Assignment 1

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Weekly Timetable</title>

<link rel="stylesheet"

href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

<style>

body { background-color: #f8f9fa; }

th { background-color: #007bff; color: white; }

td { height: 100px; vertical-align: middle; text-align: center; }

</style>

</head>

<body>

<div class="container mt-4">

<h1 class="text-center">Weekly Timetable</h1>

<table class="table table-bordered">

<thead>

<tr>

<th>Time</th>

<th>Mon</th>

<th>Tue</th>

<th>Wed</th>

<th>Thu</th>

<th>Fri</th>

</tr>

</thead>

<tbody>

<tr><td>8:00 -

9:00</td><td>Math</td><td>English</td><td>Science</td><td>History</td><td>PE</td></tr

>

<tr><td>9:00 -

10:00</td><td>Biology</td><td>Math</td><td>English</td><td>Science</td><td>Art</td></t

r>

<tr><td>10:00 -

11:00</td><td>History</td><td>Biology</td><td>Math</td><td>English</td><td>CS</td></tr

>

<tr><td>11:00 -

12:00</td><td>PE</td><td>History</td><td>Biology</td><td>Math</td><td>Science</td></t

r>

<tr><td colspan="5" class="text-center">Lunch Break</td></tr>

<tr><td>1:00 -

2:00</td><td>Science</td><td>PE</td><td>History</td><td>Biology</td><td>Math</td></tr

>

<tr><td>2:00 -

3:00</td><td>Art</td><td>Science</td><td>PE</td><td>English</td><td>History</td></tr>

</tbody>

</table>

</div>

</body>

</html>

Name- Irfan Tadvi

Prn- 122B1B273

Assignment No-02

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>User Data Validation</title>

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

<style>

/\* Center the form on the screen \*/

body {

font-family: Arial, sans-serif;

background: linear-gradient(to right, #2c3e50, #4ca1af);

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

.container {

background: #fff;

padding: 30px;

border-radius: 10px;

box-shadow: 0 0 15px rgba(0, 0, 0, 0.2);

width: 350px;

text-align: center;

}

h2 {

color: #2c3e50;

}

input {

width: 100%;

padding: 10px;

margin: 10px 0 5px;

border: 1px solid #ccc;

border-radius: 5px;

font-size: 16px;

}

.error {

color: red;

font-size: 14px;

text-align: left;

margin-bottom: 10px;

}

button {

width: 100%;

padding: 10px;

border: none;

background: #4ca1af;

color: white;

font-size: 18px;

border-radius: 5px;

cursor: pointer;

transition: 0.3s;

margin-top: 10px;

}

button:hover {

background: #2c3e50;

}

</style>

</head>

<body>

<div class="container">

<h2>User Registration</h2>

<form id="userForm">

<input type="text" id="name" placeholder="Enter Name">

<div class="error" id="nameError"></div>

<input type="password" id="password" placeholder="Enter Password">

<div class="error" id="passwordError"></div>

<input type="text" id="phone" placeholder="Enter Phone Number">

<div class="error" id="phoneError"></div>

<input type="text" id="email" placeholder="Enter Email">

<div class="error" id="emailError"></div>

<input type="text" id="address" placeholder="Enter Address">

<div class="error" id="addressError"></div>

<button type="submit">Submit</button>

</form>

</div>

<script>

$(document).ready(function () {

$("#userForm").submit(function (e) {

e.preventDefault(); // Prevent form submission

let isValid = true;

// Reset previous errors

$(".error").text("");

// Name Validation

let name = $("#name").val().trim();

if (!/^[a-zA-Z ]{3,}$/.test(name)) {

$("#nameError").text("Name must be at least 3 letters.");

isValid = false;

}

// Password Validation

let password = $("#password").val();

if

(!/(?=.[a-z])(?=.[A-Z])(?=.\d)(?=.[@$!%?&])[A-Za-z\d@$!%?&]{8,}/.test(password)) {

$("#passwordError").text("Password must be 8+ chars, with uppercase, lowercase,

number & special char.");

isValid = false;

}

// Phone Validation

let phone = $("#phone").val();

if (!/^\d{10}$/.test(phone)) {

$("#phoneError").text("Phone number must be 10 digits.");

isValid = false;

}

// Email Validation

let email = $("#email").val();

if (!/^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/.test(email)) {

$("#emailError").text("Enter a valid email address.");

isValid = false;

}

// Address Validation

let address = $("#address").val().trim();

if (address.length === 0) {

$("#addressError").text("Address cannot be empty.");

isValid = false;

}

// If form is valid, submit it

if (isValid) {

alert("Form submitted successfully!");

$("#userForm")[0].reset(); // Reset form fields

}

});

// Real-time validation

$("input").on("input", function () {

let field = $(this).attr("id");

$("#" + field + "Error").text(""); // Clear error message when user types

});

});

</script>

</body>

</html>

NAME- Irfan Tadvi

Prn -122B1B273

Assignment No- 03

Shiporder.xml

<?xml version="1.0" encoding="UTF-8"?>

<shiporder orderid="123"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:noNamespaceSchemaLocation="shiporder.xsd">

<orderperson>ABCD1</orderperson>

<shipto>

<name>Irfan Anwar Tadvi</name>

<address>Swapnapurti Society</address>

<city>Pune</city>

<country>India</country>

</shipto>

<item>

<title>9185756421</title>

<note>Special Edition</note>

<quantity>1</quantity>

<price>24555.77</price>

</item>

<item>

<title>Superbike</title>

<quantity>10</quantity>

<price>124566</price>

</item>

</shiporder>

Shiporder.xsd

<?xml version="1.0" encoding="UTF-8" ?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="shiporder">

<xs:complexType>

<xs:sequence>

<xs:element name="orderperson" type="xs:string"/>

<xs:element name="shipto">

<xs:complexType>

<xs:sequence>

<xs:element name="name" type="xs:string"/>

<xs:element name="address" type="xs:string" minOccurs="1" />

<xs:element name="city" type="xs:string"/>

<xs:element name="country" type="xs:string"/>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="item" maxOccurs="unbounded">

<xs:complexType>

<xs:sequence>

<xs:element name="title" type="xs:string"/>

<xs:element name="note" type="xs:string" minOccurs="0"/>

<xs:element name="quantity" type="xs:positiveInteger"/>

<xs:element name="price" type="xs:decimal"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

<xs:attribute name="orderid" type="xs:string" use="required"/>

</xs:complexType>

</xs:element>

</xs:schema>

Assignment 4

Name : Rohit Satish Pawar

Roll : TYCOC222

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Search Employee Name</title>

</head>

<body>

<h1>Employee Name Search</h1>

<form method="POST" action="">

<label for="name">Enter Employee Name to Search:</label>

<input type="text" name="name" id="name" required>

<input type="submit" value="Search">

</form>

<?php

// Indexed array of 20 Indian employee names

$employee\_names = [

"Amit Sharma", "Priya Verma", "Ravi Kumar", "Anjali Patel", "Vikas Singh",

"Neha Reddy", "Manoj Gupta", "Kiran Nair", "Suresh Yadav", "Pooja Mehta",

"Rajesh Kumar", "Deepika Singh", "Ajay Malhotra", "Sonali Deshmukh", "Kunal Joshi",

"Shivani Iyer", "Rakesh Kapoor", "Rekha Rathi", "Vandana Shah", "Siddharth Kapoor"

];

// Check if the form is submitted

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

// Get the name input from the user

$search\_name = $\_POST['name'];

// Search for the name in the array

if (in\_array($search\_name, $employee\_names)) {

echo "<p>The employee '$search\_name' exists in the list.</p>";

} else {

echo "<p>The employee '$search\_name' does not exist in the list.</p>";

}

}

?>

</body>

</html>

Assignment No. 5

// index.html

<html>

<head>

<script>

function changeShape() {

const option =

document.querySelector("input[name='op']:checked").value;

document.getElementById("coneCylinder").style.display = (option ===

"1" || option === "2") ? "block" : "none";

document.getElementById("sphere").style.display = (option === "3") ?

"block" : "none";

}

</script>

</head>

<body>

<form action="index.php" method="GET">

<center>

<h1>Select Shape</h1>

<label>

<input type="radio" name="op" value="1" onclick="changeShape()"

/>

Cone

</label>

<label>

<input type="radio" name="op" value="2" onclick="changeShape()"

/>

Cylinder

</label>

<label>

<input type="radio" name="op" value="3" onclick="changeShape()"

/>

Sphere

</label>

<div id="coneCylinder" style="display: none;">

<h2>For Cone & Cylinder</h2>

<table>

<tr>

<td>Enter Radius</td>

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

</tr>

<tr>

<td>Enter Height</td>

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

</tr>

<tr>

<td></td>

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

</tr>

</table>

</div>

<div id="sphere" style="display: none;">

<h2>For Sphere</h2>

<table>

<tr>

<td>Enter Radius</td>

<td><input type="text" name="r1" /></td>

</tr>

<tr>

<td></td>

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

</tr>

</table>

</div>

</center>

</form>

</body>

</html>

// index.php

<?php

define('pi', 3.14);

interface shape

{

function calc\_area($r, $h);

function calc\_vol($r, $h);

}

class sphere implements shape

{

function calc\_area($r, $h)

{

return 4 \* pi \* $r \* $r;

}

function calc\_vol($r, $h)

{

return (4 / 3) \* pi \* $r \* $r \* $r;

}

}

class cylinder implements shape

{

function calc\_area($r, $h)

{

return 2 \* pi \* $r \* ($r + $h);

}

function calc\_vol($r, $h)

{

return pi \* $r \* $r \* $h;

}

}

class cone implements shape

{

function calc\_area($r, $h)

{

return 0.5 \* $r \* $r \* $h;

}

function calc\_vol($r, $h)

{

return (1 / 3) \* pi \* $r \* $r \* $h;

}

}

$op = isset($\_GET['op']) ? $\_GET['op'] : null;

switch ($op) {

case 1:

$r = isset($\_GET['r']) ? (int)$\_GET['r'] : 0;

$h = isset($\_GET['h']) ? (int)$\_GET['h'] : 0;

$ob = new cone();

$a = $ob->calc\_area($r, $h);

$v = $ob->calc\_vol($r, $h);

echo "Area of Cone is: " . $a . "<br>Volume of Cone is: " . $v;

break;

case 2:

$r = isset($\_GET['r']) ? (int)$\_GET['r'] : 0;

$h = isset($\_GET['h']) ? (int)$\_GET['h'] : 0;

$ob = new cylinder();

$a = $ob->calc\_area($r, $h);

$v = $ob->calc\_vol($r, $h);

echo "Area of Cylinder is: " . $a . "<br>Volume of Cylinder is: " . $v;

break;

case 3:

$r = isset($\_GET['r1']) ? (int)$\_GET['r1'] : 0;

$ob = new sphere();

$a = $ob->calc\_area($r, 0); // Height is not used in sphere calculations

$v = $ob->calc\_vol($r, 0); // Height is not used in sphere calculations

echo "Area of Sphere is: " . $a . "<br>Volume of Sphere is: " . $v;

break;

default:

echo "Invalid operation.";

}

?>

Assignment No. 6

// contact.html

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>Contact Form - PHP/MySQL</title>

<script src="https://unpkg.com/@tailwindcss/browser@4"></script>

</head>

<body class="w-screen h-screen flex justify-center items-center">

<form action="contact.php" method="post" class="flex flex-col justify-center

gap-3 w-sm bg-slate-100 rounded-xl p-8">

<label class="flex flex-col gap-1">

First Name

<input type="text" name="fname" id="fname" class="rounded px-2 py-1

bg-white" />

</label>

<label class="flex flex-col gap-1">

Middle Name

<input type="text" name="mname" id="mname" class="rounded px-2 py-1

bg-white" />

</label>

<label class="flex flex-col gap-1">

Last Name

<input type="text" name="lname" id="lname" class="rounded px-2 py-1

bg-white" />

</label>

<label class="flex flex-col gap-1">

Password

<input type="password" name="pwd" id="pwd" class="rounded px-2 py-1

bg-white" />

</label>

<button type="submit" class="bg-blue-500 text-white py-2

rounded">Submit</button>

</form>

</body>

</html>

// contact.php

<?php

$con = mysqli\_connect('localhost', 'root', '', 'contact');

// get the post records

$txtfname = $\_POST['fname'];

$txtmname = $\_POST['mname'];

$txtlname = $\_POST['lname'];

$txtpwd = $\_POST['pwd'];

// database insert SQL code

$sql = "INSERT INTO registration (Id, fname, mname, lname, pwd)

VALUES ('0', '$txtfname', '$txtmname', '$txtlname', '$txtpwd')";

// insert in database

$rs = mysqli\_query($con, $sql);

if($rs)

{

echo "Contact Records Inserted <br> <br>";

}

$sql = 'SELECT \* FROM registration';

$retval = mysqli\_query($con, $sql);

while($row = mysqli\_fetch\_array($retval, MYSQLI\_ASSOC)) {

echo "EMP ID :{$row['id']} <br> ".

"EMP First NAME : {$row['fname']} <br> ".

"EMP middle name : {$row['mname']} <br> ".

"EMP Last name : {$row['lname']} <br> ".

"--------------------------------<br>";

}

echo "Fetched data successfully\n";

?>

ASSIGNMENT NO 7

Code:

1. db\_connect.php

<? php

$servername = "localhost";

$db\_username = "root"; // Default for XAMPP/WAMP

$db\_password = ""; // Leave empty for XAMPP/WAMP

$dbname = "login\_system";

Name : Atharva Toke

Div : D

PRN : 122B1B279

Batch : D3

// Create a connection

$conn = new mysqli($servername, $db\_username, $db\_password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

?>

2. login.php

<? php

session\_start();

include("db\_connect.php");

if (isset($\_SESSION['username'])) {

header("Location: home.php");

exit;

}

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$username = trim($\_POST["username"]);

$password = trim($\_POST["password"]);

// Using a prepared statement to prevent SQL injection

$stmt = $conn->prepare("SELECT \* FROM users WHERE username = ? LIMIT 1");

$stmt->bind\_param("s", $username);

$stmt->execute();

$result = $stmt->get\_result();

1

if ($result && $result->num\_rows == 1) {

$user = $result->fetch\_assoc();

// Debugging -Check if user is fetched correctly

var\_dump($user);

if (password\_verify($password, $user['password'])) {

session\_regenerate\_id(true);

$\_SESSION['username'] = $user['username'];

header("Location: home.php");

exit;

} else {

$error = "Invalid password!";

}

} else {

$error = "Invalid username!";

}

$stmt->close();

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Login</title>

</head>

<body>

<h2>Login</h2>

<?php if(isset($error)) { echo "<p style='color:red;'>$error</p>"; } ?>

<form method="post">

<label>Username:</label>

<input type="text" name="username" required><br><br>

<label>Password:</label>

<input type="password" name="password" required><br><br>

<label>Remember Me:</label>

2

<input type="checkbox" name="remember"><br><br>

<input type="submit" value="Login">

</form>

</body>

</htm l>

3. home.php

<? php

session\_start();

if (!isset($\_SESSION['username'])) {

if (isset($\_COOKIE['username'])) {

$\_SESSION['username'] = $\_COOKIE['username'];

} else {

header("Location: login.php");

exit;

}

}

?

>

<!DOCTYPE html> <html > <head> <title>Home</title> </head> <body>

<h1>Welcome, <?php echo htmlspecialchars($\_SESSION['username']); ?></h1>

<p>You are logged in.</p>

<p><a href="logout.php">Logout</a></p>

</body>

</htm l>

4.logout.php

<? php

session\_start();

session\_unset();

session\_destroy();

if (isset($\_COOKIE['username'])) {

setcookie("username", "", time() -3600, "/");

3

}

header("Location: login.php");

exit;

?>

<? php

session\_start();

session\_unset();

session\_destroy();

if (isset($\_COOKIE['username'])) {

setcookie("username", "", time() -3600, "/");

}

header("Location: login.php");

exit;

?>

Assignment 08

Name – Rohit Pawar

PRN – 122B1B222

Roll – TYCOC222

app.rb

require 'sinatra'

# Method to reverse the name

def reverse\_name(first\_name, last\_name)

"#{last\_name} #{first\_name}"

end

get '/' do

erb :index

end

post '/reverse' do

first\_name = params[:first\_name]

last\_name = params[:last\_name]

@reversed\_name = reverse\_name(first\_name, last\_name)

erb :reverse

end

index.erb

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Reverse Name</title>

</head>

<body>

<h1>Reverse Name</h1>

<form action="/reverse" method="POST">

<label for="first\_name">First Name:</label>

<input type="text" id="first\_name" name="first\_name" required><br>

<label for="last\_name">Last Name:</label>

<input type="text" id="last\_name" name="last\_name" required><br>

<button type="submit">Reverse Name</button>

</form>

</body>

</html>

reverse.erb

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Reversed Name</title>

</head>

<body>

<h1>Reversed Name</h1>

<p><%= @reversed\_name %></p>

</body>

</html>

Assignment 9

require 'mail'

# Set up SMTP settings

Mail.defaults do

delivery\_method :smtp, {

address: 'smtp.gmail.com',

port: 587,

user\_name: 'rohitpawar6012@gmail.com',

password: 'uuxb yzjz dcug baig ',

authentication: :login,

enable\_starttls\_auto: true

}

end

# Define email message

message = Mail.new do

from 'rohitpawar6012@gmail.com'

to 'rohit.pawar22@pccoepune.org'

subject 'Hello from Ruby!'

body 'This is a test email sent from Ruby.'

end

# Send email

message.deliver!