

4. User registration form for a social networking website. The form should contain username, email, HTML code.

```
<html>
  <head>
    <title> User Registration </title>
    <style>
      error { color: red; }
      form-group { margin-bottom: 15px; }
    </style>
  </head>
  <body>
    <form id="registration">
      <div class="form-group">
        <label for="username"> Username </label>
        <input type="text" id="username" name="username" required>
      </div>
      <div class="form-group">
        <label for="email"> Email </label>
        <input type="text" id="email" name="email" required>
        <span id="email_error" class="error">
          </span>
      </div>
      <div class="form-group">
        <label for="password"> Password </label>
        <input type="password" id="password" name="password" required>
      </div>
    </form>
  </body>

```

```
<form>
<script>
    document.getElementById('regForm')
        .addEventListener('submit', function(event) {
            let isValid = true;
            const email = document.getElementById('email').value;
```

```
</script>
```

```
</body>
```

```
</html>
```

output:

User Registration

4. Develop a shopping cart system using javascript
the system should handle tasks such as adding
items to cart, removing items, updating quantities
calculating total price and uploading discounts.

```
let shoppingCart = [];

function addItemCart(itemID, itemName,  
itemPrice, itemQuantity){  
    const existingItem = shoppingCart.find  
        (item => item.id === itemID);  
  
    if(existingItem){  
        existingItem.quantity += itemQuantity;  
        console.log(`Updated ${itemName}`);  
    }  
    else{  
        const newItem = {  
            id: itemID,  
            name: itemName,  
            price: itemPrice,  
            quantity: itemQuantity  
        };  
        shoppingCart.push(newItem);  
        console.log(`Added ${itemName}`);  
    }  
  
    updatedCartTotal();  
}
```

```
function updateCartTotal() {
    let totalPrice = 0;
    shoppingCart.forEach(item => {
        totalPrice += item.price * item.quantity;
    });
    console.log(`Total price: ${totalPrice} to fixed()`);
}

add Item to Cart (1, "laptop", 999.99, 1);
add Item to cart (2, "phone", 499.99, 2);
add Item to cart (3, "laptop", 999.99, 1);
```

Output:

Shopping cart
Apple \$1.00 x 3

[Remove Button]

Banana - \$ 0.50 x 5

[Remove Button]

TOTAL: \$ 5.50

Apply Discount [Button]

Enter Discount code [Input field]

2. Design a responsive blog layout using CSS. The layout should handle the task such as creating a header, main content area and footer.

<html>

<head>

<title> Blog layout </title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<header>

<h1> My Blog </h1>

<nav>

 Home

 About

 Blog

 Contact

</nav>

</header>

<div class="container">

<aside> side bar content </aside>

<main> Main content area </main>

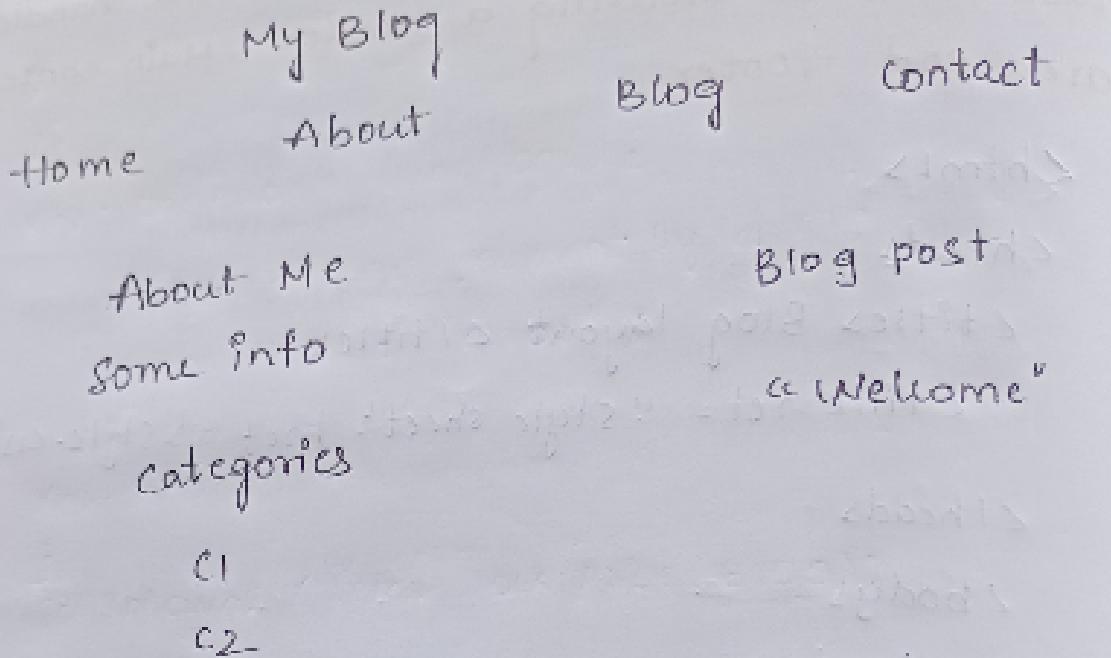
</div>

<footer> © 2024 My Blog </footer>

</body>

</html>

Output:



3. Implement a real time stock price viewer using AJAX. The system should fetch and display stock prices from a remote API. use AJAX to make a request to stock price API and display the result on the page.

<html>

<head>

<title> Real time stock Price viewer </title>

<style>

body {

font-family: Arial, sans-serif;

text-align: center;

padding: 20px;

}

\$ stock - Price {

```
font-size: 20px;  
margin-top: 20px;  
color: #333;  
}  
input { type = "text" }  
padding: 10px;  
margin-right: 10px;  
}  
button {  
padding: 10px 20px;  
}  
script  
function fetchStockPrice()  
{  
if (!companySymbol){  
alert("Please enter symbol");  
return;  
}  
const xhr = new XMLHttpRequest();  
xhr.open("Get");  
xhr.onload = function(){  
if (xhr.status == 200){  
const response = JSON.parse(xhr.responseText);  
}  
else {  
}
```

```
document.getElementById('stock-price')  
    }  
    if (xhr.status === 404) {  
        document.getElementById('stock-price').innerHTML = "Error";  
    }  
    xhr.send();  
});  
setInterval(fetchStockPrice, 6000);  
</script>  
</head>  
<body>  
<h1>Real time price</h1>  
<input type="text" id="stock-symbol"  
placeholder="Enter stock symbol" />  
</body>  
</html>
```

Output:

Stock Price viewer

Enter stock symbol : [AAPL]

Current price of AAPL : \$xxx.xx

Scenario - 1: You are building a page online resume for yourself. This page online resume for yourself, you want to style it with a specific colour scheme.

Q) How would you apply to this resume directly within the HTML file without using external CSS:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title> my resume </title>
    <style>
        body {
            margin: 0px;
            padding: 0px;
        }
        .container {
            width: 20px;
            margin: auto;
            background: #fff;
            padding: 20px;
            border-radius: 20px;
            box-shadow: 0 0 0 0px;
        }
        header {
            text-align: center;
            margin-bottom: 20px;
        }
```

Footer {

```
    text-align: center;  
    margin-top: 20px;  
    font-size: 0.9em;  
    color: #666  
}
```

```
</style>  
</head>  
<body>  
<div class = "container">  
<div class = "header">  
<h1> John </h1>  
<p> web developer </p>  
<div>  
<div class = "section">  
<h2> About me </h2>  
<p> Passionate web developer with experience  
in creating responsive and user friendly </p>  
<div>  
<div class = "section">  
<h2> skills </h2>  
<ul>  
<li> HTML, CSS, Javascript </li>  
<li> React, Node </li>  
</ul>  
</div>  
</body> </html>
```

Output:

My Resume

Deepika - web developer

Passionate web developer with
experience in creating website.

3. You have been assigned the task of establishing a development environment for a new web application project. The team is considering whether to utilize the LAMP Stack and wAMP Stack. Additionally they seek to accept the role of Apache Web Server within stack & how it integrates with other component.

Step 1: Create MySQL Database & table

CREATE DATABASE Library;

USE Library;

CREATE TABLE books (

id int AUTO_INCREMENT Primary Key,

title VARCHAR(255) NOT NULL,

author Varchar(255) NOT NULL,

published_year int NOT NULL,

);

Insert into books (title, author, published-year) value

(TO kill a mockingbird, 'Harper Lee', 1960),

('1984', 'George Orwell', 1949)

(The great gatsby, 'F.Scott Fitzgerald', 1927);

Step 2: Create PHP Script

\$servername = "localhost";

\$username = "root";

\$password = " ";

\$dbname = "Library";

\$conn = new MySQLi(\$servername, \$username, \$password, \$dbname);

```

if ($conn->connect_error) {
    die("connection failed" . $conn->connect_error);
}

$sql = "select id, title, author, published_year FROM book";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    echo "<h1>List of Books </h1>";
    echo "<table border='1'><tr><th>ID</th><th>Title</th><th>Author</th><th>Year</th></tr>";
    while ($row = $result->fetch_assoc()) {
        echo "<tr><td>" . $row["id"] . "</td><td>" . $row["title"] . "</td><td>" . $row["author"] . "</td><td>" . $row["year"] . "</td></tr>";
    }
    echo "</table>";
} else {
    echo "no books found";
}
$conn->close();

```

Step3: Save the PHP file

Step4: Access the web application.

Output:

ID	Title	Author	Year
1	To Kill a Mockingbird	Harper Lee	1960
2	1984	George Orwell	1949
3	The Great Gatsby	F. Scott Fitzgerald	1925

2) you are tasked with designing a job application form for a company's career page. The form should be user friendly & visually appealing. The form needs to capture the applicant's personal information, including their name, highest qualified degree and gender. Additionally, it should have a "submit" button to send the application and a "cancel" button to reset the form or go back to previous page.

HTML code:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=2.0">
  </head>
  <title>Job Application form</title>
  </head>
  <body>
    <h2>Job Application form</h2>
    <form>
      <label for="name">Full Name</label><br>
      <input type="text" id="name" placeholder="Enter full name" required><br><br>
      <label for="degree">Highest Qualified Degree:</label>
      <select id="degree" name="degree" required>
```

```
<option value = " " > Select your degree : <label><br>
<option value = "highschool" > High school / diploma </options>
<option value = "bachelors" > Bachelor's degree </options>
<option value = "masters" > master's degree </options>
<option value = "Phd" > phd </options>
<select> <br> <br>
<label> Gender : <label><br>
<input type = "radio" id = "male-name" name = "gender" value =
("male" required)
<label for = "male"> male </label> <br>
<input type = "radio" id = "female" name = "gender"
value = "female" required>
<label for = "female"> female </label> <br>
</form>
</body>
</html>
```

Output:

Job Application Form

Full name :

Highest Qualified degree :

Gender :

- female
- male

Submit

Cancel