PHP Fundamentals Cheat Sheet

Basic Syntax

Files start with	<?php
Single line comment	// a comment
Multi-line comment	/ comment /
End of instruction	;
Include code from another file	require_once('some_file.php');

Classes and Objects

```
class SomeClass {
    private $property;
    public $anotherProperty;
    protected $yetAnotherProperty = null;
    public function __construct($arg=null)
{
    $this->property = $arg;
}
public function someMethod()
{
    echo "Hi";
}
public function getProperty()
{
    return $this->property;
}
public function setProperty( $p )
{
    $this->property = $p;
}
}
$myObject = new SomeClass( "123" );
echo $myObject->getProperty(); // 123
$myObject->property; // ERROR:private
```

Variables

$variableName;
$variableName = "Some String";
$variableName = 'Some String';
$variableName = strtoupper('text');
$variableName = 5;
$variable = "Some {$otherVariable} info";
echo $variableName; // output
$newVar = $var1 . $var2; // concatenation

Functions

```
function multiply($arg1, $arg2)
{
    return $arg * $arg2;
}
$param = 4;
$param2 = 8;
$answer = multiply($param, $param2);
```

Control Structure: IF

```
// if something is true do something else
if( $something == true ) {
    doSomethingElse();
} elseif( $something == false ) {
    // however, if something is false, do
something
    doSomething();
} else {
    // otherwise, lets do nothing
    doNothing();
}
```

Control Structure: Loops

```
foreach( $myArray as $key => $value ) {
    echo "My array has the value {$value} stored
against the key {$key}<br />";
}
while( someCondition == true ) {
    echo 'hello';
}
```

Numerical Operations

Addition	$variable = $variable + 5;
Subtraction	$variable = $variable - 5;
Multiplication	$variable = $variable * 5;
Division	$variable = $variable / 5;

Arrays

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

Control Structure: Switch

```
switch( $someVariable ) {
    case 1:
        echo "Some variable equals 1";
        break;
    case "cheese"
        echo "Some variable equals cheese";
        break;
    default:
        echo "No idea";
        break;
}
```

PHP MySQLi Procedural Cheat Sheet

mysqli_num_rows()

```
Return the number of rows in a result set
Syntax:
mysqli_num_rows(result);
Example:
  $rowcount=mysqli_num_rows($result);
```

mysqli_fetch_assoc()

```
Fetch a result row as an associative array.
```

Opening a connection to MySQL

```
Syntax:
mysqli_connect(host,username,password,dbname,port,sock
et);
Returns an object representing the connection to the
MySQL server.
Example:
$conn =
mysqli_connect("localhost","my_user","my_password","my
_db");
```

mysqli_error()

```
Return the last error description for the most recent
function call, if any.
Syntax:
mysqli_error(connection);
Example;
echo("Error description: " . mysqli_error($conn));
```

Creating a Databasse

```
// Create connection
$conn = mysqli_connect($servername, $username,
$password);
// Check connection
if (!$conn) {
    die("Connection failed: " .
mysqli_connect_error());
}
// Create database
$sql = "CREATE DATABASE myDB";
if (mysqli_query($conn, $sql)) {
    echo "Database created successfully";
```

Creating a Databasse (cont)

```
} else {
    echo "Error creating database: " .
mysqli_error($conn);
}
mysqli_close($conn);
```

Selecting Data

```
// Create connection
$conn = mysqli_connect($servername, $username,
$password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " .
mysqli_connect_error());
}
$sql = "SELECT id, firstname, lastname FROM
MyGuests";
$result = mysqli_query($conn, $sql);
if (mysqli_num_rows($result) > 0) {
    // output data of each row
    while($row = mysqli_fetch_assoc($result)) {
        echo "id: " . $row["id"]. " - Name: " .
$row["firstname"]. " " . $row["lastname"]. "<br>";
    }
} else {
    echo "0 results";
}
mysqli_close($conn);
```

Closing a connection of MySQL

```
Syntax:
mysqli_close(connection);
Example:
mysqli_close($conn);
```

PHP MySQLi Procedural Cheat Sheet

mysqli_connect_error()

Return an error description from the last connection error, if any.
Syntax:

```
mysqli_connect_error();
```

Example:

```
die("Connection error: " . mysqli_connect_error());
```

Create a Table

```php
// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
// sql to create table
$sql = "CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg_date TIMESTAMP
)";
if (mysqli_query($conn, $sql)) {
    echo "Table MyGuests created successfully";
} else {
    echo "Error creating table: " . mysqli_error($conn);
}
mysqli_close($conn);
```

Selecting a Database

Syntax:

```
mysqli_select_db(connection,dbname);
```

Example:

```
mysqli_select_db($conn,"test");
```

mysqli_query()

Perform queries against the database.
For successful SELECT, SHOW, DESCRIBE, or EXPLAIN queries it will return a mysqli_result object. For other successful queries it will return TRUE. FALSE on failure.
Syntax:

```
mysqli_query(connection,query,resultmode);
```

Examples:

```
mysqli_query($con,"SELECT * FROM Persons");
mysqli_query($con,"INSERT INTO Persons
(FirstName,LastName,Age) VALUES
('Glenn','Quagmire',33)");
```

Inserting into Table

```php
// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";
if (mysqli_query($conn, $sql)) {
    echo "New record created successfully";
} else {
    echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}
mysqli_close($conn);
```