

A large green shape on the left side of the slide, resembling a stylized 'C' or a bracket, with a white semi-circular cutout in the center.

PHP Basics

Programming for the Internet
South Burlington High School

A solid dark blue horizontal bar with rounded ends, positioned below the text.

Comments in PHP

// This is a single-line comment.

`Phpinfo();` // Comment about this line of code.

/* This is a multiple-line comment. It is called a multiple-line comment because it can span multiple lines of code before ending with a

*/

/* Sometimes programmers add asterisks to
* the beginning of each line of a multi-line
* comment for readability */

Variables start with \$

```
$some_text = "This is a text string.";
```

It has been assigned to the variable `$some_text`.
Variables are case-sensitive.

Numeric strings do not require quotes.

```
$user_id = 987;
```

Outputting variables

```
<?
```

```
    $some_text = "This is a text string.";
```

```
    print($some_text);
```

```
?>
```

Formatting text

You can include HTML, but you need to escape quotes with a backslash.

```
print("<font face=\"Arial\" color=\"#FF0000\">  
Hello and welcome to my website.</font>");
```

Adding HTML

You can also end the PHP, insert some HTML and start the PHP up again using...

... ?> Insert some HTML code here <? ...

You can do this anywhere, even in a PHP statement (although that would likely be bad programming style).

IF - ELSE statements

```
if (something == something else)
{
    then do this statement
} else {
    do this statement instead
}
```

Note: Sometimes “IF” statements are called “IF-THEN” or “IF-ELSE” statements.

Example

```
if ($username == "webmaster") {  
    echo "Please enter your password below";  
} else {  
    echo "We are sorry but you are not a  
    recognized user";  
}
```

Multiple conditions

```
if ($name == "" || $email == "" || $password  
    == "") {  
    echo "Please fill in all the fields";  
}
```

|| is the OR operator.

Loops

```
$times = 5;  
$x = 0;  
while ($x < $times) {  
  echo "Hello World";  
  ++$x;  
}
```

\$x is the loop counter variable.

Arrays: Creating and Accessing

```
$names[0] = 'John';  
$names[1] = 'Paul';  
$names[2] = 'George';  
$names[3] = 'Ringo';  
$names[4] = 'Pete';
```

This code creates an array named “\$names” and loads it with 5 elements, \$names[0] through \$names[4].

The last line prints the third element in the array (George) to the browser.

```
echo "The third name is $names[2]";
```

Accessing array elements using a loop

Name 1 is John
Name 2 is Paul

Name 3 is George
Name 4 is Ringo

Name 5 is Pete

```
$number = 5;  
$x = 0;  
while ($x < $number) {  
  $namenumber = $x + 1;  
  echo "Name $namenumber is  
  $names[$x]<br>";  
  ++$x;  
}
```

PHP & MySQL

- MySQL is an open-source database management system (DBMS).
- MySQL is often used with PHP.
- The MySQL website is mysql.org.
- We will use MySQL and PHP to create a contacts database, a form to enter data and a page to read the data.

Managing a MySQL Database

- Can manage with PHP, but it's a bit easier to use phpMyAdmin.
 - Start Uniform Server
 - From apanel (<http://localhost/apanel/>) select phpMyAdmin under Tools.

Create MySQL Database

- Create database
 - Name: data1
 - Collation: utf-8_general_c1
- Create database table
 - Name: contacts
 - Number of fields: 8

Create MySQL Database (cont.)

Name	Type	Length	Description
id	INT	6	A unique identifier for each record
first	VARCHAR	15	The person's first name
last	VARCHAR	15	The person's last name
phone	VARCHAR	20	The person's phone number
mobile	VARCHAR	20	The person's mobile number
fax	VARCHAR	20	The person's fax number
email	VARCHAR	30	The person's e-mail address
web	VARCHAR	30	The person's web address

Collation of table is utf-8_general_ci.

Set id field as primary, auto_increment.

Enter Data Into Table

- See [dataenter.php](#)

Display Table Contents

- See [dataread.php](#)

Create a PHP Form to Enter Data

- See [dataform.php](#)