PHP Reference Sheet
Advanced

Advanced Operators

```
Instanceof - <variable> instanceof <class>
@ - @<function>() // Suppress generated errors/notices
```

Database MySQL

```
mysql_connect([<server>],[<user>],[<password>]);
mysql_create_db(<database>,[<link_identifier>]);
mysql_select_db(<database>,[<link_identifier>]);
mysql_drop_db(<database>,[<link_identifier>]);

mysql_list_dbs([<link_identifier>]);
mysql_list_tables(<database>,[<link_identifier>]);
mysql_list_fields(<database>,<table>,[<link_identifier>]);

mysql_query(<query>,[<link_identifier>]);

mysql_fetch_array(<query_result>,<result_type>);
mysql_fetch_assoc(<query_result>);
mysql_fetch_object(<query_result>,[<classname>],[<array params>]);
mysql_fetch_row(<query_result>);

// Example using fetch_array
while ($result_row = mysql_fetch_array($result,MYSQL_ASSOC)) {
    echo "Field 1 is: {$result_row['field1']}";
}

// Example using fetch_object
while ($result_obj = mysql_fetch_object($result)) {
    echo "Field 1 is: " . $result_obj->field1;
}

mysql_affected_rows([<link_identifier>]);
mysql_num_rows(<query_result>);
mysql_num_fields(<query_result>);

mysql_real_escape_string(<string>,[<link_identifier>]);
mysql_close([<link_identifier>]);
```

File Reading / Writing

```
fopen(<filename>,<mode>,[<bool use_include_path>],[<context>]);
fread(<file_handle>,<length>);
fwrite(<file_handle>,<string>,[<length>]);
fseek(<file_handle>,<offset>,[<whence>]);
fclose(<file_handle>);
is_readable(<filename>);
is_writeable(<filename>);

// Example of writing to file

$file = fopen("hello.txt","w");

if ($file) {
   fwrite($file,"This is a line of text");
   fclose($file);
}

// Example of reading from file

$fileread = fopen("hello.txt","r");

if ($fileread) {
   $text = fread($fileread,100);
   fclose($fileread);

   echo "Read in: $text";
}
```

File/directory Handling

```
file_exists(<filename>);
is_file(<filename>);
dirname(<path>);
is_dir(<filename>);
mkdir(<pathname>,[<mode>],[<recursive>],[<context>]);
rmdir(<dirname>,[<context>]);
rename(<string oldname>,<string newname>,[<context>]);
copy(<string source>,<string destination>,[<context>]);
chmod(<filename>,<mode>);
unlink(<filename,[<context>]);

// Example of using is_file, copy, and unlink to rename file.

if (is_file("hello.txt")) {
   copy("hello.txt","goodbye.txt");
   unlink("hello.txt");
}

// Example of file_exists and rename

if (file_exists("goodbye.txt")) {
   rename("goodbye.txt","helloagain.txt");
}
```

Function Handling

```
create_function(<string args>,<string code>);
call_user_func(<function_name> [,<parameter> [, <parameter> [, ... ]]]);
call_user_func(array(<class_name>,<method_name>)[,<parameter> [, <parameter> [,
... ]]]);
call_user_func_array(<functionname>,<array of parameters>);
function_exists(<functionname>);

// Example of creating anonymous function (returns the value 6)

$myNewFunction = create_function('$param1,$param2','return $param1 * $param2;');
echo "Result is: " . $myNewFunction(2,3);

// Example of calling class method using parameter

class MyClass {
  function showMessage($test = "") {
    echo "Welcome to DIC! $test";
  }
}

call_user_func(array('MyClass','showMessage'),"Enjoy your stay.");
```

Object Functions

```
$variable = get_class([<object>]);
$arrayofclassnames = get_declared_classes();
$variable = get_parent_class([<object>]);
$arrayofproperties = get_object_vars([<object>]);

is_a(<object>,<classname>);
is_subclass_of(<object>,<classname>);

class_exists(<classname>);
method_exists(<object>,<methodname>);
```

Constants

```
// Define a constant
define(<name>,<value>,[<case_sensitivity>]);
```

Magic Constants

```
__LINE__ - Current line number of the file.
__FILE__ - The full path and filename of the file. If used inside an include, the name
of the included file is returned.
__FUNCTION__ - The function name.
__CLASS__ - The class name.
__METHOD__ - The class method name. (PHP 5.0+)
```

Download More Reference Sheets & Get Programming Help @
http://www.DreamInCode.net

Edited By: hotsnoj, Martyr2
Published: December 20, 2007