

connection to mysql database

Connect to the database
`$dbc = mysqli_connect(HOST, USER, PW, DB);`

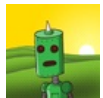
Query the database
`$result = mysqli_query($dbc, $query);`

Close the connection
`mysqli_close($dbc);`

PHP functions

<code>isset()</code>	test for variable exists
<code>empty()</code>	test for empty variable
<code>mail(\$to, \$subject, \$msg, 'From: ' . \$email)</code>	mail function
<code>mysqli_fetch_array(\$result)</code>	fetch each row of a query (in \$result)
<code>header()</code>	send a header from the server
<code>is_numeric()</code>	test to see if a value is number
<code>exit()</code>	causes script to stop immediately
<code>trim(\$string)</code>	trims leading and trailing spaces
<code>mysqli_real_escape_string(\$string)</code>	escapes special characters
<code>str_replace('a', 'b', \$string)</code>	replace a with b in a string
<code>explode(' ', \$string)</code>	make string into array
<code>implode(' ', \$string)</code>	make array into string
<code>substr(\$string, start, len)</code>	grabs a substring
<code>preg_match('regex', \$string)</code>	matches regular expressions
<code>preg_replace('regex', \$replace, \$string)</code>	replaces characters in a string by regex

Cheatographer



guslong
cheatography.com/guslong/

IF syntax

```
if (condition) {
... }
elseif (condition) {
... }
else {
... }
```

Loops

FOR loop for (initialize; condition; update) { ... }

WHILE loop while (condition) { ... }

FOREACH loop foreach (\$array as \$value) { ... }

DO WHILE do { ... ;} while (condition)

FOR (loop until a condition is met)
WHILE (loop through query results)
FOREACH (loop through an array)

CONTINUE

```
for ($i = 0; $i < 5; ++$i) {
if ($i == 2)
continue;
print "$i , ";
}
```

produces the following output:
0 , 1 , 3 , 4

SWITCH syntax

```
SWITCH ($s) {
case 1:
...
break;
case 2:
...
break;
default:
...
}
```

can be used with numbers, or strings

Cheat Sheet

This cheat sheet was published on 20th July, 2012 and was last updated on 13th August, 2012.

Regular Expressions

<code>^</code>	start of string
<code>\d</code>	number from 0 to 9
<code>\s</code>	whitespace
<code>.</code>	any letter or digit or whitespace
<code>\w</code>	any alphanumeric [0-9A-Za-z]
<code>\$</code>	end of string
<code>()</code>	group
<code>[]</code>	character class
<code>{x} {x,} {x,y}</code>	x of x or more of x to y of
<code> </code>	or
<code>*</code>	none or more
<code>?</code>	none or one
<code>+</code>	one or more
<code>\</code>	escape

Arrays

Create	<code>\$myArray = array();</code>
Push into	<code>\$myArray[] = "Something";</code>
Push to associative	<code>\$myArray['key'] = "Value";</code>
Create numeric	<code>\$myArray = array('value', 'value2');</code>
Create associative	<code>\$a = array('key'=>'val');</code>
Print from numeric	<code>echo \$myArray[0];</code>
Print from associative	<code>echo \$myArray['key'];</code>
Associative arrays	Keys are strings
Numeric arrays	Keys are numbers: 0,1,2,3,4

Sponsor

FeedbackFair, increase your conversion rate today!
Try it free!
<http://www.FeedbackFair.com>