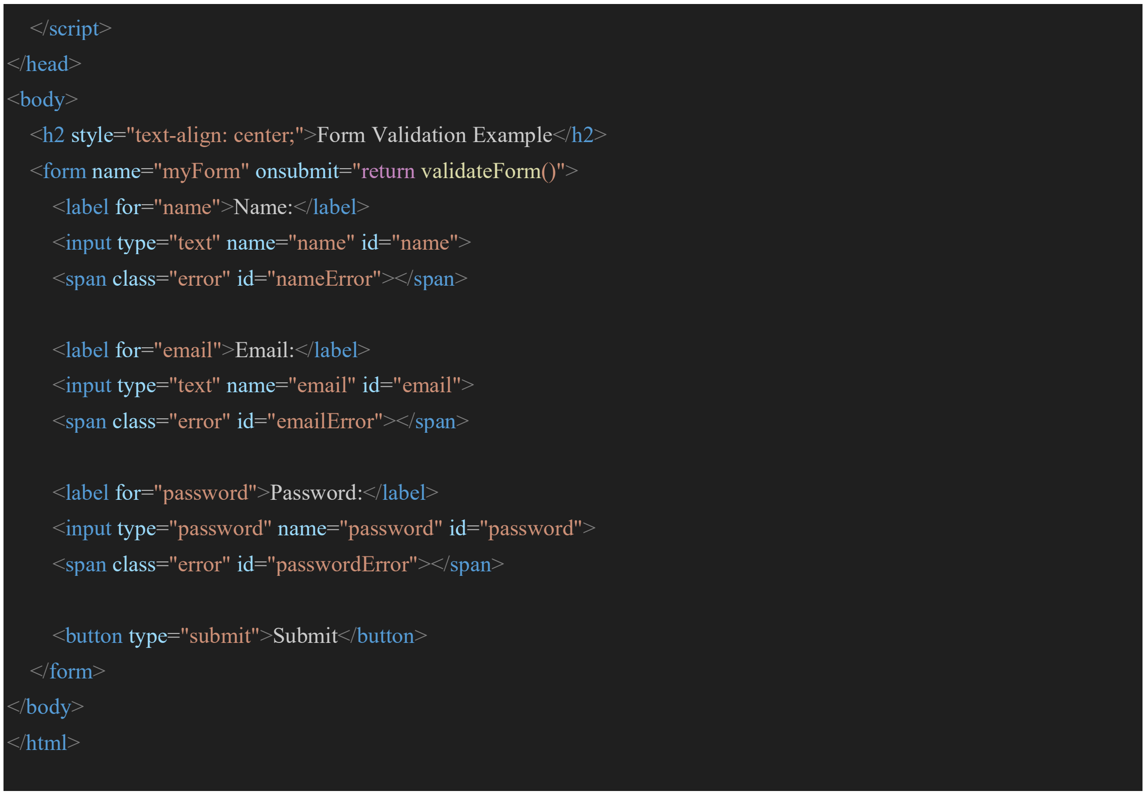
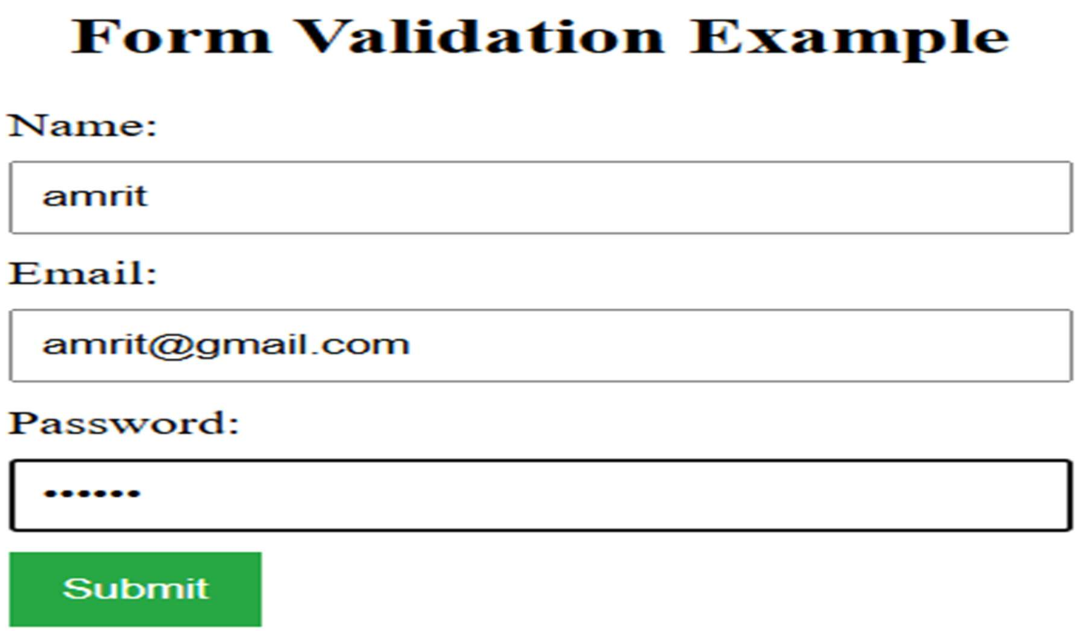
Q 11. Write a program Using Java Script validation in HTML Forms.

```
<!DOCTYPE html>
<html>
<head>
    <title>Form Validation</title>
    <script>
        function validateForm() {
            let isValid = true;

            // Clear previous error messages
            document.getElementById("nameError").innerText = "";
            document.getElementById("emailError").innerText = "";
            document.getElementById("passwordError").innerText = "";

            // Get form values
            const name = document.forms["myForm"]["name"].value;
            const email = document.forms["myForm"]["email"].value;
            const password = document.forms["myForm"]["password"].value;

            // Validate Name
            if (name === "") {
                document.getElementById("nameError").innerText = "Name is required.";
                isValid = false;
            } else if (!/^[a-zA-Z-' ]+$/.test(name)) {
                document.getElementById("nameError").innerText = "Only letters and spaces are allowed.";
                isValid = false;
            }

            // Validate Email
            if (email === "") {
                document.getElementById("emailError").innerText = "Email is required.";
                isValid = false;
            } else if (!/^\S+@\S+\.\S+$/.test(email)) {
                document.getElementById("emailError").innerText = "Invalid email format.";
                isValid = false;
            }

            // Validate Password
            if (password === "") {
                document.getElementById("passwordError").innerText = "Password is required.";
                isValid = false;
            } else if (password.length < 6) {
                document.getElementById("passwordError").innerText = "Password must be at least 6 characters long.";
                isValid = false;
            }

            return isValid; // Return true if all validations pass
        }
```



```
    </script>
</head>
<body>
    <h2 style="text-align: center;">Form Validation Example</h2>
    <form name="myForm" onsubmit="return validateForm()">
        <label for="name">Name:</label>
        <input type="text" name="name" id="name">
        <span class="error" id="nameError"></span>

        <label for="email">Email:</label>
        <input type="text" name="email" id="email">
        <span class="error" id="emailError"></span>

        <label for="password">Password:</label>
        <input type="password" name="password" id="password">
        <span class="error" id="passwordError"></span>

        <button type="submit">Submit</button>
    </form>
</body>
</html>
```

Output:



Form Validation Example

Name: amrit

Email: amrit@gmail.com

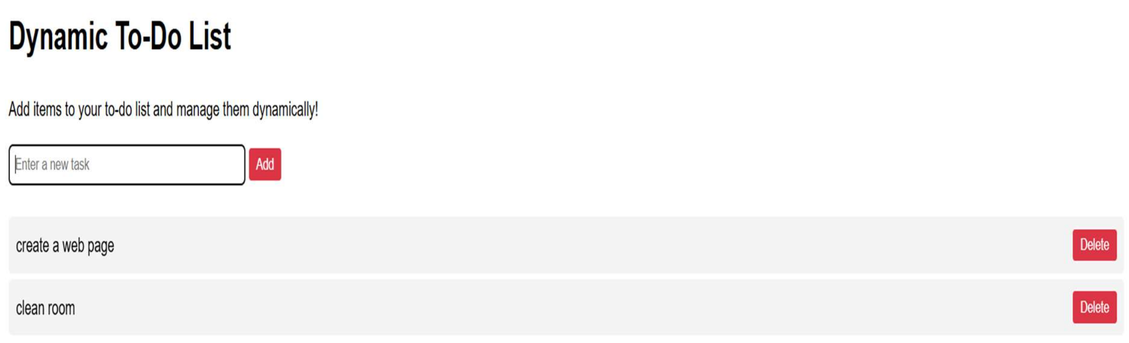
Password: ••••••

Q 12. Create a dynamic webpage with the help of JavaScript events.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Dynamic To-Do List</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 20px;
            line-height: 1.6;
        }
        #todoList {
            list-style-type: none;
            padding: 0;
        }
        #todoList li {
            padding: 10px;
            background-color: #f4f4f4;
            margin: 5px 0;
            display: flex;
            justify-content: space-between;
            align-items: center;
            border-radius: 5px;
        }
        #todoList li:hover {
            background-color: #e0e0e0;
        }
        button {
            background-color: #dc3545;
            color: white;
            border: none;
            padding: 5px 10px;
            border-radius: 3px;
            cursor: pointer;
        }
        button:hover {
            background-color: #c82333;
        }
        input {
            padding: 8px;
            width: 300px;
            margin-bottom: 10px;
            border: 1px solid #ccc;
            border-radius: 5px;
        }
    </style>
```

```
</head>
<body>
    <h1>Dynamic To-Do List</h1>
    <p>Add items to your to-do list and manage them dynamically!</p>

    <!-- Input and Add Button -->
    <input type="text" id="todoInput" placeholder="Enter a new task" />
    <button onclick="addTodo()">Add</button>

    <!-- To-Do List -->
    <ul id="todoList"></ul>

    <script>
        // Function to add a new to-do item
        function addTodo() {
            const input = document.getElementById("todoInput");
            const task = input.value.trim();

            if (task === "") {
                alert("Please enter a task!");
                return;
            }

            // Create a new list item
            const listItem = document.createElement("li");
            listItem.textContent = task;

            // Create a delete button for the list item
            const deleteButton = document.createElement("button");
            deleteButton.textContent = "Delete";
            deleteButton.onclick = function () {
                deleteTodoItem(listItem);
            };

            // Append the delete button to the list item
            listItem.appendChild(deleteButton);

            // Append the list item to the to-do list
            document.getElementById("todoList").appendChild(listItem);

            // Clear the input field
            input.value = "";
        }

        // Function to delete a to-do item
        function deleteTodoItem(item) {
            const list = document.getElementById("todoList");
            list.removeChild(item);
        }

        // Event listener for Enter key to add a to-do item
        document.getElementById("todoInput").addEventListener("keypress", function (event) {
            if (event.key === "Enter") {
                addTodo();
            }
        });
    </script>
</body>
</html>
```

Output:



Dynamic To-Do List

Add items to your to-do list and manage them dynamically!

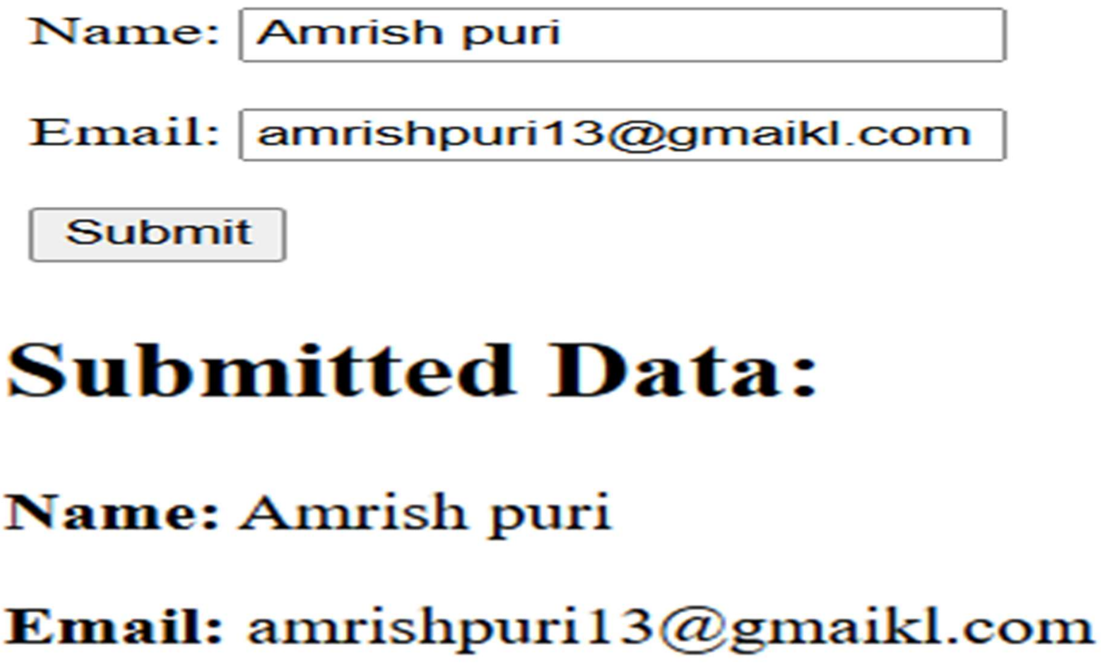
create a web page

clean room

Q 13. Introduction to JSP- Demo program.

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <title>JSP Demo - User Form</title>
</head>
<body>
    <h1>Welcome to JSP Demo</h1>
    <form method="post">
        <label for="name">Name:</label>
        <input type="text" id="name" name="name" required>
        <br><br>
        <label for="email">Email:</label>
        <input type="email" id="email" name="email" required>
        <br><br>
        <button type="submit">Submit</button>
    </form>

    <%
        // Check if form is submitted
        String name = request.getParameter("name");
        String email = request.getParameter("email");

        if (name != null && email != null) {
    %>
        <h2>Submitted Data:</h2>
        <p><strong>Name:</strong> <%= name %></p>
        <p><strong>Email:</strong> <%= email %></p>
    <%
        }
    %>
</body>
</html>
```

Output



Name: Amrish puri

Email: amrishpuri13@gmaikl.com

Submitted Data:

Name: Amrish puri

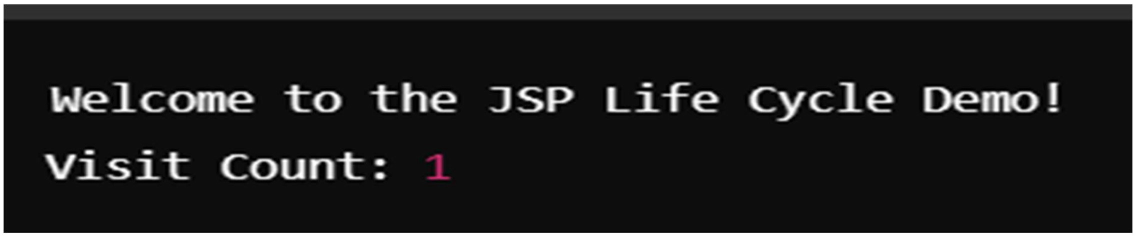
Email: amrishpuri13@gmaikl.com

Q 14. Introduction to JSP Life cycle

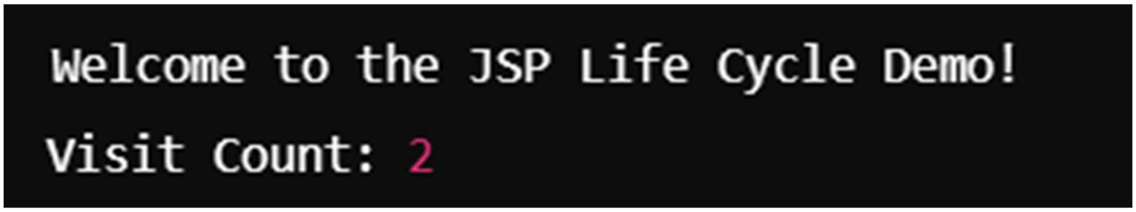
```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <title>JSP Life Cycle Demo</title>
```

```
</head>
<body>
    <h1>JSP Life Cycle Demonstration</h1>

    <%
        // JSP Life Cycle: Declaration
        int visitCount = 0; // Instance variable (maintained across requests)

        // JSP Life Cycle: Initialization
        public void jspInit() {
            visitCount = 1; // Initialize the variable
            System.out.println("JSP Initialized - jspInit() called.");
        }

        // JSP Life Cycle: Execution
        visitCount++; // Increment the visit count
        out.println("<p>Welcome to the JSP Life Cycle Demo!</p>");
        out.println("<p>Visit Count: " + visitCount + "</p>");
        System.out.println("JSP Executing - Service phase called.");

        // JSP Life Cycle: Cleanup
        public void jspDestroy() {
            System.out.println("JSP Destroyed - jspDestroy() called.");
        }
    %>
</body>
</html>
```

Output



```
Welcome to the JSP Life Cycle Demo!
Visit Count: 1
```

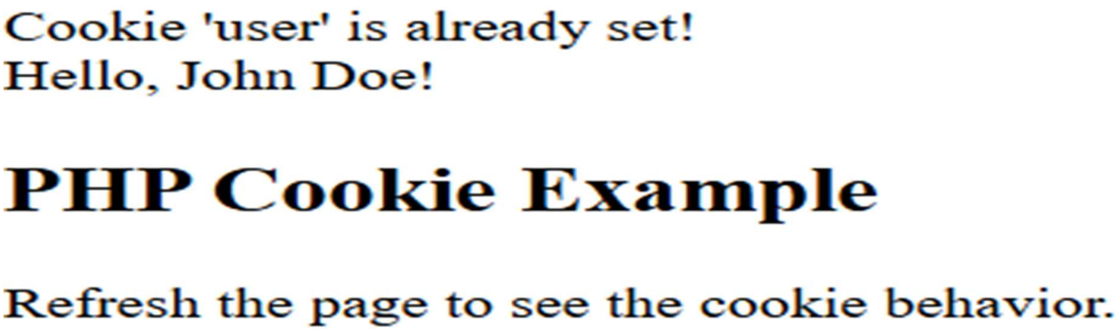


```
Welcome to the JSP Life Cycle Demo!
Visit Count: 2
```

Q 15. Write simple PHP program to set cookies and read it

```
<?php
// Set a cookie named "user" with a value "John Doe" that expires in 1 hour
if (!isset($_COOKIE['user'])) {
    setcookie('user', 'John Doe', time() + 3600, "/"); // Expires in 1 hour
    echo "Cookie 'user' has been set!<br>";
} else {
    echo "Cookie 'user' is already set!<br>";
}

// Check if the cookie is set and display the value
if (isset($_COOKIE['user'])) {
    echo "Hello, " . $_COOKIE['user'] . "!<br>";
} else {
    echo "Cookie 'user' is not set yet.<br>";
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>PHP Cookie Example</title>
</head>
<body>
    <h2>PHP Cookie Example</h2>
    <p>Refresh the page to see the cookie behavior.</p>
</body>
</html>
```

Output



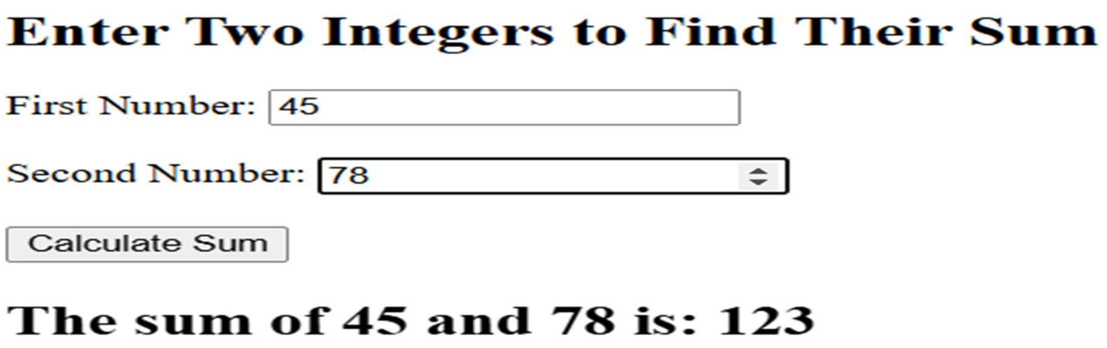
```
Cookie 'user' is already set!
Hello, John Doe!

PHP Cookie Example

Refresh the page to see the cookie behavior.
```

Q 16. Write a PHP code using function to find the sum of two integers.

```
<!DOCTYPE html>
<html lang="en">
<head>
</head>
```

```
<body>
  <h2>Enter Two Integers to Find Their Sum</h2>
  <form action="sum.php" method="post">
    <label for="num1">First Number:</label>
    <input type="number" id="num1" name="num1" required><br><br>
    <label for="num2">Second Number:</label>
    <input type="number" id="num2" name="num2" required><br><br>

    <button type="submit">Calculate Sum</button>
  </form>
  <?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  $num1 = $_POST['num1'];
  $num2 = $_POST['num2'];
  function findSum($a, $b) {
    return $a + $b;
  }

  // Call the function and calculate the sum
  $sum = findSum($num1, $num2);

  // Display the result
  echo "<h2>The sum of $num1 and $num2 is: $sum</h2>";
}
?>

</body>
</html>
```

Output



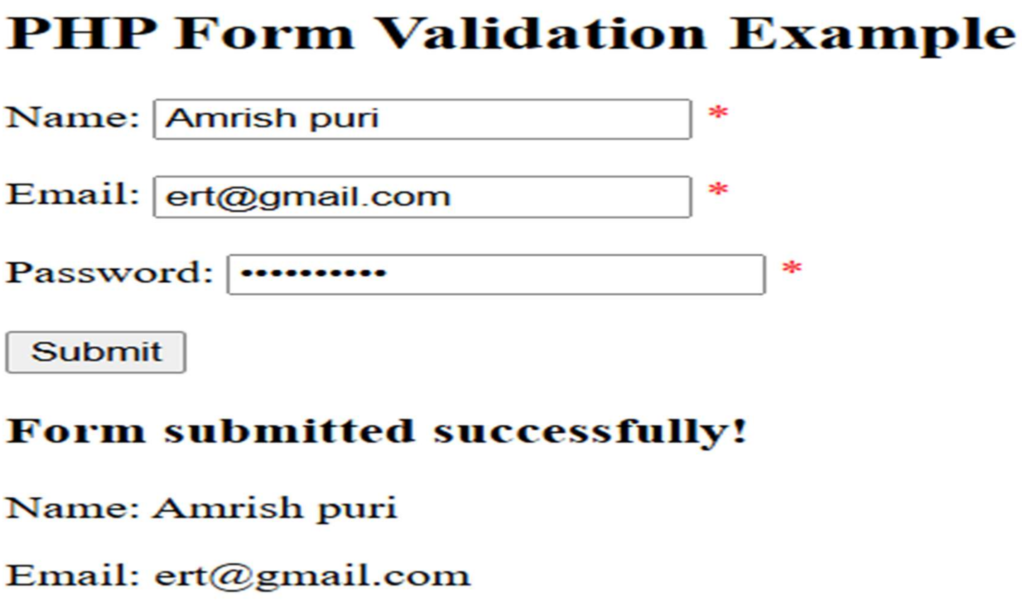
```
Enter Two Integers to Find Their Sum

First Number: 45

Second Number: 78

Calculate Sum

The sum of 45 and 78 is: 123
```

Q 17. (a) Perform from validation using PHP .(b) Write a PHP program to validate Name, Email and password

```
<!DOCTYPE html>
<html>
```

```
<head>
    <title>PHP Form Validation</title>
    <style>
        .error { color: red; }
    </style>
</head>
<body>
    <?php
    // Define variables and set to empty values
    $name = $email = $password = "";
    $nameErr = $emailErr = $passwordErr = "";

    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        // Validate Name
        if (empty($_POST["name"])) {
            $nameErr = "Name is required";
        } else {
            $name = test_input($_POST["name"]);
            if (!preg_match("/^[a-zA-Z-' ]*$/", $name)) {
                $nameErr = "Only letters and white space allowed";
            }
        }

        // Validate Email
        if (empty($_POST["email"])) {
            $emailErr = "Email is required";
        } else {
            $email = test_input($_POST["email"]);
            if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
                $emailErr = "Invalid email format";
            }
        }

        // Validate Password
        if (empty($_POST["password"])) {
            $passwordErr = "Password is required";
        } else {
            $password = test_input($_POST["password"]);
            if (strlen($password) < 6) {
                $passwordErr = "Password must be at least 6 characters long";
            }
        }
    }

    // Function to sanitize user input
    function test_input($data) {
        $data = trim($data);
        $data = stripslashes($data);
        $data = htmlspecialchars($data);
        return $data;
    }
    ?>
    <h2>PHP Form Validation Example</h2>
    <form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>">
        <label for="name">Name:</label>
        <input type="text" name="name" value="<?php echo $name; ?>">
        <span class="error">* <?php echo $nameErr; ?></span>
        <br><br>
        <label for="email">Email:</label>
        <input type="text" name="email" value="<?php echo $email; ?>">
        <span class="error">* <?php echo $emailErr; ?></span>
        <br><br>
        <label for="password">Password:</label>
        <input type="password" name="password" value="<?php echo $password; ?>">
        <span class="error">* <?php echo $passwordErr; ?></span>
        <br><br>
        <input type="submit" value="Submit">
    </form>
    <?php
    if ($_SERVER["REQUEST_METHOD"] == "POST" && !$nameErr && !$emailErr && !$passwordErr) {
        echo "<h3>Form submitted successfully!</h3>";
        echo "<p>Name: $name</p>";
        echo "<p>Email: $email</p>";
        // Note: Avoid displaying the password in a real-world scenario for security reasons.
    }
    ?>
</body>
</html>
```

Output


PHP Form Validation Example

Name: Amrish puri *

Email: ert@gmail.com *

Password: ••••••••••• *

Form submitted successfully!

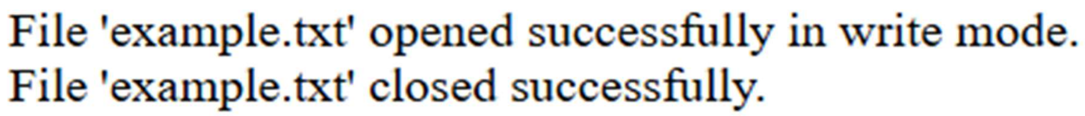
Name: Amrish puri

Email: ert@gmail.com

Q 18. File Handling in PHP (a)Create a PHP program to demonstrate opening and closing a file.

```php
<?php
// File name
$filename = "example.txt";

// Opening the file in write mode
$file = fopen($filename, "w");

// Check if the file opened successfully
if ($file) {
   echo "File '$filename' opened successfully in write mode.<br>";
   // Close the file
   fclose($file);
   echo "File '$filename' closed successfully.<br>";
} else {
   echo "Failed to open the file '$filename'.<br>";
}
?>
```

Output:

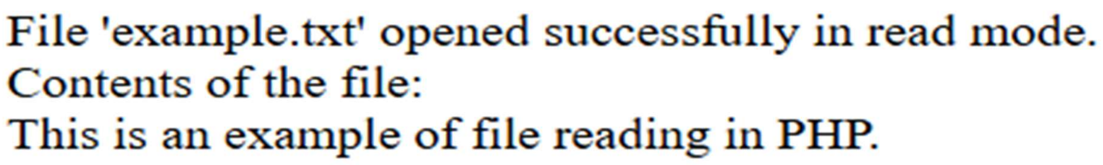

```
File 'example.txt' opened successfully in write mode.
File 'example.txt' closed successfully.
```

(b) Create a PHP program to demonstrate reading a file.

```php
<?php
```

```
// File name
$filename = "example.txt";

// Create a file and write some content for demonstration
file_put_contents($filename, "This is an example of file reading in PHP.");

// Open the file in read mode
$file = fopen($filename, "r");

// Check if the file opened successfully
if ($file) {
    echo "File '$filename' opened successfully in read mode.<br>";
    echo "Contents of the file:<br>";

    // Read the file until the end
    while (!feof($file)) {
        $line = fgets($file); // Read a line from the file
        echo $line . "<br>";
    }

    // Close the file
    fclose($file);
} else {
    echo "Failed to open the file '$filename'.<br>";
}
?>
```

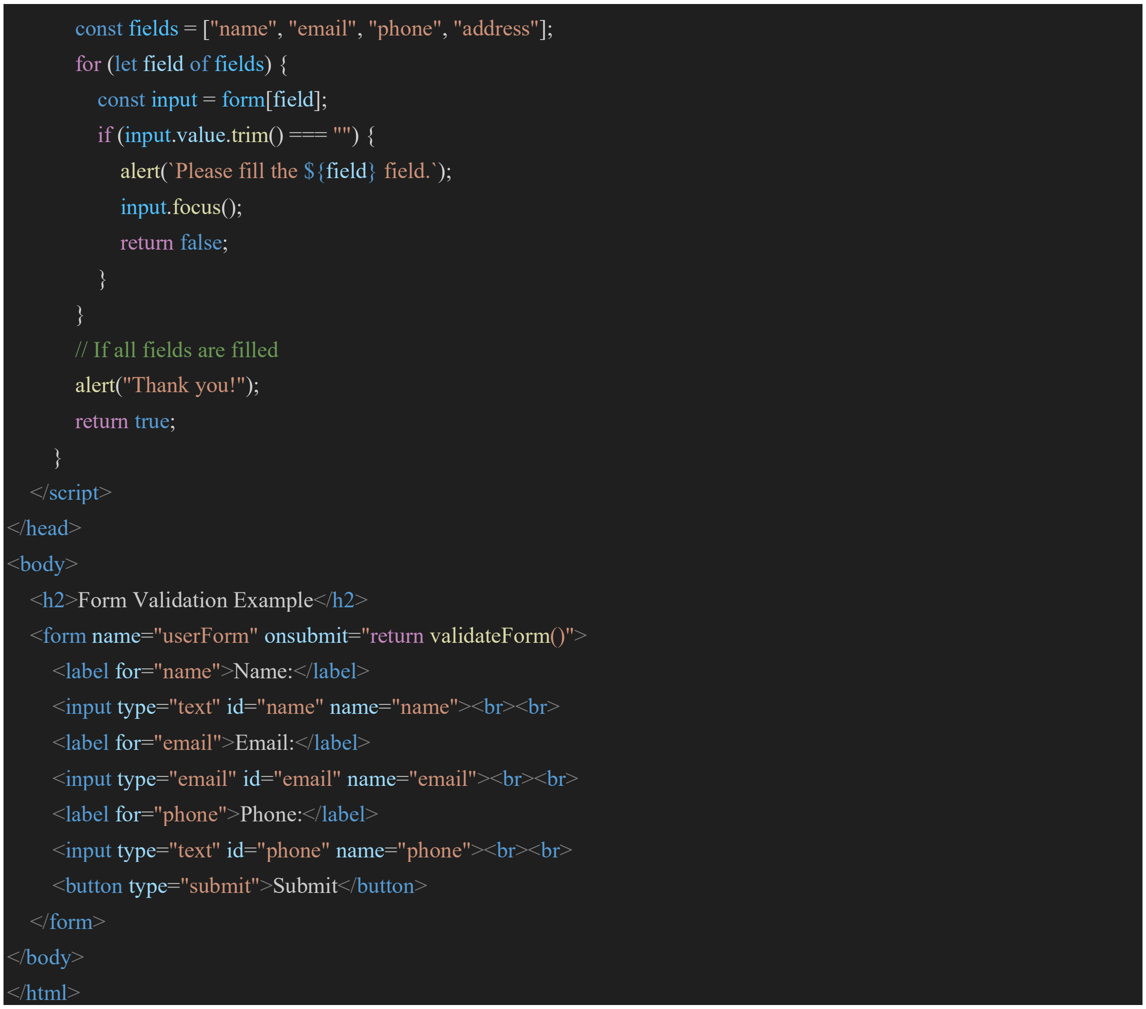
Output



```
File 'example.txt' opened successfully in read mode.
Contents of the file:
This is an example of file reading in PHP.
```

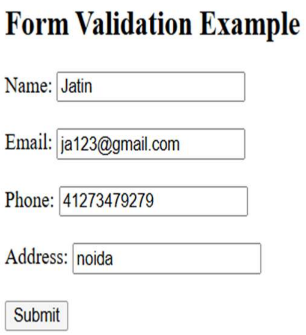
Q 19. Create an HTML form that has a number of text boxes. The user fills the textboxes with data. Write a script that verifies that all textboxes have been filled. If a text box has been left empty pop up an alert message indicating the box that has been left empty. When OK button is clicked, set focus to that specific textbox. If all the textboxes are filled, display thank you.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <script>
        function validateForm() {
            // Get all input fields
            const form = document.forms["userForm"];
```



```
            const fields = ["name", "email", "phone", "address"];
            for (let field of fields) {
                const input = form[field];
                if (input.value.trim() === "") {
                    alert(`Please fill the ${field} field.`);
                    input.focus();
                    return false;
                }
            }
            // If all fields are filled
            alert("Thank you!");
            return true;
        }
    </script>
</head>
<body>
    <h2>Form Validation Example</h2>
    <form name="userForm" onsubmit="return validateForm()">
        <label for="name">Name:</label>
        <input type="text" id="name" name="name"><br><br>
        <label for="email">Email:</label>
        <input type="email" id="email" name="email"><br><br>
        <label for="phone">Phone:</label>
        <input type="text" id="phone" name="phone"><br><br>
        <button type="submit">Submit</button>
    </form>
</body>
</html>
```

Output:



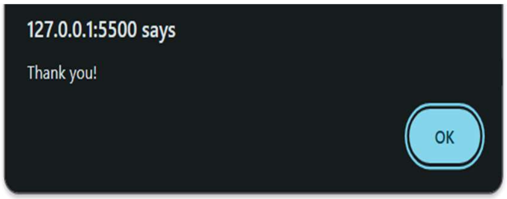
Form Validation Example

Name: Jatin

Email: ja123@gmail.com

Phone: 41273479279

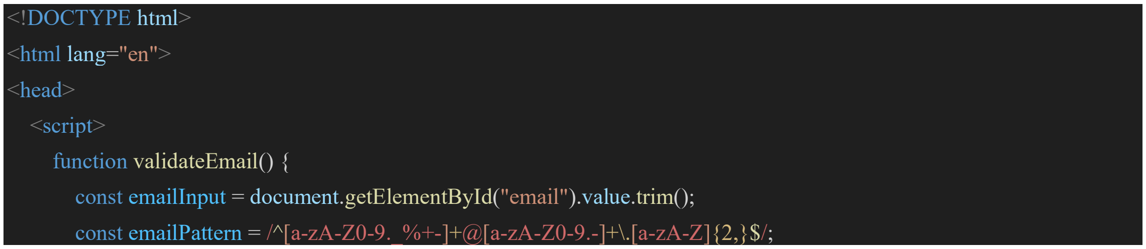
Address: noida



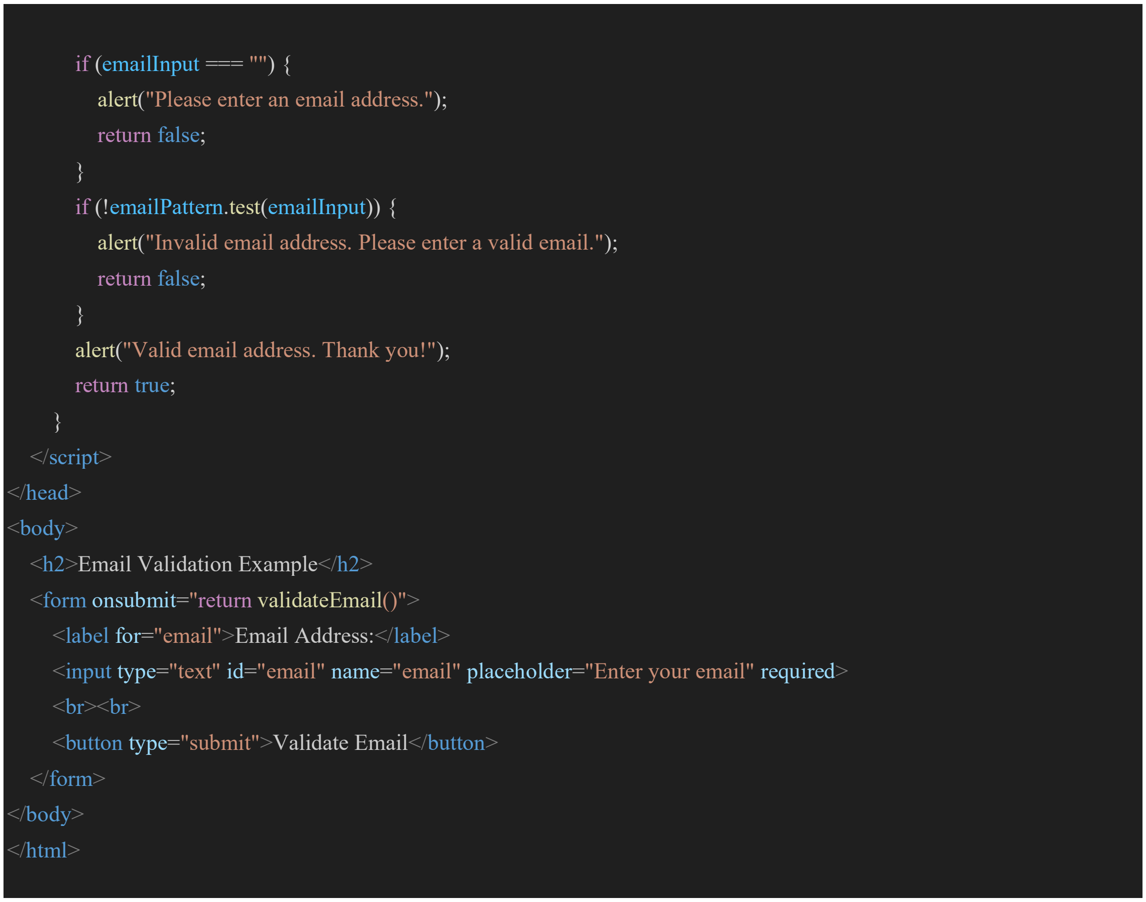
127.0.0.1:5500 says

Thank you!

Q 20. Create a program to verify whether email address provided by the user is valid or invalid.

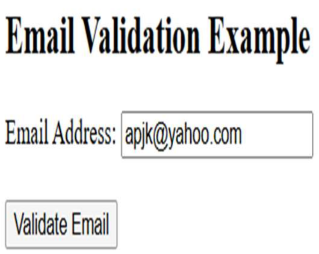


```
<!DOCTYPE html>
<html lang="en">
<head>
    <script>
        function validateEmail() {
            const emailInput = document.getElementById("email").value.trim();
            const emailPattern = /^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/;
```



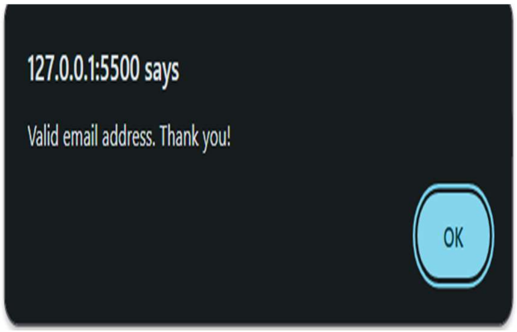
```
            if (emailInput === "") {
                alert("Please enter an email address.");
                return false;
            }
            if (!emailPattern.test(emailInput)) {
                alert("Invalid email address. Please enter a valid email.");
                return false;
            }
            alert("Valid email address. Thank you!");
            return true;
        }
    </script>
</head>
<body>
    <h2>Email Validation Example</h2>
    <form onsubmit="return validateEmail()">
        <label for="email">Email Address:</label>
        <input type="text" id="email" name="email" placeholder="Enter your email" required>
        <br><br>
        <button type="submit">Validate Email</button>
    </form>
</body>
</html>
```

Output:



Email Validation Example

Email Address: apjk@yahoo.com



127.0.0.1:5500 says

Valid email address. Thank you!