_id, row_num)`	Moves internal row pointer to specified row number. Use a fetching function to return data from that row.
`mysql_drop_db(db_name[, link_id])`	Drops specified MySQL database.
`mysql_errno([link_id])`	Returns ID of error.
`mysql_error([link_id])`	Returns text error message.
`mysql_fetch_array(result_id[, result_type])`	Fetches result set as associative array. Result type can be `MYSQL_ASSOC`, `MYSQL_NUM`, or `MYSQL_BOTH` (default).

```
<?php
echo("<TABLE>\n<TR><TH>Titles</TH></TR>\n<TR>");
$query = "SELECT title, publisher FROM books";
$result = mysql_query($query);
while ($book_row = mysql_fetch_array($result)) {
  echo("<TD>$book_row[0]</TD>\n");
}
echo("</TR></TABLE><BR>\n");
echo("<TABLE>\n<TR><TH>Publishers</TH></TR>\n<TR>");

  mysql_data_seek($result, 0);
  while ($book_row = mysql_fetch_array($result)) {
    echo("<TD>{$book_row[1]}</TD>\n");
  }
  echo("</TR></TABLE><BR>\n");
  ?>
```

```
<?php
if (isset($_POST['submit']) && $_POST['submit'] =='Submit')
{
mysql_connect("localhost", "root", "") or die("Failure to
communicate with database");
mysql_select_db("my_db");
$firstname = mysql_real_escape_string($_POST['firstname']);
$lastname= mysql_real_escape_string($_POST['lastname']);
$age = mysql_real_escape_string($_POST['age']);
$query ="INSERT INTO Faculty (FirstName,LastName,Age)
VALUES ('$firstname','$lastname',$age)";
$result = mysql_query($query);
if (mysql_affected_rows() == 1) {
echo  '<P>Your information has been recorded.</P>';
} else {
echo "<P>Something went wrong with your signup
attempt.</P>". mysql_error();
}
}
```

```
$thisfile = "p8q1.php";
$message= <<< EOMSG
<FORM METHOD="post" ACTION="$thisfile">
Firstname<INPUT TYPE=”text” SIZE=25 NAME="firstname"><BR>
Lastname<INPUT TYPE=”text” SIZE=25 NAME="lastname"><BR><
Age<INPUT TYPE=”text” SIZE=25 NAME="age"><BR><BR>
<INPUT TYPE="submit" NAME="submit" VALUE="Submit"></FORM
EOMSG;
//}
?>
<HTML>
<BODY>
<?php echo $message; ?>
</BODY>
</HTML>
```

What is a Cookie?

A cookie is often used to identify a user.
A cookie is a small file that the server embeds on the user's computer.
Each time the same computer requests a page with a browser, it will send the cookie too.

How to Create a Cookie?

The setcookie() function is used to set a cookie.
Note: The setcookie() function must appear BEFORE the <html> tag.

Syntax

```
setcookie(name, value, expire, path, domain);
```

Example 1

```
<?php
setcookie("user", "Alex Porter", time()+3600);
?>

<html>
.....
```

Example 2

```
<?php
$expire=time()+60*60*24*30;
setcookie("user", "Alex Porter", $expire);
?>

<html>
.....
```

How to Retrieve a Cookie Value?

```
<?php
// Print a cookie
echo $_COOKIE["user"];

// A way to view all cookies
print_r($_COOKIE);
?>
```

In the following example we use the isset() function to find out if a cookie has been set:

```
<html>
<body>

<?php
if (isset($_COOKIE["user"]))
  echo "Welcome " . $_COOKIE["user"] . "!<br />";
else
  echo "Welcome guest!<br />";
?>
</body>
</html>
```

How to Delete a Cookie?

When deleting a cookie you should assure that the expiration date is in the past.

Delete example:

```
<?php
// set the expiration date to one hour ago
setcookie("user", "", time()-3600);
?>
```

What if a Browser Does NOT Support Cookies?

```
<html>
<body>
<form action="welcome.php" method="post">
Name: <input type="text" name="name" />
Age: <input type="text" name="age" />
<input type="submit" />
</form>
</body>
</html>
```

Retrieve the values in the "welcome.php" file like this:

```
<html>
<body>

Welcome <?php echo $_POST["name"]; ?>.<br />
You are <?php echo $_POST["age"]; ?> years old.

</body>
</html>
```

Starting a PHP Session

Note: The session_start() function must appear BEFORE the <html> tag:

```
<?php session_start(); ?>
<html>
<body>
</body>
</html>
```

Storing a Session Variable

```
<?php
session_start();
// store session data
$_SESSION['views']=1;
?>
<html>
<body>
<?php
//retrieve session data
echo "Pageviews=". $_SESSION['views'];
?>

</body>
</html>
```

Output:

```
Pageviews=1
```

```php
<?php
session_start();

if(isset($_SESSION['views']))
$_SESSION['views']=$_SESSION['views']+1;
else
$_SESSION['views']=1;
echo "Views=". $_SESSION['views'];
?>
```

Destroying a Session

The unset() function is used to free the specified session variable:

```
<?php
unset($_SESSION['views']);
?>
```

You can also completely destroy the session by calling the session_destroy() function:

```
<?php
session_destroy();
?>
```

The PHP mail() Function

The PHP mail() function is used to send emails from inside a script.

Syntax

```
mail(to,subject,message,headers,parameters)
```

Parameter	Description
to	Required. Specifies the receiver / receivers of the email
subject	Required. Specifies the subject of the email. **Note:** This parameter cannot contain any newline characters
message	Required. Defines the message to be sent. Each line should be separated with a LF (\n). Lines should not exceed 70 characters
headers	Optional. Specifies additional headers, like From, Cc, and Bcc. The additional headers should be separated with a CRLF (\r\n)
parameters	Optional. Specifies an additional parameter to the sendmail program

PHP Simple E-Mail

```php
<?php
$to = "someone@example.com";
$subject = "Test mail";
$message = "Hello! This is a simple email message.";
$from = "someonelse@example.com";
$headers = "From: $from";
mail($to,$subject,$message,$headers);
echo "Mail Sent.";
?>
```

PHP Mail Form

```
<html>
<body>
<?php
if (isset($_REQUEST['email']))
//if "email" is filled out, send email
  {
  //send email
  $email = $_REQUEST['email'] ;
  $subject = $_REQUEST['subject'] ;
  $message = $_REQUEST['message'] ;
  mail( "someone@example.com", "Subject: $subject",
  $message, "From: $email" );
  echo "Thank you for using our mail form";
  }
else
//if "email" is not filled out, display the form
  {
  echo "<form method='post' action='mailform.php'>
  Email: <input name='email' type='text' /><br />
  Subject: <input name='subject' type='text' /><br />
  Message:<br />
  <textarea name='message' rows='15' cols='40'>
  </textarea><br />
  <input type='submit' />
  </form>";
  }
?>
</body>
</html>
```

```
<?php
session_start();
?>
<HTML><HEAD><TITLE>Greetings</TITLE></HEAD>
<BODY>
<H2>Welcome to the Center for Content-free Hospitality</H2>
<?php
if (!IsSet($_SESSION['visit_count'])) {
echo "Hello, you must have just arrived.
Welcome!<BR>";
$_SESSION['visit_count'] = 1;
}
else {
$visit_count = $_SESSION['visit_count'] + 1;
echo "Back again are ya? That makes $visit_count times now ".
"(not that anyone's counting)<BR>";
$_SESSION['visit_count'] = $visit_count;
}
```

```
$self_url = $_SERVER['PHP_SELF'];
$session_id = SID;
if (IsSet($session_id) &&
$session_id) {
$href = "$self_url?$session_id";
}
else {
$href = $self_url;
}
echo "<BR><A HREF=\"$href\">Visit us again</A> sometime";
?>
</BODY></HTML>
```

Eg in Notes