

ANIT SARA SANTHOSH
S7, INT MCA
Roll No.9

①

1(b) <!DOCTYPE html>
<html>
<head>
<title>Online Application </title>
</head>
<script>
function validateform()
{
 var name = document.getElementById("name").
 value;
 var namelen = name.length();
 var phone = document.getElementById("phone").
 value;
 var phonelen = phone.length();
 if (namelen == 0)
 {
 if (name == "")
 {
 alert("Please complete the ^{name} field");
 return false;
 }
 }
}

①

```
if (!name.match(^[a-zA-Z]+$))  
{  
    alert("Name should contain  
    alphabets only");  
    return false;  
}
```

```
if (phone == "")  
{  
    alert("Complete the field phone");  
    return false;  
}
```

~~if phlen >= 10~~

```
if ((phlen < 1) || (phlen > 10))  
{  
    alert("Please enter a valid number");  
    return false;  
}
```

```
if (isNaN(phone))  
{  
    alert("Please enter a valid number");  
    return false;  
}
```

③

```
if (!phone.match(/^[\d-9]{10}$/))
{
    alert("Enter a valid phone number");
    return false;
}
</script>
</head>
<body>
<h1>Online Application Form</h1>
<form action="#" onsubmit="return validateform()">
<table align="center">
<tr>
<td><label name="name">Name:</td>
<td><input type="text" name="name" id="name"></td>
</tr>
<tr>
<td><label name="password">Password:</td>
<td><input type="password" name="password" id="password" required pattern="[\d-zA-Z-0-9@#.!]{7,12}">
</td>
```

</tr>

<td>

<td><label name="email"> Email Id </label>
</td>

<td><input type="email" name="email"
id="email" required></td>
placeholder =*

</td><td>

<td><label name="phone"> Phone Number
</label></td>

<td><input type="text" name="phone"
id="phone">

</td><td><input type="submit" value="Submit">

</table>

</form>

</body>

</html>

2 (b) (i) JSON:

- JSON stands for Javascript Object Notation.
- It is a way to store information in an organized, easy-to-access manner.
- JSON is a lightweight format that is used for data interchanging.
- The extension of JSON file is .json.
- JSON has two structures
 - A collection of name/value pairs.
 - An ordered list of values.
- The uses of JSON are:
 - It is used while writing Javascript based applications that include browser extensions and websites.
 - It can be used with modern programming languages.
 - JSON format is used for serializing and transmitting structure data over

(6)

network connection.

- Web services and APIs use JSON format to provide public data.
- It is primarily used to transmit data between a server and web applications.

<script>

~~Ex:~~ var Emp = {
 "name": "Abc",
 "age": 21,
 "city": "Mumbai"
};

 alert(Emp.name);

</script>

② JQuery

- JQuery is a small and light weight Javascript library.
- JQuery means "write less do more".
- It is cross platform

- JQuery is a fast, small, feature rich javascript library.
- It is designed to simplify the client side scripting of HTML.
- It is making things like HTML document traversal and manipulation, event handling, animation and Ajax much simpler with an easy-to-use API that works across a multitude of browsers.
- The main purpose of JQuery is to provide an easy way for using javascript and making it more attractive and interactive.
- It is also used to add animation.
- It has added benefit of being cross browser compatible.

Eg: `$(".demo").html("Welcome");`

AJAX

- AJAX is short for Asynchronous Javascript And XML.
- AJAX is a technique for creating better, faster and more interactive web application with the help of XML, HTML, CSS and JavaScript.
- AJAX uses XHTML for content, CSS for presentation along with DOM and Javascript for dynamic content display.
- AJAX is the act of exchanging data with server, and update part of the webpage without reloading the whole page again.

- AJAX just uses combination of a browser built-in XMLHttpRequest object, Javascript and HTML DOM.
- AJAX allows webpages to be updated asynchronously by exchanging data with a web server behind the scenes.
- The AJAX can update webpage without reloading the page again and again.

(10)

MyPHP.php

3) <!DOCTYPE html>

```

<html>
<head>
<title>
    Online Application
</title>
</head>
<body>
<?php
    if (isset($_POST['submit'])) {
        $name = $_POST['name'];
        $email = $_POST['email'];
        $dist = $_POST['district'];
        $colou = $_POST['colou'];
        $gen = $_POST['gender'];
        $lang = $_POST['lang'];

        echo "Name: ". $name . "<br>";
        echo "Email: ". $email . "<br>";

        echo
        if (isset($_POST['gender']))
    }

```

```
echo "District : \"$_POST["district"]". "  
echo " colour : " ;  
foreach ($_POST["colour"] as $col)  
{  
    echo "$col " ;  
}  
echo "<br>" ;  
echo "Gender: ". $gen . "<br>" ;  
if (!empty($_POST["lang"]))  
{  
    echo "Language known: " ;  
    foreach ($_POST["lang"] as $lang)  
{  
        echo $lang . " , " ;  
    }  
    echo "<br>" ;  
}  
}  
}
```

?>

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

(12)

online app.html

```
<!DOCTYPE html>
<html>
<head>
<title>
    Form
</title>
</head>
<body>
    <form method="POST" action="My Php-  
    Php">
        <label name="name">Name </label>
        <input type="text" name="name"
               id="name" required>
        <br>
        <label name="email">Email Id
        </label>
        <input type="email" name="email"
               id="email" required>
        <br>
```

(13)

<label name="district"> District : </label>

<input type="

<select name="district" id="district" required>

<option value="Pathanamthitta">

Pathanamthitta </option>

<option value="Kottayam">

Kottayam </option>

<option value="Kollam">

Kollam </options>

</select>

<label name="colour"> Col

Favourite - colour

</label>

<input

<select name="colour" id="colour" required multiple>

<option value="Red"> Red </option>

<option value="Black"> Black </option>

<option value="white"> white </option>

<option value="Others"> Others </option>

</select>

<be>

<label name="gender"> Gender: </label>

<input type="radio" name="gender" value="Male">

<input type="radio" name="gender" value="Female"> Female

<be>

<label name="lang"> Language - Known </label>

<input type="checkbox" name="lang[]"
value="C"> C

<input type="checkbox" name="lang[]"
value="C++"> C++

<input type="checkbox" name="lang[]"
value="Java"> Java

<input type="checkbox" name="lang[]"
value="Others"> Others

<be>

<input type="submit" name="submit" value="Submit">

</form>
</body>
</html>

4) (a). In order to store or access the data inside a MySQL database, we need to connect to the MySQL database server.

- We can use mysqli and PDO extensions
- PDO extension is more portable and supports more than twelve different databases;
- mysqli supports MySQL databases only and it is easier to connect and operate.
- To connect to mysql database server the function used is mysqli_connect().
- The communication between PHP and mysql database server takes

- plan through this connection. ⑯
- We can establish connection by mysqli procedural way or object oriented way and PAP Data Object Way.
 - Here mysqli object oriented way is selected.

Eg: syntax

\$conn = mysqli_connect()

\$conn = new mysqli ("hostname",
"username", "password",
"database");

Eg: Procedural way

<?php

\$conn = mys

Eg: Object Oriented Way

<?php

\$con = new mysqli ("localhost",
"root", "", "tej");

?>

- To close connection

- In Object Oriented way :

~~\$con~~

~~\$con ->close();~~

- Then after establishing connection we can start creating database.

(b) <!DOCTYPE html>

<html>

<head>

<title>

PHP

</title>

</head>

<body>

<?php

\$con = new mysqli("localhost",

if (isset(\$_POST["submit"])) {
 \$name = \$_POST["name"];
 \$email = \$_POST["email"];
 \$sql = "INSERT INTO stud (sname, email) Values ('\$name', '\$email');"

(18)

```
if ($::con) { → query ($sql) == true ) {  
    echo " Inserted Successfully";  
}  
else  
{  
    echo " Something went wrong");  
}  
}
```

??

```
< Name:  
< input type="text" name="name" required  
      id="name">  
<br>  
< input type="email" name="email"  
      id="email" required>  
<br>  
< input type="submit" name="submit"  
      value="Submit">  
<br>  
</body>  
</html>
```

(c) <!DOCTYPE html>

(19)

<html>

<head>

<title>

Data

</title>

</head>

<body>

<?php

\$con = new mysqli ("localhost",
"root", " ", "tbl_std");

\$sql = "SELECT * FROM std"

\$sql = "SELECT * FROM Student";

\$result = \$con->query (\$sql);

if (\$result->num_rows > 0) {

echo "<table><tr><th>AdmNo </th><th>Name
<th>Course</th><th>";

while (\$row = \$result->fetch_

echo "<tr><td>"; \$row[0]); {

echo "<td>"; \$row[1]. "

echo "<td>"; \$row[2]. "

"<td><td>";

```

echo "Course : ". $row["course"];
^" </td></tr>";  } echo "</table>";  echo  else {      echo "No data found";  }  ??  </body>  </html> |
```

(Q2) These are all application servers.

- The application server is a service that hosts application.
- These help to create web application and provide environment to run them.

(d) phpmyadmin:

(21)

- This is an open source software tool written in php.
- This helps to handle the operate mysql on the web.
- We can create, drop, delete, update, alter etc using this software.
- It can display multiple result set using queries etc.
- Also this can track the changes done on databases etc.
- It provides the facility to backup the database into different formats.
- It helps us to control user's permission and operate several servers at the same time.

• MySQL

- It is the database management system software used for managing database.
- It is open source database software.
- It is fast, scalable and easy to use.
- It allows us to update the table indexes automatically.
- It allows us to implement database operations on tables, rows, columns and indexes.
- It uses many SQL queries and combines useful information from multiple tables for the end-users.

• XAMPP

- It can be used by any operating system.
- Its full form is x-os, Apache, MySQL, PHP and Perl.

- It is open source, cross platform web server solution pack and is developed by Apache Friends.
- It is simple, lightweight and easy to use for developers in creating local web servers.
- XAMPP comes with extra feature like supporting of PHP, Mozilla, mercury mail and some other scripts.