PHP is a Personal Home Page Tools; PHP : Hypertext Preprocessor.

PHP also cross platform to develop, to deploy and to use. That means we can put PHP on a Windows server, on a Mac server, or on a Linux server, and run the same PHP code with no problems or any differences.

* server side scripting language.
* PHP code does not need to be compiled. It's executed by the web server exactly as it's written. C or Java, require the code to be compiled or translated into another form before it can be used.
* Designed for use with HTML. It can be embedded in our HTML/to generate HTML

In the end PHP is going to return HTML to the web browser php => html

* Provide more flexibility than HTML alone
* A PHP lets us create dynamic pages. And page content can change based on conditions. Such as interactions with the user, or data stored in a database. You can think of PHP as turbo charging your html. PHP syntax is going to be very similar to C, Java, and Perl.

Script

* runs in response to an event
* performs instructions from top to bottom
* little or no user interaction

Program

* runs even when not responding to events.
* Jumps around instructions
* Lots of user interaction.

server-side

* code run on our web server

PHP runs on a web server, that means it generally can't run on its own. We'll need to have a running web server in order to use PHP

client-side

* code runs on the user's computer

JavaScript is an example of another popular client side scripting language

Static web pages

* All visitors to web page see the same page all the time

Dynamic web Pages

* A PHP lets us create dynamic pages. And page content can change based on conditions. Such as interactions with the user, or data stored in a database. You can think of PHP as turbo charging your html. PHP syntax is going to be very similar to C, Java, and Perl.

Webserver

A web server is going to listen for page requests from the web browser, and then serve up pages back to web browser in response. While processing the PHP that's in those page at the same time. Apache's the most widely-used web server, especially for PHP development.

<?php echo “hello world”; ?>

You can think of it like printing it to the users browser and we'll use it inside our PHP tags with whatever we want to echo back.

To echo to page

<?php echo “hello world” ?>

concatenate

<?php echo “hello” . “world” ?>

For page break - regular html <br />

<?php echo 2 + 3; ?>

Variable: symbolic representation of a value.

* need to start with a dollar sign
* that needs to be followed by either letter or an underscore
* can contain letters, numbers, underscores or dashes.
* cannot contain any spaces and
* case sensitive.

Strings

* A string is a set of characters. Those characters can be letters, numbers, symbols – text
* defined inside of quotation marks, either single quotes or double quotes
* inline replacements or in place substitution

echo “{$phrase} Again<br />”;

we can use var inside the strings $phrase = “hello”; echo “$phrase World !!”use only with double quotes

String Functions

* strlen(variable/string);
* str\_repeat(char to repeat, #times)); --- str\_repeat(‘=’, 12));

Arrays

Defining an array $name =[‘name1’, ‘name2’, ‘name3’];

Associative Arrays

Key value maps - $artist\_country = [‘goethe’ => ‘Germany’, ‘stringberg’ => ‘sweden’];

array\_shift($numbers);array\_unshift($numbers, "first");array\_pop($numbers);array\_push($numbers, 'last');

Date:

time();

checkdate(12,31,2000)? 'true': 'false';strtotime("November 21, 2016");strtotime("+1 day");strtotime("last Monday"); echo strftime("the date today is %m/%d/%y", time());

String:

str\_replace('\*', '', $no\_zeros);

Super Globa echo "mysql format date : " . $mysql\_datetime;

echo "Server details: <br />";

echo "Server\_name: ".$\_SERVER['SERVER\_NAME']."<br />";

echo "Server\_ADDR: ".$\_SERVER['SERVER\_ADDR']."<br />";

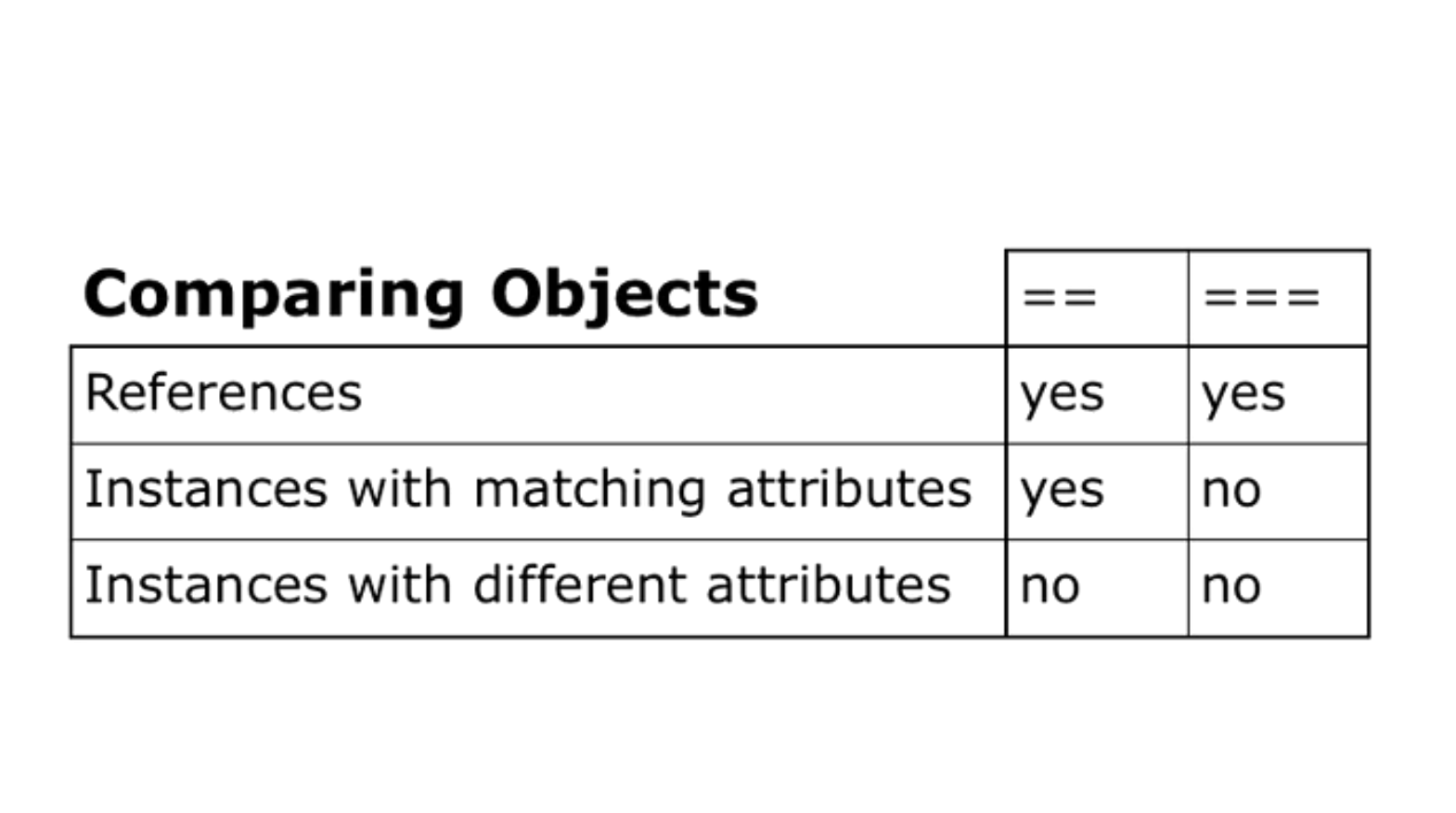
echo "Server\_PORT: ".$\_SERVER['SERVER\_PORT']."<br />";

echo "page details:<br /> ".$\_SERVER['PHP\_SELF']."<br />"; - redirect url

echo "script filename details:<br /> ".$\_SERVER['SCRIPT\_FILNAME']."<br />";ls: - include or require use director

$\_GET; $\_POST;$\_COOKIE; $\_SESSION;$\_SERVER

Objects:

* Grouping code by common theme.
* Abstractions of code structure.
  + Functions
  + Attributes
* The public side will have an overview of all the images in the gallery and if we have more than, let's say, 10 to 20 images on a page, we want to havepagination, which would mean that we would have a link that says Next, and that would let us get the next 10 or 20 images. So that's called pagination and we'll be seeing how to do that.

Benefits:

https://www.linkedin.com/in/padmajakondeti Dedicated, hardworking and highly motivated web developer seeking a consulting/full-time position. Four years of extensive experience in developing websites using JavaScript, jQuery, HTML, CSS, PHP, MySQL , Adobe Target & Analytics, Photoshop, Flash and Boot Strap. Hands on experience with new technologies like Nodejs, Express, Handlebar and React etc. from the Rutgers Coding Boot Camp. Eligible to work in USA.

* Better code organization and maintainability.
* It adds clarity and reduces complexity.
* Simple rules allow complex interactions
* Emphasizes data over procedure
* Code modularity
* Code reusability
* Well suited for databases

Class/Object functions:

* get\_declared\_classes
* get\_class\_methods
* get\_class
* is\_a bool **is\_a** ( object $object , string $class\_name [, bool $allow\_string = FALSE ] )
* bool class\_exists ( string $class\_name [, bool $autoload = true ] )
* bool method\_exists ( [mixed](http://php.net/manual/en/language.pseudo-types.php#language.types.mixed) $object , string $method\_name )
* Invoking a function: $person->say\_hello()[function call]

Instantiation: Creating an instance of the class. Occurrence of a person. Different instance.

$person = new Person();

Properties, attributes, instance variables

Conditional statements

* if () {} elseif(){} else{}; - if($age < 18) echo “you are an young fellow” else echo “you are an adult”;
* foreach(){}; - foreach($nameList as $name) {echo “$name is a great artist”;}

Public, private and protected

Static methods you can’t use $this!!

Student::$total\_student;

Scope resolution operator.

MySql database connection string

Server=myServerAddress;Port=1234;Database=myDataBase;Uid=myUsername;  
Pwd=myPassword

**Step1:**Open db connection

public function open\_connection() {

$this->connection = **mysqli\_connect(DB\_SERVER, DB\_USER, DB\_PASS, DB\_NAME);**

if(mysqli\_connect\_errno()) {

die("Database connection failed: " .

mysqli\_connect\_error() .

" (" . mysqli\_connect\_errno() . ")"

);

}

}

**Step2:** perform database query

public function query($sql) {

$result = **mysqli\_query($this->connection, $sql);**

$this->confirm\_query($result);

return $result;

}

private function confirm\_query($result) {

if (!$result) { die("Database query failed.");}}

public function mysql\_prep($string) {

$escaped\_string = mysqli\_real\_escape\_string($this->connection, $string);

return $escaped\_string;

}

To make the string prepares the values for sql - clean and accessible

mysqli::real\_escape\_string -- mysqli\_real\_escape\_string — Escapes special characters in a string for use in an SQL statement, taking into account the current charset of the connection

// "database neutral" functions

public function fetch\_array($result\_set) {

return mysqli\_fetch\_array($result\_set);

}

public function num\_rows($result\_set) {

return mysqli\_num\_rows($result\_set);

}

public function insert\_id() {

// get the last id inserted over the current db connection

return mysqli\_insert\_id($this->connection);

}

public function affected\_rows() {

return mysqli\_affected\_rows($this->connection);

}

Instance method and static method

$User = new User();

// get\_object\_vars returns an associative array with all attributes

// (incl. private ones!) as the keys and their current values as the value

$object\_vars = get\_object\_vars($this);

**Step4:**Close db connection

public function close\_connection() {

if(isset($this->connection)) {

**mysqli\_close($this->connection);**

unset($this->connection);

}

}

Config file

New db connection

array\_shift($result\_array) : to pull first element out of the array

function \_\_autoload(){} – user is not defined then looks for the classes through autoload

session – is going to be maintained on the server as a file.