

KONGU ENGINEERING COLLEGE, PERUNDURAI 638 060  
CONTINUOUS ASSESSMENT TEST - 3  
Regulations 2014

Month and Year : October 2019		Roll Number :
Programme	: B.E/B.Tech	Date : 11.10.2019
Branch	: CSE/IT	Time : 9.15 AM to 10.45 AM
Semester	: V	
Course Code	: 14ITT52	Duration : 1 ½ Hours
Course Name	: Web Technology	Max. Marks : 50

PART - A (10 × 2 = 20 Marks)  
ANSWER ALL THE QUESTIONS

- Distinguish between traditional web application and Ajax application. [CO4,K2]
- Indicate the responsibilities of XMLHttpRequest object. [CO4,K2]
- Mention the different status code of XMLHttpRequest object along with its description. [CO4,K1]
- Identify the role of web servers and name any two web servers. [CO4,K1]
- Interpret preg\_match () and preg\_replace () methods. [CO5,K2]
- Apply regular expression pattern to display the string containing letter 'A' in the given sentence String s1= "Doctor says that Apple is good for health". [CO5,K3]
- Write a regular expression to check whether the rollnumber given by the user is in the following format: 17CSR001. [CO5,K3]
- Define cookies. [CO5,K1]
- Write a PHP script to initialize and display session variables. [CO5,K3]
- Host machine can act as local web servers. Justify your answer. [CO4,K2]

PART – B (3 × 10 = 30 Marks)  
ANSWER ANY THREE QUESTIONS

- Explain the multitier architecture of a web application with a neat sketch. (10) [CO4,K2]
- Consider that you are developing an online application to maintain the following information about various training institutes
  - ❖ Name of the institutes
  - ❖ Address
  - ❖ Telephone NO
  - ❖ Email-ID of the manager
  - ❖ Official website address
  - ❖ Brief description about the institute
 The above information is available inside a database table named "Institutes". Write appropriate AJAX & PHP code to fetch the above information from the database and display the same in a table. Provide links to the website of the institutes. (6) [CO3,K3]
- a) Apply regular expression in PHP to generate appropriate pattern and validate the following
  - ❖ Email\_id(validemail-Id: xyz\_123@gmail.com,  
AB12xy@xyz.edu,  
ABC12@xyz.ac.in,  
ABC@xyz.co.in)
  - ❖ MAC address of a device : (valid MAC address:00:0A:83:B1:C0:8E,  
0A1:C5A:74:E3:58:2F1, 35:447:5:A:26:7C)
 b) Consider the string "PHP is a server side scripting language". Write a PHP script using regular expression to perform the following tasks (4)
  - Find the words that starts with letter 'S' in the above string and display the same.
  - Replace the white spaces with the character '\*' and display the resultant string.
- What is session tracking? Name any three session variables. Create a login page with user name and email as fields. Perform the following task for session tracking (10) [CO5,K3]
  - Set name and email as session variable
  - Check whether the session variable is set and display the value of session variables
  - Destroy the session variable.

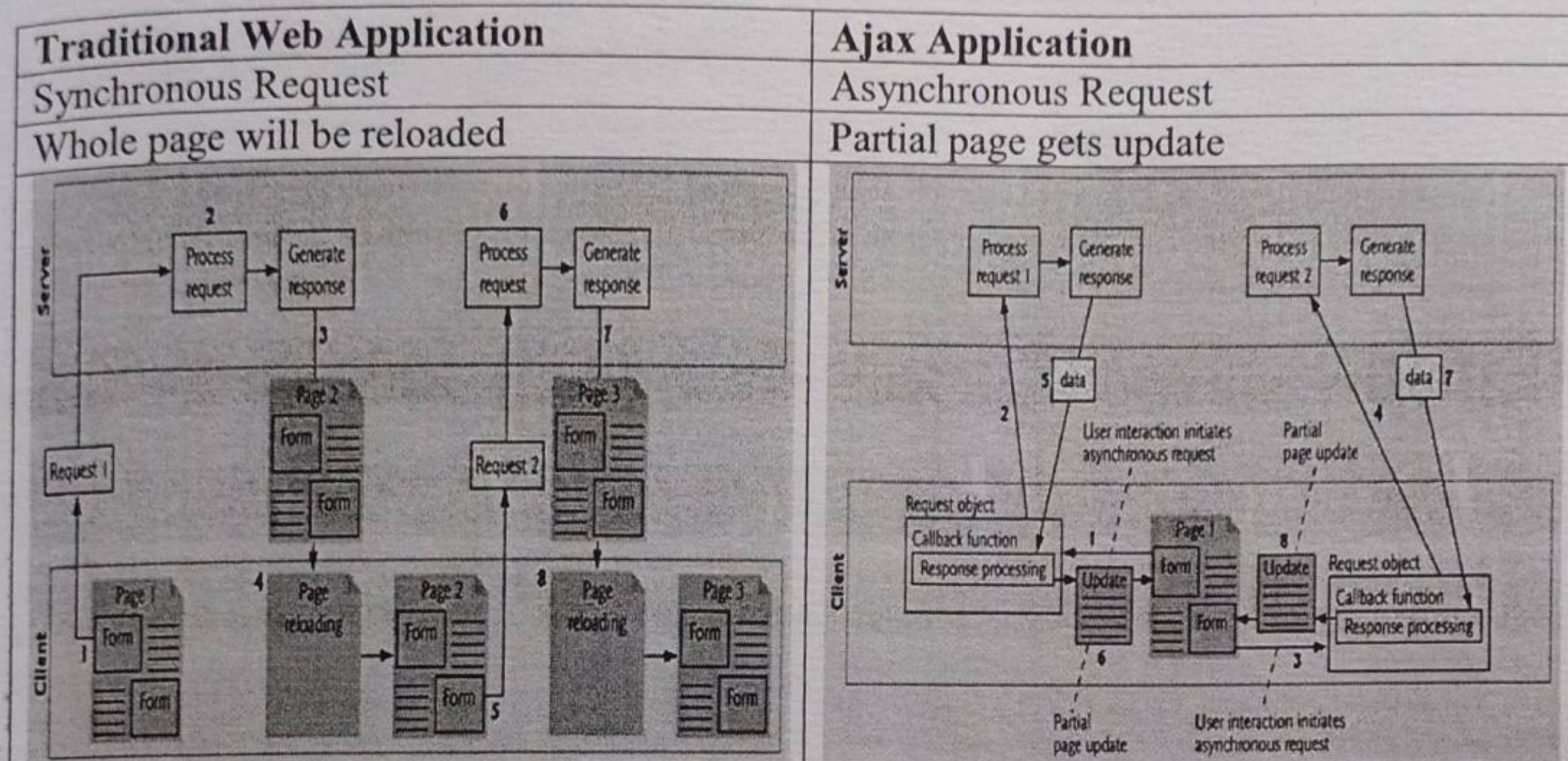
Bloom's Taxonomy Level	Remembering (K1)	Understanding (K2)	Applying (K3)	Analysing (K4)	Evaluating (K5)	Creating (K6)
Percentage	10	30	60	--	--	--

## Web Technology

### Cat3 Answer Key

#### Part A

1.



2. XMLHttpRequest object is used to manage a request. It sends the request to and awaits the response from the server. When the server responds the XMLHttpRequest object that issues the request invokes a callback function to update the page.

3.

Status Code	Description
200	Request was successful
404	Requested resource was not found
500	Error

4. The primary function of a web server is to store, process and deliver web pages to clients

Two Web Server

- Apache HTTP Server
- Microsoft's Internet Information Services Express

5. **preg\_match** – this function is used to perform a pattern match on a string. It returns true if a match is found and false if a match is not found.

**preg\_replace** – this function is used to perform a pattern match on a string and then replace the match with the specified text.

6. `preg_match( "/ [a-zA-Z]*A[a-zA-Z]*/", $s1,$out)`

7. preg\_match("[0-9]{2}[a-zA-Z]{3}[0-9]{2,3}", \$s1, \$out1)

8. A cookie is a small file that the server embeds on the user's computer. Cookies are kept for tracking purposes. A cookie is often used to identify a user.

9. <?php

```
// start a session  
session_start();
```

```
// initialize session variables
```

```
$SESSION['id'] = '1';  
$SESSION['name'] = 'Tutsplus';
```

```
// access session variables
```

```
echo $SESSION['id'];  
echo $SESSION['name'];
```

```
?>
```

10. Local web servers can be accessed through the name "localhost" which refers to the local machine. The name "localhost" is transferred to the IP address 127.0.0.1

## Part- B

### 11. Explanation

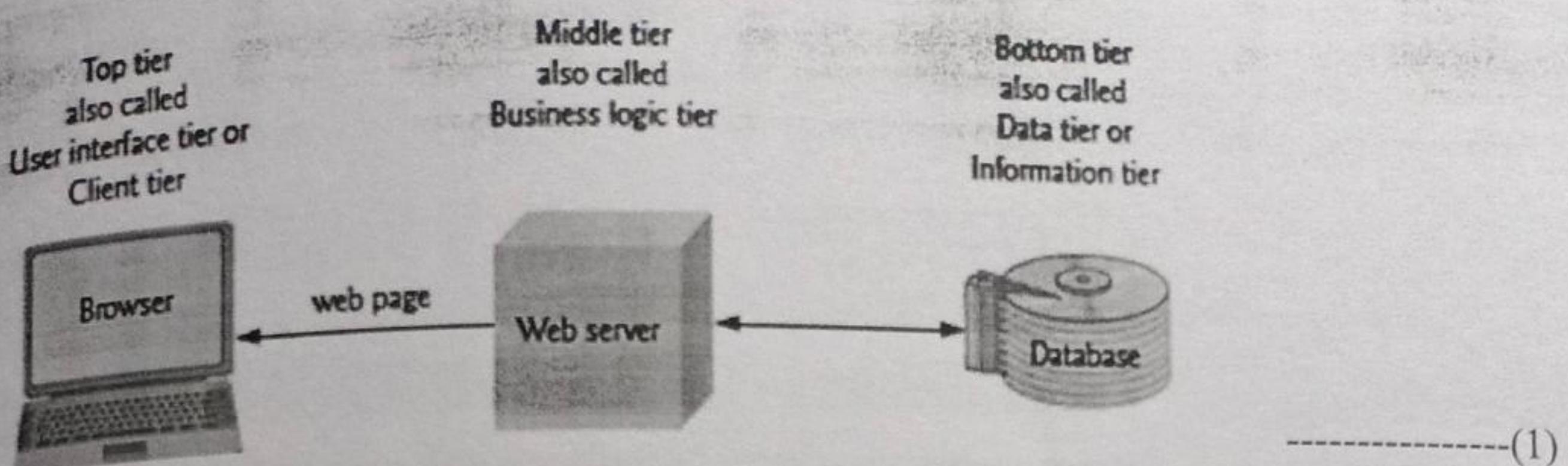
Divides the functionality of web-based applications into separate **tiers** (i.e., logical groupings of functionality) and these tiers can exist on separate computers

The **bottom tier** (also called the data tier or the information tier) maintains the application's data. This tier typically stores data in a relational database management system (RDBMS) - (3)

The **middle tier** implements business logic, controller logic and presentation logic to control interactions between the application's clients and its data.

- **Controller logic** processes client requests and retrieves data from the database
- **Presentation logic** then processes data from the information tier and presents the content to the client.
- **Business logic** enforces **business rules** and ensures that data is reliable before the application updates a database or presents data to users. ----- (3)
- 

The **top tier**, or client tier, is the application's user interface (a web browser or a mobile device), which gathers input and displays output ----- (3)



12. Creation of application form using different html tags ----- (3)  
 PHP and AJAX code to fetch the details----- (5)  
 Creation of link to websites----- (1)  
 Output----- (1)

13. Email:

`preg_match("/^([a-zA-Z0-9\+\_\-\-]+)(\.[a-zA-Z0-9\+\_\-\-]+)*@[([a-zA-Z0-9\-\-]+\.)+[a-zA-Z]{2,6}\$/ix", $str) - (3)`

Mac Address `^[a-fA-F0-9:]{17}:[a-fA-F0-9]{12}\$` ----- (3)

Display the words that start with s:

```
while (preg_match( "/^b(s|[alpha:]\+)\b/", $search, $match ) )
{
    print( $match[ 1 ] . " " );
    $search = preg_replace( "/" . $match[ 1 ] . "/", "", $search);----- (3)
}

str_replace( " ", "*", $arr, $i) ----- (1)
```

14. Session tracking, session variables and task using session tracking

A way to store user information (in variables) to be used across multiple pages  
 Session tracking – Definition ----- (2)  
 Session Variables : `session_start(); session_destroy(); session_unset();` ----- (1)  
 Creation of login page using Appropriate tags and event handling mechanism----- (2)

Program for setting , displaying and destroying session variables

```
<html>
<body>
<?php
// Set session variables ----- (2)
$_SESSION["Name"] = $_POST["UserName"];
$_SESSION["email"] = $_POST["Mail"];
echo "Session variables are set.";
?>
</body>
</html>
// displaying session variables ----- (2)
echo "Name " . $_SESSION["Name"] . "<br>";
```

```
echo "Email" . $_SESSION["email"] . ".";
// destroy the session -----
session_destroy();-----(1)
?>
</body>
</html>
```

18.10.2019  
Date

18.10.2019  
Date

Name and signature of Hall Supdt. with Date



# KONGU ENGINEERING COLLEGE

PERUNDURAI ERODE - 638 060.

(Autonomous)



Name of the Student	M. AKASH	Register No.	I	T	C	S	R	O	D	O	9
Programme	B.E	Branch & Semester	CSE	V							
Course Code and Name	141TT52 WEB TECHNOLOGY	Date	11.10.19	No. of Pages Used							

## MARKS TO BE FILLED IN BY THE EXAMINER

PART - A		PART - B		Grand Total Max. Marks : 50
Question No.	Max Marks : 2	Question No.	Max Marks : 10	
1	✓	11	i) 9	
2	✓	ii)		
3	✓	12	i) 10	
4	✓	ii)		
5	✓	13	i)	
6	✓	ii)		
7	✓	14	i) 10	49
8	✓	ii)		
9	✓			
10	✓			
<b>TOTAL</b>	<b>20</b>	<b>TOTAL</b>	<b>29</b>	

Total marks in words : **Four Nine**

### INSTRUCTION TO THE CANDIDATE

- Check the Question Paper, Programme, Course Code, Branch Name etc., before answering the questions.
- Use both sides of the paper for answering questions.
- POSSESSION OF ANY INCRIMINATING MATERIAL AND MALPRACTICE OF ANY NATURE IS PUNISHABLE AS PER RULES.

Name of the Examiner

Signature of the Examiner with Date

## CONTINUOUS ASSESSMENT TEST - II

### NBB TECHNOLOGY

Answer the following.

#### 1. TRADITIONAL WEB APPLICATION

\* It is a client-server approach to retrieve and respond to request.

\* When requesting for new page source, it will retrieve from server and display it in new page removing old contents.

\* All Browsers support this traditional web application.

#### AJAX APPLICATION

\* It is a XMLHttpRequest method for retrieving and responding for request.

\* AJAX requests and respond to a client and retrieve data in the already displaying page itself.

\* Browsers with newer versions support AJAX, for old browsers Active XMLHttpRequest is used.

#### 2. XMLHttpRequest Object:

\* Creates Request object when the object is created and provided the new URL in its open() method.

\* Processes the given Request in the web server and checks its Response.

\* Retrieve the Response from the Server and display the Response within the page where user has given its location.

### 3. STATUS CODE FOR XML HTTP REQUEST:

- 1) 200: "Response is ready i.e connected to share"
- 2) 403: "Forbidden" - problem on loading
- 3) 404: "Page not found"

### 4. WEB SERVERS:

Web Servers mainly process the client Request by validating the correct Request send by the client and process the response on database and retrieve its content and displays it to the user. Some famous Web servers are,

\* Apache

\* Microsoft IIS

### 5. preg-match(): matches certain string value with the given string, if it matches it returns true.

```
preg_match ("expression", $string);
```

preg-replace(): checks the given string with the condition, if it matches it will replace the first occurrence.

```
preg_replace ("expression", "replace string", given $string);
```

### b. <?php

```
$str = "Doctor says that Apple is good for health";
```

```
$s1 = preg-split ("\\s", " /\\s/", $str);
```

```
foreach ($s1 as $temp)
```

```
{ if (preg-match ("/[A]/", $temp))
```

```
echo "$temp";
```

```
?>
```

7. <?php

```
$roll = $_POST['rollno'];
if (!preg_match ("^\\d{1-9}\\d{2} [A-Z]{3}[0-9]{3}$", $roll))
    echo "Valid Roll Number";
else
    echo "Invalid Roll Number";
?>
```

8.

Cookies are small data which stores on the side data storage. Cookies helps to get information whenever page is loaded. Cookies retrieves and manages to display stored data in cookies are small it contains max capacity of 4kb. Login pages, cookies are used as when user again went to login page, cookies helps to give the necessary user name and password easily. Cookies will not destroy. Cookies are started automatically.

```
setcookie ("username", $name, time() + 3600) // lasts for 60 seconds.
```

9. <?php

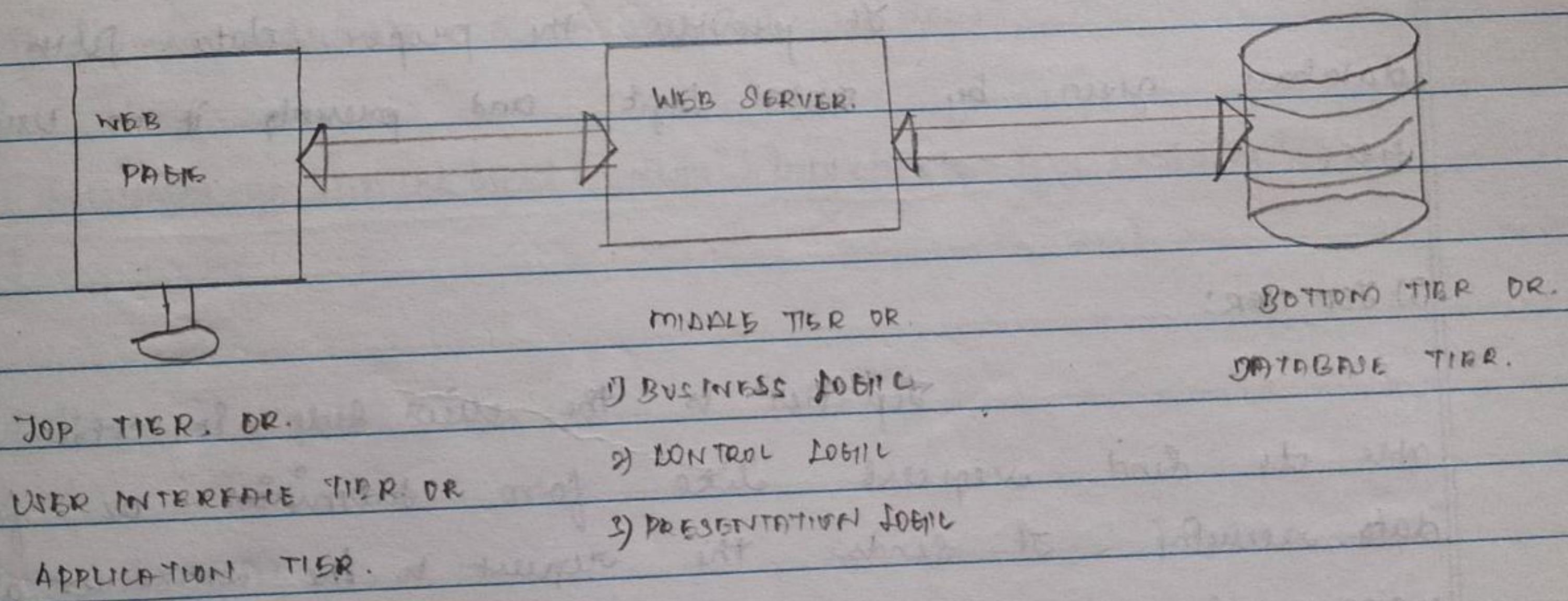
```
session_start();
$_SESSION['id'] = $_POST['name'];
if (isset ($_SESSION['id']))
    echo "Welcome ". $_SESSION['id'];
else
    echo "Session NOT Initialized";
session_destroy();
```

10. Host machines can also have web servers like Apache, XAMP, Tomcat etc... which can able to manage the register database if the computer itself can act as a Local Host. When the webserver and database are in same system, webserver use the database as localhost.

1. Answer the following.

2. Multi Tier Architecture:

Multi-Tier architecture in web server defines that the client and server can be on different range and web servers in that time can act upon the process. Multi-tier architecture can be improvise the user html page and improves the speed and quality.



1) Bottom - TIER:

Bottom tier is also known as database tier. It contains the particular page information which can able to process the request. Bottom tier stores all the necessary data for the web page.

## 2) MIDDLE-TIER:

Middle Tier contains the web services which handle the response and request of the client and server.

### CONTROL LOGIC:

Control Logic provides on fetching the data from the bottom tier i.e. retrieving the data from database based on response given.

### BUSINESS LOGIC:

Business Logic performs the intermediate analysis checks the appropriate request from the client which can be valid or not and provides the response part to control tier.

### PRESENTATION LOGIC:

It provides the proper data fetch from the database given by control logic and presents it in User Interface.

## 3) TOP TIER:

Top-tier is the client side on which client sends the client side request like form submission or further data entry. It sends the request to the web server and waits for the manager to provide it.

12. Edit main.html:

```
<html>
<head>
<title> Institution Information </title>
<script>
var any = new XMLHttpRequest();
function getcontent(url)
{
try {
    any.onreadystatechange = stateChange();
    any.open('GET', url, true);
    any.send(null);
} catch(exception) { alert("Problem");}
}
function stateChange()
{
if (any.readyState == 4 && any.status == 200)
{
    document.getElementById("demo").innerHTML = any.responseText;
}
}
</script>
</head>
<body> <center>
<h1> Details of Institutions </h1>
<button onclick="getcontent('info.php')"> Click Here </button>
<p id="demo" style="border: 1px solid black; color: black"> </p> </center>
</body>
</html>
```

db1.php :

```
<html>
<?php
$host = "localhost"; $user = ""; $password = "sample";
$qry = "Select * from Institute";
$result = mysqli_query($con, $qry);
if (mysqli_num_rows($result) > 0)
{
    while ($row = mysqli_fetch_array($result))
    {
        echo "<table><tr><th>% Name of the Institute </th><th> Address </th>
        phone no </th><th> Email of Manager </th><th> Website Address </th><th> Description </th>
        </tr><tr><td>" . $row['name'] . "</td><td>" . $row['Address'] . "</td><td>" .
        $row['mob'] . "</td><td>" . $row['email'] . "</td><td>" . $row['webs'] . "</td><td>" .
        $row['description'] . "</td></tr>";
    }
    echo "</table>";
}
else
{
    echo "No Rows";
}
?>
</html>
```

In database: attributes are: name, address, mob, email, webs, description

DETAILS OF INSTITUTION

CLICK HERE

{Before clicking}

After clicking :

DETAILS OF INSTITUTION

CLICK HERE

Name of Institution	Address	Telephone No.	Email of Manager	Website Address	Description
KONGU ENGINEERING COLLEGE	Parmudurai	0122-450789	abc@gmail.com	<u>www.kongu.ac.in</u>	Chennai college.

<link> method      Nefined as a Link.

#### 14. SESSION TRACKING:

Session Tracking in web page helps you to identify the user at any time when accessing the page. Session tracking is a server side program where we can maintain how many people access the page or who access the pages to database.

Session destroys automatically when the client closes the browser. When we want to use the data of who accessed the session. We can use database and when user login, we can save the login details in session at particular timing.

Session tracking is local means, the variables which can be utilized for storing data. Hence session tracking creates session and can pass through all the pages related to other pages can use this session.

Maximum of 128MB can be used in session. To start a session we use session\_start(); Session variables can be assigned as, \$SESSION['name'], \$SESSION['id'], \$SESSION['email'].  
Login.php:

```
<html>
<head>
<title> Session Program </title>
<?php
session_start();
if (isset($_REQUEST['METHOD']) == "post") {
    $SESSION['name'] = $_POST['name']; // Session Variable for Name
```

```

$_SESSION['email'] = $_POST['email']; // setting session variable for email
if (isset($_SESSION['name']) & & $_SESSION['email']) // checking whether session is set
    header("location: welcome.php");
else {echo "session Not set";}
}

?> </head>
<body> <h1> Login </h1>
<form action="" method="POST">
    name: <input type="text" name="name">
    email: <input type="text" name="email">
    <input type="submit" value="submit">
</form>
</body>
</html>.

```

Welcome.php:

```

<html>
@? <?php
    echo "Welcome ". $_SESSION['name'];
    echo " your Mail id ". $_SESSION['email'];
    echo "<a href='logout.php'> Logout </a> ";
?>
</html>

```

Use unset(\$\_SESSION['name'])  
for particular session  
destroy

Logout.php:

```

<?php
    session_destroy(); // For Destroying all variables
    echo "Destroyed Session Successfully"; ?>

```

Output will be displayed at:

doge:

User Name:

Email:

Welcome Akash

Your Email: akash@gmail.com

Logout

User Name:

Email:

Destroyed session successfully.