Homework #2

1.What does it mean for PHP to be an interpreted language? How does this differ from a compiled language? : An interpreted language is a language is ran line by line by an interpreter or virtual machine every time it is ran, while a compiled language is compiled into machine code once and therefore is more efficient. This is since its not being converted into machine code every time its ran.

2.Discuss the advantages and disadvantages of using an interpreted language like PHP over a compiled language. : The disadvantages of using an interpreted language is slower runtime as every time a program is ran using PHP, it has to be recompiled/interpreted into byte code and then machine code. The main advantages however are easy compatibility across platforms, easier to debug as you aren't stuck looking at memory stacks or machine code, and dynamic typing which leads to easier development.

PHP Arrays

3.Explain the difference between indexed arrays and associative arrays in PHP. Provide examples of both. : Indexed arrays are arrays that use indices, or numbers to reference data in the array. An associative array is like a dictionary or hash map, in which you access data via keys that are set when you input data into arrays.

4.How can you iterate through an associative array in PHP? Provide code demonstrating at least two different methods. :

```
<?php
Foreach ($array as $i => $value) {
        //Code here
}
//or
$key_array = array_keys($array)
For ($i = 0; count(array_keys($array); ++$i){
        echo $array[$keys[$i]];
}
?>
```

5What will the following code output, and why?

```
$arr = [1 => 'apple', 2 => 'banana', 'key' => 'orange'];
echo $arr[1];
echo $arr['key'];


/*

Outputs:

appleorange

*/
```

Homework #2

PHP Classes

6.Define a class in PHP that represents a Car with properties like make, model, and year. Include methods for starting the car and stopping it.

```
<?php
class car {
	public $make;
	public $model;
	public $year;

	public function __construct($make,$model,$year) {
		$this->make = $make;
		$this->model = $model;
		$this->year = $year;
	}
	public function start() {
		echo "Starting your " . $this->year . ", " . $this->make . " " . $this->model . "\n";
	}
	public function stop() {
		echo "Stopping your " . $this->year . ", " . $this->make . " " . $this->model . "\n";
	}
}
?>
```

7.What is the purpose of the __construct() method in PHP classes? How is it different from regular methods? : The purpose of the construct method is to provide a constructor for a class, where the user can instantiate a new object and then provide their variables immediately instead of using a setter in a function, if even provided, to change the variables. It differs from regular methods because it will always be called when an object is created, and also only when an object is created. It also cannot be used as a normal function, as it is only ran at object creation.

PHP Variables

8.What are the different types of variables in PHP? : The different types of variables are strings, integers, floats, Booleans, arrays, objects, none/null, and resources.

9.What does the $this keyword refer to in PHP? Provide an example of how it is used within a class. : The $this keyword refers to a pointer/reference to the object's variable that we are in. For example,

```
<?php
	public function start() {
		echo "Starting your " . $this->year . ", " . $this->make . " " . $this->model . "\n";
	}
?>
```

This snippet from the car class above shows the function, which would be called by the object and it references its own “year” variable by using the $this operator.

PHP If Statements

10.Write an if-else statement that checks whether a number is positive, negative, or zero, and prints the appropriate message. :

```
if ($number > 0) {
        echo “Positive Number”;
} elseif ($number < 0) {
        echo “Negative Number”;
} else {
        Echo “Number is Zero”;
}
```

11.Explain the difference between == and === in PHP if statements. Provide an example where they would behave differently. : == Only compares values and the === compares both value and type.

```
If (5 == “5”) {
//This returns Bool True since the values are the same since “5” is automatically converted to an int
}
If (5 === “5”) {
//This returns bool false since it’s a integer versus a string.
}
```

12.What is a ternary operator in PHP? Rewrite the following if-else statement using the ternary operator: A ternary operator is a short-hand version of an if-else statement. You are able to write a if true do this otherwise do this, shorthand

Copy code

```
if ($a > $b) {
  echo "a is greater";
} else {
 echo "b is greater";
}

//Rewriten
Echo ($a > $b) ? “a” : “b”;
```

Homework #2

13.Write an if-else statement in PHP that checks whether a variable contains a string with more than 10 characters and prints a message accordingly.

echo (strlen($string) > 10) ? “Length greater than 10” : “Length less than 10”;

instanceof Operator

14.What does the instanceof operator do in PHP? Provide an example where it is used.

The instanceof operator is used to check if an object belongs to a class specified.

```
<?php
class car {
	public $make;
	public $model;
	public $year;

	public function __construct($make,$model,$year) {
		$this->make = $make;
		$this->model = $model;
		$this->year = $year;
	}
	public function start() {
		echo "Starting your " . $this->year . ", " . $this->make . " " . $this->model . "\n";
	}
	public function stop() {
		echo "Stopping your " . $this->year . ", " . $this->make . " " . $this->model . "\n";
	}
}

$obj  = 5;

If ($obj instanceof car) {
//Test
}
?>
```

15Why might you use instanceof in an object-oriented PHP application? Give a scenario where this would be beneficial.

You may use instanceof in an object oriented application if you need to ensure user input or database output is of a certain class type for processing. For example, if you tried to start a fruit, the function from the car class, it would fail and cause failure of the program. You can ensure this never happens with the instanceof operator.

16What will the following code output, and why?

Homework #2

```
class Animal {}
    22. class Dog extends Animal {}
    23. $dog = new Dog();
    24. var_dump($dog instanceof Animal);
/*Output:

bool(true)

The conditional returns a Boolean value, true, and then var_dumps prints the info of that result.

*/
```