Introduction to Web Development (4340704)

Computer Department
AVPTI Rajkot

Teaching and Examination Scheme

Theory Marks		Practical Marks		Total Marks
ESE	CA	ESE	CA	**150**
70	30	25	25	

Lecture	Tutorial	Practical	Credit
3	0	4	5

COURSE OUTCOMES (COs)

- Develop PHP scripts using variables, operators and control structures.
- Develop PHP scripts using arrays and functions.
- Develop PHP scripts by applying object oriented concepts.
- Develop web pages using form controls with validation to collect user inputs in PHP.
- Develop and host interactive websites using PHP and MySQL database.

Unit – I

Introduction to PHP

Unit Outcomes

- Write simple scripts using variables, constants, and operators.
- Write simple scripts using decision making statements to solve the given problem.
- Write simple scripts using loop controls to solve the given problem.

Introduction to Static and Dynamic Websites

- **What is Website?**

- A website is a grouping of related web pages, including text, graphics, audio, and video. The home page is the initial page of a website. You must input a website's unique internet address (URL) into your browser to visit it.

- A website is hosted on one or more servers and may accessed over a computer network by visiting its homepage. The website owner, which can be an individual, a company, or an organization, is in charge of it.

- There are two types of websites:
  - **Static Website** and
  - **Dynamic Website**

What is a static website?

- A static website is made up of a fixed number of pre-built files stored on a web server. These files are written in HTML, CSS, and JavaScript, which are called “client-side” languages because they execute in the user’s web browser.
- When a user requests a page from the server with a URL, the server returns the HTML file that is specified by the URL.
- During this exchange, the web server does not alter the files before they’re shipped to the user, so the web page will look the exact same to everyone who requests it. The content is “static” — the only way to change how the website looks is by manually changing the content of the files.

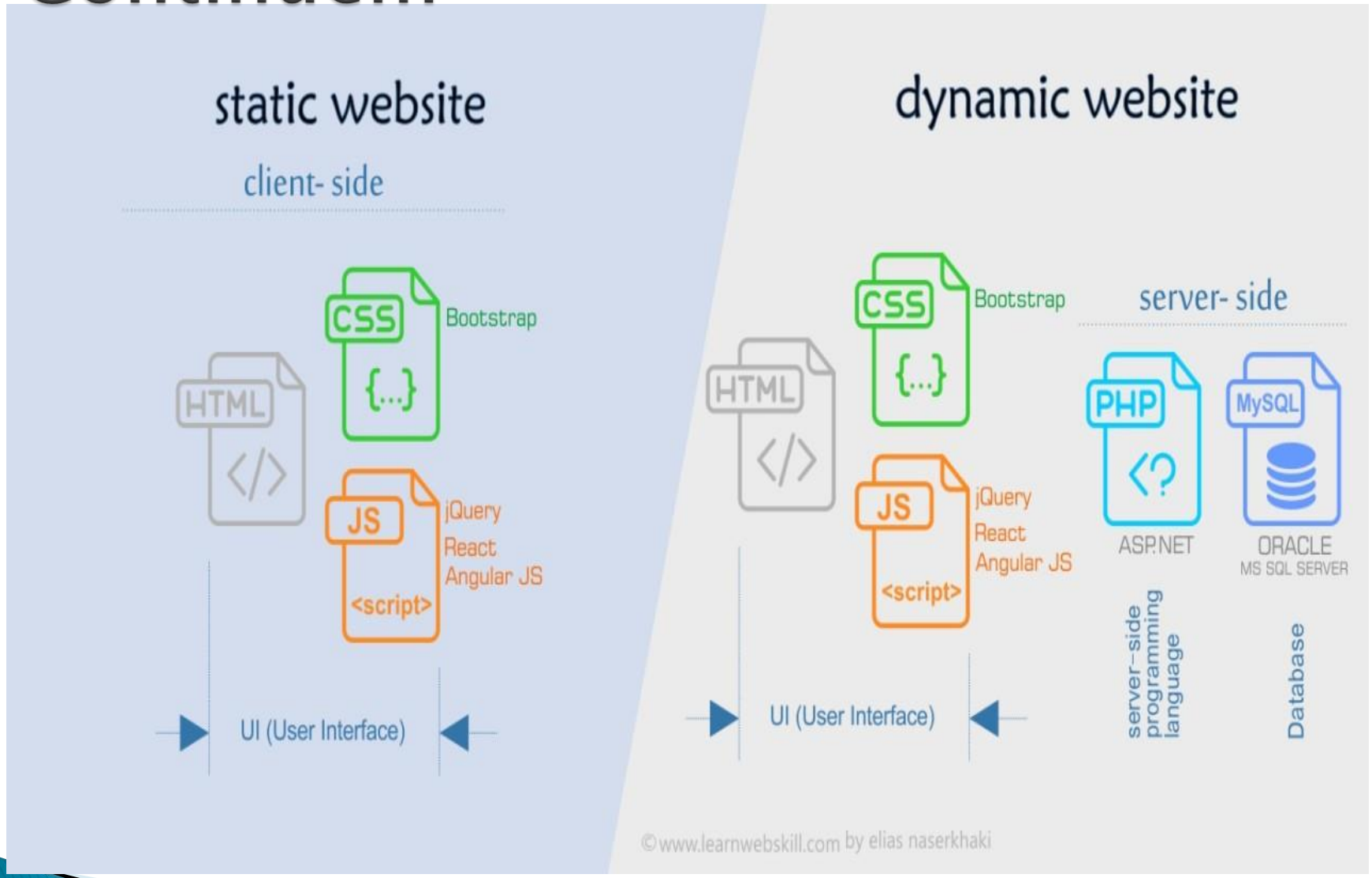
What is a Dynamic website?

- a dynamic website presents different information to different visitors. The content that a visitor sees can be determined by several factors, such as their location, local time, settings and preferences, and/or actions they're taken on the website.
- dynamic websites use server-side scripting languages like PHP, Python, Ruby, or server-side JavaScript, in addition to client-side languages (HTML, CSS, and JavaScript).

Static Vs Dynamic Website

Static Website	Dynamic Website
Content of Web pages can not be change at runtime.	Content of Web pages can be changed.
No interaction with database possible.	Interaction with database is possible
It is faster to load as compared to dynamic website.	It is slower than static website.
Cheaper Development costs.	More Development costs.
No feature of Content Management.	Feature of Content Management System.
HTML, CSS, Javascript is used for developing the website.	Server side languages such as PHP, Node.js are used.
Same content is delivered everytime the page is loaded.	Content may change every time the page is loaded.

Continue…


static website
client- side
HTML
CSS
Bootstrap
JS
jQuery
React
Angular JS
<script>
UI (User Interface)
dynamic website
HTML
CSS
Bootstrap
JS
jQuery
React
Angular JS
<script>
server- side
PHP
ASP.NET
MySQL
ORACLE
MS SQL SERVER
server-side programming language
Database
UI (User Interface)
©www.learnwebskill.com by elias naserkhaki

Introduction to PHP

- PHP (**recursive acronym for *PHP: Hypertext Preprocessor***) is a widely-used **open source** general-purpose server side **scripting language** that is especially suited **for web development** and can be **embedded into HTML**.
- It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, Oracle, Microsoft SQL Server.

What Can PHP Do?

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can be used to control user-access
- PHP can encrypt data.

Why PHP?

- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases
- PHP is free. Download it from the official PHP resource: www.php.net
- PHP is easy to learn and runs efficiently on the server side

History Of PHP

- PHP development began in **1995** when **Rasmus Lerdorf** wrote several Common Gateway Interface (CGI) programs in C which he used to maintain his personal homepage. He extended them to work with web forms and to communicate with databases, and called this implementation "**Personal Home Page/Forms Interpreter" or PHP/FI.**

Continue…

- PHP/FI could be used to build simple, dynamic web applications. To accelerate bug reporting and improve the code, Lerdorf initially announced the release of PHP/FI as "Personal Home Page Tools (PHP Tools) version 1.0“on June 8, 1995.
- PHP/FI 2 officially released in November 1997.

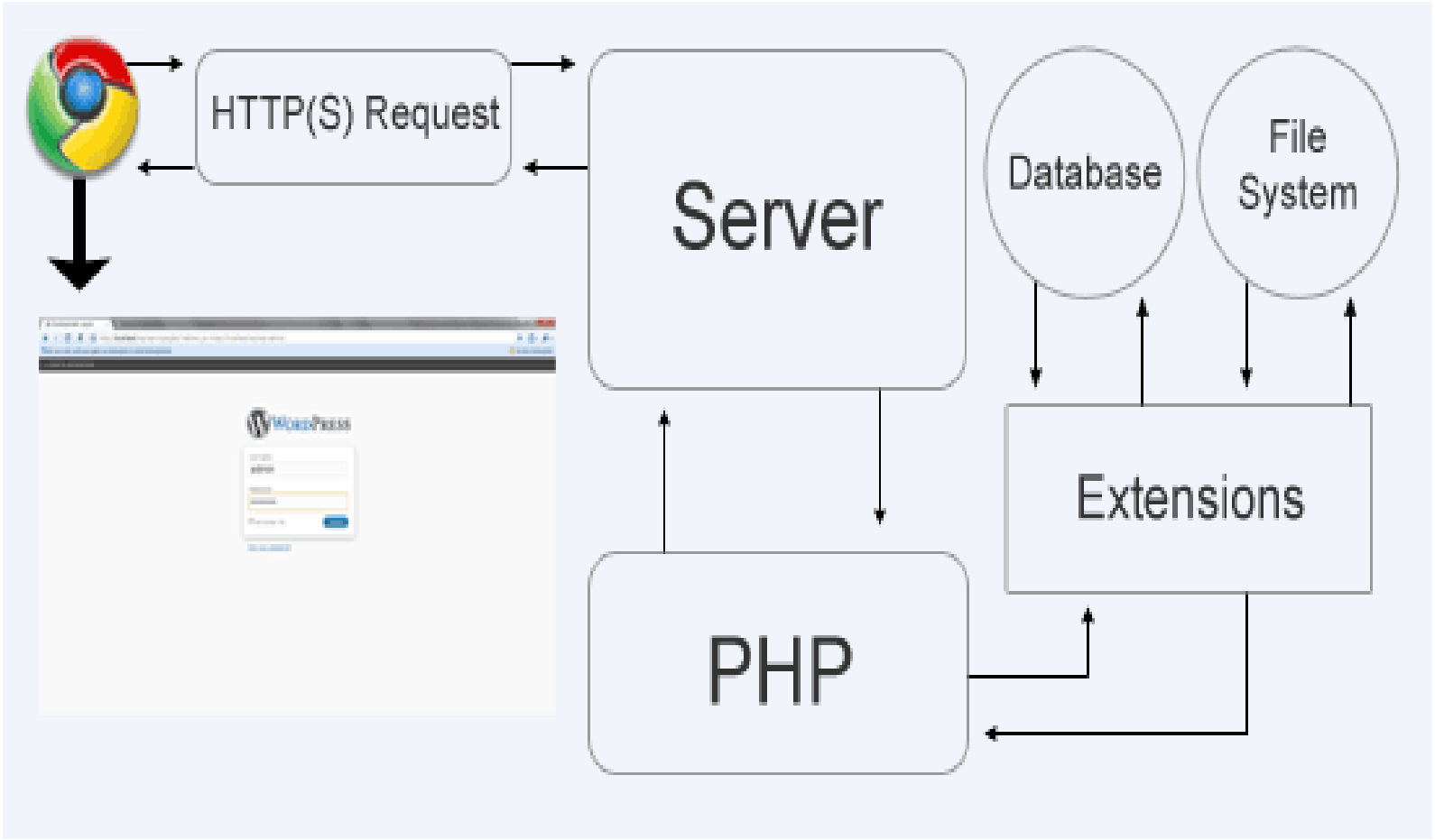
Continue…

- **PHP 3 and 4**
- Zeev Suraski and Andi Gutmans rewrote the parser in 1997 and formed the base of PHP 3, changing the language's name to the recursive acronym *PHP: Hypertext Preprocessor*.
- Afterwards, public testing of PHP 3 began, and the official launch came in June 1998.
- On May 22, 2000, PHP 4, was released As of August 2008 this branch reached version 4.4.9.

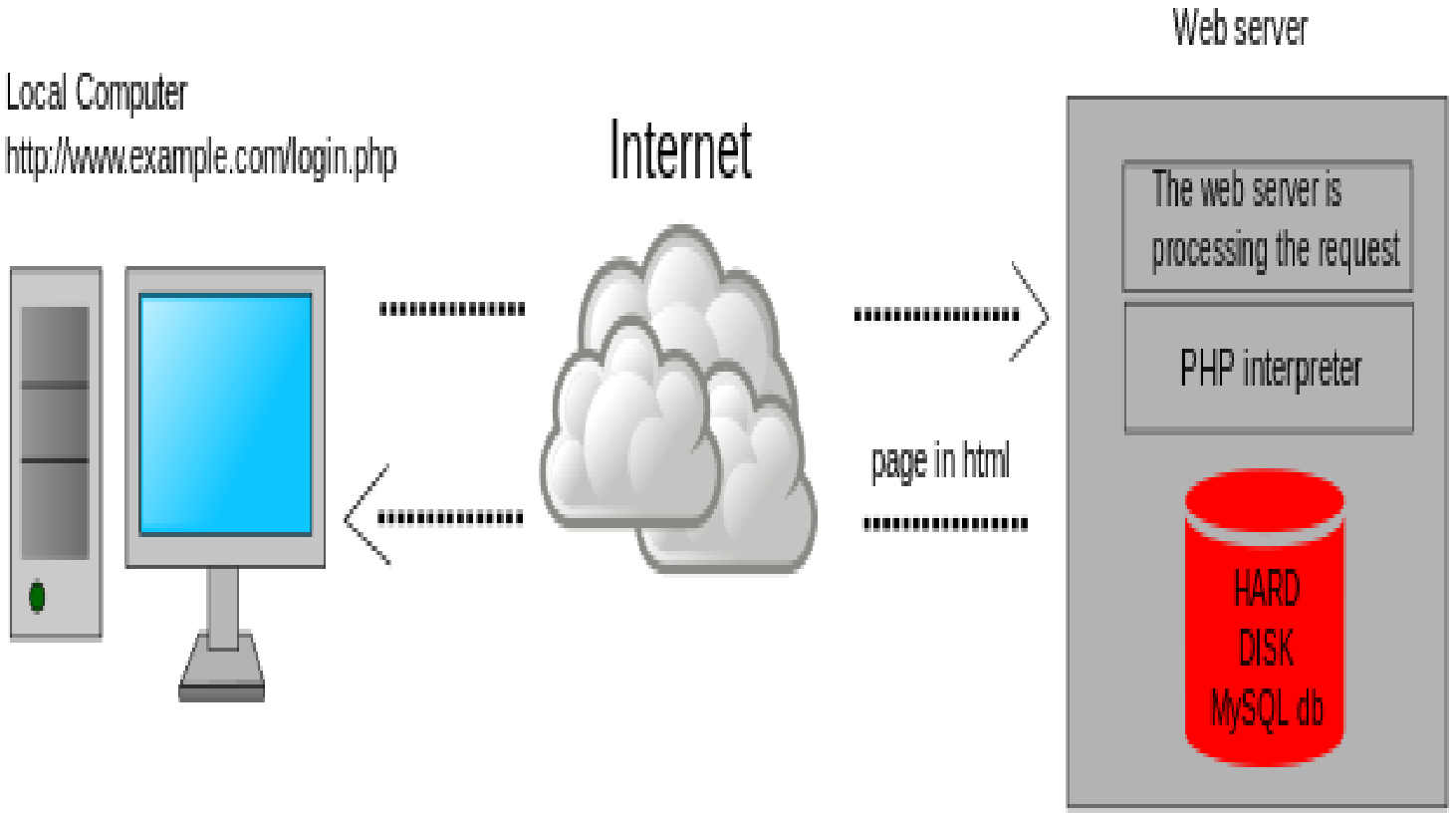
Continue…

- **PHP 5**
- On July 14, 2004, PHP 5 was released, PHP 5 included new features such as improved support for object-oriented programming, the PHP Data Objects (PDO) extension (which defines a lightweight and consistent interface for accessing databases).
- **PHP 6 and 7**
- PHP 6 is not released and 2015 PHP 7 was released.
- **PHP 8** officially released on November 26, 2020

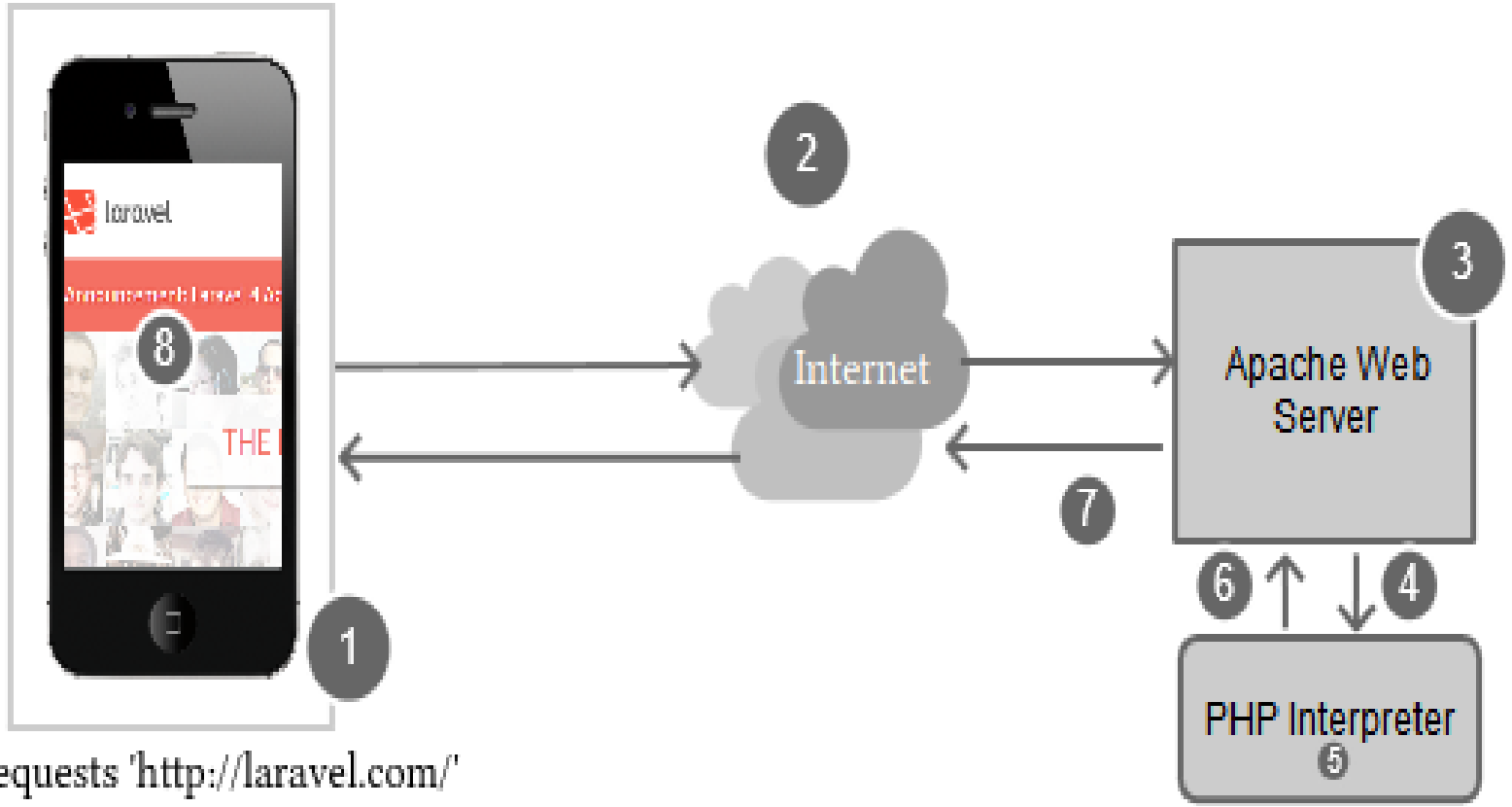
how php works



Continue…


Web server
Local Computer
http://www.example.com/login.php
Internet
The web server is processing the request
PHP interpreter
page in html
HARD DISK
MySQL db

Continue...



User requests 'http://laravel.com/'

Continue…

- The PHP software works with the *web server*, which is the software that delivers web pages to the world.
- When you type a URL into your web browser's address bar, you're sending a message to the web server at that URL, asking it to send you an HTML file.
- The web server responds by sending the requested file. Your browser reads the HTML file and displays the web page.

Create and Save PHP File

- **Creating PHP file**
- PHP file can be created with any text editor. (Notepad, Notepad 2, Notepad++, etc.) Put Following Code in Text Editor.
- ***Example:***
- **<?php echo "Hello PHP!"; ?>**
- Saving PHP file
- Save the file as **simple_php_file.php**
- Make sure that file **extension is php** otherwise it will not execute.

PHP file Structure,PHP start and end tags, Commenting codes

- A PHP script starts with <?php and ends with ?>
- A PHP script can be placed anywhere in the document between this two tags.
- <?php
  // PHP code goes here
  ?>
- A PHP file normally contains HTML tags, and some PHP scripting code.
- You can have as many PHP sections in a script as you need.

Comments in PHP

- A comment in PHP code is a line that is not read/executed as part of the program.
- **Comments can be used to**:
- Let others understand what you are doing.
- Remind yourself of what you did.
- You can also write description about your page in comment.
- // This is a single-line comment
- # This is also a single-line comment
- /*
  This is a multiple-lines comment block
  that spans over multiple
  lines
  */
- You can also use comments to leave out parts of a code line
  $x = 5 /* + 15 */ + 5;

Output statement, echo and print statement

- In PHP there are two basic ways to get output: echo and print.
- **The PHP echo Statement**
- The echo statement can be used with or without parentheses: echo or echo().
- **The PHP print Statement**
- The print statement can be used with or without parentheses: print or print().
- Both statements can be used to display text as well as variable.

Difference between echo and print

- **echo can take multiple** parameters while **print can take one** argument.
- **echo has no return** value while **print has a return value of 1** so it can be used in expressions.
- echo is marginally faster than print.

WAMP

- WAMP is sometimes used as an abbreviated name for the software stack **Windows, Apache, MySQL, PHP**. It is derived from LAMP which stands for **Linux, Apache, MySQL, and PHP**. As the name implies, while LAMP is used on Linux servers, WAMP is used on Windows servers.
- The “A” in WAMP stands for Apache. Apache is server software that is used to serve webpages. Whenever someone types website’s URL, Apache is the software that “serves” your webpages.

Continue…

- The “M” in WAMP stands for MySQL. MySQL is a database management system. It’s job in the software stack is to store all of your website’s content, user profiles, comments, etc.
- The “P” in WAMP stands for PHP. PHP is the programming language.It runs as a process in Apache and communicates with the MySQL database to dynamically build your webpages.

Xampp

- Xampp is an acronym for X(any Operating System), Apache(Web server), MySQL Database, PHP Language and PERL.
- **PERL (Practical Extraction and Report Language**.)
- Perl is a high level general-purpose programming language which is used for web development, network programming, GUI development, and more.
- XAMPP is more Powerful ans support all OS.
- WAMP provides support for MySQL and PHP.
- XAMPP provides support for MySQL, PHP and PERL.

How to run PHP program

- Write this program in a notepad and save as firstprogram.php
- Now to run this do the following steps.
- First download the xampp from http://www.apachefriends.org/en/xampp.html
- Extract the file and double click the exe. This will install Mysql and Apache servers.
- Once the installation is complete, you will find XAMPP under Start / Programs / XAMPP. You can use the XAMPP Control Panel to start/stop all servers. Start Mysql and Apache servers.
- **Copy firstprogram.php to C:/Program Files/XAMPP/htdocs/**
- To run the php file, you just need to brows http://localhost/firstprogram.php

- **Note**: You can also create any folders inside htdocs folder. For ex. create a folder called “test” and copy firstprogram.php to “test”, then the address will become http://localhost/test/firstprogram.php

PHP Variables

- Variables are "containers" for storing information:
  Rules for PHP variables:
- A variable starts with the **$ sign**, followed by the name of the variable
- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are **case sensitive** ($y and $Y are two different variables)
- A variable can have a short name (like x and y) or a more descriptive name (age, carname, total_volume).

Creating (Declaring) PHP Variables

- PHP has no command for declaring a variable. A variable is created the moment you first assign a value to it:
- $name=“hiren”; //var become string type
- $x=10; //var become int type
- $y=15.10; // var become float type
- **PHP is a Loosely Type Language**
- In the example above, we have not specify the data type for variable.
- PHP automatically converts the variable to the correct data type, depending on its value.

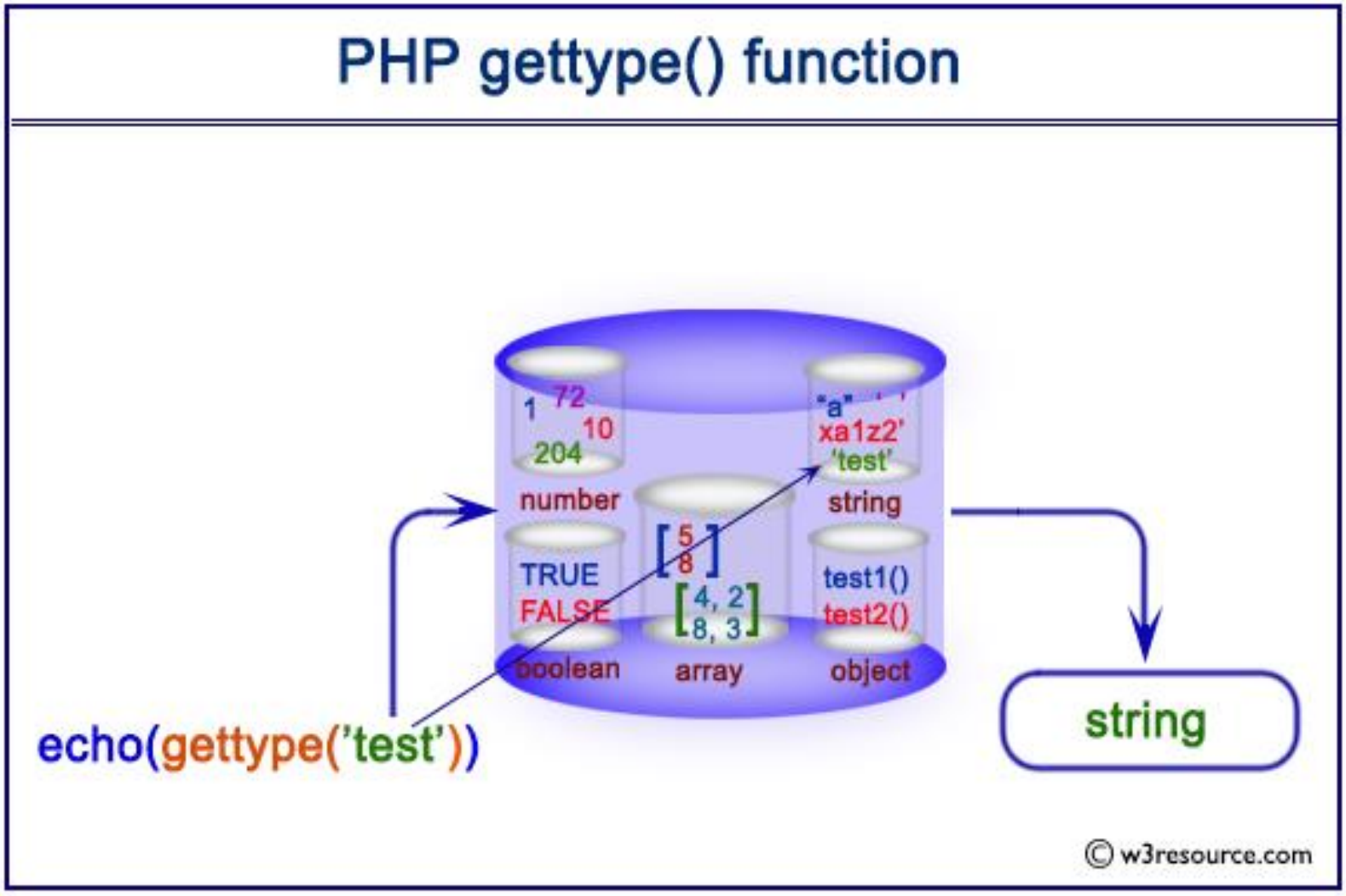
PHP Data Types

- **PHP supports the following data types:**
- **The predefined data types are:**
  - Integer—Used for whole numbers
  - double—Used for real numbers
  - String—Used for strings of characters
  - Boolean—Used for true or false values
- **The user-defined (compound) data types are:**
  - Array—Used to store multiple data items
  - Object—Used for storing instances of classes
- **The special data types are:**
  - NULL—These are special types of variables that can hold only one value i.e.NULL
  - Resource —Resource variables hold special handles for files and database connections.

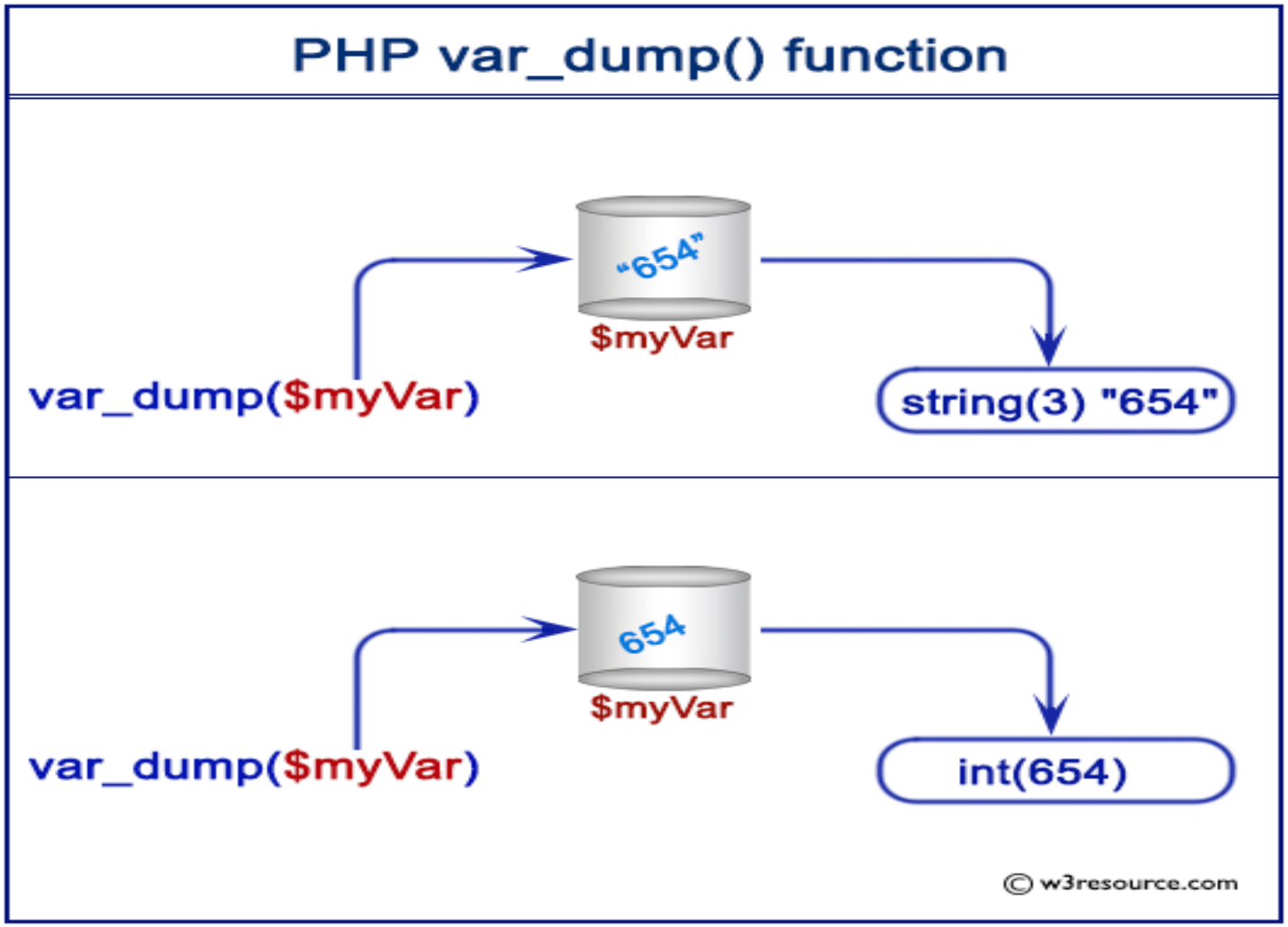
gettype () and settype(),var_dump()

- **The gettype()** function is used to get the type of a variable.
- **gettype(var_name)**
- It will return integer,double,float etc.
- **The settype()** function is used to set the type of a variable(used for type casting).
- **settype(var_name, var_type)**
- **The var_dump()** function is used to display structured information (type and value) about one or more variables.
- **var_dump(variable1, variabl2, ...variablen)**

Continue…


PHP gettype() function
1
72
10
204
number
TRUE
FALSE
boolean
[5, 8]
[4, 2]
[8, 3]
array
"a"
xa1z2'
'test'
string
test1()
test2()
object
echo(gettype('test'))
string
© w3resource.com

Continue…


PHP var_dump() function
"654"
$myVar
var_dump($myVar)
string(3) "654"
654
$myVar
var_dump($myVar)
int(654)
© w3resource.com

PHP operators

- Operator is one type of symbol which is used to perform different types of operations on operand(variable or value).
- For example a+b , here a and b are operands, + is operator , which perform addition operation of a and b.
- **Operators Categories**
- Unary operators, which use a single operand.
- Binary operators, which take two operands and perform a variety of arithmetic and logical operations.
- The conditional operator (a ternary operator), which takes three operands.

Types of Operator

PHP divides the operators in the following groups:

- Arithmetic operators
- Assignment operators
- Bit wise operators
- Comparison operators
- Increment/Decrement operators
- Logical operators
- Conditional operators

Operator precedence

- Operator precedence (priority) determines the order in which the particular expression to be evaluated.
- Certain operators have higher priority than others so while evaluating the expression you have to consider the priority of different operators.
- for example, the multiplication operator has higher precedence than the addition operator

Continue…

- operator precedence (from highest to lowest)

Category	Operator	Associativity
Unary	! ++ --	Right to left
Multiplicative	* / %	Left to right
Additive	+ -	Left to right
Relational	< <= > >=	Left to right
Equality	== !=	Left to right
Logical AND	&&	Left to right
Logical OR	||	Left to right
Conditional	?:	Right to left
Assignment	= += -= *= /= %=	Right to left

Constant

- Constants are like variables except that once they are defined they cannot be changed or undefined.
- A constant is an identifier (name) for a simple value. The value cannot be changed during the script.
- A valid constant name starts with a letter or underscore (no $ sign before the constant name)and usually it in upper case.
- Unlike variables, constants are automatically global across the entire script.
- You can define a constant by using the **define()** ***function*** or by using the ***const keyword***

Coninue…

- define(“CONSTANT_NAME”,”value”);
- const constVariablename=value;
- Difference between constant and variable

Constant	Variable
Name don’t start with $ symbol	Name starts with $ symbol
Scope is global	Access only in its scope
Not redefined or undefined	Redefine
Initialization only one time	More than one time Initialization

Predefine (Magical) constant in PHP

- PHP provides large numbers of predefined constant to any script which it runs.
- There are main seven predefine constant in PHP and they are also known as a magical constants.
- The magic constants are case-insensitive.

List of predefine constants

Name	Description
__FILE__	The full path and filename of the file
__DIR__	The directory of the file
__FUNCTION__	The function name
__CLASS__	The class name
__METHOD__	The class method name
__NAMESPACE__	The name of the current name space
__LINE__	The current line number of the file

PHP Conditional Statements

- The statements which control the flow of execution of program based on some conditions is known as a Conditional statements or control statements
- Conditional statements are used to perform different actions based on different conditions.
- In PHP there are following conditional statements:
- **if statement** – executes some code if one condition is true
- **if...else statement** – executes some code if a condition is true and another code if that condition is false
- **if...else if....else statement** – executes different codes for more than two conditions
- **switch statement** – selects one of many blocks of code to be executed

PHP – The if Statement

- The if statement executes some code if one condition is true.
- if (*condition*)

{

code to be executed if condition is true;

}

The if...else Statement

The if....else statement executes some code if a condition is true and another code if that condition is false.

if (*condition*)

{

code to be executed if condition is true;

}

else

{

code to be executed if condition is false;

}

PHP – The if...elseif....else Statement

- The if....else if...else statement executes different codes for more than two conditions.
- if (*condition*)

```
{
    code to be executed if this condition is true;
}
else if (condition)
{
    code to be executed if this condition is true;
}
else
{
    code to be executed if all conditions are false;
 }
```

PHP – The switch Statement

- The switch statement is used to perform different actions based on different conditions.
- Use the switch statement to **select one of many blocks of code to be executed.**
- switch (*n*)

```
{
    case label1:
        code to be executed if n=label1;
        break;
    case label2:
        code to be executed if n=label2;
        break;
    ...
    default:
        code to be executed if n is different from all
        labels;
}
```

PHP Loop statments

- The statements which is used to repeat the execution of the same code more than one time is known as a looping statements.
- In PHP, there are following looping statements:
- **while** – loops through a block of code as long as the specified condition is true
- **do...while** – loops through a block of code once, and then repeats the loop as long as the specified condition is true
- **for** – loops through a block of code a specified number of times
- **foreach** – loops through a block of code for each element in an array

The PHP while Loop

- The while loop executes a block of code as long as the specified condition is true.
- **while** (*condition is true*)

{

code to be executed until the condition is true

}

do...while Loop

The do...while loop will always execute the block of code once, it will then check the condition, and repeat the loop while the specified condition is true.

Continue…

- **do**
  {
  code to be executed at least once and than execute until condition is true ;
  } **while** (*condition is true*);

for Loop

The for loop is mainly used when you know in advance how many times the script should run.

for (*init counter; test counter; increment counter*)
{
code to be executed;
}

- in loop statements we can use one loop inside another loop, this is known as a **nesting of loops.**

Break and continue statements

- Break ends the execution of the current for,
  for each ,while , do while and switch statements .
- *break* accepts an optional numeric argument which tells it how many nested enclosing structures are to be broken out of.
- The default value is *1*, only the immediate enclosing structure is broken out of.
- ***continue*** is used within looping structures to skip the rest of the current loop iteration and continue execution at the condition evaluation and then the beginning of the next iteration.