

Gujarat Technological University
G.H.Patel College of Engineering & Technology
BE Semester-VI Mid-Semester Examination A. Y. 2018-19
2160708 – Web Technology

Solution

Date : 13/03/2019 Max-Marks: 20

Time : 2:00 PM to 3:00 PM

Note: 1. Attempt all questions.

2. Make suitable assumptions wherever necessary.

3. Figures to the right indicate full marks.

Q.1 1. How does the browser and the server interact for transferring the web pages?

2

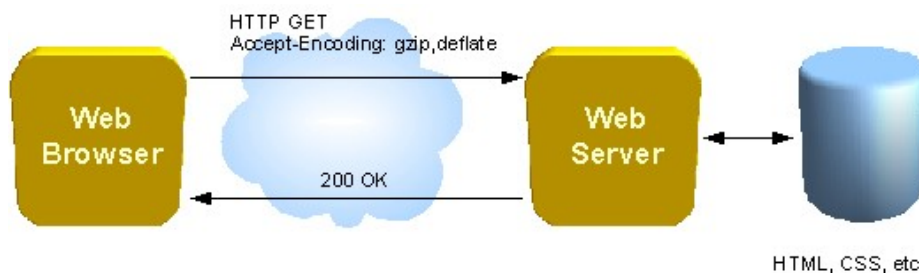
Ans.

Client-Server Network Design and the Web

Web browsers and web servers function together as a [client-server](#) system. In computer networking, client-server is a standard method for designing applications where data is kept in central locations (server computers) and efficiently shared with any number of other computers (the clients) on request. All web browsers function as clients that request information from websites (servers).

Numerous web browser clients can request data from the same website. Requests can happen at all different times or simultaneously. Client-server systems conceptually call for all requests to the same site to be handled by one server. In practice, however, because the volume of requests to web servers can sometimes grow very large, web servers are often built as a distributed pool of multiple server computers.

For very large websites popular in different countries around the world, this web server pool is geographically distributed to help improve the response time to browsers. If the server is closer to the requesting device, it would follow that the time it takes to deliver the content is faster than if the server were further away.



Network Protocols for Web Browsers and Servers

Web browsers and servers communicate via [TCP/IP](#). Hypertext Transfer Protocol ([HTTP](#)) is the standard application protocol on top of TCP/IP supporting web browser requests and server responses.

Web browsers also rely on [DNS](#) to work with [URLs](#). These protocol standards enable different brands of web browsers to communicate with different brands of web servers without requiring special logic for each combination.

Like most internet traffic, web browser and server connections normally run through a series of intermediate [network routers](#).

A basic web browsing session works like this:

- The user specifies a URL in their browser (either from a bookmark or by typing it in)
- The browser initiates a TCP connection to the web server or server pool (using port 80 by default) via its [IP address](#) as published in DNS. As part of this process, the browser also makes DNS lookup requests to convert the URL to an IP address
- After the server completes acknowledgment of its side of the TCP connection, the browser sends HTTP requests to the server to retrieve the content
- After the server replies with content for the page, the browser retrieves it from the HTTP packets and displays it accordingly. Content can include embedded URLs for advertising banners or other third-party content that in turn triggers the browser to issue new TCP connection requests to those locations. The browser may also save temporary information about its connections to local files on the client computer called cookies

Marking Strategy:

- 0.5 marks if diagram is drawn
- 1.5 marks for description and 2 full marks if right description

Enlist web design issues. Explain any two.

3

Ans.

1. Browser
2. Bandwidth and Cache
3. Display resolution
4. Look and Feel of the Website
5. Page Layout and linking
6. User centric design
7. Sitemap

Marking Strategy:

- 1 mark for list of issues
- 1 mark for explanation of each one

2. Explain following HTML tags with syntax:

2

1. img 2. anchor 3. fieldset 4. form

Ans.

1. img

The tag defines an image in an HTML page.

The tag has two required attributes: src and alt.

```

```

2. anchor

The <a> tag defines a hyperlink, which is used to link from one page to another.

The most important attribute of the <a> element is the href attribute, which indicates the link's destination.

By default, links will appear as follows in all browsers:

- An unvisited link is underlined and blue
- A visited link is underlined and purple
- An active link is underlined and red

```
<a href="https://www.w3schools.com">Visit W3Schools.com!</a>
```

3. fieldset

The <fieldset> tag is used to group related elements in a form.

The <fieldset> tag draws a box around the related elements.

```
<fieldset>  
  <legend>Personalia:</legend>  
  Name: <input type="text"><br>  
  Email: <input type="text"><br>  
  Date of birth: <input type="text">  
</fieldset>
```

4. form

The <form> tag is used to create an HTML form for user input. It has main two attributes action and method.

The <form> element can contain one or more of the following form elements:

- <input>
- <textarea>
- <button>
- <select>
- <option>
- <optgroup>
- <fieldset>
- <label>
- <output>

```
<form action="/action_page.php" method="get">  
  First name: <input type="text" name="fname"><br>  
  Last name: <input type="text" name="lname"><br>  
  <input type="submit" value="Submit">  
</form>
```

Marking Strategy:

- 0.5 mark for each HTML tag for correct use and syntax.

Q.2(A) Explain the following CSS properties with example: (Any Four)

4

1. margin 2. background 3. position 4. padding 5. border

Ans.

1. margin

The CSS margin properties are used to create space around elements, outside of any defined borders.

With CSS, you have full control over the margins. There are properties for setting the margin for each side of an element (top, right, bottom, and left).

```
<!DOCTYPE html>  
<html>  
<head>  
<style>  
div {  
  border: 1px solid black;  
  margin-top: 100px;
```

```
margin-bottom: 100px;
margin-right: 150px;
margin-left: 80px;
background-color: lightblue;
}
</style>
</head>
<body>
```

<h2>Using individual margin properties</h2>

<div>This div element has a top margin of 100px, a right margin of 150px, a bottom margin of 100px, and a left margin of 80px.</div>

```
</body>
</html>
```

2. background

The background has many related properties like color, image, repeat, position and attachment.

The background-color property specifies the background color of an element.

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
background-color: lightblue;
}
</style>
</head>
<body>
```

<h1>Hello World!</h1>

<p>This page has a light blue background color!</p>

```
</body>
</html>
```

3. position

The position property specifies the type of positioning method used for an element (static, relative, absolute, fixed, or sticky).

```

<html>
<head>
<style>
h2 {
  position: absolute;
  left: 100px;
  top: 150px;
}
</style>
</head>
<body>

```

```

<h1>The position Property</h1>

```

```

<h2>This is a heading with an absolute position</h2>

```

```

<p>With absolute positioning, an element can be placed anywhere on a page. The
heading below is placed 100px from the left of the page and 150px from the top of the
page.</p>

```

```

</body>
</html>

```

4. padding

An element's padding is the space between its content and its border.

The padding property is a shorthand property for:

- [padding-top](#)
- [padding-right](#)
- [padding-bottom](#)
- [padding-left](#)

```

<html>
<head>
<style>
p.ex1 {
  border: 1px solid red;
  padding: 35px;
}

```

```

p.ex2 {
  border: 1px solid red;
  margin: 35px;
}

```

```

}

</style>
</head>
<body>

<h1>The padding Property</h1>

<p class="ex1">This paragraph has a padding of 35 pixels on all four sides.</p>

<p class="ex2">This paragraph has no specified padding, but a margin of 35 pixels on
all four sides.</p>

<p><strong>Note:</strong> Padding creates extra space within an element, while
margin creates extra space around an element!</p>

</body>
</html>
5. border

```

The border property is a shorthand property for:

- [border-width](#)
- [border-style](#) (required)
- [border-color](#)

If border-color is omitted, the color applied will be the color of the text.

```

<html>
<head>
<style>
h1 {
  border: 5px solid red;
}

h2 {
  border: 4px dotted blue;
}

div {
  border: double;
}
</style>
</head>

```

```

<body>

<h1>A heading with a solid red border</h1>

<h2>A heading with a dotted blue border</h2>

<div>A div element with a double border.</div>

</body>
</html>

```

Marking Strategy:

- 1 mark for description and example of each property

Q.2 (B) Design HTML page for login functionality. Also, write PHP script to authenticate that user is valid or not. (Assume that database contains login information table with required details). **4**

Ans.

Login.php

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Login Page</title>

</head>

<body>

<center>

<h1>Login</h1><br />

<form method="post" action="check.php">

<label>User Name</label>

<input type="text" name="uid" /><br /><br />

<label>Password </label>

<input type="text" name="pwd" /><br /><br />

```



```
<input type="submit" name="submit" />
</form>
</center>
</body>
</html>
```

check.php

<?php

```
$uid=$_POST['uid'];
$pwd=$_POST['pwd'];
if (isset($_POST['submit']) && !empty($_POST['uid']) && !empty($_POST['pwd']))
{
    mysql_connect("localhost","root","");
    mysql_select_db("test");

    $q=mysql_query("select * from login_info where uid=$uid");

    while($row = mysql_fetch_array($q))
    {
        $pwd2=$row['pass'];
    }
    if($pwd==$pwd2)
    {
        header("Location:welcome.php");
    }
}
```

```

        else
        {
            header("Location:index.php");
        }
    }
?>

```

Marking Strategy:

- 2 marks for HTML page
- 2 marks for PHP page

OR

Q.2 Design HTML page for signing up the new user. Also, write PHP script to store the details of new user in the database. (Assume that database contains login information table with fields like first name, last name, email, mobile and password). **4**

Ans.

Register.php/register.html

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Untitled Document</title>

</head>

<body>

<center>

<form method="post" action="insert.php">

<table width="445" height="353" border="1">

```

```

<tr>

  <td><div align="right">Name:</div></td>

  <td><input type="text" name="name" /></td>

</tr>

<tr>

  <td><div align="right">Password:</div></td>

  <td><input type="text" name="pwd" /></td>

</tr>

<tr>

  <td><div align="right">Email:</div></td>

  <td><input type="email" name="email" /></td>

</tr>

<tr>

  <td><div align="right">Mobile:</div></td>

  <td><input type="text" name="mobile" /></td>

</tr>

<tr>

  <td><div align="right">Gender</div></td>

  <td>

    <input type="radio" name="gen" value="male"/>Male

    <input type="radio" name="gen" value="female" />Female  </td>

</tr>

<tr>

  <td><input type="submit" name="Register" /></td>

  <td><input type="reset" name="clear" /></td>

</tr>

```

```
</table>
</form>
</center>
</body>
</html>
```

insert.php

```
<?php
$uid=$_POST['name'];
$pwd=$_POST['pwd'];
$email=$_POST['email'];
$mob=$_POST['mobile'];
$gen=$_POST['gen'];

if (isset($_POST['submit']) && !empty($_POST['uid']) && !empty($_POST['pwd'])
&& !empty($_POST['email']) && !empty($_POST['mobile']) &&
!empty($_POST['gen']))
{
mysql_connect("localhost","root","");
mysql_select_db("test");

mysql_query("insert into login_info values($uid,$pwd,$email,$mob,$gen)");

header("Location:login.php");
}
?>
```

Marking Strategy:

- 2 marks for HTML page
- 2 marks for PHP page

Q.3 Write HTML/PHP code to upload the file on the server. Make sure that it allows to upload only .pdf or .doc and file size must not exceed than 100K. **5**

Ans.

```
<form method="post" enctype="multipart/form-data" action="<?php echo
$_SERVER['PHP_SELF'] ?>">
```

```
<input type="file" name="xyz">
```

```
<input type="submit" value="upload">
```

```
</form>
```

```
<?php
```

```
    $name=$_FILES['xyz']['name'];
```

```
    $type=$_FILES['xyz']['type'];
```

```
    $size=$_FILES['xyz']['size'];
```

```
    $tmp_name=$_FILES['xyz']['tmp_name'];
```

```
    $x=substr($name,(strpos($name,".")+1));
```

```
    if(($x=="DOC"||$x=="PDF"||$x=="doc"||$x=="pdf") && $size<100000)
```

```
    {
```

```
        $path='upload/';
```

```
        move_uploaded_file($tmp_name,$path.$name);
```

```
    }
```

```
    else
```

```
    {
```

```
        echo "file should have extension pdf or doc and less than 100 KB";
```

```
    }
```

?>

Marking Strategy:

- 2 Marks for HTML code
- 3 Marks for PHP

OR

- Q.3** 1. Design a dropdown menu using HTML and CSS, which becomes visible when user mouse over on menu item. (For ex. When user puts mouse curser on department then it will show the list of departments' name) **3**

Ans.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-
1" />

<title>Untitled Document</title>

<style>

.dropbtn {

color: white;

padding: 16px;

font-size: 16px;

border: none;

cursor: pointer;

background-color:red;

}

/* The container <div> - needed to position the
```

```

dropdown content */

.dropdown {
position: relative;
display: inline-block;
}

.dropdown-content {
display: none;
position: absolute;
background-color: #f9f9f9;
min-width: 160px;
box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
z-index: 1;
}

/* Links inside the dropdown */
.dropdown-content a {
color: black;
padding: 12px 16px;
text-decoration: none;
display: block;
}

/* Change color of dropdown links on hover */
.dropdown-content a:hover
{background-color: yellow;
text-decoration: underline;}

.dropdown:hover .dropdown-content {

```

```

display: block;

}

/* Change the background color of the dropdown button when the
dropdown content is shown */

.dropdown:hover .dropbtn {
background-color: #666666;
}*/

</style>

</head>

<body>

<div class="dropdown">

<button class="dropbtn">Dropdown</button>

<div class="dropdown-content">

<a href="#">Link 1</a>

<a href="#">Link 2</a>

<a href="#">Link 3</a>

</div>

</div>

</body>

</html>

```

Marking Strategy:

- 1 Mark for HTML
- 2 Marks for CSS

2. Explain any two input types in HTML with example.

2

Ans.

Input Type Text

```
<input type="text">
```



```
<form>
First name:<br>
<input type="text" name="firstname"><br>
Last name:<br>
<input type="text" name="lastname">
</form>
```

Input Type Password

```
<input type="password">

<form>
User name:<br>
<input type="text" name="username"><br>
User password:<br>
<input type="password" name="psw">
</form>
```

Input Type Submit

```
<input type="submit">

<form action="/action_page.php">
First name:<br>
<input type="text" name="firstname" value="Mickey"><br>
Last name:<br>
<input type="text" name="lastname" value="Mouse"><br>
<input type="submit" value="Submit">
</form>
```

You can write

- <input type="button">
- <input type="checkbox">
- <input type="color">
- <input type="date">
- <input type="datetime-local">
- <input type="email">
- <input type="file">
- <input type="hidden">
- <input type="image">
- <input type="month">
- <input type="number">
- <input type="password">
- <input type="radio">
- <input type="range">
- <input type="reset">
- <input type="search">
- <input type="tel">
- <input type="time">
- <input type="url">
- <input type="week">

Marking Strategy:

- One mark for each input type