Getting Started with OOP   
*The Object of My Affections*

**Terms**

Class

Instantiation

Encapsulation

Getter

Setter

Functions – methods

Variables – properties

*[I used many websites and books for this code. Kevork’s book is excellent. Some code below is based on info found at killerphp and code2learn.]*

A class is similar to a blueprint. You can make new objects based on the class. This process is called instantiation.

The class has methods. These are functions which the object can perform.

The class has properties. These are written as variables.

Having all the processes for an object to be used (its own data and methods) is called encapsulation.

cat\_lib.php

<?php

/\*\*\*\*\*

file: cat\_lib.php

used with cat.php

\*\*\*\*\*\*\*\*\*\*\*\*/

class Cat

{

var $name;

var $breed;

function setName($nameofcat)

{

$this -> name = $nameofcat;

}

function getName()

{

return $this -> name;

}

function setBreed($breedname)

{

$this -> breed = $breedname;

}

function getBreed()

{

return $this -> breed;

}

function antics()

{

echo "This kitty chases imaginary mice.";

}

}

?>

cat.php

<?php

/\*\*\*\*\*\*\*\*\*\*\*

cat.php

\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

include('cat\_lib.php');

$kitty = new Cat();

$kitty -> setName('Mrkosa');

echo "My kitty is called ".$kitty -> getName()." .";

$kitty -> setBreed('Persian');

echo "My kitty is a ".$kitty -> getBreed(). " .";

$kitty -> antics();

?>

Run the files and examine the code. You should see this on your screen:

**My kitty is called Mrkosa .My kitty is a Persian .This kitty chases imaginary mice.**

**Assignment:** Create two new files. When they are run, you should see this on your screen:

**My dog has the name of Poochie . The breed is Brittany Spaniel .This dog dreams a lot.**

<?php

/\*\*\*\*\*\*\*\*\*\*\*\*

file: dog.inc

called from dog.php

\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

class Dog

{

var $name;

var $breed;

function setName($nameofdog)

{

$this -> name = $nameofdog;

}

function getName()

{

return $this ->name;

}

function setBreed($breedofdog)

{

$this -> breed = $breedofdog;

}

function getBreed()

{

return $this -> breed;

}

function sleep()

{

echo "This dog dreams a lot. ";

}

}

?>

dog.php

<?php

/\*\*\*\*\*\*

file: dog.php

\*\*\*\*\*\*\*\*\*\*/

include ("dog.inc");

$myDog = new Dog();

$myDog -> setName('Poochie');

echo "My dog has the name of ".$myDog -> getName(). " .";

$myDog -> setBreed('Brittany Spaniel');

echo " The breed is ".$myDog -> getBreed(). " .";

$myDog -> sleep();

?>

**Part 2.**

In this section you will gather data from variables posted in an HTML form.

<html>

<!--

calc.html

calls calc.php

-->

<head>

<title>Calculations</title>

<link rel="stylesheet" type="text/css" href = "calc.css" />

</head>

<body>

<form action="calc.php" method=post>

<fieldset>

<legend>Calculations </legend>

<p ><label >Integer 1</label></p>

<p > <input type="text" name="numOne" /></p>

<p ><label >Integer 2</label></p>

<p > <input type="text" name="numTwo" /></p>

<p class ="submit" > <input type = "submit" value = "Calculate" /> </p>

</fieldset>

</form>

</body>

</html>

Here is an incomplete calc.inc file. Work out the missing lines of code.

<?php

/\*\*\*\*\*\*\*\*\*\*\*\*\*

file: calc.inc

provide class for calc.php

\*\*\*\*\*\*\*\*\*\*\*/

class Calc

{

var $input;

var $input2;

var $output;

function setInput1($int)

{

$this -> input = $int;

}

function setInput2($int)

{

$this -> input2= $int;

}

function getProduct()

{

*missing lines*

}

function getDifference()

{

*missing lines*

}

function getSum()

{

$this->output = $this->input + $this ->input2;

return $this->output;

}

function getQuotient()

{

*missing lines*

}

}

?>

**Full code**

<?php

/\*\*\*\*\*\*\*\*\*\*\*\*\*

file: calc.inc

provide class for calc.php

\*\*\*\*\*\*\*\*\*\*\*/

class Calc

{

var $input;

var $input2;

var $output;

function setInput1($int)

{

$this -> input = $int;

}

function setInput2($int)

{

$this -> input2= $int;

}

function getProduct()

{

$this->output = $this-> input \* $this->input2;

return $this -> output;

}

function getDifference()

{

$this->output = $this->input - $this->input2;

return $this->output;//return $this -> output . "<br /> \n" ;

}

function getSum()

{

$this->output = $this->input + $this ->input2;

return $this->output;

}

function getQuotient()

{

$this -> output = $this->input / $this ->input2;

return $this->output;

}

}

?>

Below is an incomplete file for calc.php. Add the missing lines of code.

<?php

/\*\*\*\*\*\*\*\*\*\*\*\*

file: calc.php

uses calc.inc

called from calc.html

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

include('calc.inc');

$numOne = $\_POST['numOne'];

$numTwo = $\_POST['numTwo'];

$calc = new Calc();

$calc -> setInput1($numOne);

$calc -> setInput2($numTwo);

//add them and show the sum

echo "The sum of the two integers is ".$calc -> getSum()." .";

//display the difference

*missing code*

//display the product

*missing code*

//display the quotient

*missing code*

?>

**Full code for calc.php**

<?php

/\*\*\*\*\*\*\*\*\*\*\*\*

file: calc.php

uses calc.inc

called from calc.html

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

include('calc.inc');

$numOne = $\_POST['numOne'];

$numTwo = $\_POST['numTwo'];

$calc = new Calc();

$calc -> setInput1($numOne);

$calc -> setInput2($numTwo);

//add them and show the sum

echo "The sum of the two integers is ".$calc -> getSum()." .";

//display the difference

echo " The difference is ".$calc -> getDifference()." .";

//display the product

echo " The product is ".$calc -> getProduct()." .";

//display the quotient

echo " The quotient is ".$calc -> getQuotient()." .";

?>

The CSS file is calc.css

html

{

margin:0;

padding:0;

}

body{

font-family: arial;

font-size: 16px;

margin:2em 0 0 5em;

padding:0;

font-family: 'Palatino Linotype', Garamond, serif;

background: #406E72; /\* pastel greenish \*/

color: #FFF;

}

label {

display: block;

width: 200px;

float: left;

padding-left: 5px;

text-align: right;

font-weight: bold;

}

input {

float: clear;

display: block;

width: 150px;

float: left;

position: relative;

margin-left: 4px;

margin-bottom: 15px;

}

fieldset

{

border: 1px solid #BF3044;/\*reddish\*/

width: 30em;

padding-bottom: 20px;

padding-left: 10px;

}

legend

{

color: #fff;

background: #A8A8B3; /\*pastel light green \*/

border: 1px solid #FFFF66; /\*reddish\*/

padding: 2px 6px;

}

.submit input

{

color: #fff;

background: #5877B1;/\*gold \*/

border: 2px outset #FFFF66; /\*outset so it looks like a button \*/

margin-left: 4em;

margin-top: 2em;

font-weight: bold;

width: 100px;

padding-bottom: 5px;

}

h1{

color: #666633;

}

a{

text-decoration: none;

}

a:link{

color:#000; /\* black \*/

}

a:visited{

color: #000; /\* black \*/

}

a:hover {

color: #DAE492; /\*grey \*/

}