**Q.1)Create student registration form with name,address,contact,gender,course,languages known,email,password(encrypted) with name ,email and password required.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Student Registration Form</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

margin: 0;

padding: 0;

}

.container {

max-width: 500px;

margin: 20px auto;

padding: 20px;

background-color: #fff;

border-radius: 5px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

h2 {

text-align: center;

}

label {

font-weight: bold;

}

input[type="text"],

input[type="email"],

input[type="password"] {

width: 100%;

padding: 10px;

margin-bottom: 10px;

border: 1px solid #ccc;

border-radius: 5px;

}

input[type="radio"] {

margin-right: 5px;

}

input[type="submit"] {

background-color: #007bff;

color: #fff;

padding: 10px 20px;

border: none;

border-radius: 5px;

cursor: pointer;

}

input[type="submit"]:hover {

background-color: #0056b3;

}

</style>

</head>

<body>

<div class="container">

<h2>Student Registration Form</h2>

<form action="#" method="post">

<label for="name">Name:</label><br>

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

<label for="address">Address:</label><br>

<input type="text" id="address" name="address"><br>

<label for="contact">Contact:</label><br>

<input type="text" id="contact" name="contact"><br>

<label for="gender">Gender:</label><br>

<input type="radio" id="male" name="gender" value="male">

<label for="male">Male</label>

<input type="radio" id="female" name="gender" value="female">

<label for="female">Female</label>

<input type="radio" id="other" name="gender" value="other">

<label for="other">Other</label><br>

<label for="course">Course:</label><br>

<input type="text" id="course" name="course"><br>

<label for="languages">Languages Known:</label><br>

<input type="text" id="languages" name="languages"><br>

<label for="email">Email:</label><br>

<input type="email" id="email" name="email" required><br>

<label for="password">Password:</label><br>

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

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

</form>

</div>

</body>

</html>

**Q.2) Create stripped time table of your class with suitable headings and footer.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Class Time Table</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

margin: 0;

padding: 0;

}

.container {

max-width: 800px;

margin: 20px auto;

background-color: #fff;

padding: 20px;

border-radius: 5px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

h1, h2 {

text-align: center;

}

table {

width: 100%;

border-collapse: collapse;

margin-bottom: 20px;

}

th, td {

border: 1px solid #ddd;

padding: 10px;

text-align: left;

}

th {

background-color: #007bff;

color: #fff;

}

tfoot {

font-weight: bold;

}

</style>

</head>

<body>

<div class="container">

<h1>Class Time Table</h1>

<table>

<thead>

<tr>

<th>Time</th>

<th>Monday</th>

<th>Tuesday</th>

<th>Wednesday</th>

<th>Thursday</th>

<th>Friday</th>

</tr>

</thead>

<tbody>

<tr>

<td>8:00 AM - 9:00 AM</td>

<td>Mathematics</td>

<td>Science</td>

<td>English</td>

<td>History</td>

<td>Geography</td>

</tr>

<tr>

<td>9:00 AM - 10:00 AM</td>

<td>Physics</td>

<td>Chemistry</td>

<td>Art</td>

<td>Physical Education</td>

<td>Mathematics</td>

</tr>

<tr>

<td>10:00 AM - 11:00 AM</td>

<td>Break</td>

<td>Break</td>

<td>Break</td>

<td>Break</td>

<td>Break</td>

</tr>

<!-- Add more rows for additional periods -->

</tbody>

<tfoot>

<tr>

<td colspan="6">Class timings are subject to change.</td>

</tr>

</tfoot>

</table>

</div>

</body>

</html>

**Q.3) Divide the web page into 3 different frameset and render each HTML document in separate frame.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Frameset Example</title>

</head>

<frameset cols="33%,33%,33%">

<frame src="frame1.html" name="frame1">

<frame src="frame2.html" name="frame2">

<frame src="frame3.html" name="frame3">

<noframes>

<body>Your browser does not support frames.</body>

</noframes>

</frameset>

</html>

**Frame 1:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Frame 1 Content</title>

</head>

<body>

<h1>Frame 1 Content</h1>

<p>This is the content for frame 1.</p>

</body>

</html>

**Frame 2:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Frame 2 Content</title>

</head>

<body>

<h1>Frame 2 Content</h1>

<p>This is the content for frame 2.</p>

</body>

</html>

**Frame 3:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Frame 3 Content</title>

</head>

<body>

<h1>Frame 3 Content</h1>

<p>This is the content for frame 3.</p>

</body>

</html>

**Q.4) Create audio and video gallery.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Audio and Video Gallery</title>

<style>

.gallery {

display: grid;

grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));

grid-gap: 20px;

}

.item {

border: 1px solid #ddd;

padding: 10px;

border-radius: 5px;

text-align: center;

}

video, audio {

width: 100%;

border-radius: 5px;

}

</style>

</head>

<body>

<h1>Audio and Video Gallery</h1>

<div class="gallery">

<div class="item">

<h2>Video 1</h2>

<video controls>

<source src="video1.mp4" type="video/mp4">

Your browser does not support the video tag.

</video>

</div>

<div class="item">

<h2>Video 2</h2>

<video controls>

<source src="video2.mp4" type="video/mp4">

Your browser does not support the video tag.

</video>

</div>

<div class="item">

<h2>Audio 1</h2>

<audio controls>

<source src="audio1.mp3" type="audio/mpeg">

Your browser does not support the audio tag.

</audio>

</div>

<div class="item">

<h2>Audio 2</h2>

<audio controls>

<source src="audio2.mp3" type="audio/mpeg">

Your browser does not support the audio tag.

</audio>

</div>

</div>

</body>

</html>

**Q.5) Create image gallery render at least 8 images.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Image Gallery</title>

<style>

.gallery {

display: grid;

grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));

grid-gap: 10px;

}

.item {

overflow: hidden;

border-radius: 5px;

}

.item img {

width: 100%;

height: auto;

transition: transform 0.3s ease;

}

.item:hover img {

transform: scale(1.1);

}

</style>

</head>

<body>

<h1>Image Gallery</h1>

<div class="gallery">

<div class="item">

<img src="image1.jpg" alt="Image 1">

</div>

<div class="item">

<img src="image2.jpg" alt="Image 2">

</div>

<div class="item">

<img src="image3.jpg" alt="Image 3">

</div>

<div class="item">

<img src="image4.jpg" alt="Image 4">

</div>

<div class="item">

<img src="image5.jpg" alt="Image 5">

</div>

<div class="item">

<img src="image6.jpg" alt="Image 6">

</div>

<div class="item">

<img src="image7.jpg" alt="Image 7">

</div>

<div class="item">

<img src="image8.jpg" alt="Image 8">

</div>

</div>

</body>

</html>

**Q.6) Design web page showing your introduction with photo.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Introduction to ChatGPT</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

background-color: #f4f4f4;

}

.container {

max-width: 800px;

margin: 20px auto;

padding: 20px;

background-color: #fff;

border-radius: 5px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

h1 {

text-align: center;

}

p {

text-align: justify;

line-height: 1.6;

}

.image-container {

text-align: center;

margin-top: 20px;

}

img {

max-width: 100%;

border-radius: 5px;

}

</style>

</head>

<body>

<div class="container">

<h1>Introduction to ChatGPT</h1>

<p>

Hi there! I'm ChatGPT, your friendly AI language model created by OpenAI. I'm designed to assist you with a wide range of tasks, from answering questions and generating text to providing suggestions and more.

</p>

<p>

I'm trained on a vast amount of data and can converse with you on many topics. Whether you need help with homework, want to brainstorm ideas, or simply chat for fun, I'm here for you!

</p>

<div class="image-container">

<img src="chatgpt\_photo.jpg" alt="ChatGPT Photo">

</div>

</div>

</body>

</html>

**Q.7) Write different 5 paragraphs and apply different color with transparency.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Colorful Paragraphs</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

background-color: #f4f4f4;

}

.container {

max-width: 800px;

margin: 20px auto;

padding: 20px;

background-color: #fff;

border-radius: 5px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

p {

padding: 10px;

border-radius: 5px;

margin-bottom: 20px;

}

.color1 {

background-color: rgba(255, 0, 0, 0.5); /\* Red with 50% transparency \*/

}

.color2 {

background-color: rgba(0, 255, 0, 0.5); /\* Green with 50% transparency \*/

}

.color3 {

background-color: rgba(0, 0, 255, 0.5); /\* Blue with 50% transparency \*/

}

.color4 {

background-color: rgba(255, 255, 0, 0.5); /\* Yellow with 50% transparency \*/

}

.color5 {

background-color: rgba(255, 0, 255, 0.5); /\* Magenta with 50% transparency \*/

}

</style>

</head>

<body>

<div class="container">

<p class="color1">This is a paragraph with a red background color and 50% transparency.</p>

<p class="color2">This is a paragraph with a green background color and 50% transparency.</p>

<p class="color3">This is a paragraph with a blue background color and 50% transparency.</p>

<p class="color4">This is a paragraph with a yellow background color and 50% transparency.</p>

<p class="color5">This is a paragraph with a magenta background color and 50% transparency.</p>

</div>

</body>

</html>

Q.8) Design web page which have position values sticky,fixed ,absolute and relative.

Ans:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Positioning Example</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

}

.container {

width: 100%;

height: 1000px; /\* Just for demonstration \*/

background-color: #f4f4f4;

position: relative;

}

.sticky {

position: -webkit-sticky;

position: sticky;

top: 20px;

background-color: #007bff;

color: #fff;

padding: 10px;

margin-bottom: 20px;

}

.fixed {

position: fixed;

top: 20px;

right: 20px;

background-color: #dc3545;

color: #fff;

padding: 10px;

}

.absolute {

position: absolute;

top: 50%;

left: 50%;

transform: translate(-50%, -50%);

background-color: #28a745;

color: #fff;

padding: 10px;

}

.relative {

position: relative;

top: 50px;

left: 50px;

background-color: #ffc107;

color: #333;

padding: 10px;

}

</style>

</head>

<body>

<div class="container">

<div class="sticky">Sticky Element - Sticks to the top of the container when scrolling</div>

<div class="fixed">Fixed Element - Stays fixed at the top-right corner of the viewport</div>

<div class="absolute">Absolute Element - Positioned relative to the container's top-left corner</div>

<div class="relative">Relative Element - Positioned relative to its normal position in the document flow</div>

</div>

</body>

</html>

**Q.9) Define and call a function factorial using PHP.**

**Ans:**

<?php

// Function to calculate factorial

function factorial($n) {

if ($n === 0) {

return 1;

} else {

return $n \* factorial($n - 1);

}

}

// Call the factorial function and print the result

$number = 5; // Change this to the desired number

$result = factorial($number);

echo "Factorial of $number is: $result";

?>

**Q.10) Write a program to implement any 5 string manipulation functions.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>String Manipulation</title>

</head>

<body>

<h2>String Manipulation</h2>

<form action="" method="post">

<label for="string">Enter a string:</label><br>

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

<label for="oldSubstring">Enter the old substring:</label><br>

<input type="text" id="oldSubstring" name="oldSubstring"><br><br>

<label for="newSubstring">Enter the new substring:</label><br>

<input type="text" id="newSubstring" name="newSubstring"><br><br>

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

</form>

<?php

// Function to convert string to uppercase

function convertToUpper($str) {

return strtoupper($str);

}

// Function to convert string to lowercase

function convertToLower($str) {

return strtolower($str);

}

// Function to get the length of a string

function getStringLength($str) {

return strlen($str);

}

// Function to reverse a string

function reverseString($str) {

return strrev($str);

}

// Function to replace a substring in a string

function replaceSubstring($str, $oldSubstr, $newSubstr) {

return str\_replace($oldSubstr, $newSubstr, $str);

}

// Handle form submission

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

$string = $\_POST["string"];

$oldSubstring = $\_POST["oldSubstring"];

$newSubstring = $\_POST["newSubstring"];

echo "<h3>Results:</h3>";

echo "Original string: $string <br>";

echo "Uppercase: " . convertToUpper($string) . "<br>";

echo "Lowercase: " . convertToLower($string) . "<br>";

echo "Length: " . getStringLength($string) . "<br>";

echo "Reversed: " . reverseString($string) . "<br>";

if (!empty($oldSubstring) && !empty($newSubstring)) {

echo "Replace '$oldSubstring' with '$newSubstring': " . replaceSubstring($string, $oldSubstring, $newSubstring) . "<br>";

}

}

?>

</body>

</html>

**Q.11) Write a program to implement string compare and search operations.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>String Operations</title>

</head>

<body>

<h2>String Operations</h2>

<form action="" method="post">

<label for="string1">Enter string 1:</label><br>

<input type="text" id="string1" name="string1" required><br><br>

<label for="string2">Enter string 2:</label><br>

<input type="text" id="string2" name="string2" required><br><br>

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

</form>

<?php

// Function to compare two strings

function compareStrings($str1, $str2) {

return strcmp($str1, $str2);

}

// Function to search for a substring in a string

function searchString($str, $substr) {

return strpos($str, $substr);

}

// Handle form submission

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

$string1 = $\_POST["string1"];

$string2 = $\_POST["string2"];

echo "<h3>Results:</h3>";

echo "String 1: $string1 <br>";

echo "String 2: $string2 <br>";

// Compare strings

$comparison = compareStrings($string1, $string2);

if ($comparison < 0) {

echo "String 1 is less than String 2 <br>";

} elseif ($comparison > 0) {

echo "String 1 is greater than String 2 <br>";

} else {

echo "String 1 is equal to String 2 <br>";

}

// Search for substring

$substring = "world";

$searchResult = searchString($string1, $substring);

if ($searchResult !== false) {

echo "Substring '$substring' found at position: $searchResult <br>";

} else {

echo "Substring '$substring' not found in String 1 <br>";

}

}

?>

</body>

</html>

**Q.12) Write a program to check whether a number is prime or not in php.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Prime Number Checker</title>

</head>

<body>

<h2>Prime Number Checker</h2>

<form action="" method="post">

<label for="number">Enter a number:</label><br>

<input type="number" id="number" name="number" required><br><br>

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

</form>

<?php

// Function to check if a number is prime

function isPrime($num) {

if ($num <= 1) {

return false;

}

for ($i = 2; $i <= sqrt($num); $i++) {

if ($num % $i == 0) {

return false;

}

}

return true;

}

// Handle form submission

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

$number = $\_POST["number"];

if (is\_numeric($number)) {

if (isPrime($number)) {

echo "<p>$number is a prime number.</p>";

} else {

echo "<p>$number is not a prime number.</p>";

}

} else {

echo "<p>Please enter a valid number.</p>";

}

}

?>

</body>

</html>

**Q.13) Write a program to display Fibonacci series of N terms.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Fibonacci Series</title>

</head>

<body>

<h2>Fibonacci Series</h2>

<form action="" method="post">

<label for="terms">Enter the number of terms:</label><br>

<input type="number" id="terms" name="terms" required><br><br>

<input type="submit" value="Generate Series">

</form>

<?php

// Function to generate Fibonacci series

function fibonacciSeries($terms) {

$fibonacci = [];

$fibonacci[0] = 0;

$fibonacci[1] = 1;

for ($i = 2; $i < $terms; $i++) {

$fibonacci[$i] = $fibonacci[$i - 1] + $fibonacci[$i - 2];

}

return $fibonacci;

}

// Handle form submission

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

$terms = $\_POST["terms"];

if (is\_numeric($terms) && $terms > 0) {

echo "<h3>Fibonacci Series of $terms terms:</h3>";

$fibonacciSeries = fibonacciSeries($terms);

echo "<p>";

for ($i = 0; $i < $terms; $i++) {

echo $fibonacciSeries[$i];

if ($i != $terms - 1) {

echo ", ";

}

}

echo "</p>";

} else {

echo "<p>Please enter a valid positive integer.</p>";

}

}

?>

</body>

</html>

**Q.14) Write a program to swap the numbers using call by value.**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Swap Numbers</title>

</head>

<body>

<h2>Swap Numbers using Call by Value</h2>

<form action="" method="post">

<label for="num1">Enter the first number:</label><br>

<input type="number" id="num1" name="num1" required><br><br>

<label for="num2">Enter the second number:</label><br>

<input type="number" id="num2" name="num2" required><br><br>

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

</form>

<?php

// Function to swap two numbers

function swapNumbers($num1, $num2) {

$temp = $num1;

$num1 = $num2;

$num2 = $temp;

echo "After swapping: <br>";

echo "First number: $num1 <br>";

echo "Second number: $num2 <br>";

}

// Handle form submission

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

$num1 = $\_POST["num1"];

$num2 = $\_POST["num2"];

echo "Before swapping: <br>";

echo "First number: $num1 <br>";

echo "Second number: $num2 <br>";

// Call the swapNumbers function

swapNumbers($num1, $num2);

}

?>

</body>

</html>

**Q.15) Write a program to swap the numbers using call by references**

**Ans:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Swap Numbers</title>

</head>

<body>

<h2>Swap Numbers using Call by Reference</h2>

<form action="" method="post">

<label for="num1">Enter the first number:</label><br>

<input type="number" id="num1" name="num1" required><br><br>

<label for="num2">Enter the second number:</label><br>

<input type="number" id="num2" name="num2" required><br><br>

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

</form>

<?php

// Function to swap two numbers using call by reference

function swapNumbers(&$num1, &$num2) {

$temp = $num1;

$num1 = $num2;

$num2 = $temp;

echo "After swapping: <br>";

echo "First number: $num1 <br>";

echo "Second number: $num2 <br>";

}

// Handle form submission

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

$num1 = $\_POST["num1"];

$num2 = $\_POST["num2"];

echo "Before swapping: <br>";

echo "First number: $num1 <br>";

echo "Second number: $num2 <br>";

// Call the swapNumbers function with call by reference

swapNumbers($num1, $num2);

}

?>

</body>

</html>