USN :

CMR Institute of Technology, Bangalore
DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
II - INTERNAL ASSESSMENT

Semester: 6-CBCS 2018   Date: 26 Jun 2021
Subject: WEB TECHNOLOGY AND ITS APPLICATIONS (18CS63)   Time: 09:00 AM - 10:30 AM
Faculty: Ms Priyadharshini A   Max Marks: 50

Q.No		Answer any 5 question(s)	Marks	CO	PO	BT/CL
1		Compare absolute, relative and fixed positioning with suitable examples	10	CO2	PO2	L2
2	a	Give an example to make thumbnails using hover pseudo-class	4	CO2	PO2	L2
	b	Describe how to modify an element and to change an element's style using DOM	6	CO3	PO2	L2
3	a	Generate a CSS file using SASS preprocessor	5	CO2	PO3	L3
	b	Design a responsive web page using CSS media queries.	5	CO2	PO3	L3
4		Explain different event types in JavaScript with suitable examples	10	CO3	PO1	L1
5	a	Write a PHP code to display "Have a good morning!" if the current time is less than 10, and "Have a good day!" if the current time is less than 20. Otherwise display "Have a good night!"	4	CO3	PO3	L3
	b	Compare pass by value and pass by reference in PHP with suitable examples.	6	CO3	PO2	L2
6		Explain A, M and P components of a LAMP stack.	10	CO3	PO1	L1

Scheme Of Evaluation
Internal Assessment Test 2 – June 2021

Sub:	Web Technology and its applications						Code:	18CS63
Date: 26 / 06 / 2021	Duration:	90mins	Max Marks:	50	Sem:	VI	Branch:	ISE

Note: Answer Any FIVE Questions

Question #		Description	Marks Distribution		Max Marks
1		Compare absolute, relative and fixed positioning with suitable examples Comparing and explaining all 3 properties	4m 3m 3m	10 m	10 M
2	(a)	Write a program make thumbnails using hover pseudo-class	4m	4M	10 M
	(b)	Modifying the element with program Changing an elements style	4m 2m	6m	
3	a)	Write a program which generates CSS file	5m	5m	10 M
	b)	Responsive design using Media queries – Explanation with examples	5m	5m	
4		4 different types Mouse, Keyboard, Form and frames events Explanation with all types	3m 3m 2m 2m	10m	10 M
5	(a)	Write a PHP program using if loop	4m	4m	10m
	(b)	Explanation and comparison of pass by values and pass by parameters with example	3m 3m	6m	
6		Explain A,M and P components of a LAMP stack A component M component P component	3m 3m 4m	10M	10M

1. Compare absolute, relative and fixed positioning with suitable examples
   Solution :

Type	Description
absolute	The element is removed from normal flow and positioned in relation to its nearest positioned ancestor.
fixed	The element is fixed in a specific position in the window even when the document is scrolled
relative	The element is moved relative to where it would be in the normal flow.

Relative Positioning

In **relative positioning** an element is displaced out of its normal flow position and moved relative to where it would have been placed normally. The other contents around the relatively positioned element remain in its old position in the flow; thus the space the element would have occupied is preserved as shown in the example below.

Eg:

```
<html>
<head>
<style>
figure {
position: relative; top: 150px;
left: 200px;
}
</style>
</head>
<body>
<p>Sachin Tendulkar, who is considered one of the greatest players in modern-day cricket was born exactly 47 years ago in a small apartment in Bombay, India. He was the youngest one amongst four siblings and hailed from a middle-class family.

Surprisingly, he was named after his family's favourite music director, Sachin Dev Burman. Sachin had little interest in music ironically, but was always an outstanding athlete and schoolboys were always awestruck by his extraordinary batting skills.</p>

<figure>

<img src="HOUSE.png" width="200" height="100" />
<figcaption>Home</figcaption>
</figure>
<p>Sachin Tendulkar, who is considered one of the greatest players in modern-day cricket was born exactly 47 years ago in a small apartment in Bombay, India. He was the youngest one amongst four siblings and hailed from a middle-class family.

Surprisingly, he was named after his family's favourite music director, Sachin Dev Burman. Sachin had little interest in music ironically, but was always an outstanding athlete and schoolboys were always awestruck by his extraordinary batting skills.</p>

</body>
```

</html>

Absolute Positioning

When an element is positioned absolutely, it is removed completely from normal flow. Here, space is not left for the moved element, as it is no longer in the normal flow. Its position is moved in relation to its container block. In the below example, <figure> block's container is body block. The moved block can overlap the content in the underlying normal flow.

```
<html>
<head>
<style>
figure {
position: absolute;
 top: 150px;
left: 200px;
}
</style>
</head>
<body>
<p>Sachin Tendulkar, who is considered one of the greatest players in modern-day cricket was born exactly 47 years ago in a small apartment in Bombay, India. He was the youngest one amongst four siblings and hailed from a middle-class family.

Surprisingly, he was named after his family's favourite music director, Sachin Dev Burman. Sachin had little interest in music ironically, but was always an outstanding athlete and schoolboys were always awestruck by his extraordinary batting skills.</p>

<figure>

<img src="HOUSE.png" width="200" height="100" />
<figcaption>Home</figcaption>
</figure>
<p>Sachin Tendulkar, who is considered one of the greatest players in modern-day cricket was born exactly 47 years ago in a small apartment in Bombay, India. He was the youngest one amongst four siblings and hailed from a middle-class family.

Surprisingly, he was named after his family's favourite music director, Sachin Dev Burman. Sachin had little interest in music ironically, but was always an outstanding athlete and schoolboys were always awestruck by his extraordinary batting skills.</p>

</body>
</html>
```

Fixed Position

The fixed position value is used relatively infrequently. The element is positioned in relation to the viewport (i.e., to the browser window). Elements with **fixed positioning** do not move when the user scrolls up or down the page.

The fixed position is used to ensure that navigation elements or **advertisements are always visible.**

```
<html>
<head>
<style>
figure {
position: fixed;
 top: 150px;
left: 200px;
}
</style>
</head>
<body>
<p>Sachin Tendulkar, who is considered one of the greatest players in modern-day cricket was born exactly 47 years ago in a small apartment in Bombay, India. He was the youngest one amongst four siblings and hailed from a middle-class family.

Surprisingly, he was named after his family's favourite music director, Sachin Dev Burman. Sachin had little interest in music ironically, but was always an outstanding athlete and schoolboys were always awestruck by his extraordinary batting skills.</p>

<figure>

<img src="HOUSE.png" width="200" height="100" />
<figcaption>Home</figcaption>
</figure>
<p>Sachin Tendulkar, who is considered one of the greatest players in modern-day cricket was born exactly 47 years ago in a small apartment in Bombay, India. He was the youngest one amongst four siblings and hailed from a middle-class family.

Surprisingly, he was named after his family's favourite music director, Sachin Dev Burman. Sachin had little interest in music ironically, but was always an outstanding athlete and schoolboys were always awestruck by his extraordinary batting skills.</p>

</body>
</html>
```

2. (a) Give an example to make thumbnails using hover pseudo-class.

```
<html>
  <head>
    <title>Title of the document</title>
    <style>
      img {
        border: 2px solid #C0C0C0;
        padding: 5px;
      }
      img:hover {
        box-shadow: 5px 4px 10px #8B0000;
      }
    </style>
  </head>
  <body>
    <atarget="_blank"
href="https://www.metoffice.gov.uk/binaries/content/gallery/metofficegovuk/hero-
images/weather/autumn/autumn-leaves-against-a-blue-sky-photo-andrew-small.jpg">

    <imgsrc="https://www.metoffice.gov.uk/binaries/content/gallery/metofficegovuk/hero-
images/weather/autumn/autumn-leaves-against-a-blue-sky-photo-andrew-small.jpg"
alt="Autumn" width="200">
    </a>
  </body>
</html>
```

(b) Describe how to modify an element and to change an element's style using DOM

Element Node Object and its modification

The document.getElementById() method returns an object of **element node**. The method takes the 'ID' of element as parameter,whose control is returned. Since IDs must be unique in an HTML document, getElementByID() returns a single node.

Some properties can be used on these element nodes to modify, create or alter the node properties. They are

Property	Description
id	The current value for the id of this element.
innerHTML	Represents all the things inside of the tags. This can be read or written to and is the primary way in which we update particular <div>, <p> elements using JavaScript.
style	The style attribute of an element. We can read and modify this property.
tagName	The tag name for the element.

Eg:

Suppose the document contains these elements –

```
<div id="div01">
<p>By Ricardo on <time>September 15, 2012</time></p>
<p>Easy on the HDR buddy.</p>
</div>
/******IN JavaScript *************/
var latest = document.getElementById("div01"); var oldMessage = latest.innerHTML;
```

```
                latest.innerHTML = oldMessage + "<p>Updated this div with JS</p>";
```

Now the document has been modified to reflect that change.
<div id="div01">
<p>By Ricardo on <time>September 15, 2012</time></p>
<p>Easy on the HDR buddy.</p>
<p>Updated this div with JS</p>
        </div>

Changing an Element's Style
The CSS style associated with a particular block of elements can be modified by using 'style' or className property of the Element node,
Eg:
1) To change a node's background color

var x= document.getElementById("specificTag"); x.style.backgroundColour = "#FFFF00";
2) To add a three-pixel border.

        var x= document.getElementById("specificTag");
        commentTag.style.borderWidth="3px";

3. (a) Generate a CSS file using SASS preprocessor

```scss
$colorSchemeA: #796d6d;
$colorSchemeB: #9c9c9c;
$paddingCommon: 0.25em;


footer {
   background-color: $colorSchemeA;
   padding: $paddingCommon * 2;
}


@mixin rectangle($colorBack, $colorBorder) {
   border: solid 1pt $colorBorder;
   margin: 3px;
   background-color: $colorBack;
}


fieldset {
   @include rectangle($colorSchemeB, $colorSchemeA);
}


.box {
   @include rectangle($colorSchemeA, $colorSchemeB);
   padding: $paddingCommon;
}
```
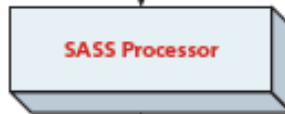
SASS source file, e.g. `source.scss`

This example uses SASS (Syntactically Awesome Stylesheets). Here three variables are defined.

You can reference variables elsewhere. SASS also supports math operators on its variables.

A mixin is like a function and can take parameters. You can use mixins to encapsulate common styling.

A mixin can be referenced/called and passed parameters.

SASS Processor

The processor is some type of tool that the developer would run.

The output from the processor is a normal CSS file that would then be referenced in the HTML source file.
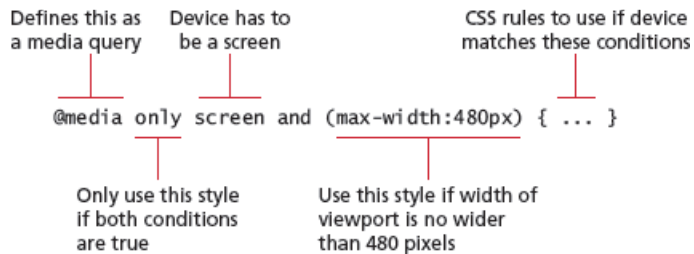
```css
footer {
   padding: 0.50em;
   background-color: #796d6d;
}


fieldset {
   border: solid 1pt #796d6d;
   margin: 3px;
   background-color: #9c9c9c;
}


.box {
   border: solid 1pt #9c9c9c;
   margin: 3px;
   background-color: #796d6d;
   padding: 0.25em;
}
```

Generated CSS file, e.g., `styles.css`

(b) Design a responsive web page using CSS media queries.

Media Queries

The other key component of responsive designs is **CSS media queries**. A media query is a way to apply style rules based on the medium that is displaying the file. Use these queries to look at the capabilities of the device, and then define CSS rules to target that device.

Example of media query

@media only screen and (max-width: 480px) {……}

Defines this as   Device has to   CSS rules to use if device
a media query   be a screen   matches these conditions

    @media only screen and (max-width:480px) { ... }

      Only use this style   Use this style if width of
      if both conditions   viewport is no wider
      are true   than 480 pixels

These queries are Boolean expressions and can be added to your CSS files or to the <link> element to conditionally use a different external CSS file based on the capabilities of the device.

Few elements of the browser features that can be examined with media queries are –

Feature	Description
width	Width of the viewport
height	Height of the viewport
device-width	Width of the device
device-height	Height of the device
orientation	Whether the device is portrait or landscape
color	The number of bits per color

4. Explain different event types in JavaScript with suitable examples.

There are several classes of event, with several types of event within each class. The classes are mouse events, keyboard events, form events, and frame events.

Mouse Events

Mouse events are defined to capture a range of interactions driven by the mouse. These can be further categorized as mouse click and mouse move events. The possible mouse events are –

Events	Description
onclick	The mouse was clicked on an element
ondblclick	The mouse was double clicked on an element
onmousedown	The mouse was pressed down over an element
onmouseup	The mouse was released over an element
onmouseover	The mouse was moved (not clicked) over an element
onmouseout	The mouse was moved off of an element
onmousemove	The mouse was moved while over an element

Keyboard Events

Keyboard events occur when taking inputs from the keyboard. These events are most useful within input fields. The possible keyboard events

Events	Description
onkeydown	The user is pressing a key (this happens first)
onkeypress	The user presses a key (this happens after onkeydown)
onkeyup	The user releases a key that was down (this happens last)

The input box above, for example, could be listened to and each key pressed echoed back to the user as an alert.

```
document.getElementById("keyExample").onkeydown = function myFunction(e){
var keyPressed=e.keyCode; //get the raw key code
var character=String.fromCharCode(keyPressed); //convert to string
alert("Key " + character + " was pressed");
    }
```

Form Events

Forms are the main means by which user input is collected and transmitted to the server. The events triggered by forms allow us to do some timely processing in response to user input. The most common JavaScript listener for forms is the onsubmit event. The different form events are -

Events	Description
onblur	A form element has lost focus (that is, control has moved to a different element), perhaps due to a click or Tab key press.
onchange	Some <input>, <textarea>, or <select> field had their value change. This could mean the user typed something, or selected a new choice
onfocus	This is triggered when an element gets focus (the user clicks in the field or tabs to it).
onreset	HTML forms have the ability to be reset. This event is triggered when reset (cleared).
onselect	When the users selects some text. This is often used to try and prevent copy/paste.
onsubmit	When the form is submitted this event is triggered. We can do some prevalidation when the user submits the form in JavaScript before sending the data on to the server

Eg: If the password field (with id pw) is blank, we prevent submitting to the server
document.getElementById("loginForm").onsubmit = function(e){
var pass = document.getElementById("pw").value; if(pass=="")
{
alert ("enter a password"); e.preventDefault();
}
  }

Frame Events
Frame events are the events related to the browser frame that contains the web page. The most important event is the onload event, which triggers when an object is loaded. The frame events are –

Events	Description
onabort	An object was stopped from loading
onerror	An object or image did not properly load
onload	When a document or object has been loaded
onresize	The document view was resized
onscroll	The document view was scrolled
onunload	The document has unloaded

5. (a) Write a PHP code to display "Have a good morning!" if the current time is less than 10, and "Have a good day!" if the current time is less than 20. Otherwise, display "Have a good night!".

```php
<?php
$t = date("H");

if ($t < "10") {
  echo "Have a good morning!";
} elseif ($t < "20") {
  echo "Have a good day!";
} else {
  echo "Have a good night!";
}
?>
```

(b) Compare pass by value and pass by reference in PHP with suitable examples.

Passing Parameters by Reference

By default, arguments passed to functions are **passed by value** in PHP. This means that PHP passes a copy of the variable so if the parameter is modified within the function, it does not change the original.

In the below example, notice that even though the function modifies the parameter value, the contents of the variable passed to the function remain unchanged after the function has been called.

```php
function changeParameter($arg) {
$arg += 300;
echo "arg=" . $arg;
}
$initial = 15;
echo "<br/>initial=" . $initial; changeParameter($initial);
echo "<br/>initial=" . $initial; // value not changed
```

Output:

initial=15 arg =315 initial=15

PHP allows arguments to functions to be **passed by reference**, which will allow a function to change the contents of a passed variable. A parameter passed by reference points the local variable to the same place as the original, so if the function changes it, the original variable is changed as well. The mechanism in PHP to specify that a parameter is passed by reference is to add an ampersand (&) symbol next to the parameter name in the function definition.

Eg:

```php
function changeParameter(&$arg) {
$arg += 300;
echo "arg=" . $arg;
}
$initial = 15;
        echo "<br/>initial=" . $initial;
changeParameter($initial);
echo "<br/>initial=" . $initial; // value changed
```

Output:

initial=15 arg =315 initial=315

The memory differences between pass-by value and pass-by reference is illustrated below.

Memory and Output

```
$initial=15;
```
`$initial | 15`

```
echo "initial=" . $initial;
```
`initial=15`

Functions

```
changeParameter($initial);
```
`$initial | 15`   `$arg | 15`

```
// passing by value
function changeParameter($arg) {
    $arg += 300;
}
```

`$arg | 315`

```
echo "initial=" . $initial;
```
`initial=15`

```
changeParameter($initial);
```
`$initial | 15`   `$arg | 15`

```
// passing by reference
function changeParameter(&$arg) {
    $arg += 300;
}
```

`$initial | 315`   `$arg | 315`

```
echo "initial=" . $initial;
```
`initial=315`
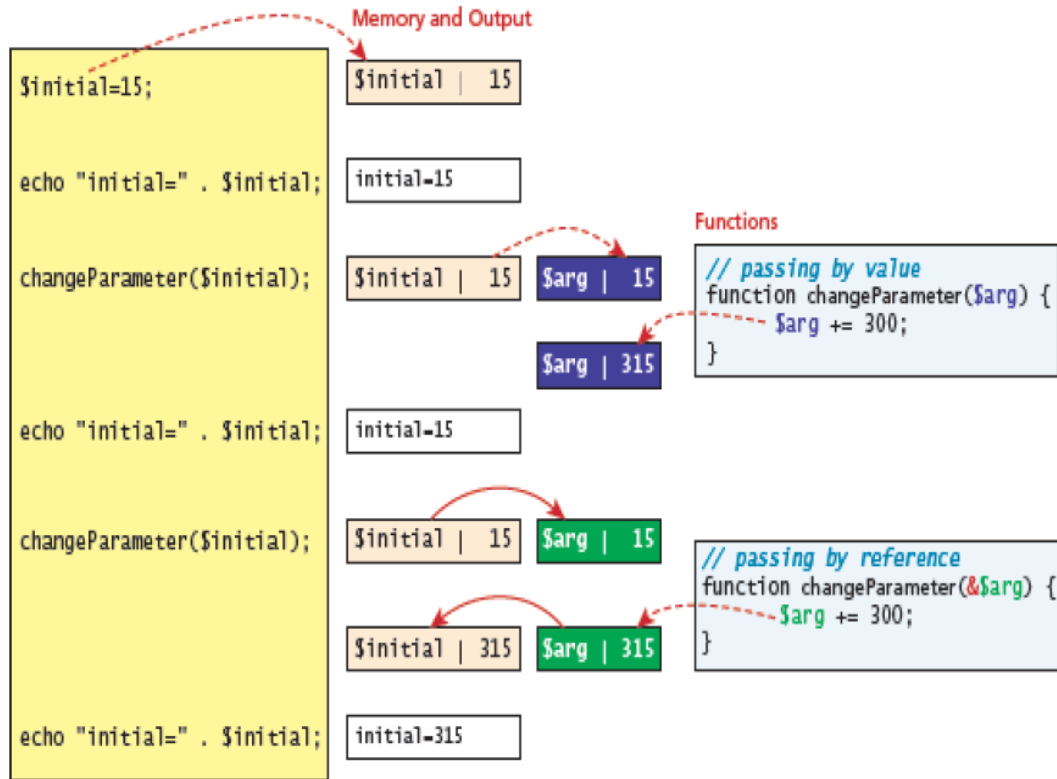
6. Explain A,M and P components of a LAMP stack.

LAMP stack

WAMP, MAMP, ...

You will be using the LAMP software stack

- **L**inux operating system

- **A**pache web server

- **M**ySQL DBMS

- **P**HP scripting language

Apache and Linux

LA

Consider the **Apache** web server as the intermediary that interprets HTTP requests that arrive through a network port and decides how to handle the request, which often requires working in conjunction with PHP.
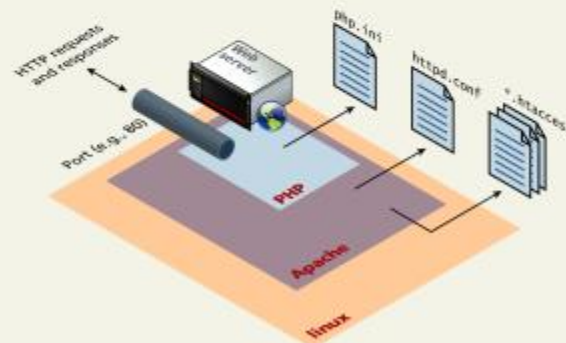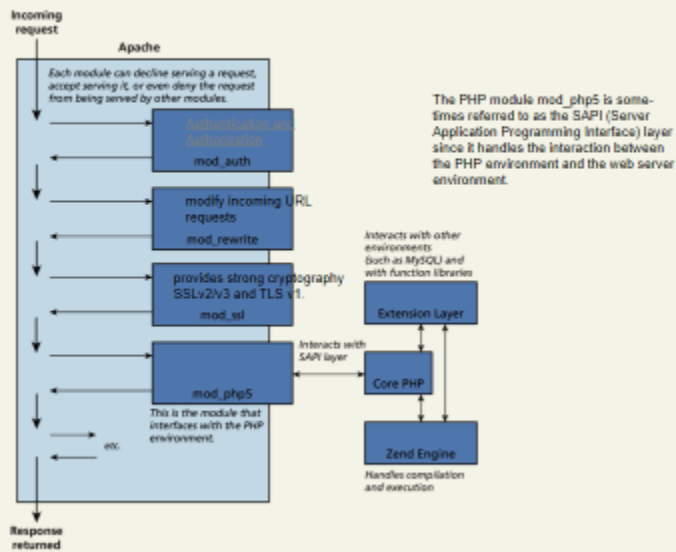
Apache
Continued

Apache runs as a daemon on the server. A **daemon** is an executing instance of a program (also called a **process**) that runs in the background, waiting for a specific event that will activate it.

When a request arrives, Apache then uses modules to determine how to respond to the request.

In Apache, a **module** is a compiled extension (usually written in the C programming language) to Apache that helps it *handle* requests. For this reason, these modules are also sometimes referred to as **handlers**.

Apache and PHP
PHP Module in Apache

Apache Threads

Multi-thread and multi-process

Apache runs in two possible modes:

- **multi-process** (also called **preforked**). the term fork refers to the operating system creating a copy of an already running process. Since forking is time intensive, Apache will prefork a set number of additional processes in advance of their being needed.

  - key advantage of multi-processing mode is that each process is insulated from other processes;

- **multi-threaded** (also called **worker**) a smaller number of Apache processes are forked. Each of the processes runs multiple threads. A thread is like a light-weight process that is contained within an operating system process. A thread uses less memory than a process, and typically threads share memory and code; as a consequence, the multi-threaded mode typically scales better to large loads.

  - All modules running within Apache have to be thread-safe. Unfortunately, not every PHP module is thread-safe, and the thread safety of PHP in general is quite disputed.
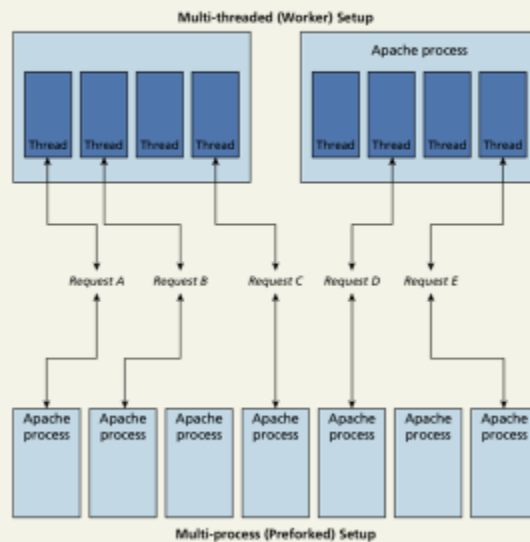
The default installation of Apache runs using the multi-process mode.

Apache Threads

Multi-thread and multi-process

PHP Internals

PHP itself is written in C
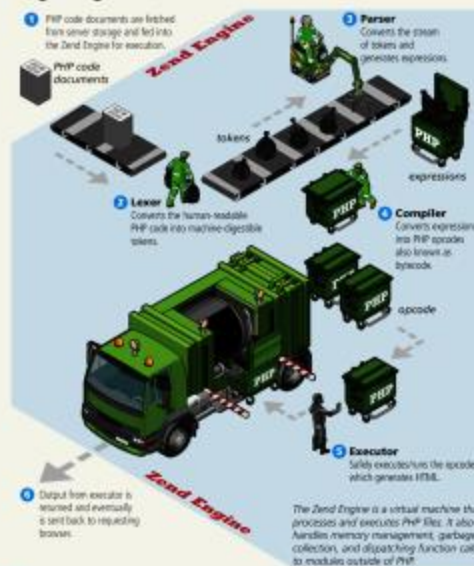
There are 3 main modules

1. **PHP core**. The Core module defines the main features of the PHP
    environment, including essential functions for variable handling,
    arrays, strings, classes, math, and other core features.

2. **Extension layer**. This module defines functions for interacting
    with services outside of PHP. This includes libraries for MySQL,
    FTP, SOAP web services, and XML processing, among others.

3. **Zend Engine**. This module handles the reading in of a requested
    PHP file, compiling it, and executing it.

Zend Engine

No, your code is not garbage.