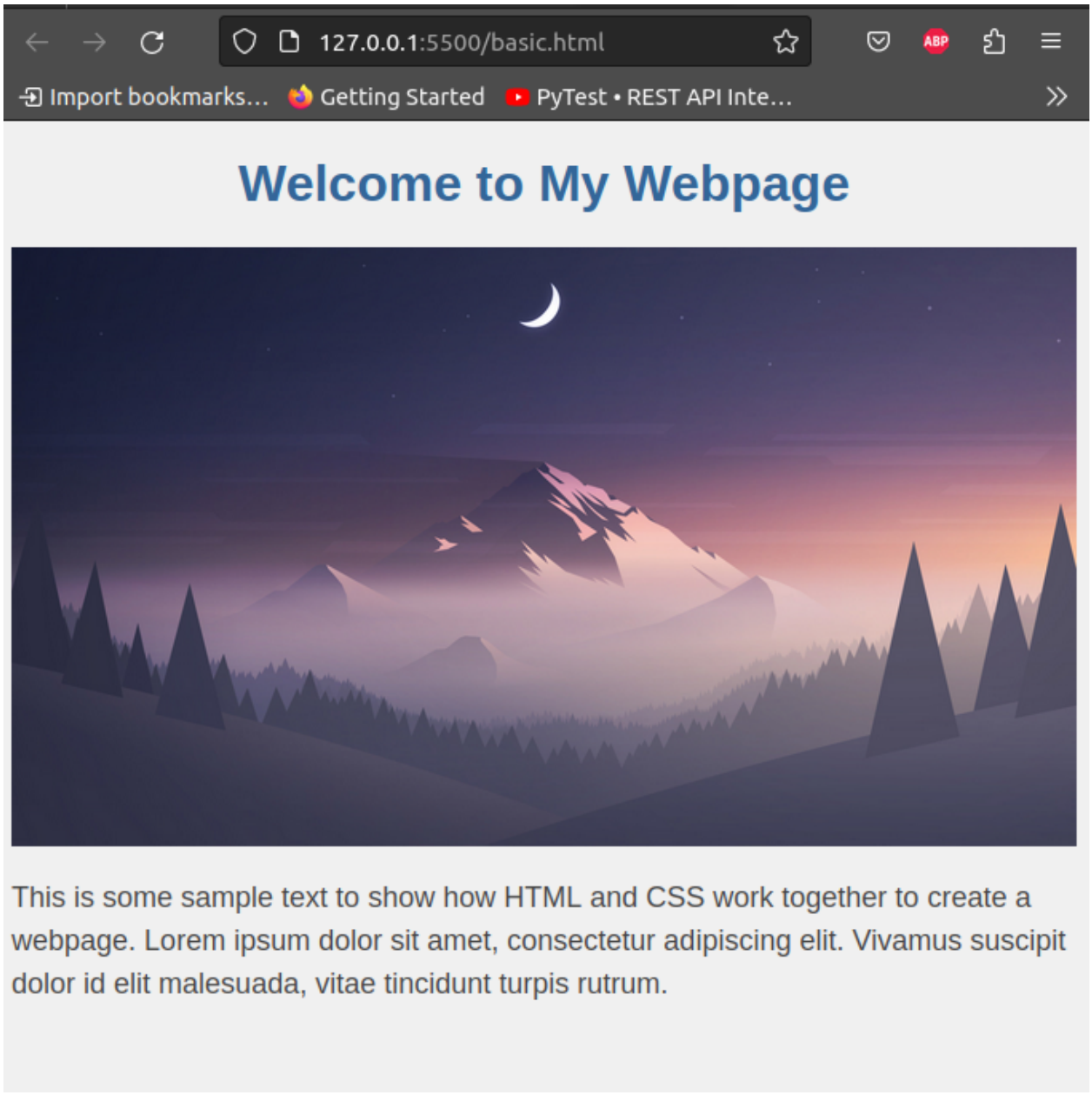
Table of Contents

Implementation of HTML tags and CSS properties

```
<!DOCTYPE html>
<html>
<head>
  <title>My Webpage</title>
  <style>
    body {
      background-color: #f0f0f0;
      font-family: Arial, sans-serif;
    }

    h1 {
      color: #336699;
      text-align: center;
    }

    p {
      color: #444444;
      font-size: 18px;
      line-height: 1.5;
      margin-bottom: 20px;
    }

    img {
      display: block;
      margin: 0 auto;
      max-width: 100%;
      height: auto;
    }
  </style>
</head>
<body>
  <h1>Welcome to My Webpage</h1>
  <img src="./15.jpg" alt="Placeholder Image">
  <p>This is some sample text to show how HTML and CSS work together to create a webpage. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus suscipit dolor id elit malesuada, vitae tincidunt turpis rutrum.</p>
</body>
</html>
```



Types of CSS

Inline CSS

```
<!DOCTYPE html>
<html>
<head>
 <title>Inline CSS Example</title>
```

```
</head>
<body>
 <h1 style="color: blue; font-size: 2em;">This is a heading with inline CSS</h1>
 <p style="color: red;">This is a paragraph with inline CSS</p>
</body>
</html>
```

Internal CSS

```
<!DOCTYPE html>
<html>
<head>
<title>Internal CSS Example</title>
<style>
body {
background-color: lightblue;
font-family: Arial, sans-serif;
font-size: 16px;
color: white;
}
h1 {
font-size: 30px;
color: navy;
text-align: center;
margin-top: 50px;
}
</style>
</head>
<body>
<h1>Welcome to my website!</h1>
<p>This is an example of internal CSS.</p>
</body>
</html>
```

External CSS

#html

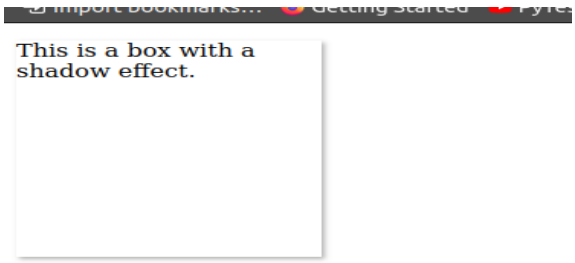
```
<!DOCTYPE html>
<html>
<head>
<title>External CSS Example</title>
<link rel="stylesheet" type="text/css" href="styles.css">
</head>
```

```
<body>
<h1>Welcome to my website!</h1>
<p>This is an example of external CSS.</p>
</body>
</html>
#css

body {


background-color: lightblue;

font-family: Arial, sans-serif;

font-size: 16px;

color: white;
}

h1 {

font-size: 30px;

color: navy;

text-align: center;

margin-top: 50px;
}
```

Box Shadow in CSS

```
<!DOCTYPE html>
<html>
<head>
 <title>Box Shadow Example</title>
 <style>
   .box {
     width: 200px;
     height: 200px;
     background-color: #fff;
     box-shadow: 2px 2px 4px rgba(0,0,0,0.3);
   }
 </style>
</head>
```

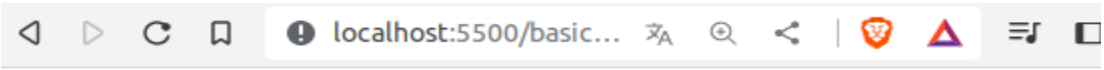
```
<body>
 <div class="box">
  <p>This is a box with a shadow effect.</p>
 </div>
</body>
</html>
```



Responsive Designs using media queries (media types, viewports)

```
<!DOCTYPE html>
<html>
<head>
 <title>Responsive Design Example</title>
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <style>
  /* Styles for desktop devices */
  h1 {
   font-size: 3em;
   margin-bottom: 20px;
  }

  p {
   font-size: 1.5em;
   line-height: 1.5;
   margin-bottom: 20px;
  }

  /* Styles for mobile devices */
  @media screen and (max-width: 767px) {
   h1 {
    font-size: 2em;
```

```
    }

    p {
      font-size: 1.2em;
    }
  }
 </style>
</head>
<body>
 <h1>Welcome to our website</h1>
 <p>This is our website.</p>
</body>
</html>
```



Welcome to our website

This is our website.

Inline Javascript

```
<!DOCTYPE html>
<html>
<head>
  <title>Inline JavaScript</title>
</head>
<body>
  <button onclick="alert('Hello, world!')">Click me</button>
</body>
</html>
```

External Javascript

```
// script.js
function changeColor() {
  document.getElementById("myText").style.color = "red";
}
#html
<!-- index.html -->
<!DOCTYPE html>
<html>
<head>
	<title>External JavaScript</title>
	<script src="script.js"></script>
</head>
<body>
	<p id="myText" onclick="changeColor()">Click me to change my color!</p>
</body>
</html>
```

Javascript Functions

```
function addNumbers(a, b) {
  return a + b;
}

console.log(addNumbers(2, 3)); // Output: 5
```

Document Object Model (DOM)

Finding HTML elements by ID

```
<!DOCTYPE html>
<html>
<head>
	<title>Find Element by ID</title>
</head>
<body>
	<p id="myParagraph">Hello, world!</p>
	<script>
		var element = document.getElementById("myParagraph");
		element.style.color = "red";
	</script>
</body>
</html>
```

Finding elements by TagName

```
<!DOCTYPE html>
<html>
<head>
	<title>Find Elements by Tag Name</title>
</head>
<body>
	<p>Paragraph 1</p>
	<p>Paragraph 2</p>
	<script>
		var elements = document.getElementsByTagName("p");
		for (var i = 0; i < elements.length; i++) {
			elements[i].style.color = "blue";
		}
	</script>
</body>
</html>
```

Finding HTML elements by Class Name

```
<!DOCTYPE html>
<html>
<head>
	<title>Find Elements by Class Name</title>
	<style>
		.myClass {
			color: green;
		}
	</style>
</head>
<body>
	<p class="myClass">Paragraph 1</p>
	<p class="myClass">Paragraph 2</p>
	<script>
		var elements = document.getElementsByClassName("myClass");
		for (var i = 0; i < elements.length; i++) {
			elements[i].style.fontWeight = "bold";
		}
	</script>
</body>
</html>
```

JavaScript Objects and Methods

```
var person = {
```

```
  firstName: "John",
  lastName: "Doe",
  age: 30,
  fullName: function() {
    return this.firstName + " " + this.lastName;
  }
};

console.log(person.firstName); // Output: John
console.log(person.age); // Output: 30
console.log(person.fullName()); // Output: John Doe
```

```
# output
John
30
John Doe
```

Creating empty object and adding properties

```
var car = {};
car.make = "Honda";
car.model = "Civic";
car.year = 2022;

console.log(car.make); // Output: Honda
console.log(car.year); // Output: 2022
```

Implementing ‘this’ keyword in object

```
var person = {
  firstName: "John",
  lastName: "Doe",
  fullName: function() {
    return this.firstName + " " + this.lastName;
  }
};

console.log(person.fullName()); // Output: John Doe
```

Conditional Statements in JavaScript

1. **Example of if condition**
2. **if- else condition**
3. **if- else ladder**
4. **Switch statement**

1. If condition

```
var age = 18;

if (age >= 18) {
  console.log("You are eligible to vote.");
}
```

2. If-else condition

```
var age = 15;

if (age >= 18) {
  console.log("You are eligible to vote.");
} else {
  console.log("You are not eligible to vote yet.");
}
```

3. If-else ladder

```
var score = 85;

if (score >= 90) {
  console.log("You got an A!");
} else if (score >= 80) {
  console.log("You got a B!");
} else if (score >= 70) {
  console.log("You got a C!");
} else if (score
```

4. Switch case

```
var day = "Monday";

switch (day) {
  case "Monday":
    console.log("Today is Monday.");
    break;
  case "Tuesday":
    console.log("Today is Tuesday.");
    break;
  case "Wednesday":
```

```
    console.log("Today is Wednesday.");
    break;
  case "Thursday":
    console.log("Today is Thursday.");
    break;
  case "Friday":
    console.log("Today is Friday.");
    break;
  case "Saturday":
    console.log("Today is Saturday.");
    break;
  case "Sunday":
    console.log("Today is Sunday.");
    break;
  default:
    console.log("Invalid day.");
    break;
}
```

Loops in JavaScript

for loop
while loop
Break statement
continue statement

1. **For loop with break**

```
for (var i = 1; i <= 10; i++) {
  if (i === 5) {
    break;
  }
  console.log(i);
}
```

2. **For loop with continue**

```
for (var i = 1; i <= 10; i++) {
  if (i === 5) {
   continue;
  }
  console.log(i);
}
```

3. **while loop with continue**

```
var i = 0
while(i < 10)
```

```
{
if (i == 5){
continue:
}
console.log(i);
i++;
}
```

Functions in JavaScript

```
function findMax(numbers) {
  var max = numbers[0];
  for (var i = 1; i < numbers.length; i++) {
    if (numbers[i] > max) {
      max = numbers[i];
    }
  }
  return max;
}

var myNumbers = [3, 7, 2, 10, 5];
var maxNumber = findMax(myNumbers);

console.log("The maximum number is " + maxNumber);
```

```
# Output
The maximum number is 10
```

52
54

Methods used in array

1. push ()
```
var fruits = ["apple", "banana", "orange"];
fruits.push("grape");
console.log(fruits);
// Output: ["apple", "banana", "orange", "grape"]
```
2. pop()
```
var fruits = ["apple", "banana", "orange"];
var lastFruit = fruits.pop();
console.log(lastFruit); // Output: "orange"
console.log(fruits);
 // Output: ["apple", "banana"]
```
3. sort()
```
var fruits = ["orange", "apple", "banana", "grape", "mango"];
fruits.sort();
console.log(fruits);
// Output: ["apple", "banana", "grape", "mango", "orange"]
```

4. slice()
   var fruits = ["apple", "banana", "orange", "grape", "mango"];
   var slicedFruits = fruits.slice(2, 4);
   console.log(slicedFruits); // Output: ["orange", "grape"]

Form validation in HTML with Javascript

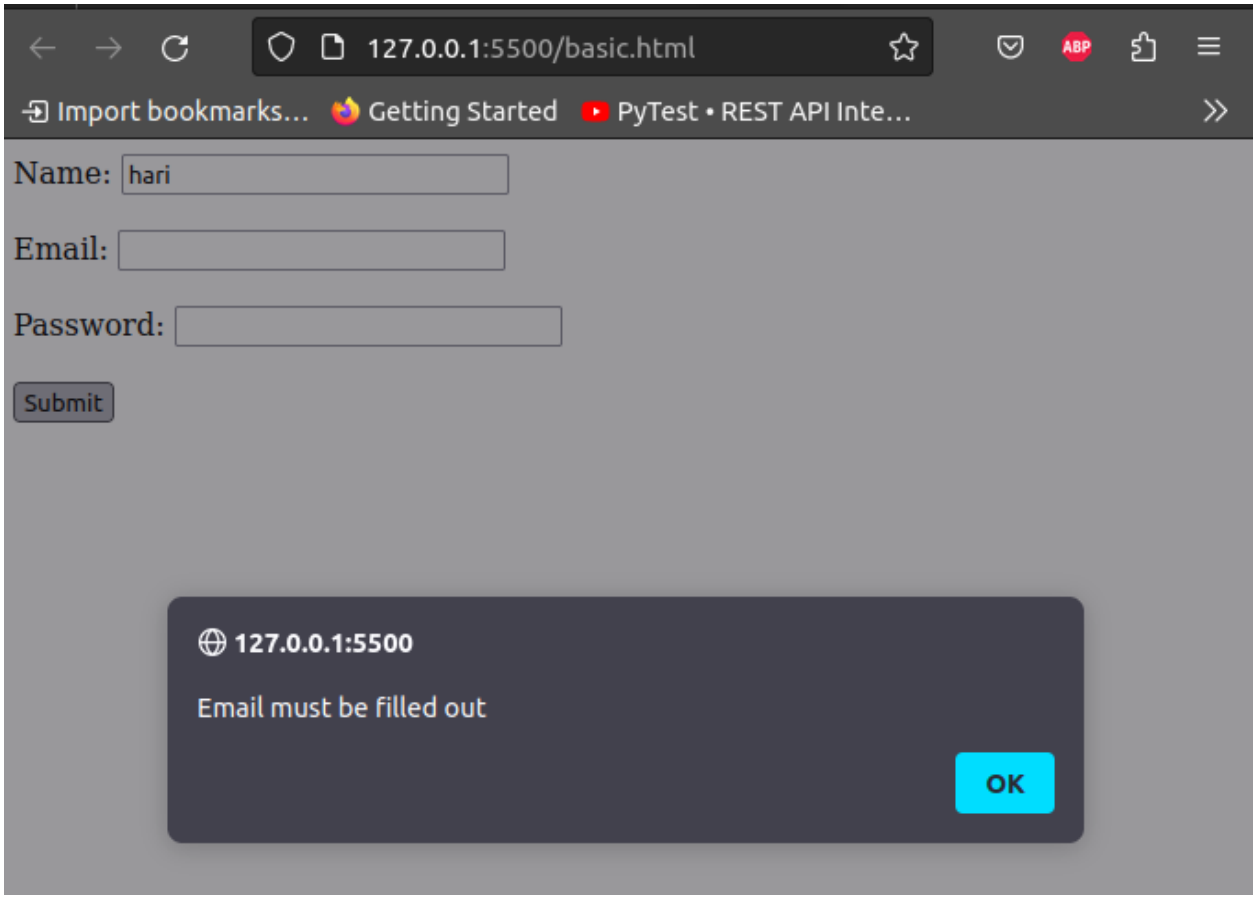
html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <script src="./inline_js.js"></script>
</head>
<body>

<form>
 <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="password">Password:</label>
 <input type="password" id="password" name="password"><br><br>
 <button type="button" onclick="validateForm()">Submit</button>

</form>
</body>

</html>
```

javascript

```
function validateForm() {
  var name = document.forms[0]["name"].value;
  var email = document.forms[0]["email"].value;
  var password = document.forms[0]["password"].value;
  if (name == "") {
    alert("Name must be filled out");
    return false;
  }
  if (email == "") {
    alert("Email must be filled out");
    return false;
  }
```

```
  if (password == "") {
    alert("Password must be filled out");
    return false;
  }
  if (password.length < 8) {
    alert("Password must be at least 8 characters long");
    return false;
  }
  alert("Form validated successfully");
  return true;
}
```



Jquery

```
<!DOCTYPE html>
<html>
<head>
  <title>jQuery Example</title>
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
  <script>
    $(document).ready(function() {
      $("button").click(function() {
        $("p").hide();
      });
    });
  </script>
</head>
<body>

  <h1>jQuery Example</h1>

  <p>This is a paragraph.</p>
  <p>This is another paragraph.</p>

  <button>Hide Paragraphs</button>

</body>
</html>
```

XML document creation

```
<?xml version="1.0" encoding="UTF-8"?>
<employees>
  <employee>
    <id>1001</id>
    <name>John Doe</name>
    <department>Marketing</department>
    <salary>50000</salary>
  </employee>
  <employee>
    <id>1002</id>
    <name>Jane Smith</name>
    <department>Finance</department>
    <salary>60000</salary>
  </employee>
</employees>
```

XSL transformation

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  <xsl:template match="/">
    <html>
      <body>
        <h2>Employee Information</h2>
        <table border="1">
          <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Department</th>
            <th>Salary</th>
          </tr>
          <xsl:for-each select="employees/employee">
            <tr>
              <td><xsl:value-of select="id"/></td>
              <td><xsl:value-of select="name"/></td>
              <td><xsl:value-of select="department"/></td>
              <td><xsl:value-of select="salary"/></td>
            </tr>
          </xsl:for-each>
        </table>
      </body>
    </html>
  </xsl:template>

</xsl:stylesheet>
```

Internal DTD creation

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE catalog [
  <!ELEMENT catalog (book+)>
  <!ELEMENT book (title, author, year, price)>
  <!ELEMENT title (#PCDATA)>
  <!ELEMENT author (#PCDATA)>
  <!ELEMENT year (#PCDATA)>
  <!ELEMENT price (#PCDATA)>
]>
<catalog>
  <book>
    <title>Harry Potter and the Philosopher's Stone</title>
    <author>J.K. Rowling</author>
    <year>1997</year>
    <price>20.00</price>
  </book>
  <book>
    <title>The Great Gatsby</title>
    <author>F. Scott Fitzgerald</author>
    <year>1925</year>
    <price>15.00</price>
  </book>
</catalog>
```

External DTD creation

 # sample.dtd

```
<!ELEMENT catalog (book+)>
<!ELEMENT book (title, author, year, price)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT author (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT price (#PCDATA)>
```

 # sample.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE catalog SYSTEM "books.dtd">
<catalog>
  <book>
    <title>Harry Potter and the Philosopher's Stone</title>
```

```
    <author>J.K. Rowling</author>
    <year>1997</year>
    <price>20.00</price>
  </book>
  <book>
    <title>The Great Gatsby</title>
    <author>F. Scott Fitzgerald</author>
    <year>1925</year>
    <price>15.00</price>
  </book>
</catalog>
```

XML schema creation

#sample.xsd

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="catalog">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="book" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="title" type="xs:string"/>
              <xs:element name="author" type="xs:string"/>
              <xs:element name="year" type="xs:int"/>
              <xs:element name="price" type="xs:decimal"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

#sample.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<catalog xmlns:xsi=”www.w3.com”  xsi:noNamespaceSchemaLocation="books.xsd">
  <book>
    <title>Harry Potter and the Philosopher's Stone</title>
    <author>J.K. Rowling</author>
    <year>1997</year>
```

```
    <price>20.00</price>
  </book>
  <book>
    <title>The Great Gatsby</title>
    <author>F. Scott Fitzgerald</author>
    <year>1925</year>
    <price>15.00</price>
  </book>
</catalog>
```

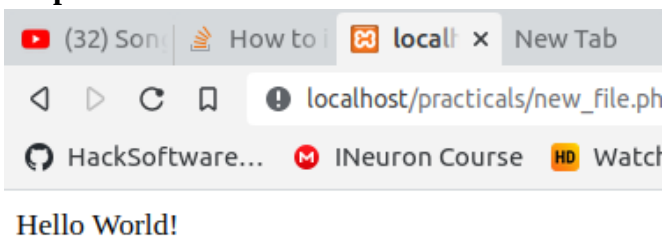
Lab Report on PHP

PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages.PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP.

Basic PHP Program: **Printing Hello World**

```
<?php
echo "Hello World!";
?>
```

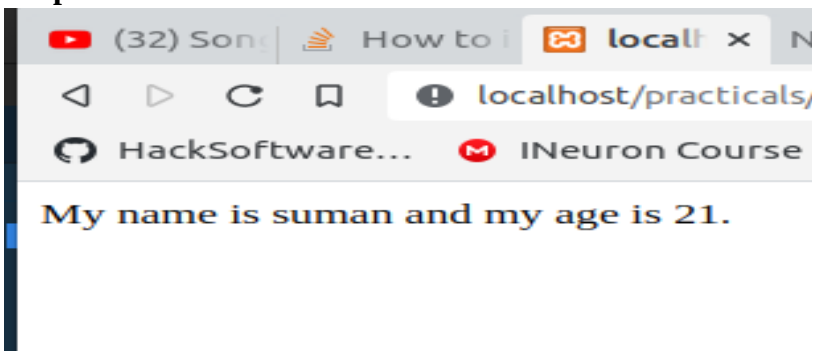
Output



Using Variable in PHP

```
<?php
$name="suman";
$age=21;

echo "My name is $name and my age is $age.";
?>
```

Output



PHP Data Types

Data Types define the type of data a variable can store. PHP allows eight different types of data types. All of them are discussed below. There are pre-defined, user-defined, and special data types.

The predefined data types are:

- Boolean
- Integer
- Double
- String

The user-defined (compound) data types are:

- Array
- Objects

The special data types are:

- NULL
- resource

String functions

There are various functions that is associated with strings in PHP. But some of the mostly used functions are:

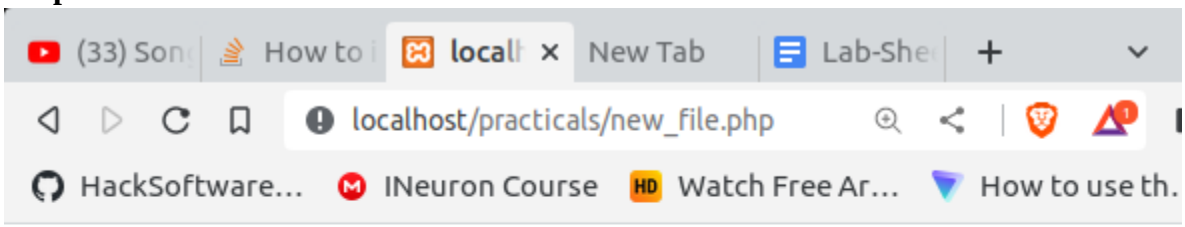
strlen(): This function gives the length of the given string.
str_word_count(): This function gives the number of words present in the given string.
strrev(): This function reverses the given string.
strpos($var, “word”): This function finds the position of the given word in the given variable.
str_replace(“word1”, “word2”, $var): This function replaces word1 with word2 in the given variable.

Implementation of some of the functions of string in PHP

```
<?php
$stmt="It's a skateboarding penguin with a sunhat!";
echo $stmt;
echo "<br>";

echo "The length of the statement is ".strlen($stmt);
echo "<br>";
echo "No. of words= ".str_word_count($stmt);
echo "<br>";
echo "Reverse of the above statement: ".strrev($stmt);
echo "<br>";
echo "Where does the word skateboarding starts from? ".strpos($stmt,
"skateboarding");
```

```
echo "<br>";
echo str_replace("sunhat", "sunglass", $stmt);
echo "<br>";

?>
```

Output



It's a skateboarding penguin with a golegang!
The length of the statement is 45
No. of words= 7
Reverse of the above statement: !gnagelog a htiw niugnep gnidraobetaks a s'tI
Where does the word skateboarding starts from? 7
It's a skateboarding penguin with a tamanggang!

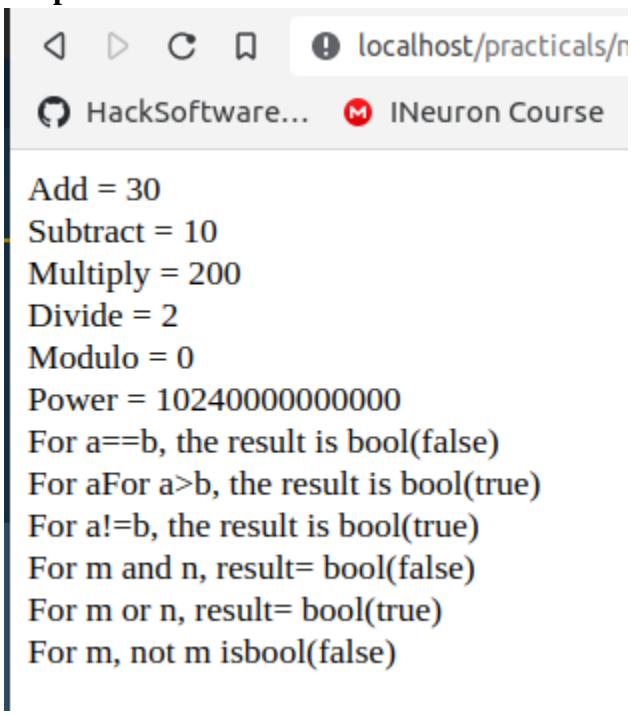
Operators in PHP

Operators are same in PHP as in other programming languages. Implementation of some operators in PHP is as follows:

```
<?php
        $a=20;
        $b=10;
        //Arithmetic Operators
        echo "Add= ". ($a+$b)."<br>";          //Addition
        echo "Subtract= ". ($a-$b)."<br>";
        echo "Multiply= ". ($a*$b)."<br>";
        echo "Divide= ". ($a/$b)."<br>";
        echo "Modulo= ". ($a%$b)."<br>";
        echo "Power= ". ($a**$b)."<br>";

        //Comparison Operators
        echo "For a==b, the result is ";
```

```
        echo var_dump($a==$b);
        echo "<br>";

        echo "For a<b, the result is ";
        echo var_dump($a<$b);
        echo "<br>";

        echo "For a>b, the result is ";
        echo var_dump($a>$b);
        echo "<br>";

        echo "For a!=b, the result is ";
        echo var_dump($a<>$b);
        echo "<br>";

        //Logical Operator
        $m=true;
        $n=false;

        echo "For m and n, result= ";
        echo var_dump($m&&$n);
        echo "<br>";

        echo "For m or n, result= ";
        echo var_dump($m||$n);
        echo "<br>";

        echo "For m, not m is";
        echo var_dump(!$m);
        echo "<br>";

?>
```

Output



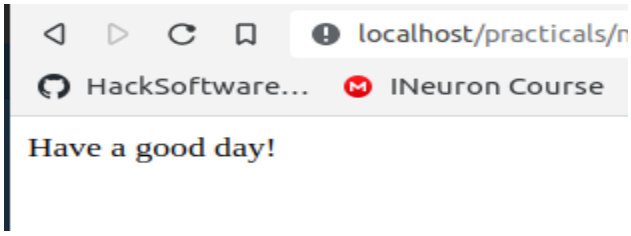
Control Structure in PHP

if- else condition

To print “Have a good day” if time is less than 20:00 and “Have a good day” if time is more tha 20:00

```
<?php
$t = date("H");
if ($t < "22") {
        echo "Have a good day!";
} else {
        echo "Have a good night!";
}
?>
```

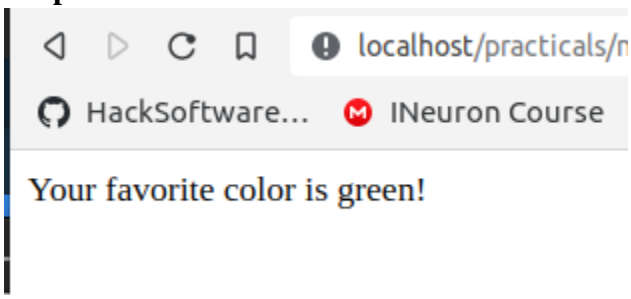
Output



Switch statement

```
<?php
$favcolor = "red";

switch ($favcolor) {
  case "red":
    echo "Your favorite color is red!";
    break;
  case "blue":
    echo "Your favorite color is blue!";
    break;
  case "green":
    echo "Your favorite color is green!";
    break;
  default:
    echo "Your favorite color is neither red, blue, nor green!";
}
```

Output



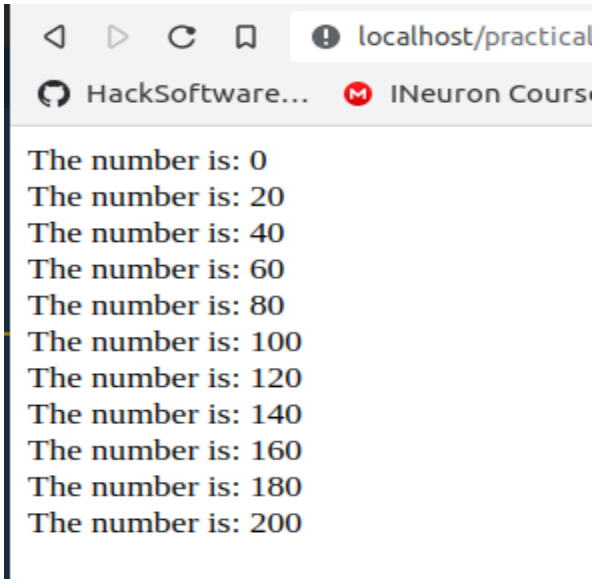
PHP Loops

While loop

To print the multiple of 10 less than 100

```
<?php
$x = 0;

while($x <= 100) {
  echo "The number is: $x <br>";
  $x+=10;
}
?>
```

Output

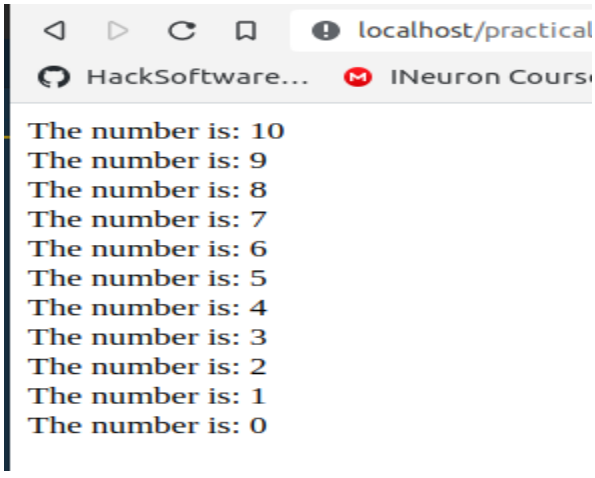


do-while loop

To print the numbers from 10 to 0

```
<?php
$x = 10;

do {
  echo "The number is: $x <br>";
  $x--;
} while ($x >= 0);
?>
```

Output

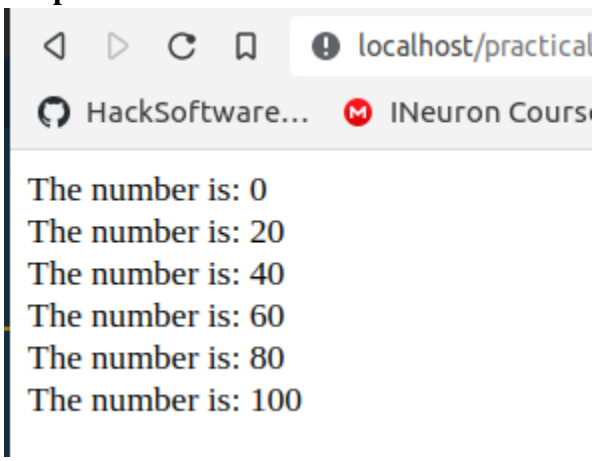


for loop

Prints the multiple of 20 less than 100

```
<?php
for ($x = 0; $x <= 100; $x+=20) {
  echo "The number is: $x <br>";
}
?>
```

Output

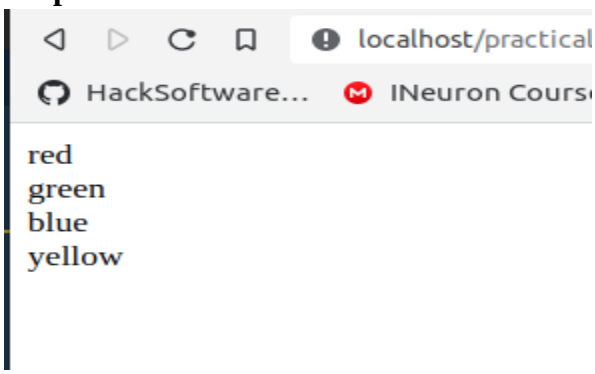


for-each loop

Prints all the elements of an array

```
<?php
$colors = array("red", "green", "blue", "yellow");

foreach ($colors as $value) {
  echo "$value <br>";
}
?>
```

Output

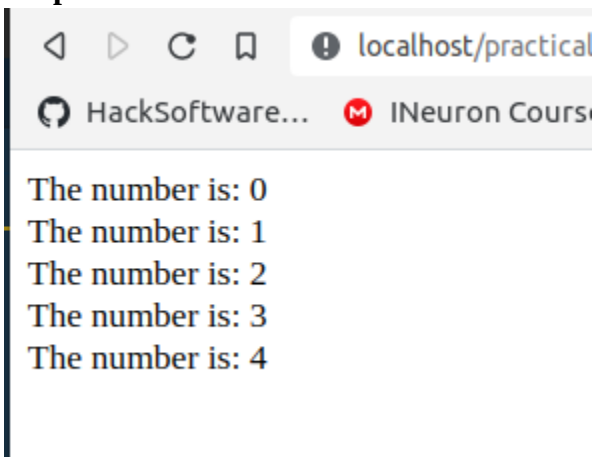


Break

This example jumps out of the loop when **x** is equal to **5**

```
<?php
for ($x = 0; $x < 10; $x++) {
  if ($x == 4) {
    break;
  }
  echo "The number is: $x <br>";
}
?>
```

Output



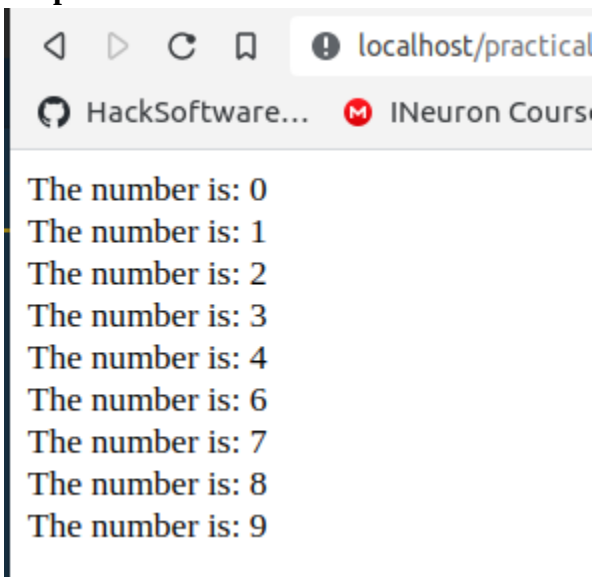
Continue

The continue statement breaks one iteration (in the loop), if a specified condition occurs, and continues with the next iteration in the loop.

This example skips the value of **4**.

```
<?php
for ($x = 0; $x < 10; $x++) {
  if ($x == 5) {
    continue;
  }
  echo "The number is: $x <br>";
}?>
```

Output

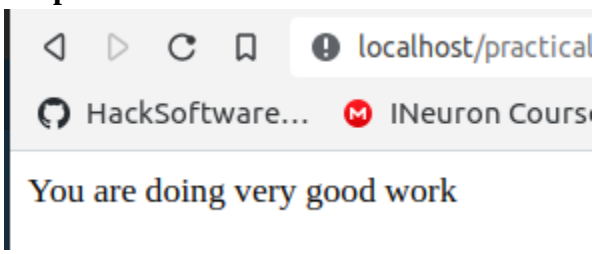


Functions in PHP

Function to greet

```
<?php
function writeMsg() {
  echo "You are doing very good work";
}

writeMsg();
?>
```

Output

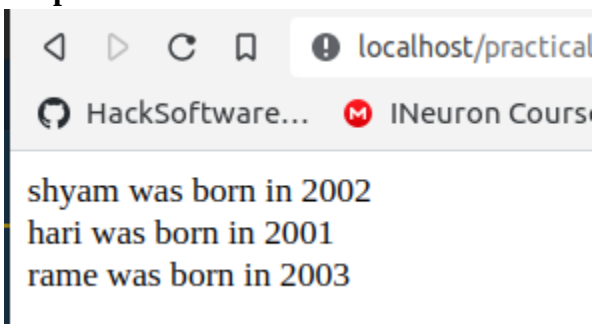


Function with Arguments

```
<?php
function familyName($fname, $year) {
 echo "$fname was born in $year <br>";
}
familyName("shyam", "2002");
familyName("hari", "2001");
```

```
familyName("rame", "2003");
?>
```

Output

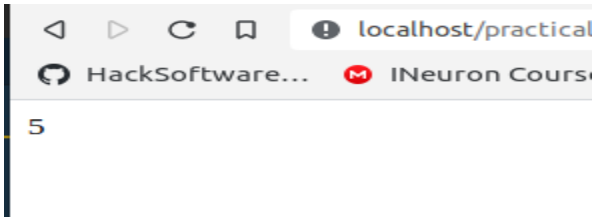


Array

count(): returns the number of elements in an array

```
<?php
$cars = array("Volvo", "BMW", "Toyota", "Mahindra", "Tata");
echo count($cars);
?>
```

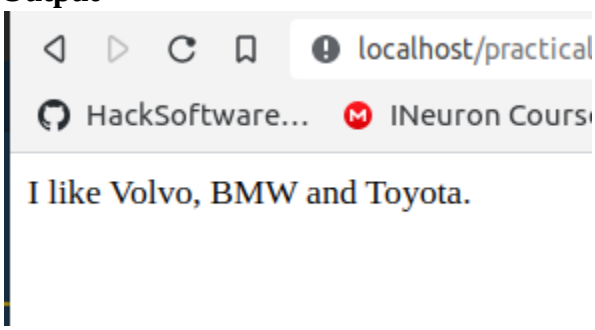
Output



Indexed Array example

```
<?php
$cars = array("Volvo", "BMW", "Toyota");
echo "I like " . $cars[0] . ", " . $cars[1] . " and " . $cars[2] . ".";
?>
```

Output

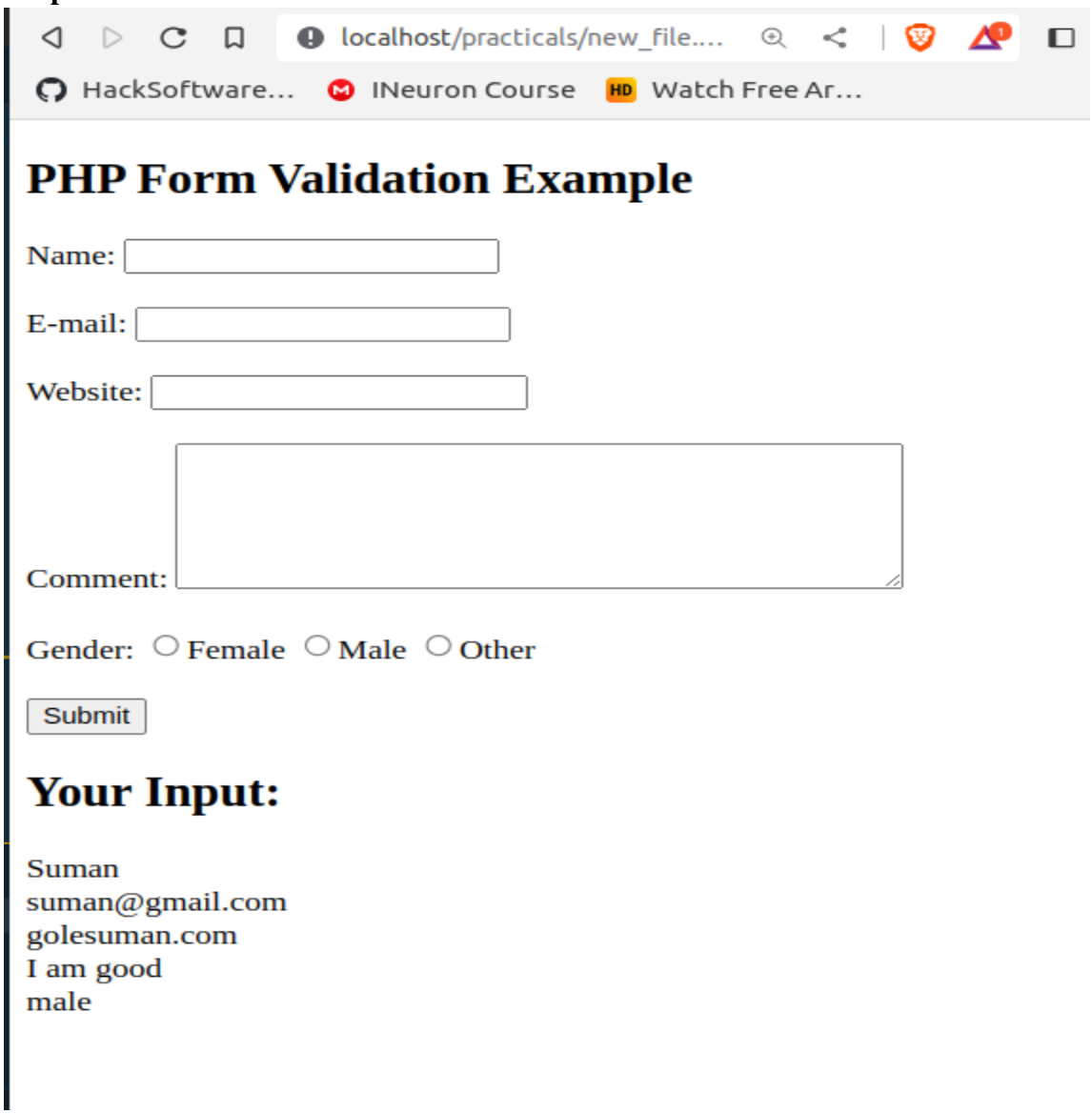


Form Validation using PHP

```
<!DOCTYPE HTML>
<html>
<head>
</head>
<body>
	<?php
	// define variables and set to empty values
	$name = $email = $gender = $comment = $website = "";

	if ($_SERVER["REQUEST_METHOD"] == "POST") {
		$name = test_input($_POST["name"]);
		$email = test_input($_POST["email"]);
		$website = test_input($_POST["website"]);
		$comment = test_input($_POST["comment"]);
		$gender = test_input($_POST["gender"]);
	}

	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"]); ?>">
		Name: <input type="text" name="name">
		<br><br>
		E-mail: <input type="text" name="email">
		<br><br>
		Website: <input type="text" name="website">
		<br><br>
		Comment: <textarea name="comment" rows="5"
cols="40"></textarea>
		<br><br>
		Gender:
```

```
            <input type="radio" name="gender" value="female">Female
            <input type="radio" name="gender" value="male">Male
            <input type="radio" name="gender" value="other">Other
            <br><br>
            <input type="submit" name="submit" value="Submit">
        </form>

        <?php
        echo "<h2>Your Input:</h2>";
        echo $name;
        echo "<br>";
        echo $email;
        echo "<br>";
        echo $website;
        echo "<br>";
        echo $comment;
        echo "<br>";
        echo $gender;
        ?>

</body>

</html>
```

Output



After Submitting,

Add data to Database with PHP

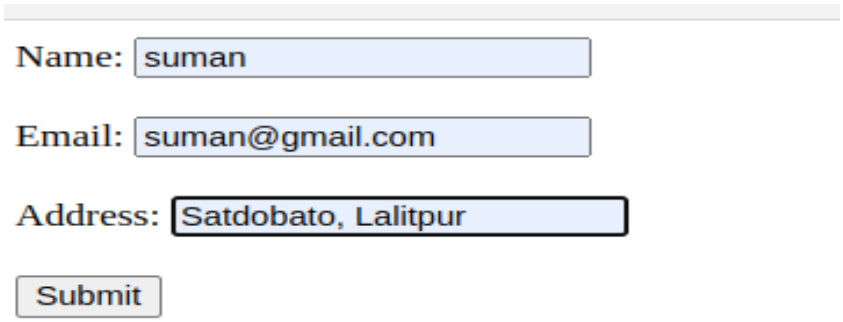
form

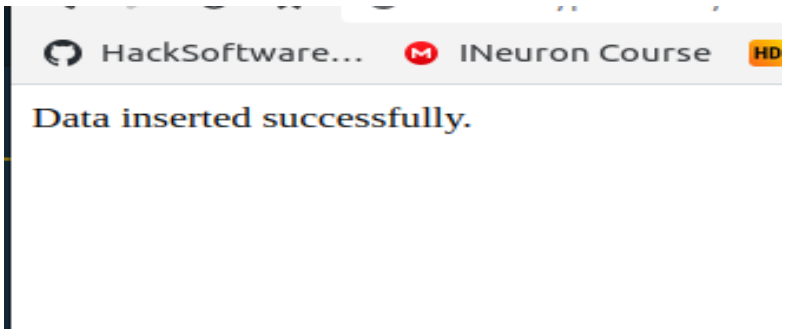
```
<!DOCTYPE html>
<html>
<head>
  <title>Form Example</title>
</head>
<body>
  <form action="./students.php" method="post">
    <label>Name:</label>
    <input type="text" name="name"><br><br>

    <label>Email:</label>
    <input type="email" name="email"><br><br>

    <label>Address:</label>
    <input type="text" name="address"><br><br>

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

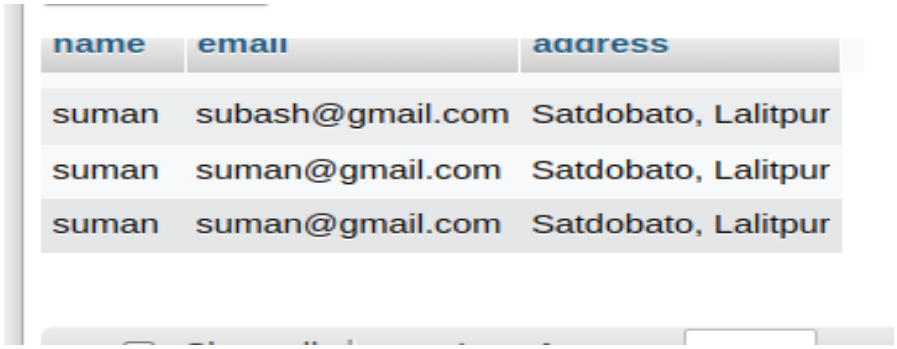
php

```
<?php

// Connect to the database
$host = 'localhost';
$user = 'root';
$password = '';
$dbname = 'test_db';
$conn = mysqli_connect($host, $user, $password, $dbname);

// Check connection
if (!$conn) {
 die("Connection failed: " . mysqli_connect_error());
}

// Retrieve form data
```

```
$name = $_POST['name'];
$email = $_POST['email'];
$address = $_POST['address'];

// Insert data into database
$sql = "INSERT INTO students (name, email, address) VALUES ('$name', '$email', '$address')";

if (mysqli_query($conn, $sql)) {
 echo "Data inserted successfully.";
} else {
 echo "Error: " . mysqli_error($conn);
}

// Close connection
mysqli_close($conn);
```







name	email	address
suman	subash@gmail.com	Satdobato, Lalitpur
suman	suman@gmail.com	Satdobato, Lalitpur
suman	suman@gmail.com	Satdobato, Lalitpur