PHP Notes

Purpose of PHP:

* Evaluate data sent from browser
* Build web content to serve browser
* Send and receive cookies->(little packets of data that browser uses to remember things,ex. If you’re logged into a website)

-PHP code is written in between <?php and ?>

Javascript vs PHP:

-Javascript only runs on browser(client) so it only knows whats going on in browser plus info it gets from websites it connected to

-PHP runs on same computer as the website you’re visiting known as the server

**Print a line through PHP:**

<?php

echo ”Here is the line that will show up”;

?>

Can write PHP into HTML like so:

<body> <body>

<p> <p>

<?php <h1>

echo “PHP Line”; **OR** <?php

?> echo ”Test”;

</p> ?>

</body> </h1>

</p>

</body>

Concatenating strings:

echo “Hello,”. “ “. “world”. “!”; (use dot(.) operator for concatenation)

Multiplication:

echo 17 \* 23; (2091 outputted to screen)

Creating Variables:

<?php

$myName = “Mayez”;

?>

Comments:

Use two forward slashes // (ex. //This is a comment)

Print variables:

$myAge = 20;

echo $myAge;

**NOTE: IF STATEMENTS EXACTLY THE SAME AS C EXCEPT INSTEAD OF IF/ELSE IF/ELSE THEY USE IF/ELIF/ELSE**

Switch Statements example:

<?php

switch (2) {

case 0:

echo 'The value is 0';

break;

case 1:

echo 'The value is 1';

break;

case 2:

echo 'The value is 2';

break;

default:

echo "The value isn't 0, 1 or 2";

}

?>

**OR**

<?php

$fruit = "Apple";

switch ($fruit) {

case 'Apple':

echo "Yummy.";

break;

}

?>

Arrays in PHP:

<?php

$array = array("Egg", "Tomato", "Beans");

?>

Remove elements from array:

<?php

$array = array("red", "blue", "green");

unset($array[2]);

?>

Remove entire array:

<?php

unset($array);

?>

For Loop in PHP:

<?php

for ($leap = 2004; $leap < 2050; $leap = $leap + 4) {

echo "<p>$leap</p>";

}

?>

Foreach loop in PHP:

<?php

$langs = array("JavaScript",

"HTML/CSS", "PHP",

"Python", "Ruby");

foreach ($langs as $lang) {

echo "<li>$lang</li>";

}

unset($lang);

?>

**NOTE: WHILE AND DO-WHILE LOOPS EXACTLY THE SAME AS C**

Functions in PHP:

* strlen()
* substr() (return part of string)

Ex.

$myname = "David";

*// you can manipulate strings easily*

*// with built-in funtions too*

$partial = substr($myname, 0, 3);

print $partial;

* strpos(“emiliy, “e”) (finds first occurrence of substring in a string)
* round(M\_PI, 4) (second parameter is number of places to round to)
* rand(1,10)

Adding to array:

$fav\_bands = array();

array\_push($fav\_bands, "Maroon 5");

array\_push($fav\_bands, "Bruno Mars");

array\_push($fav\_bands, "Nickelback");

array\_push($fav\_bands, "Katy Perry");

array\_push($fav\_bands, "Macklemore");

count()<-(count number of elements in array)

ex.

print count($fav\_bands); *// prints 5*

sort()<- (Arrange in ascending order)

$array = array(5, 3, 7, 1);

sort($array);

print join(", ", $array);

*// prints "1, 3, 5, 7"*

rsort() <- (Arrange is descending order)

$array = array(5, 3, 7 ,1);

rsort($array);

print join(":", $array);

*// prints "7:5:3:1"*

Creating a class in PHP:

<?php

class Person {

}

?>

Create new object:

$me = new Person('boring', '12345', 12345);

Creating a method/function in PHP:

public $isAlive = true;

public $firstname;

public $lastname;

public $age;

// Assigning the values

public function \_\_construct($firstname, $lastname, $age) {

$this->firstname = $firstname;

$this->lastname = $lastname;

$this->age = $age;

}

// Creating a method (function tied to an object)

public function greet() {

return "Hello, my name is " . $this->firstname . " " . $this->lastname . ". Nice to meet you! :-)";

}

echo $me-> greet(); // print out what greet returns

Creating a constructor class (used for making object/ template for each object):

public function \_\_construct($prop1, $prop2) {

$this->prop1 = $prop1;

$this->prop2 = $prop2;

}

**NOTE: FOR** function \_\_ construct leave space in between function and underscore as well as use 2 underscores before construct

Making a method:

public function funcname($optionalParameter) {

*// Do something*

}

OR

public function greet(){

return "Hello, my name is ". $this->firstname." ".$this->lastname.". Nice to meet you! :-)";

}

$teacher = new Person("boring","12345",12345);

$student = new Person("Bruh","yo",95);

echo $teacher->greet();

echo $student->greet();

Calling a Method:

$obj1 -> meth1();

Built in functions in php:

is\_a()

- find out if a particular object is an instance of a given class

-Ex. is\_a($me, "Person")

property\_exists()

-see if an object has a given property

property\_exists($me, "name")

method\_exists()

- see if an object has a given method

method\_exists($me, "dance")

Class inheritance/ extending classes:

class Shape {

public $hasSides = true;

}

class Square extends Shape {

}

$square = new Square();

// Add your code below!

if ( ) {

echo "I have sides!";

}

Final keyword: stops subclasses from overriding methods

Ex.

class Vehicle {

final public function drive() {

return "I'm drivin' here!";

}

}

const alive = true;

Value will never change

**NOTE: IF U WANT TO PRINT PROPERTIES INSIDE METHODS AND CLASSES, DO SO OUTSIDE THE CLASS BRACKETS**

**-**The static keyword lets you use a class' property or method without having to create an instance of that class

class Person {

public static $isAlive = "Yep!"

public static function greet() {

echo "Hello there!";

}

}

echo Person::$isAlive;

*// prints "Yep!"*

Person::greet();

*// prints "Hello there!"*

Associative Arrays:

Arrays you can access without using integers

* done using the key => value pairs

$myArray = array(2012, 'blue', 5);

// ...and this is an associative array:

$myAssocArray = array('year' => 2012,

'colour' => 'blue',

'doors' => 5);

// This code will output "blue"...

echo $myArray[1];

echo '<br />';

// ... and this will also output "blue"!

echo $myAssocArray['colour'];

Print newline:

echo ‘<br />’;

Multidimensional Arrays:

$deck = array(array('2 of Diamonds', 2),

array('5 of Diamonds', 5),

array('7 of Diamonds', 7));

// Imagine the first chosen card was the 7 of Diamonds.

// This is how we would show the user what they have:

echo 'You have the ' . $deck[2][0] . '!';