

CBCS SCHEME

USN

--	--	--	--	--	--	--	--	--	--

Seventh Semester B.E. Degree Examination, July/August 2021 Web Technology and It's Applications

Time: 3 hrs.

Max. Marks: 100

Note: Answer any FIVE full questions.

1. a. ✓ What are HTML Elements and Attributes? Explain. (06 Marks) 14
b. ✓ Explain the different Relative Link Type Referencing with suitable example. (08 Marks)
c. ✓ What is CSS? List and explain benefits of CSS. (06 Marks)
2. a. Explain ordered and unordered list with example. (05 Marks)
b. With an example explain CSS Box model. (08 Marks)
c. ✓ List the different selectors available in CSS and explain in detail. (07 Marks)
3. a. ✓ Create a table that correctly uses the caption, thead, tfoot and tbody elements. Briefly discuss the role of each of these elements. (10 Marks)
b. ✓ What is a responsive design? Why it is important? (05 Marks)
c. ✓ Explain how rowspan and colspan attributes are used? (05 Marks)
4. a. Describe how block level elements are different from inline elements. Be sure to describe any two different types of inline elements with simple example. (10 Marks)
b. In what situations would you use a radio button and a checkbox? With an example explain briefly. (05 Marks)
c. Explain the role of CSS preprocessors in the web development workflow. (05 Marks)
5. a. Define software layer? Explain the various common software design layers in Javascript with a neat diagram. (08 Marks)
b. What are form events in Javascript? List and explain different form events. (05 Marks)
c. Demonstrate the use of inline, external and embedded Javascript with an example for each. (07 Marks)
6. a. What is Fail-Safe design and why does it matters? (04 Marks)
b. Explain Document Object Model. Demonstrate the DOM tree with an example. (08 Marks)
c. What are server-side include files? Why are they important in PHP? (08 Marks)
7. a. What are the superglobal arrays in PHP? What function is used to determine if a value was sent via Query string? (10 Marks)
b. How do you read or write a file on the server from PHP? Explain with suitable example. (10 Marks)

- 8 a. Define Class and Object. Interpret the concept of data encapsulation, Inheritance, Polymorphism and Object interface with respect to OOP. (10 Marks)
b. Explain `_construct()` and `_destruct()` with example or each. (10 Marks)
- 9 a. What are HTTP Cookies? How do you handle them in PHP? (08 Marks)
b. Why is state is a problem for web application? Explain. (08 Marks)
c. What does `$()` shorthand stand for in jQuery? (04 Marks)
- 10 a. Explain how sessions stored between requests. (05 Marks)
b. Write a jQuery selector to get all the `<P>` that contain the word "Hello". (05 Marks)
c. What are the commonly used animations in jQuery? Explain with suitable example. (10 Marks)

33

SUB: WEB TECHNOLOGY & ITS APPLICATIONS [18CS63]

1.a) HTML Elements are called as Tags of a webpage. HTML documents are composed of textual content & HTML Elements. HTML element is more expensive term which includes a element name within angle brackets. [6M]

* HTML element is identified in document by tags. It consists of element name within angle brackets.

* HTML elements contain attributes. An attribute is a name = value pair that provides more information abt HTML element.

Ex:- `VJIT `
Element name attribute Hypertext closing Tag

`<body>` `<p>` This is some `` text ``.
child. `</p>` `<h1>` Title goes here `</h1>`.
sibling. `<div>` `<p>` This is `` important ``.
 `</p>` `</div>`.
Ancestor. `</body>`

b) The different Relative link Type References are-

* Same Directory :-

``

* child directory :

``

- * Grandchild Directory
``
- * Parent Directory
``
- * Sibling Directory
``
- * Root Reference
``
- * Default Document.
``

c) Cascading Style Sheets is a W3C standard for describing the appearance of HTML elements. It can be added directly to HTML element within `<head>` element or a separate text file contain only CSS.

[6m]

Benefits of CSS :-

- * Improved Control over formatting :-
 CSS gives web authors to fine grained control over the appearance of their web content.
- * Improved Site maintainability :-
 It allows us to make site-wide visual modifications by changing a single file
- * Improved Accessibility :-
 CSS driven sites are more accessible.
- * Improved page download speed :-
 It allows to be quicker to download coz of size of file.
- * Improved output flexibility :-
 It can be used to adopt a page for different output media.

2. a) Ordered List :-

Collections of items have set of order, it has [SM] default values as in browser is a numbered list.

```
<OL>
  <LI> Introduction </LI>
  <LI> Background </LI>
  <LI> My Solution </LI>
  <LI> <OL>
    <LI> Methodology </LI>
    <LI> Results </LI>
    <LI> Discussion </LI>
  </OL>
  <LI> Conclusion </LI>
</OL>
```

o/r.

1. Introduction
2. Background
3. My Solution
 1. Methodology
 2. Results
 3. Discussion
4. Conclusion

* Unordered List :-

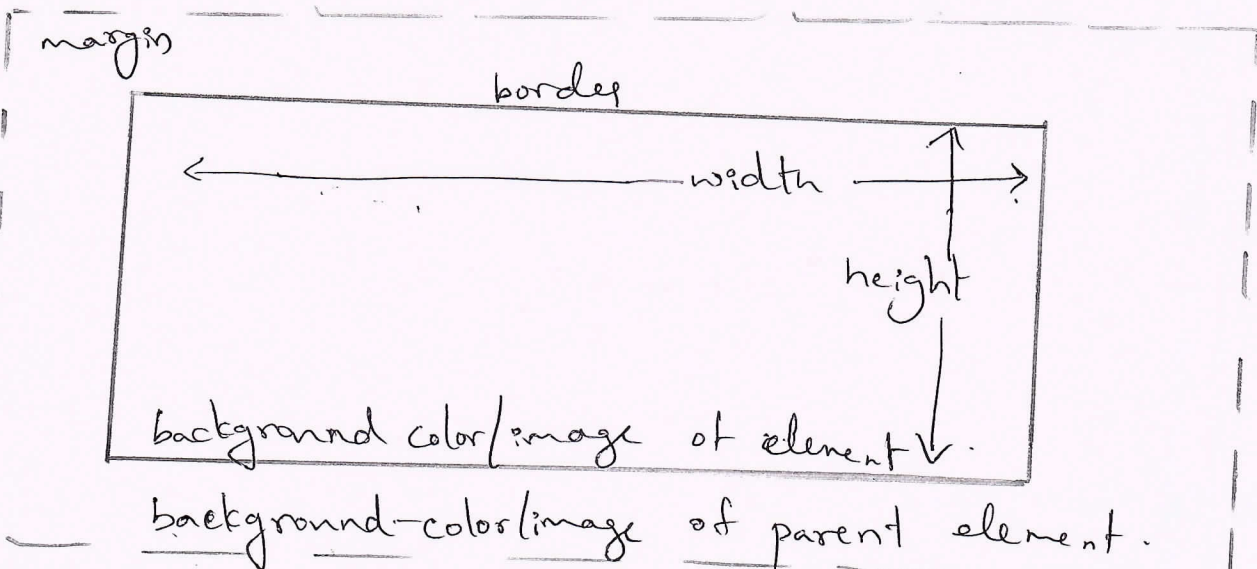
Collections of items in no-particular order, ∴ default rendered by browser is bulleted list.

```
<UL>
  <LI> <a href="index.html"> Home </a> </LI>
  <LI> About us </LI>
  <LI> Products </LI>
  <LI> Contact us </LI>
</UL>
```

output

- Home
- About us
- Products
- Contact us

b) All HTML elements exist within an Element Box.



[SM]

* There are different background properties :-
- background, background-attachment, background-color, background-image, background-size, background-repeat.

* Borders :- It provide a way to visually separate elements. We can put 4 sides of an element.

eg:- body { background: white url(../images/logo.gif) no-repeat; background-position: 300px 50px; border-top-color: red; }

* Margins & Padding :- These are essential properties for adding white space to web page, which can differentiate one element from another.

eg:- p { border: solid 4pt teal; margin: 5px; width: 200px; padding: 10px; height: 100px; }

e) The different selectors are :-

* Element Selectors :- selects all instances of given HTML elements. [EM]

eg:- p, div, aside { margin: 0; padding: 0; }

p { margin: 10px; padding: 10px; }

* class Selectors :-

Allows us to target different HTML elements regardless of their position in document tree.

eg:- <style>

first { font-style: italic; }

</style> } color: red;

<p class="first">

----- </p>

<h1 class="first">-----</h1>

* ID Selectors :-

Allows to target a specific element by its id attribute of its type or position.

* It can have the form.
with pound (#) followed by
id name.

eg:- `<style #com {
color: red; }
</style>
<div id="com">
<p>Camp on HDR buddy
</p> </div>`

* Attribute Selectors :-

It provides a way to select HTML elements either
by presence of an element attribute.

eg:- `<style
[title] { cursor: help;
padding-bottom: 3px;
border-bottom: 2px dotted blue;
</style> text-decoration: none;`

`<div>
----- </div>`

* Pseudo-Element & Pseudo-class Selectors :-

It is a way to select explicitly as an element in
HTML document tree. Pseudo-class selector to variety
of family relationships.

eg:- `a:link { text-decoration: underline;
color: blue; }
a:visited { text-decoration: none;
font-weight: bold; }
a:active { background-color: yellow; }`

* Contextual Selectors :-

It allows to select elements based on their
ancestors, descendants or siblings.

` Canada
 Germany
<div id="man"> ----- </div>
<div> <p> By <time> Oct 1 2015 </time> </p> </div>`

3.07) <Caption> element is used to add the heading of the table. <thead> ⇒ Table header potentially include other <tr> elements. <tfoot> which can be added before the body. <tbody> ⇒ which display the actual content of a web page. [10M]

* <table>

<caption> French Paintings </caption>

<thead> <tr> <th> Title </th>

<th> Artist </th> <th> Year </th> </tr>

</thead>

<tfoot> <tr> <td colspan="2"> Total No. of painting </td>
<td> 2 </td> </tr> </tfoot>

<tbody> <tr> <td> Death of Marat </td>

<td> Jacques - David </td>

<td> 1793 </td> </tr>

</tbody>. </table>. ----- // create two or more rows.

b). The responsive design, which responds to changes in the browser size that go beyond the width [5M] scaling of a liquid layout.

* One of the main problems of a liquid layout is that images & horizontal navigation elements tend to be fixed size & when the browser window shrinks to size of a mobile browser, liquid layouts can become unusable.

* So In responsive design, images will be scaled down & navigation elements will be replaced as browser shrinks. ∴ Responsive Design is very important.

c) The rowspan & colspan can be used in a element attribute. These are 2 things about tables.

* First is all content must appear within <td> or <th> container. Second is each row must have same no. of <td> or <th> container. [5M]

eg:- <table> <tr> <th> Title </th> <th> Artist </th>
<th> Year </th> <th colspan="2"> Size </th> </tr>
<tr> <td> Death of Mozart </td> <td> David </td>
<td> 1793 </td> <td> 162cm </td> <td> 128cm </td>
</tr> - - - </table>.

4.a) Block-level elements such as <p>, <div>, <h2> & <table> are each contained on their own line. Because [10M]
block-level elements begin with a line break, without styling, two block-level elements can't exist on same line.

* It use CSS box model, with margin, paddings, background colors & borders.

* Inline-elements don't form their own blocks but are displayed within lines. Normal text in an HTML document is inline, as are elements such as , <a>, & .

* There are two different types of inline elements:-

→ Replaced inline Elements. → Nonreplaced "

eg:- <p> This photo of Pond in Central Park, New York city was taken on ^{time} Oct 22 2015. </p> </p>.

& functions.

* A CSS preprocessor which takes code written in some type of preprocessed language & converts that code into normal CSS. [SM]

* One of the best way to use preprocessor of a CSS is with colors.

* CSS preprocessors such as LESS, SASS & Stylus provide these type of functionality.

eg:-

```
$colorSchemeA: #796d6d;
```

```
footer { background-color: $colorSchemeA; }
```

SASS Preprocessor

```
footer { background-color: #796d6d; }
```

S.a) Software layer is a way of conceptually grouping programming classes which has similar functionality & dependencies.

Common s/w design layer names are:-

* Presentation Layer:-

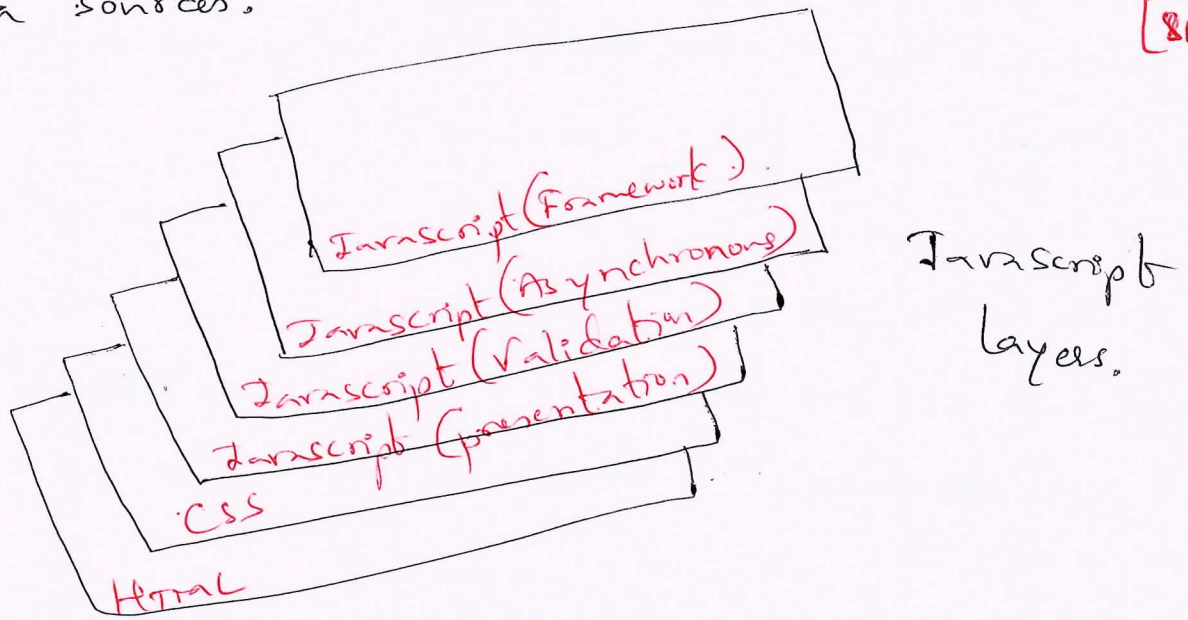
classes focused on the user interface. It includes creating, hiding & showing divs, using tabs to show multiple views or arrows to page through result sets.

* Business Layer:-

classes that model real world entities such as customers, products & sales.

* Data layer :-

Classes that handle the interaction with data sources.



b) Form events: are the main which user input is collected & transmitted to the server.

The form events are :- onblur, onchange, onfocus, onreset, onselect & onsubmit.

onblur :- Form element has lost focus.

onchange :- some `<input>`, `<textarea>` or `<select>` field had their value change.

onfocus :- Complementing onblur event, this event is triggered to get focus.

onreset :- HTML forms have ability to be reset.

onselect :- When user selects some text.

onsubmit :- When form is submitted this event is triggered.

c) There are 3 different ways to link JS to HTML.

* Inline :- includes JS within certain HTML attributes.

eg:- `More `
`<input type = "button" value = "alert ('Are you ready')" />`

* Embedded JS :- It refers to placing of JS code within `<script>` element. Like inline JS, embedded scripts can be difficult to maintain. [3M]

eg:- `<script type="text/javascript">`
`alert("Hello Fondz!!");`
`</script>`

* External JS :- Externally we can include the JS code in HTML. By convention, JS external files have extension `.js`.

eg:- `<head>`
`<script type="text/javascript" src="greet.js">`
`</script>`
`</head>`

6.a) The approach of adding functional replacements for ~~the~~ web application without Javascript is also referred to as FAIL-SAFE DESIGN. [4M]

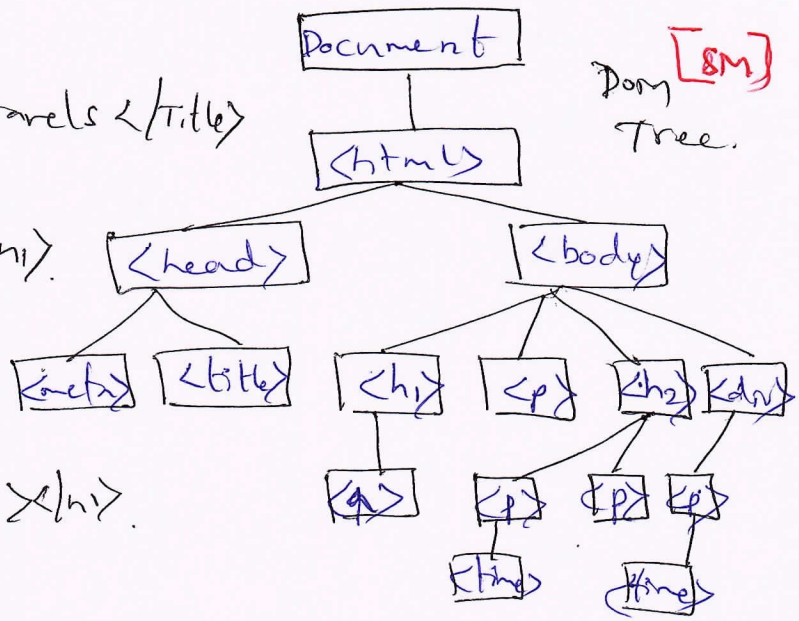
* It is a phrase with a meaning beyond the web development. It means when a plan fails then the system's design will still work properly.

b) The different way of programmatically accessing the elements & attributes within the HTML.

This is accomplished through a programming interface called DOCUMENT OBJECT MODEL (DOM).

```

eg :- <html> <head>
      <meta> <title> Travels </title>
      </head>
      <body> <h1> Your Travels </h1>
      <p> photo -----
      </p>
      
      <h2> Reviews </h2> <p> ----- </p> </h2>
      <div> <p> ----- </p>
      <p> ----- </p>
      </div> </body> </html>
  
```



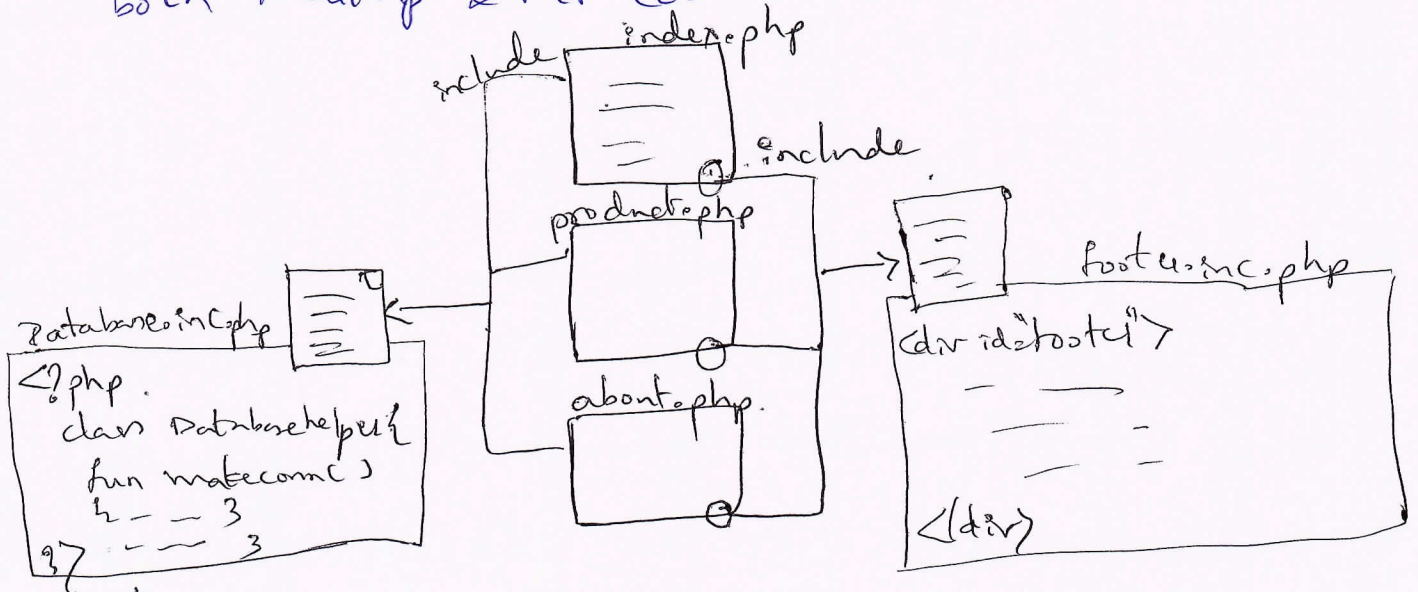
DOM [SM]
Tree.

Server-side include files are type of encapsulation, nothing but it is like copying & pasting by the server.

We can embed different types of files within the web application. [SM]

PHP does have one imp facility to insert or include content from one file into another.

Include files provide a mechanism for reusing both markup & PHP code.



- * If the included file doesn't exist or server doesn't have the permission to access, then the include files display an error msg & stops execution.
- * The process will not be halted, so it continues the execution of web application.

7a) PHP uses special predefined associative arrays called Superglobal Arrays, which allow the programme to easily access HTTP headers, query string parameters & commonly needed information.

* The Superglobal Arrays are -

[10m]

→ `$_GLOBAL`, `$_COOKIE`, `$_ENV`, `$_FILES`, `$_GET`, `$_POST`, `$_REQUEST`, `$_SESSION`, `$_SERVER`

* The `$_GET` & `$_POST` methods are used to determine the value sent via Query String.

* `isset()` function is used to determine the value sent via query string.

```

<?php
if($_SERVER["REQUEST_METHOD"] == "POST") {
    if(isset($_POST["name"])) {
        echo "User login Details:";
        echo " — here we could redirect or
              authenticate";
        echo " & hide login form or something
              else";
    }
}

```

?? 3.

b) There are two basic techniques for read/writing files in C++.

* Stream Access :- Read just a small portion of file at a time. It is most-memory-efficient approach when reading very large files. [Com]

* The functions `fopen()`, `fclose()` & `fgets()` are used to access the content of file.

* To write data to file by using `fwrite()` in which the same as `fgets()` passing the file handle & string to write.

eg:-

```
$f = fopen("sample.txt", "r");
$ln = 0; while($line = fgets($f)) {
    $ln++;
    printf("%d\n", $ln);
} echo $line - "\n";
fclose($f);
```

* In-memory File Access :- It is simple & easiest way to use to read/write file. There are 3 functions.

→ `file()` - Reads entire file into an array
→ `file_get_contents()` ⇒ Reads " " a string variable
→ `file_put_contents()` - write contents of a string " out to a file.

eg:-

```
$$sStr = file_get_contents(FILENAME);
```

// Read contents of file

`file_put_contents(FILENAME, $some);` // write contents of file.

8.a) Class which consists of properties & methods.
Each property in class using different access
Specifiers.

Eg- class Artist {
public firstName;
public lastName;
getName();

* Object is instantiation³ of a class. to make use
of properties & methods of a class using
new keyword.

\$a = new Artist();

* DATA ENCAPSULATION :-

which refers to restricting access to an object's
internal components. There are 2 methods for
accessing & modifying properties : Getter & Setter.

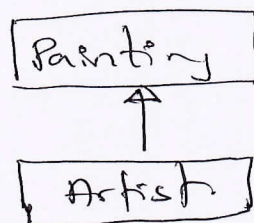
```
public function getName() { return $this->firstName; }  
public function setName($n)  
{ $this->firstName = $n; }
```

* INHERITANCE :-

A class inherit from another class is called
subclass or derived class. It inherits all of its
public & protected methods & properties.

eg:- class Painting extends Artist { ----- }

```
$p = new Painting;  
echo $p->getName();  
echo $p->setName("Poo");
```



* POLYMORPHISM :- The method have many forms of its definition. An object can be multiple things at the same time.

[10M]

```
$p = new Artist ("Pablo");  
$q = new Painting ("1937", $p); $r = new Sculpture ("Chicago");  
$p → addwork($q);  
$p → addwork($r);
```

* Object Interface :- It is way of defining a formal list of methods that a class must implement without specifying implementation.

```
interface viewable {  
    public function getSize();  
    public function getPNG();  
}
```

* Interface contains no properties & its methods do not have method bodies defined.

b> Constructors which specify parameters during instantiation to initialize properties within a class. [10M]

* In PHP constructors are defined as functions with the name --construct().

* In the constructor each parameter is assigned to an internal class variable using \$this → pro_name.

class Artist {

function __construct(\$fname, \$lname, \$city).

{ \$this → firstName = \$fname; \$this → city = \$city;

\$this → lastName = \$lname;

```
$p = new Artist("Fero", "Shaikh", "Gulbarga");  
$d = new Artist("Mohit", "Jain", "Mumbai");
```

* Destructor :-

* It is called when object is destructed or script is stopped or exited. Whenever we create a --destruct PHP will automatically call this function at the end of script.

```
eg:- public function __destruct()  
{  
    echo "Artist name: " . $this->name;  
}
```

Q.2) HTTP Cookies are a client-side approach for persisting state information. ~~While~~ The cookie information is stored and retrieved by browser information in a cookie travel within the HTTP header. [8M]

* HTTP Cookies are handled in PHP by two ways :-

⇒ Browser will delete a cookie which are beyond their expiry date.

⇒ If the cookie doesn't have a expiry date specified the browser will delete it when the browser closes.

* Cookies can be created & retrieved using

setcookie() function.

* If cookie has expired then client's browser would not send anything.

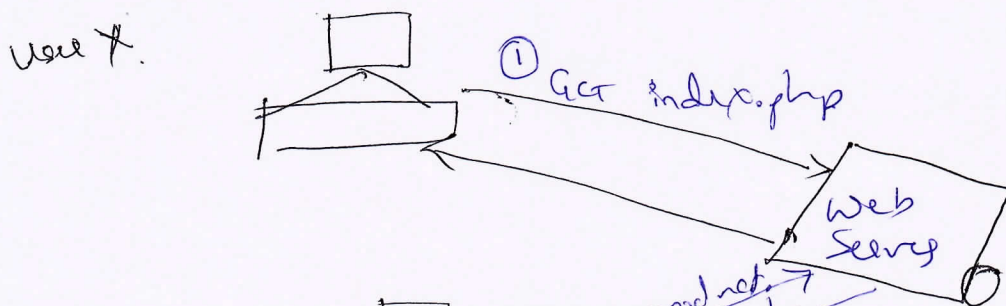
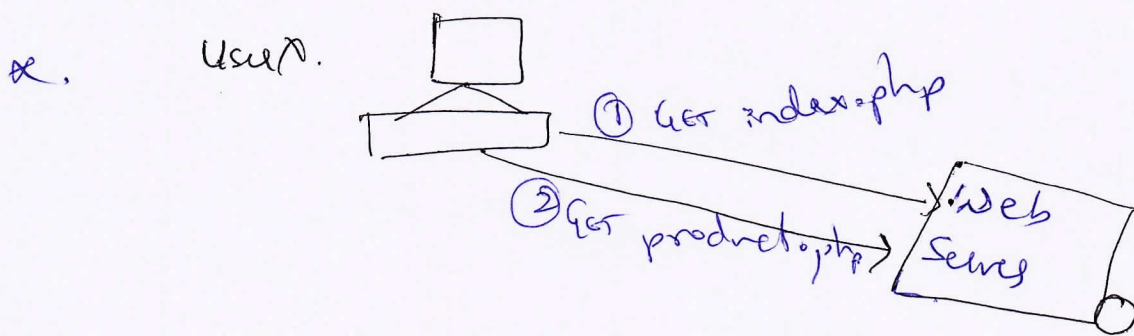
9.6) State is a problem for web application because of sharing information with another request/user.

* All web applications need to be processed with user inputs, output information & read/write from databases or other storage media.

* Single user - desktop applications do not have problems to access the information stored in memory.

* Web application consists of series of disconnected HTTP requests to a web server where each request for a server page is request to run separate program.

* HTTP protocol doesn't distinguish two requests by one src from two requests from two different sources.

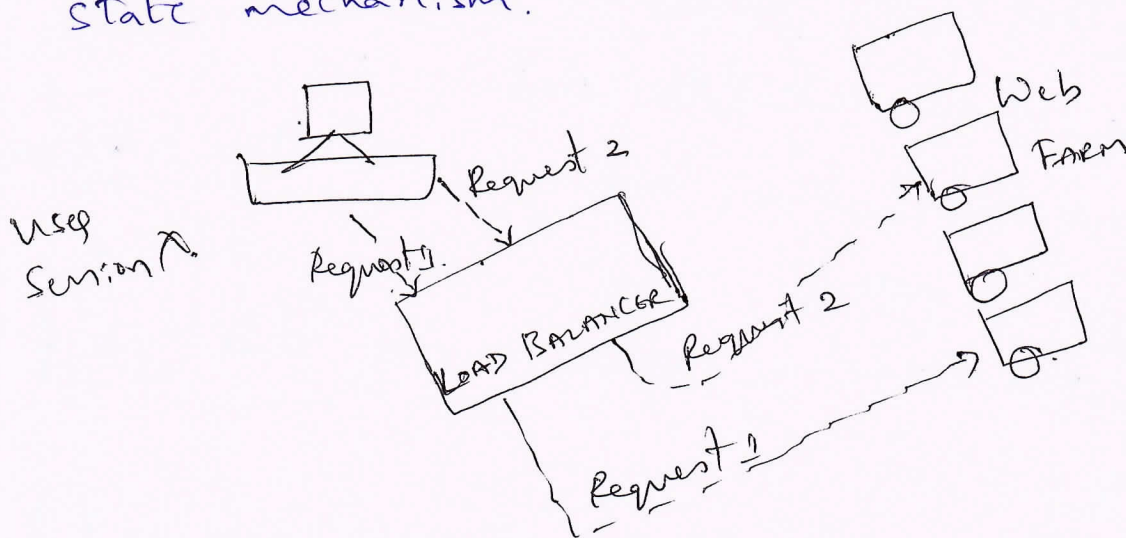


c) $\$() \rightarrow$ can be confusing to PHP developers, at first because the $\$$ is used for variable declaration. [4M]
 * It selects DOM object which match CSS attributes.
 * Pass in the string of CSS selector to $\$()$ & result will be set of DOM objects matching the selectors.

* Eg:- $\$(" * ")$, $\$(" tag ")$, $\$(" . class ")$, $\$(" # id ")$;

10.a) The sessions are stored between requests in 2 different ways:- [5M]

* Configuring the load balance to be "Session aware" & relate all requests using session to same server.
 * Use a shared location to store session either in a database, memcache or some other shared session state mechanism.



b) jquery to select all the $\langle p \rangle$ that contains word "Hello". [5M]

var all = $\$(" p * : contains ('Hello') . ")$;

c) Animation used commonly in jQuery are :-
* hide() & show(). * fadeIn & fadeOut()
* SlideUp() & SlideDown() * Toggle.

* hide() & show() methods can be called with no arguments to perform a default animation. The new version takes 2 parameters: duration & callback method to execute on completion.

* FadeIn() & FadeOut() :- which used to control the opacity of an element. It takes two parameters as duration & callback method.

* SlideUp/SlideDown() :- These method don't touch the opacity of an element rather gradually change its height.

* Toggle Methods :- jQuery has toggle method to get the visible & hidden states.

* Toggle methods between fading in & fading out use fadeToggle() method, toggling between the two sliding states can be achieved using slideToggle().

eg) - \$(this).hide();
\$(this).html("Visit ");
\$(this).show(1000);

Verifed by
Prof. K. Ramesh



Dean, Academics.

Prepared By :-
Prof. K. Ramesh, N, N.