MIT ART DESIGN & TECHNOLOGY UNIVERSITY

MIT College of Management (MITCOM), Pune

# PROGRAMME: MASTER OF COMPUTER APPLICATION (MCA CC /DS)

**CERTIFICATE**

This is to certify that, Mr./Miss ­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_has submitted a Practical Report on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ to MIT – ADT University, Pune for the partial fulfillment of Master in Computer Application (Data Science/ Cloud Computing) submitted during the academic year 2024-25.

PRN No. MCA Year MCA Sem. –

Dr.Alkawati Magadum Dr.Sangita Phunde

Subject Incharge HOD MCA Principal

**External Examiner** **Sign of Examiners:**

**1.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

**Internal Examiner** **Sign of Examiners:**

**2.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

MIT ART DESIGN & TECHNOLOGY UNIVERSITY

MIT College of Management (MITCOM), Pune

**Declaration**

I undersigned hereby declares that, the Journal of assignments solved by me and it is executed as per the course requirement of MCA program of MIT-ADT University, Pune. This report has not submitted by me or any other person to any other University or Institution for a degree or diploma course. This is my own and original work.

Place: MITCOM, Pune Date:

Sign of the student: ----------------------------

Name of the Student

# MIT ART DESIGN & TECHNOLOGY UNIVERSITY

**MIT College of Management (MITCOM), Pune**

**Sub:-**

**Name: -** Div:- MCA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Title of the Practicals** | **Page** | **Date** | **Record Sign** |
| 1. | Write a PHP Program in CodeIgniter to determine given number is Even or ODD. |  |  |  |
| 2. | Write a PHP Program in CodeIgniter to check if a given number is divisible by 3, and display an appropriate message. |  |  |  |
| 3. | Write a PHP Program in CodeIgniter to displays the name of the day based on a given number. |  |  |  |
| 4. | Write a PHP Program in CodeIgniter to evaluate a score and display the corresponding grade using CodeIgniter. |  |  |  |
| 5. | Write a PHP Program in CodeIgniter to calculates the sum of natural numbers up to a specified limit. |  |  |  |
| 6 | Write a PHP Program in CodeIgniter to generates and displays a multiplication table for a specified number using do while loop. |  |  |  |
| 7 | Write a PHP Program in CodeIgniter to calculates the factorial of a given number using a for loop. |  |  |  |
| 8 | Write a PHP Program in CodeIgniter to that generates the Fibonacci series up to a specified number of terms. |  |  |  |
| 9 | Write a PHP Program in CodeIgniter to that iterates through an array of student names and displays them using simple array. |  |  |  |
| 10 | Write a PHP Program in CodeIgniter to Write a PHP program to create an indexed array of fruits and display them. |  |  |  |
| 11 | Write a PHP Program in CodeIgniter to calculate the length of String. |  |  |  |
| 12 | Write a PHP Program in CodeIgniter to count the number of words in string without using string functions |  |  |  |
| 13 | Write a PHP Program in CodeIgniter to to demonstrate use of various built-in string functions. |  |  |  |
| 14 | Create a CodeIgniter PHP program that demonstrates inheritance with an Animal superclass (with properties name and age and a speak() method ) and a Dog subclass that overrides speak() to include the dog's name and age. |  |  |  |
| 15 | Write a PHP Program in CodeIgniter to Create a Car\_model class with a constructor to initialize properties like make, model, and year etc. |  |  |  |
| 16 | Write a PHP program in CodeIgniter to design a web page featuring a text box for name input, radio buttons for selecting a contact method (Email or Phone), check boxes for choosing interests (Sports, Music, Reading), and buttons for submitting or resetting the form. |  |  |  |
| 17 | Write a simple PHP program in CodeIgniter that demonstrates introspection and serialization. Use a class to create an object, and then showcase how to inspect its properties and methods using PHP's reflection |  |  |  |
| 18 | Write a PHP program in CodeIgniter to implement session management and cookie handling for a user login system |  |  |  |
| 19 | Write a PHP program in CodeIgniter to perform the following tasks: a) Create a form to enter user information (name and email) and save this data into a database. b) Retrieve and display the saved user information in a table format on a separate page. |  |  |  |
| 20 | Write a PHP program in CodeIgniter to develop a simple application that allows users to Update existing records by modifying user information (e.g., name and email). |  |  |  |

**Practical NO 1:**

**Write a PHP Program in CodeIgniter to determine given number is Even or ODD.**

**1) Controller** (NumberCheck)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class NumberCheck extends CI\_Controller {

public function index() {

$this->load->view('number\_check\_form');

}

public function check() {

$number = $this->input->post('number');

if ($number % 2 == 0) {

$result = "$number is Even.";

} else {

$result = "$number is Odd.";

}

$data['result'] = $result;

$this->load->view('number\_check\_result', $data);

}

}

?>

**2) View:-**

* Number\_check\_form:

<!DOCTYPE html>

<html>

<head>

<title>Even or Odd Checker</title>

</head>

<body>

<h1>Even or Odd Checker</h1>

<form method="post" action="<?php echo site\_url('NumberCheck/check'); ?>">

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

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

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

</form>

</body>

</html>

* Number\_check\_result:

<!DOCTYPE html>

<html>

<head>

<title>Result</title>

</head>

<body>

<h1>Result</h1>

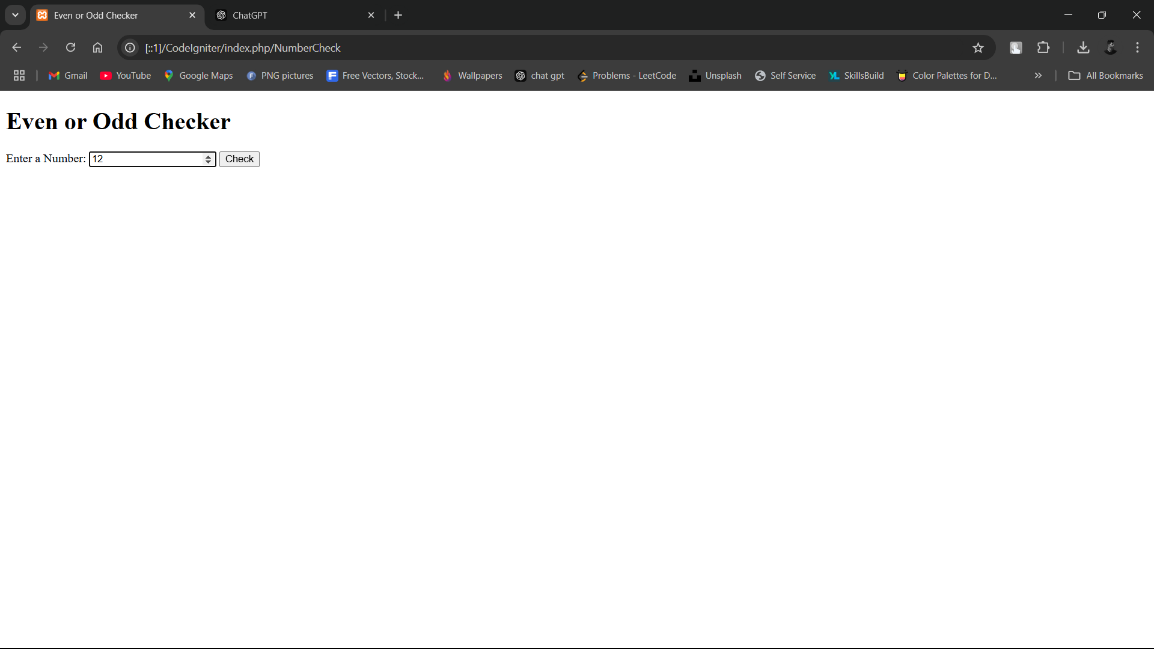
<p><?php echo $result; ?></p>

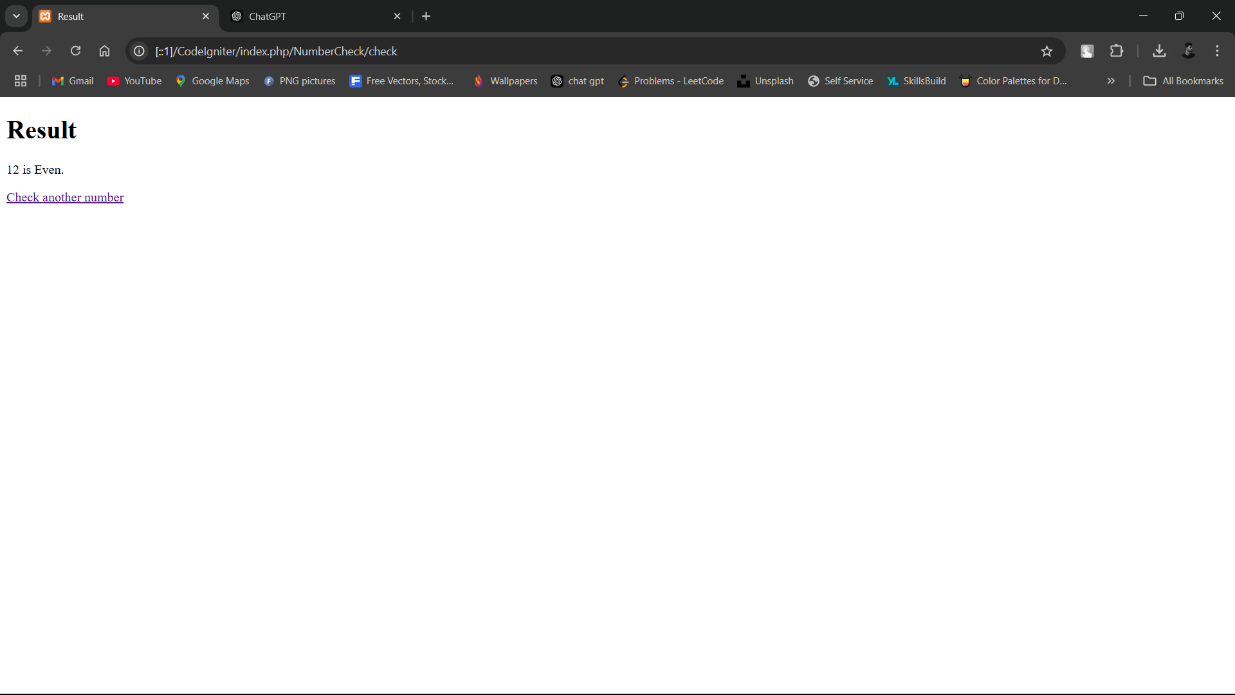
<a href="<?php echo site\_url('NumberCheck'); ?>">Check another number</a>

</body>

</html>

**Output**:





**Practical NO 2:**

**Write a PHP Program in CodeIgniter to check if a given number is divisible by 3, and display an appropriate message.**

1) **Controller** (DivisibilityCheck.php)

<?php

defined('BASEPATH') OR *exit*('No direct script access allowed');

class DivisibilityCheck extends CI\_Controller {

    public function index() {

        $this->load->view('divisibility\_check\_form');

    }

    public function check() {

        $number = $this->input->post('number');

*if* ($number % 3 == 0) {

            $result = "$number is divisible by 3.";

        } *else* {

            $result = "$number is not divisible by 3.";

        }

        $data['result'] = $result;

        $this->load->view('divisibility\_check\_result', $data);

    }

}

?>

2) **View**:-

* divisibility\_check\_form:

<!DOCTYPE html>

<html>

<head>

    <title>Divisibility Checker</title>

</head>

<body>

    <h1>Divisibility Checker</h1>

    <form method="post" action="<?php echo site\_url('DivisibilityCheck/check'); ?>">

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

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

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

    </form>

</body>

</html>

* divisibility\_check\_result:

<!DOCTYPE html>

<html>

<head>

<title>Result</title>

</head>

<body>

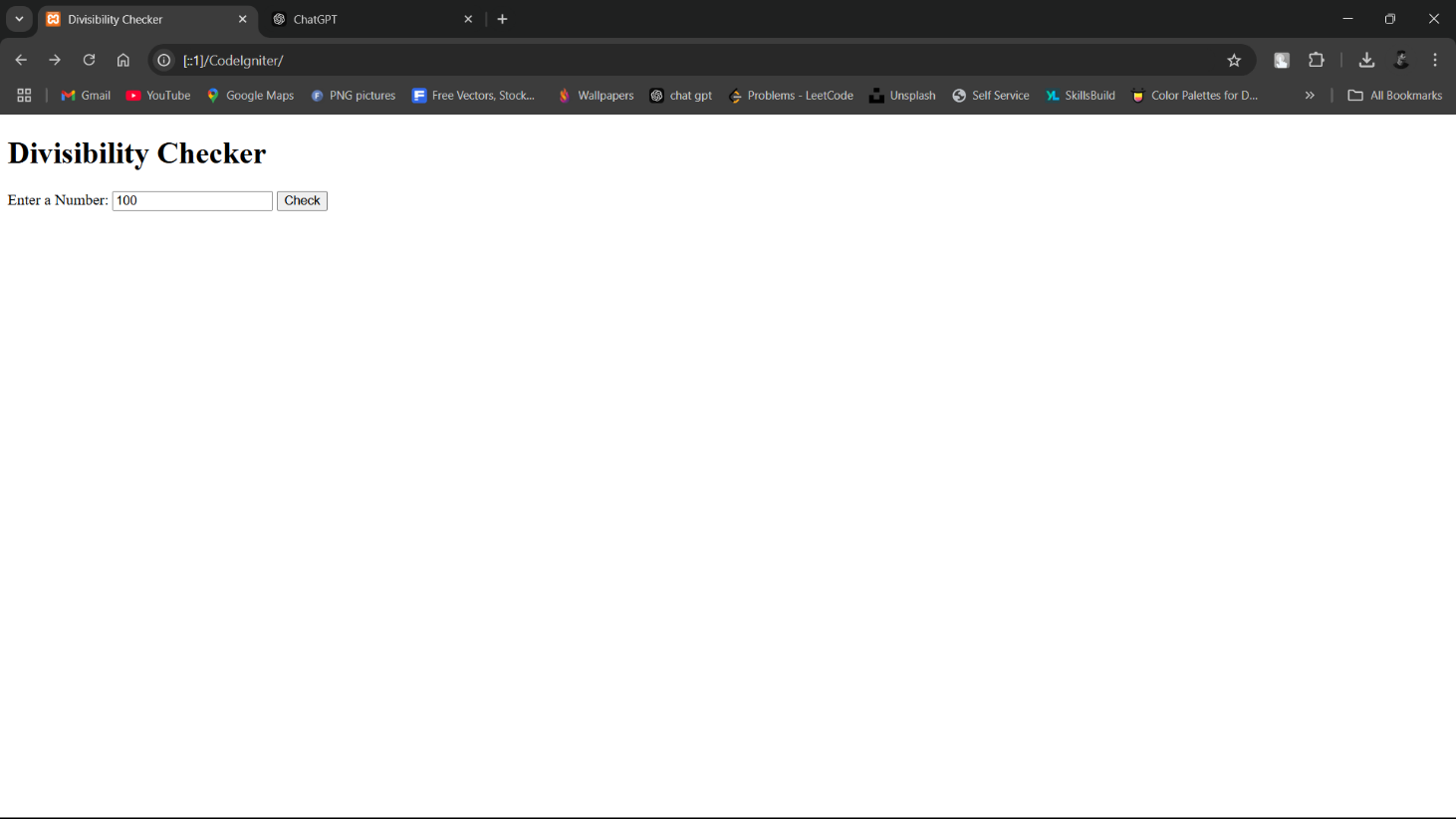
<h1>Result</h1>

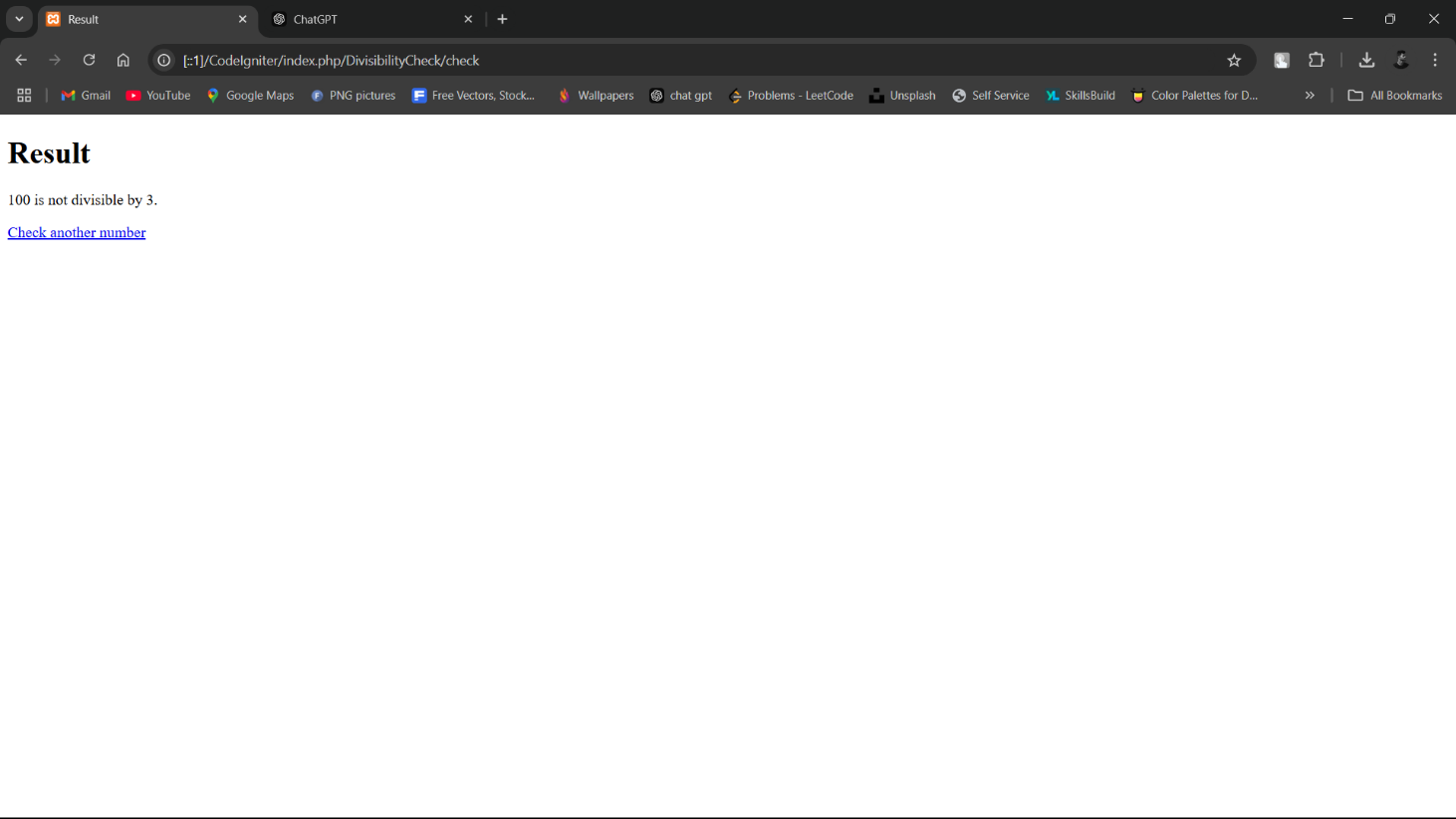
<p><?php echo $result; ?></p>

<a href="<?php echo site\_url('DivisibilityCheck'); ?>">Check another number</a>

</body></html>

**Output**:





**Practical NO 3:**

**Write a PHP Program in CodeIgniter to displays the name of the day based on a given number.**

1) **Controller** (DayName.php)

<?php

defined('BASEPATH') OR *exit*('No direct script access allowed');

class DayName extends CI\_Controller {

    public function index() {

        $this->load->view('day\_name\_form');

    }

    public function check() {

        $dayNumber = $this->input->post('day\_number');

        $days = [

            1 => "Sunday",

            2 => "Monday",

            3 => "Tuesday",

            4 => "Wednesday",

            5 => "Thursday",

            6 => "Friday",

            7 => "Saturday"

        ];

        $result = isset($days[$dayNumber]) ?

       $days[$dayNumber] : "Invalid number! Please enter a number between 1 and 7.";

        $data['result'] = $result;

        $this->load->view('day\_name\_result', $data);

    }

}

2) **View**:

* day\_name\_form:

<!DOCTYPE html>

<html>

<head>

<title>Day Name Finder</title>

</head>

<body>

<h1>Find the Name of the Day</h1>

<form method="post" action="<?php echo site\_url('DayName/check'); ?>">

<label for="day\_number">Enter a number (1-7):</label>

<input type="number" name="day\_number" min="1" max="7" required>

<input type="submit" value="Get Day Name">

</form>

</body>

</html>

* day\_name\_result:

<!DOCTYPE html>

<html>

<head>

<title>Result</title>

</head>

<body>

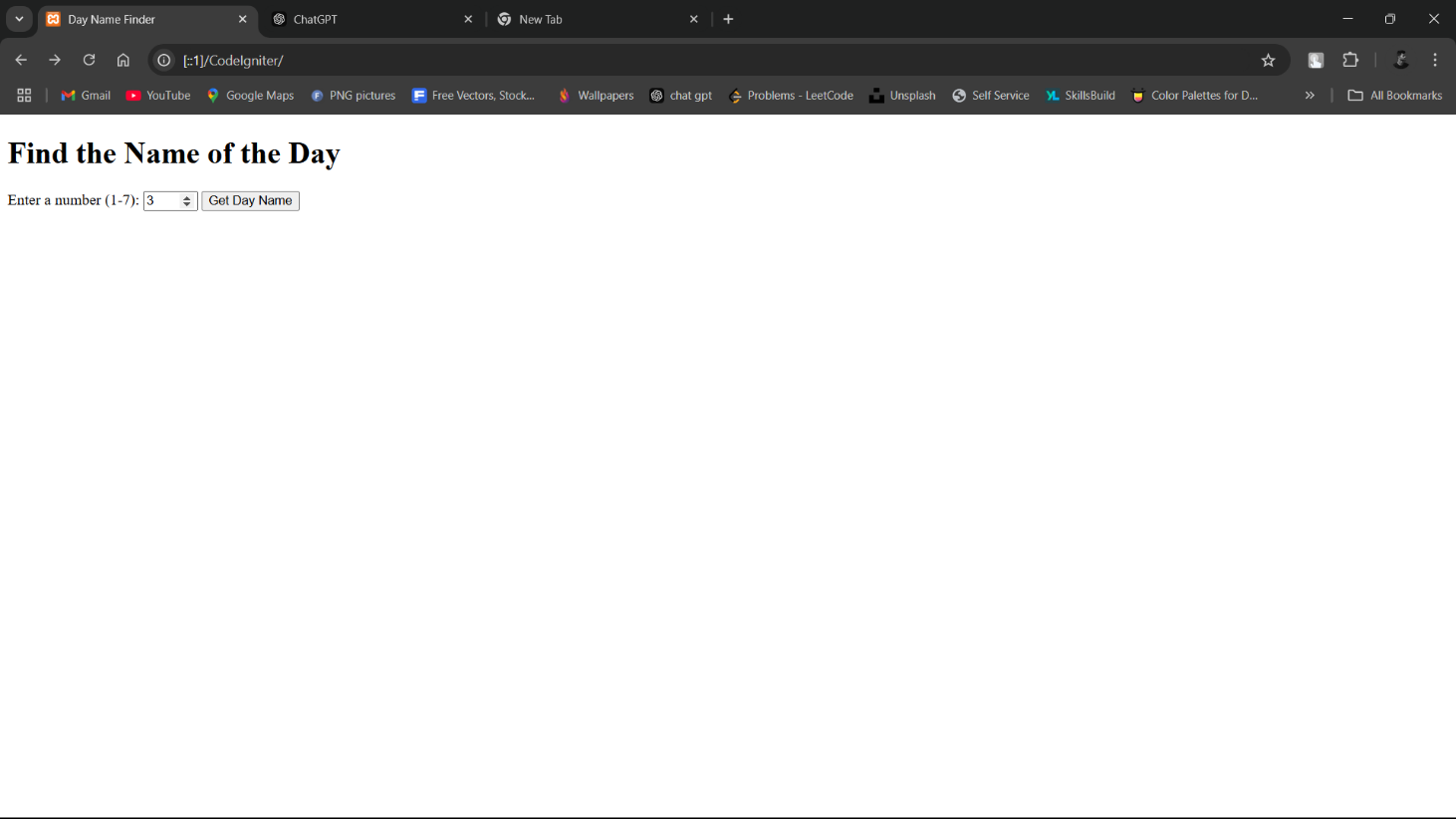
<h1>Result</h1>

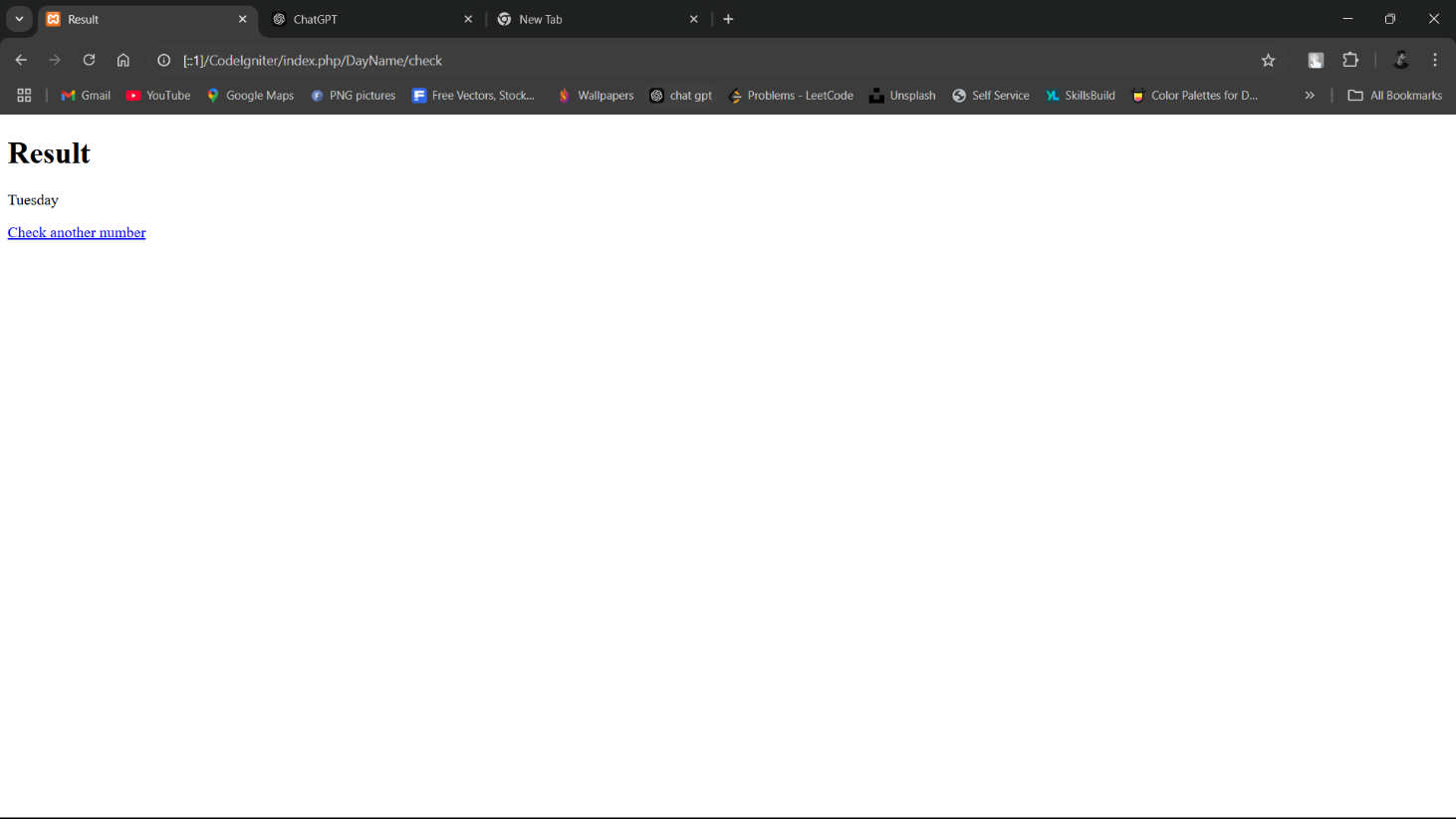
<p><?php echo $result; ?></p>

<a href="<?php echo site\_url('DayName'); ?>">Check another number</a>

</body></html>

**Output**:





**Practical NO 4:**

**Write a PHP Program in CodeIgniter to evaluate a score and display the corresponding grade using CodeIgniter.**

1) **Controller** (GradeEvaluator.php):

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class GradeEvaluator extends CI\_Controller {

public function index() {

$this->load->view('grade\_evaluator\_form');

}

public function evaluate() {

$score = $this->input->post('score');

// Determine the grade based on the score

if ($score >= 90 && $score <= 100) {

$grade = 'A';

} elseif ($score >= 80) {

$grade = 'B';

} elseif ($score >= 70) {

$grade = 'C';

} elseif ($score >= 60) {

$grade = 'D';

} elseif ($score >= 0) {

$grade = 'F';

} else {

$grade = 'Invalid score! Please enter a score between 0 and 100.';

}

$data['result'] = "Score: $score, Grade: $grade";

$this->load->view('grade\_evaluator\_result', $data);

}

}

?>

2) **view**:

* grade\_evaluator\_form.php

<!DOCTYPE html>

<html>

<head>

<title>Grade Evaluator</title>

</head>

<body>

<h1>Grade Evaluator</h1>

<form method="post" action="<?php echo site\_url('GradeEvaluator/evaluate'); ?>">

<label for="score">Enter the Score (0-100):</label>

<input type="number" name="score" min="0" max="100" required>

<input type="submit" value="Evaluate Grade">

</form>

</body>

</html>

* grade\_evaluator\_result.php

<!DOCTYPE html>

<html>

<head>

<title>Result</title>

</head>

<body>

<h1>Result</h1>

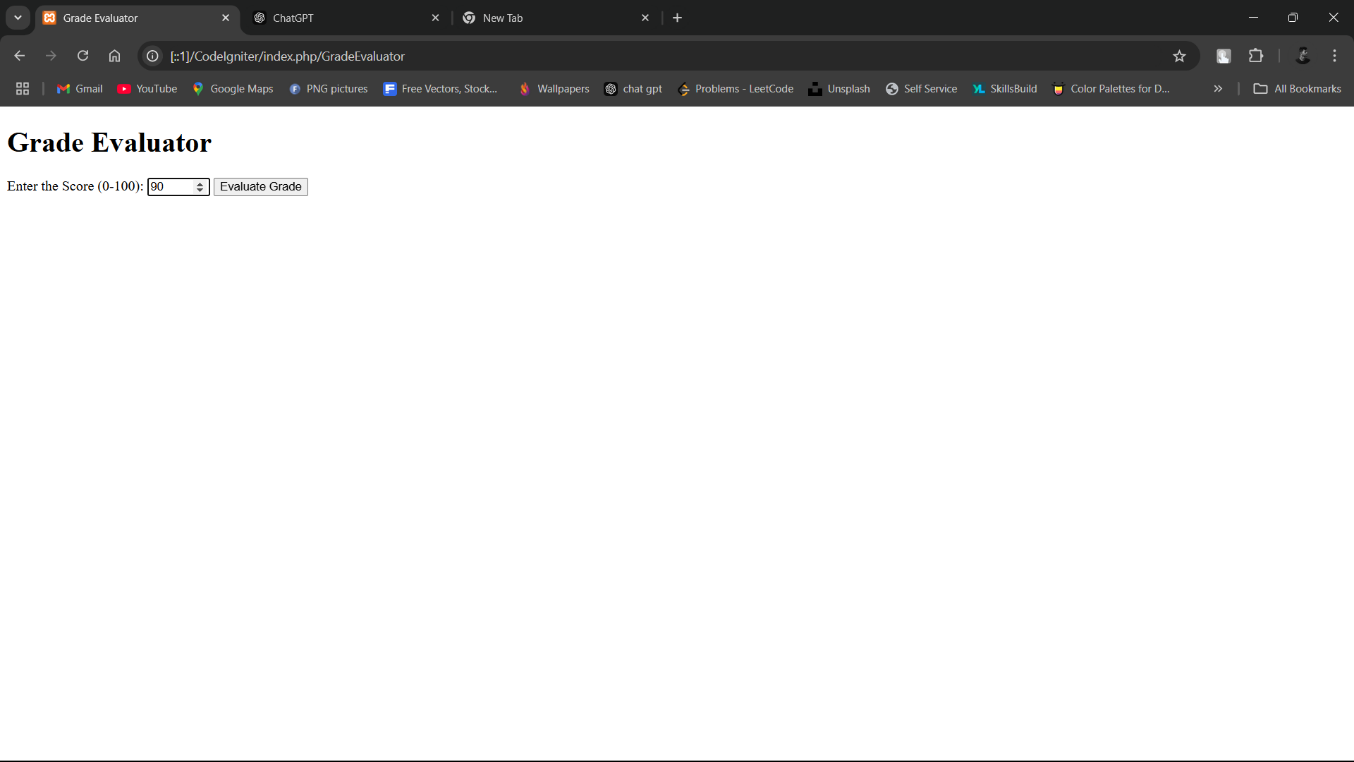
<p><?php echo $result; ?></p>

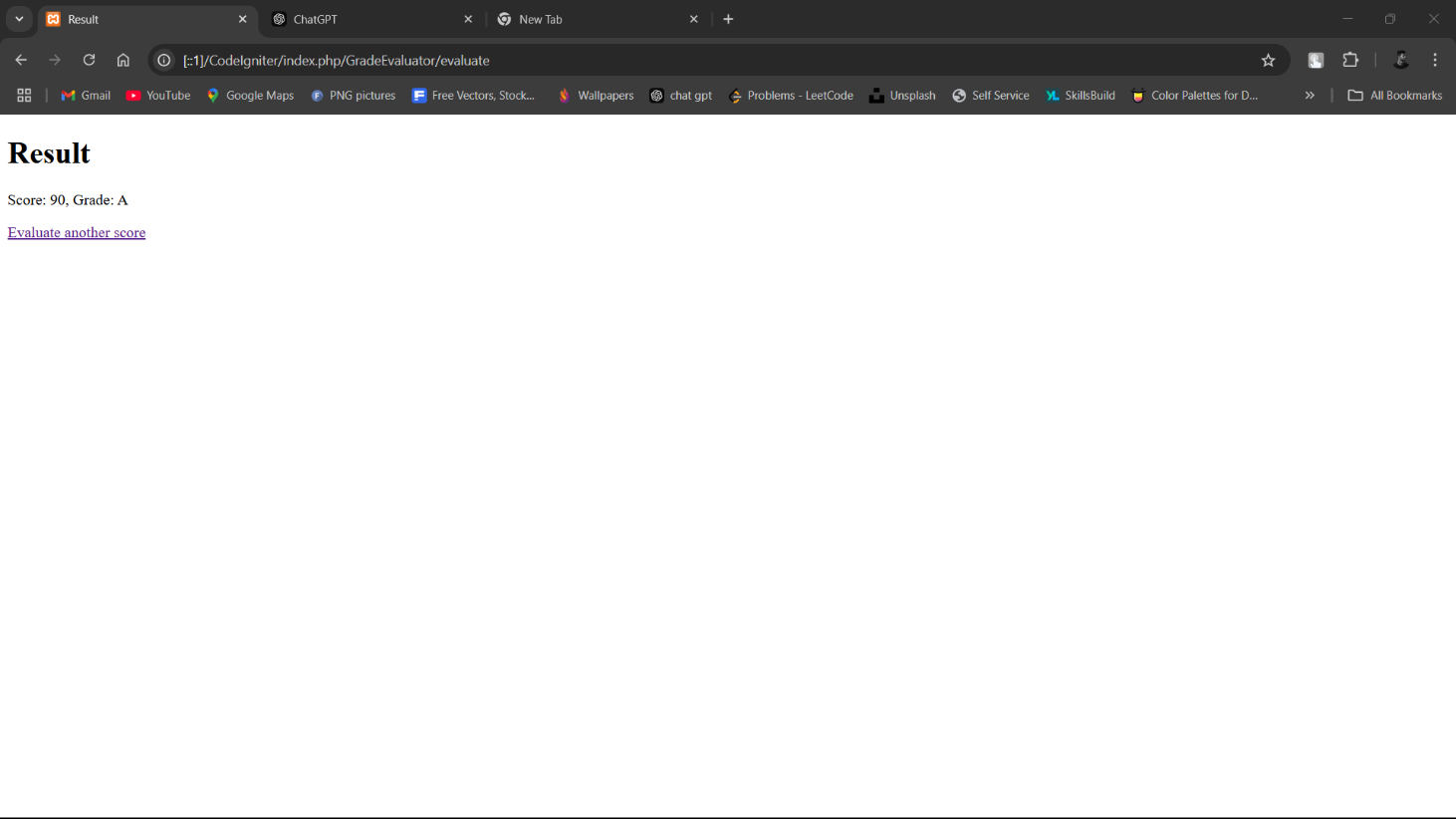
<a href="<?php echo site\_url('GradeEvaluator'); ?>">Evaluate another score</a>

</body>

</html>

**Output**:





**Practical NO 5:**

**Write a PHP Program in CodeIgniter to calculates the sum of natural numbers up to a specified limit.**

1) **Controller** (SumNaturalNumber.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class SumNaturalNumbers extends CI\_Controller {

public function index() {

$this->load->view('sum\_natural\_numbers\_form');

}

public function calculate() {

$limit = $this->input->post('limit');

// Validate input

if ($limit < 0) {

$result = "Please enter a non-negative number.";

} else {

// Calculate the sum of natural numbers

$sum = ($limit \* ($limit + 1)) / 2;

$result = "The sum of natural numbers up to $limit is $sum.";

}

$data['result'] = $result;

$this->load->view('sum\_natural\_numbers\_result', $data);

}

}

?>

2) **View**

* sum\_natural\_numbers\_form.php

<!DOCTYPE html>

<html>

<head>

<title>Sum of Natural Numbers</title>

</head>

<body>

<h1>Calculate Sum of Natural Numbers</h1>

<form method="post" action="<?php echo site\_url('SumNaturalNumbers/calculate'); ?>">

<label for="limit">Enter the Limit:</label>

<input type="number" name="limit" min="0" required>

<input type="submit" value="Calculate Sum">

</form>

</body>

</html>

* sum\_natural\_numbers\_result.php

<!DOCTYPE html>

<html>

<head>

<title>Result</title>

</head>

<body>

<h1>Result</h1>

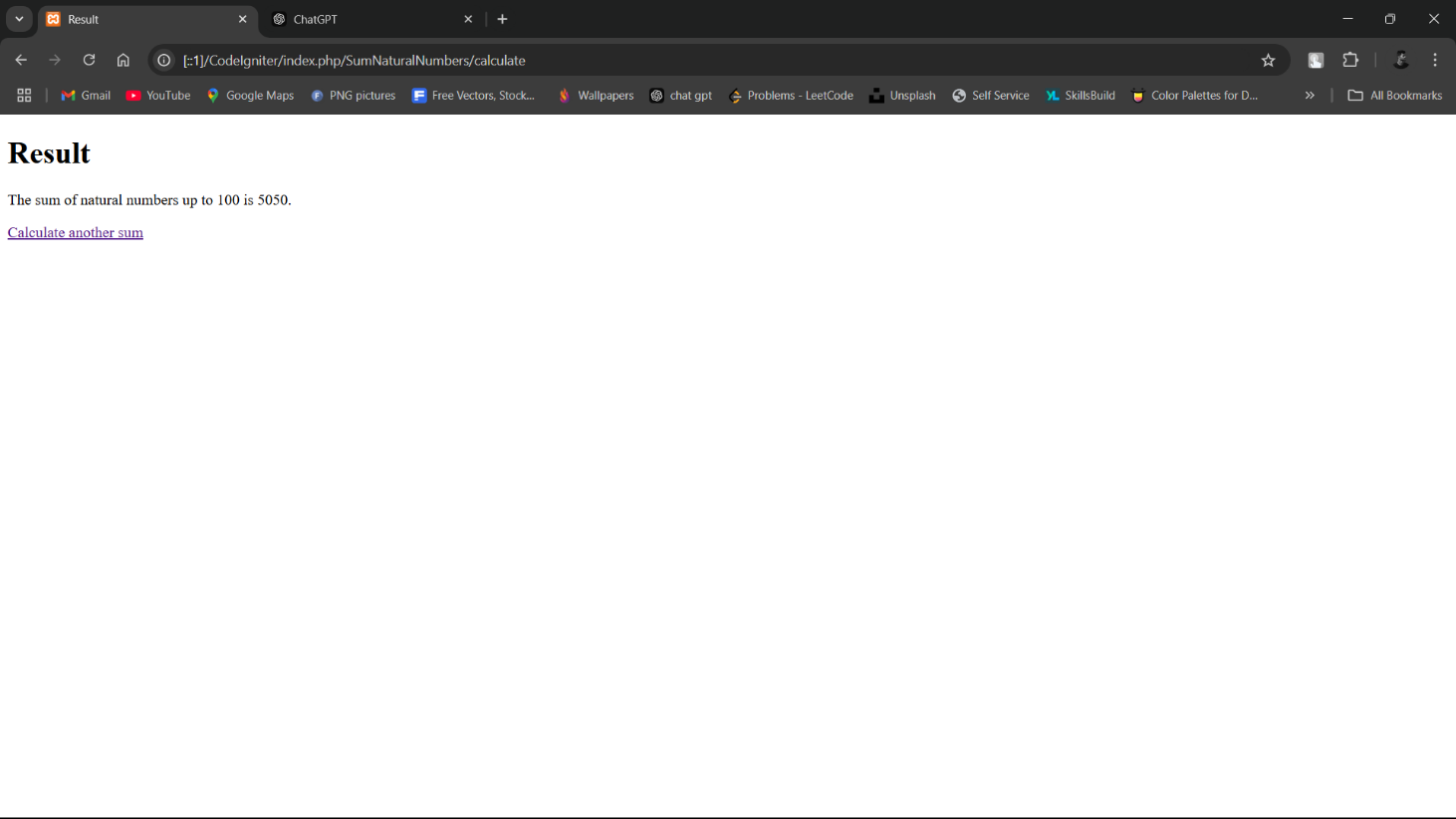
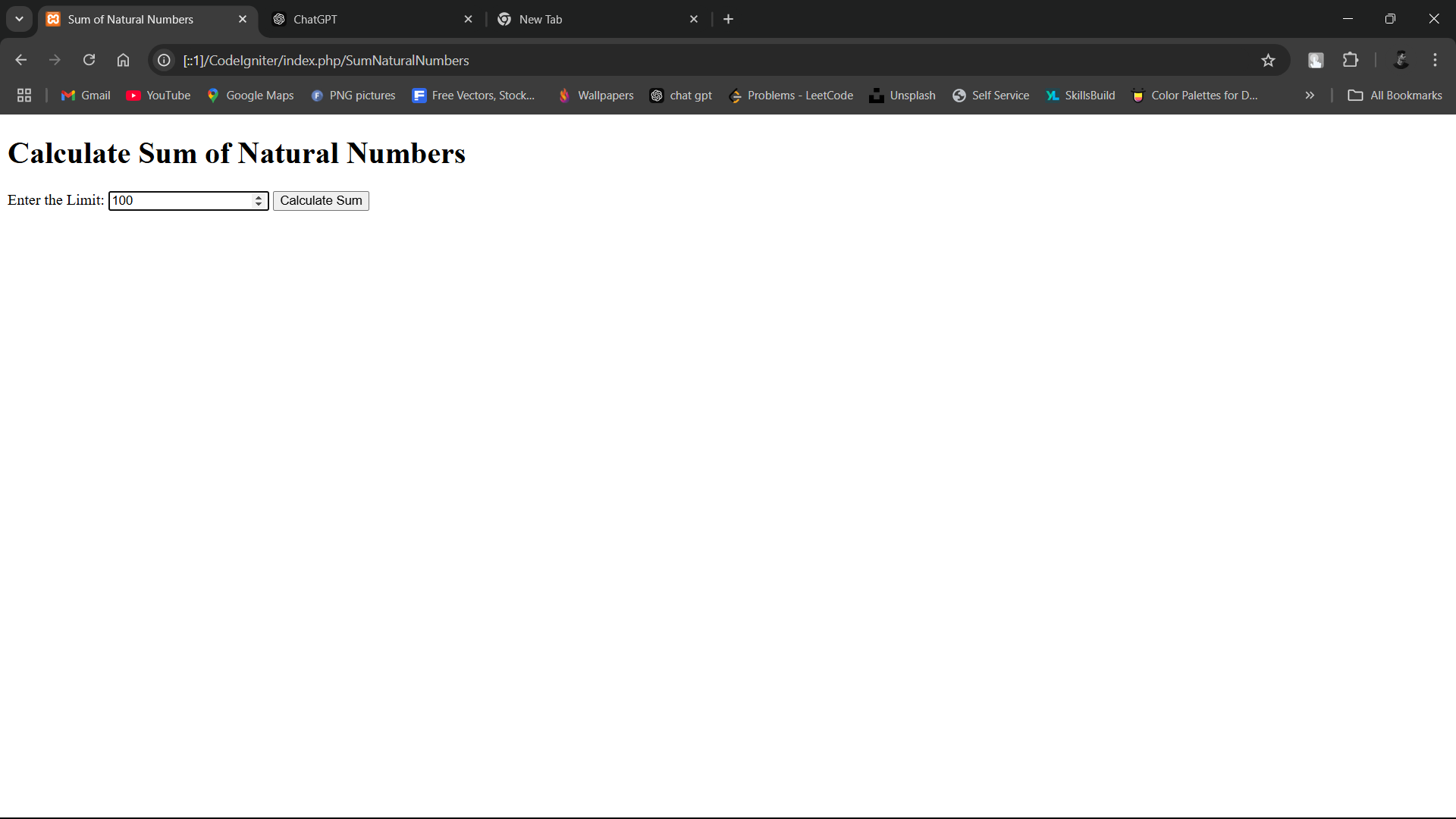
<p><?php echo $result; ?></p>

<a href="<?php echo site\_url('SumNaturalNumbers'); ?>">Calculate another sum</a>

</body>

</html>

**Output**:



**Practical NO 6:**

**Write a PHP Program in CodeIgniter to generates and displays a multiplication table for a specified number using do while loop.**

1) **Controller** (MultiplicationTable.php):

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class MultiplicationTable extends CI\_Controller {

public function index() {

$this->load->view('multiplication\_table\_form');

}

public function generate() {

$number = $this->input->post('number');

$table = [];

// Generate multiplication table using do while loop

$i = 1;

do {

$table[] = "$number x $i = " . ($number \* $i);

$i++;

} while ($i <= 10);

$data['table'] = $table;

$this->load->view('multiplication\_table\_result', $data);

}

}

?>

2) **View**:

* multiplication\_table\_form.php

<!DOCTYPE html>

<html>

<head>

<title>Multiplication Table</title>

</head>

<body>

<h1>Generate Multiplication Table</h1>

<form method="post" action="<?php echo site\_url('MultiplicationTable/generate'); ?>">

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

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

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

</form>

</body>

</html>

* multiplication\_table\_result.php

<!DOCTYPE html>

<html>

<head>

<title>Multiplication Table Result</title>

</head>

<body>

<h1>Multiplication Table</h1>

<ul>

<?php foreach ($table as $line): ?>

<li><?php echo $line; ?></li>

<?php endforeach; ?>

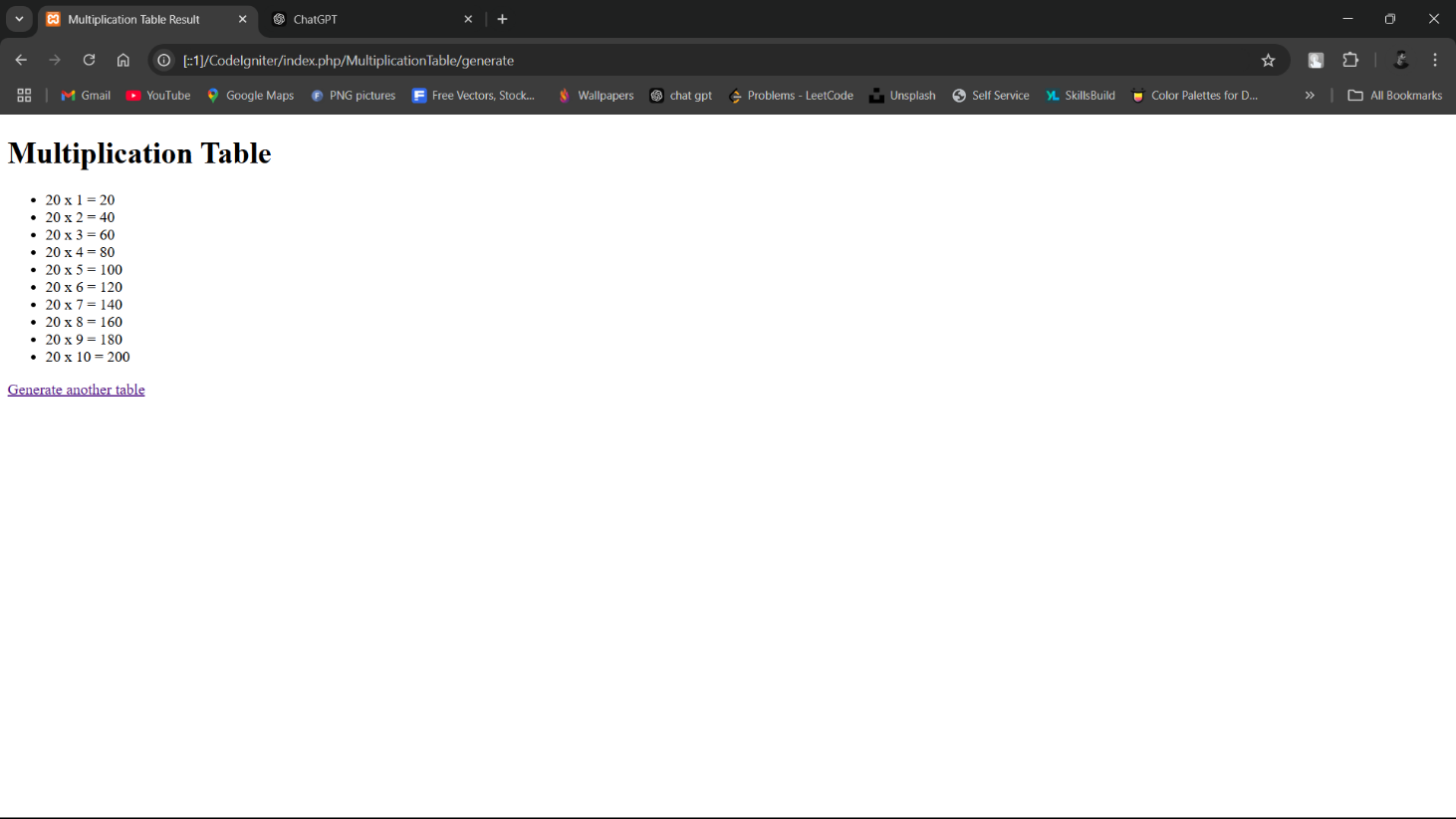
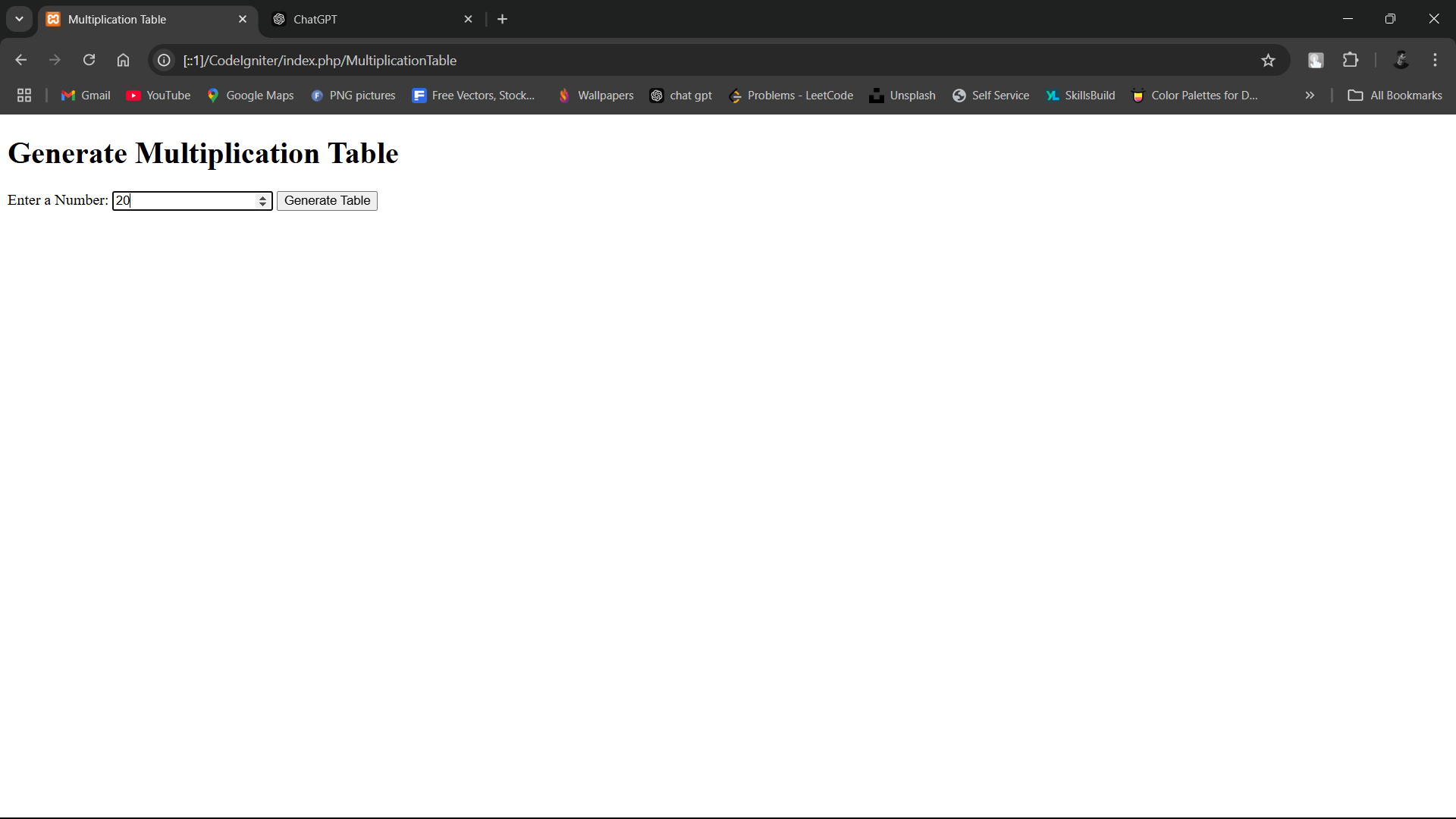
</ul>

<a href="<?php echo site\_url('MultiplicationTable'); ?>">Generate another table</a>

</body>

</html>

**Output**:



**Practical NO 7:**

**Write a PHP Program in CodeIgniter to calculates the factorial of a given number using a for loop.**

1) **Controller** (FactorialCalculator.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class FactorialCalculator extends CI\_Controller {

public function index() {

$this->load->view('factorial\_form');

}

public function calculate() {

$number = $this->input->post('number');

$factorial = 1;

// Calculate factorial using a for loop

if ($number < 0) {

$result = "Factorial is not defined for negative numbers.";

} else {

for ($i = 1; $i <= $number; $i++) {

$factorial \*= $i;

}

$result = "The factorial of $number is $factorial.";

}

$data['result'] = $result;

$this->load->view('factorial\_result', $data);

}

}

?>

2) **View**:

* factorial\_form.php

<!DOCTYPE html>

<html>

<head>

<title>Factorial Calculator</title>

</head>

<body>

<h1>Calculate Factorial</h1>

<form method="post" action="<?php echo site\_url('FactorialCalculator/calculate'); ?>">

<label for="number">Enter a Non-Negative Integer:</label>

<input type="number" name="number" min="0" required>

<input type="submit" value="Calculate Factorial">

</form>

</body>

</html>

* factorial\_result.php

<!DOCTYPE html>

<html>

<head>

<title>Factorial Result</title>

</head>

<body>

<h1>Result</h1>

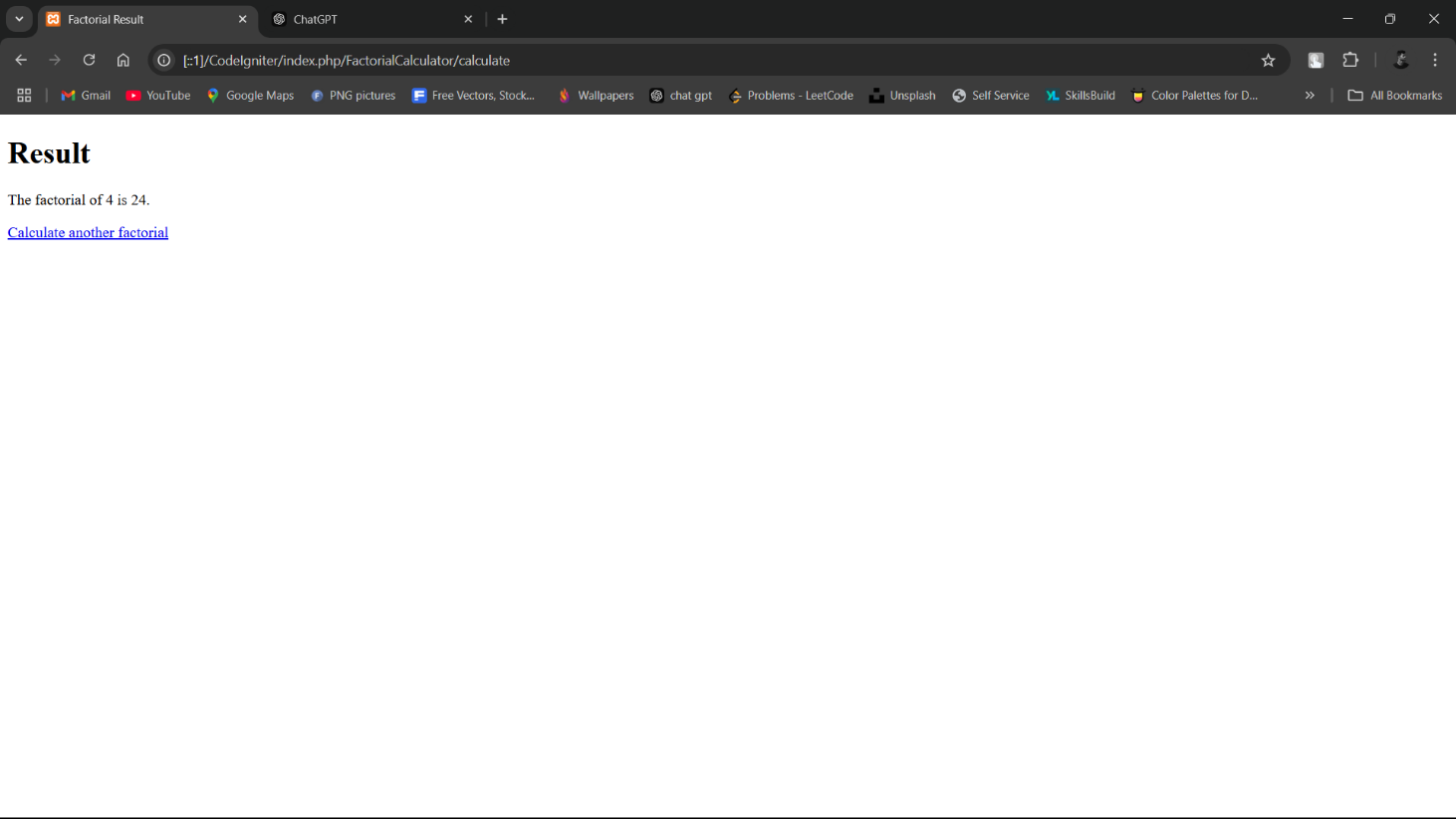
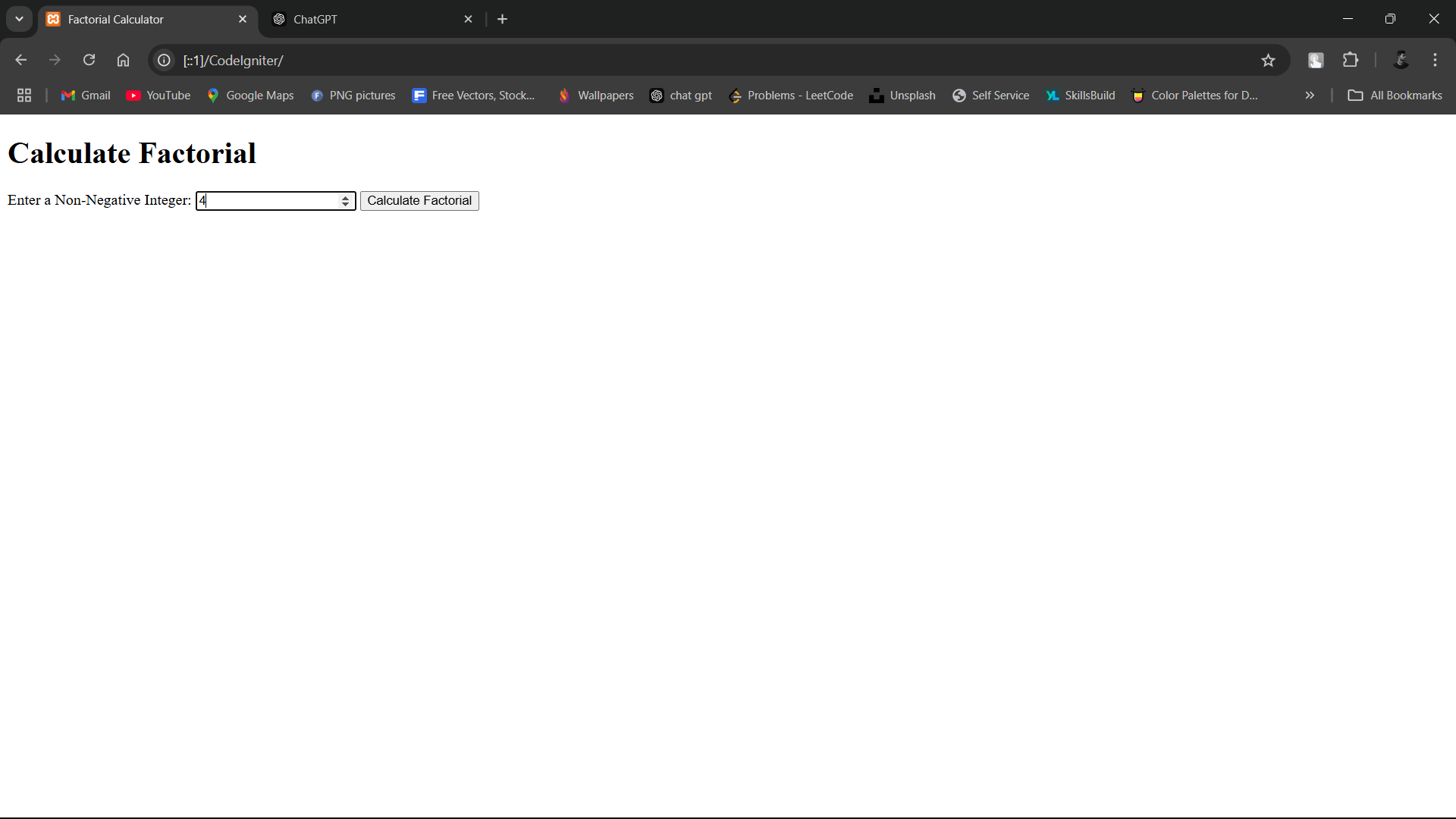
<p><?php echo $result; ?></p>

<a href="<?php echo site\_url('FactorialCalculator'); ?>">Calculate another factorial</a>

</body>

</html>

**Output**:



**Practical NO 8:**

**Write a PHP Program in CodeIgniter to that generates the Fibonacci series up to a specified number of terms.**

1) **Controller** (FibonacciSeries.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class FibonacciSeries extends CI\_Controller {

public function index() {

$this->load->view('fibonacci\_form');

}

public function generate() {

$terms = $this->input->post('terms');

$fibonacci = [];

// Generate Fibonacci series

if ($terms <= 0) {

$result = "Please enter a positive integer.";

} else {

$fibonacci[0] = 0;

if ($terms > 1) {

$fibonacci[1] = 1;

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

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

}

}

$result = "Fibonacci series up to $terms terms: " . implode(", ", $fibonacci);

}

$data['result'] = $result;

$this->load->view('fibonacci\_result', $data);

}

}

?>

2) **View**

* fibonacci\_form.php:

<!DOCTYPE html>

<html>

<head>

<title>Fibonacci Series Generator</title>

</head>

<body>

<h1>Generate Fibonacci Series</h1>

<form method="post" action="<?php echo site\_url('FibonacciSeries/generate'); ?>">

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

<input type="number" name="terms" min="1" required>

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

</form>

</body>

</html>

* fibonacci\_result.php:

<!DOCTYPE html>

<html>

<head>

<title>Fibonacci Series Result</title>

</head>

<body>

<h1>Result</h1>

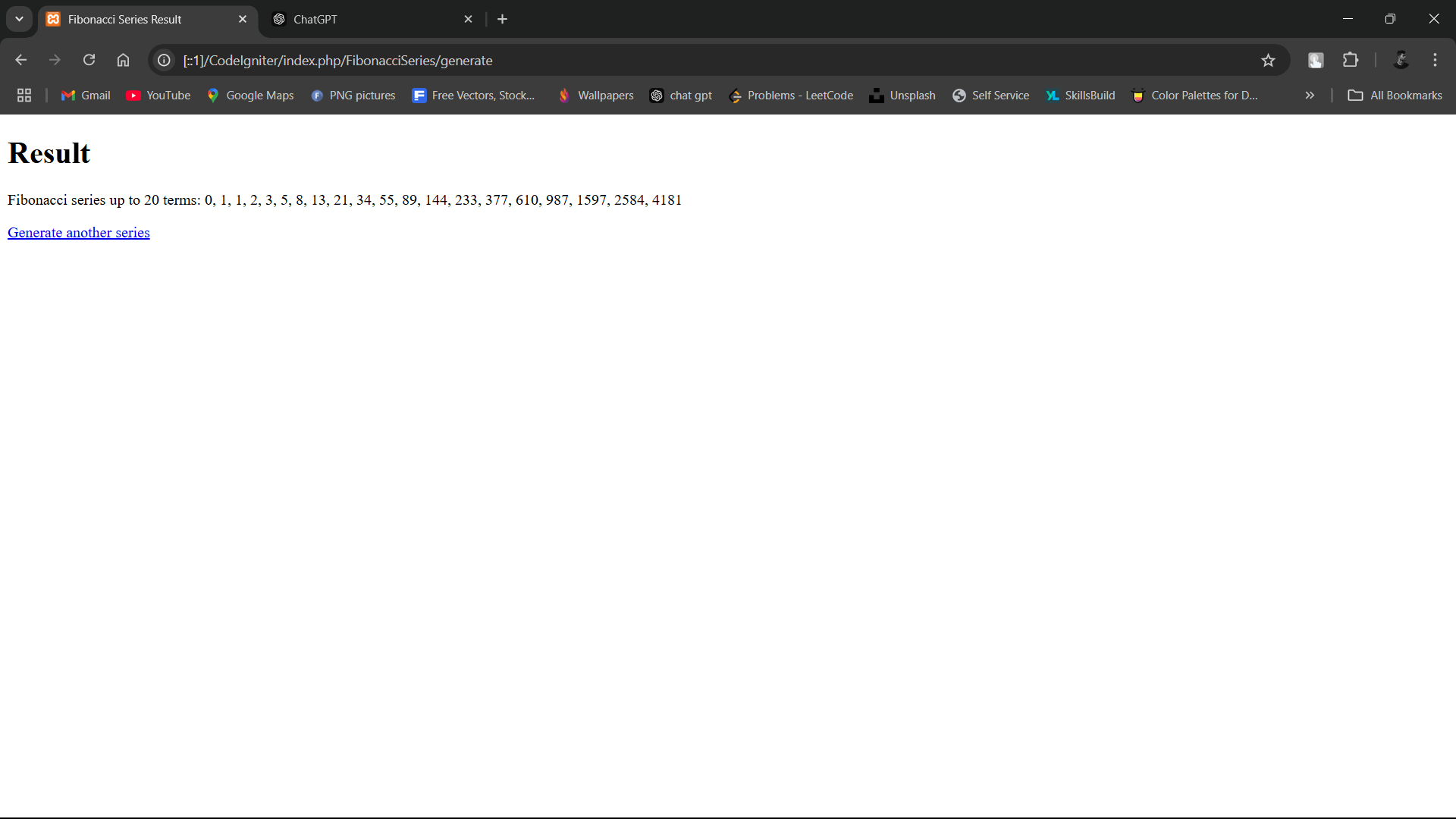
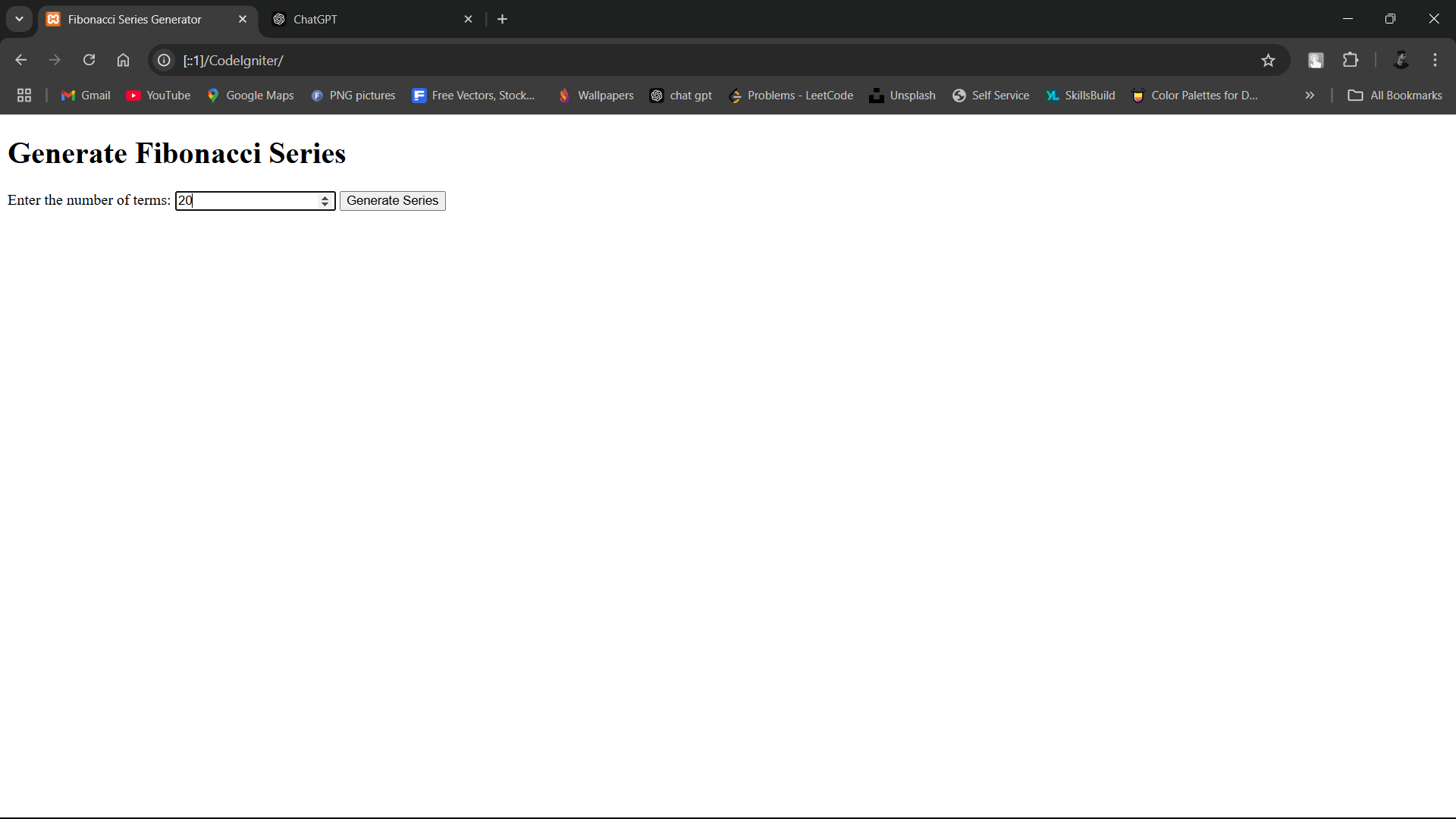
<p><?php echo $result; ?></p>

<a href="<?php echo site\_url('FibonacciSeries'); ?>">Generate another series</a>

</body>

</html>

**Output**



**Practical NO 9:**

**Write a PHP Program in CodeIgniter to that iterates through an array of student names and displays them using simple array.**

**1) Controller** (StudentList.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class StudentList extends CI\_Controller {

public function index() {

$students = ["Alice", "Bob", "Charlie", "David", "Eva"];

$data['students'] = $students;

$this->load->view('student\_list', $data);

}

}

?>

**2) View**

* student\_list.php:

<!DOCTYPE html>

<html>

<head>

<title>Student List</title>

</head>

<body>

<h1>List of Students</h1>

<ul>

<?php foreach ($students as $student): ?>

<li><?php echo $student; ?></li>

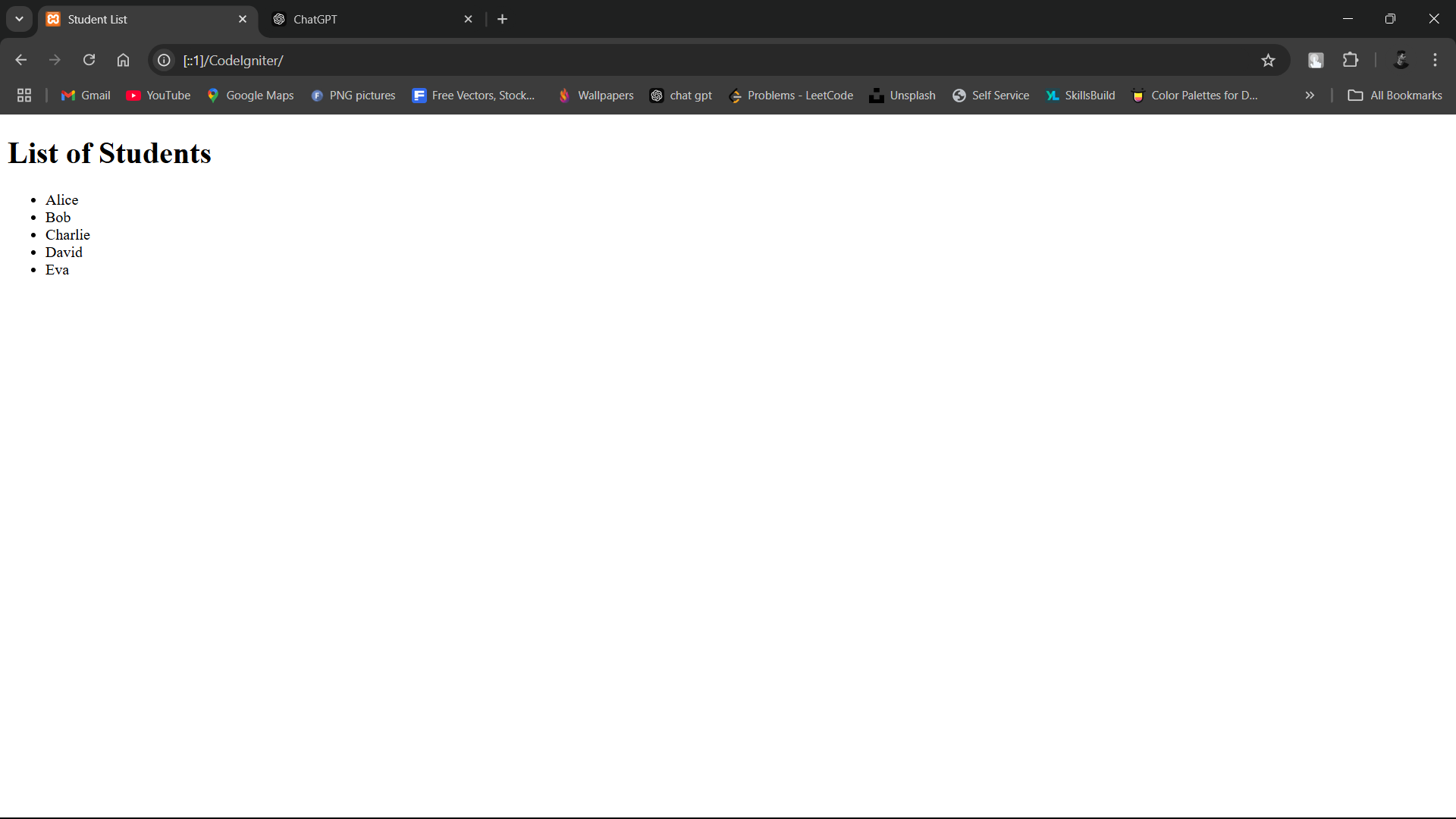
<?php endforeach; ?>

</ul>

</body>

</html>

**Output**



**Practical NO 10:**

**Write a PHP Program in CodeIgniter to Write a PHP program to create an indexed array of fruits and display them.**

1) **Controller** (Fruits.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class Fruits extends CI\_Controller {

public function index() {

// Create an indexed array of fruits

$fruits = array("Apple", "Banana", "Cherry", "Date", "Elderberry");

// Load the view and pass the fruits array

$this->load->view('fruits\_view', ['fruits' => $fruits]);

}

}

?>

2) **View**

* fruits\_view.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Fruits List</title>

</head>

<body>

<h1>List of Fruits</h1>

<ul>

<?php foreach ($fruits as $fruit): ?>

<li><?php echo $fruit; ?></li>

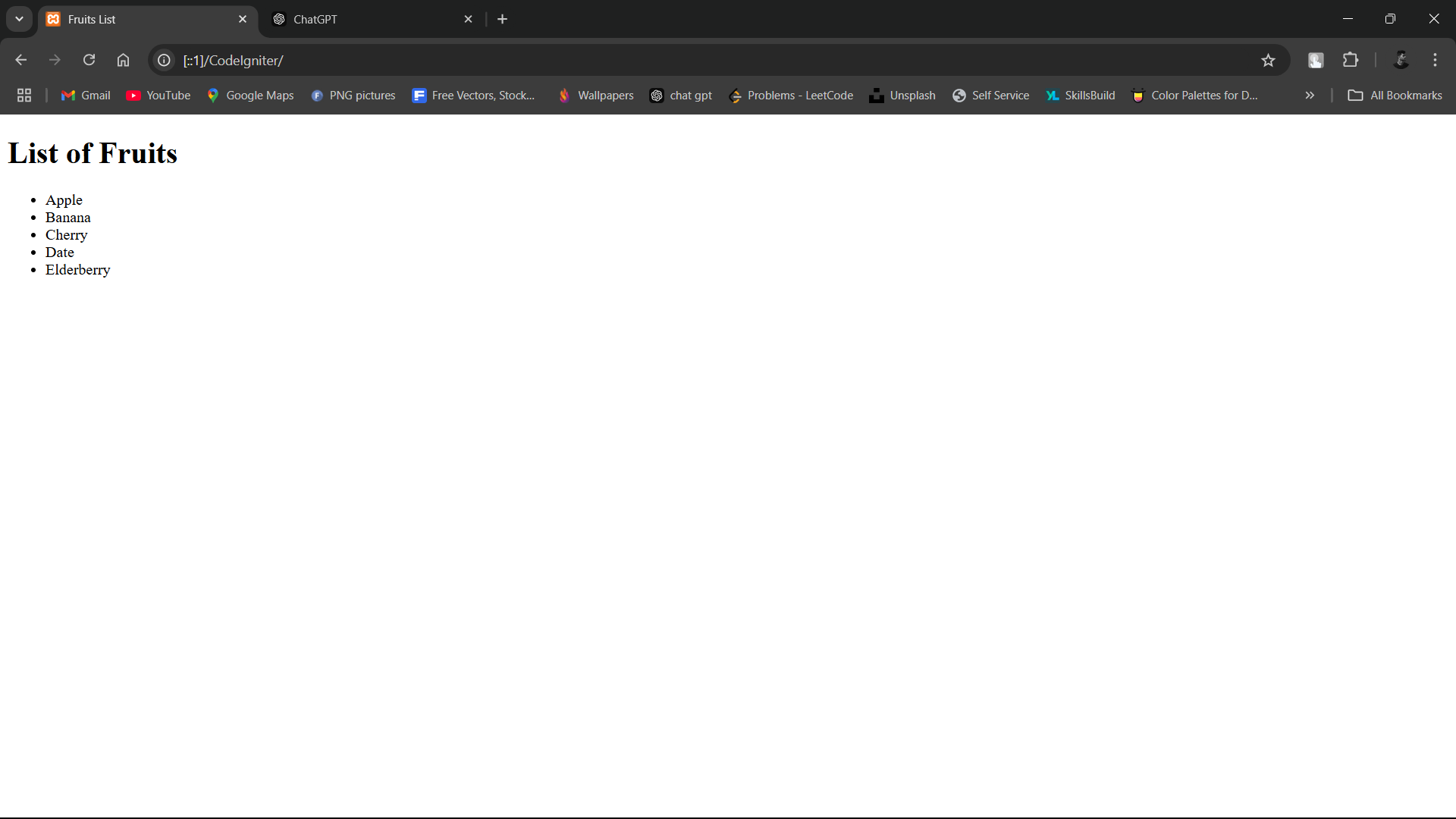
<?php endforeach; ?>

</ul>

</body>

</html>

**Output**



**Practical NO 11:**

**Write a PHP Program in CodeIgniter to calculate the length of String.**

1) **Controller** (StringLength.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class StringLength extends CI\_Controller {

public function index() {

$this->load->view('string\_length\_form');

}

public function calculate() {

$input\_string = $this->input->post('input\_string');

$length = strlen($input\_string);

$data['length'] = $length;

$data['input\_string'] = $input\_string;

$this->load->view('string\_length\_result', $data);

}

}

?>

2) **View**

* string\_length\_form.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>String Length Calculator</title>

</head>

<body>

<h1>Calculate String Length</h1>

<form action="<?php echo site\_url('stringlength/calculate'); ?>" method="post">

<label for="input\_string">Enter a string:</label>

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

<input type="submit" value="Calculate Length">

</form>

</body>

</html>

* string\_length\_result.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>String Length Result</title>

</head>

<body>

<h1>String Length Result</h1>

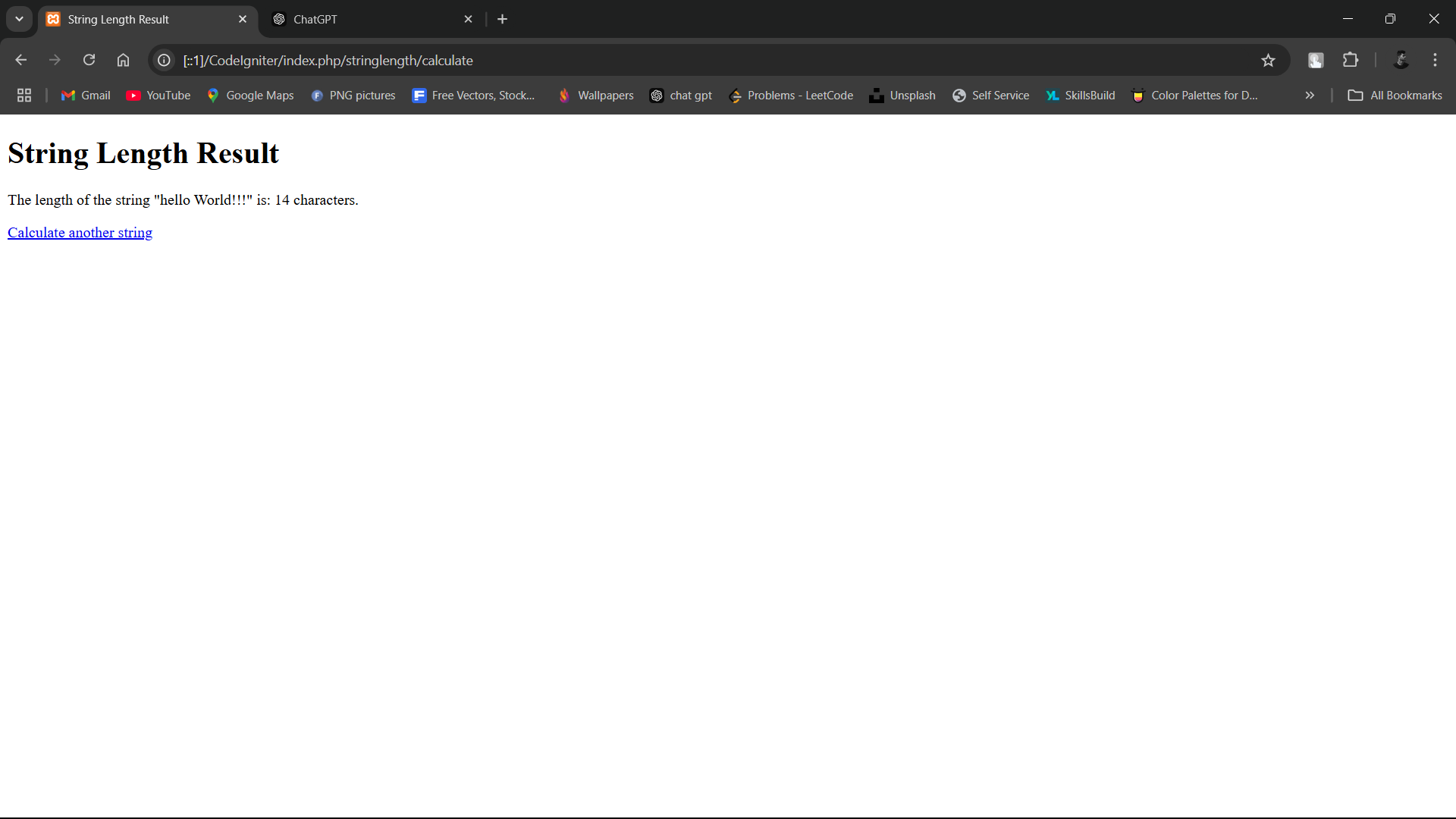
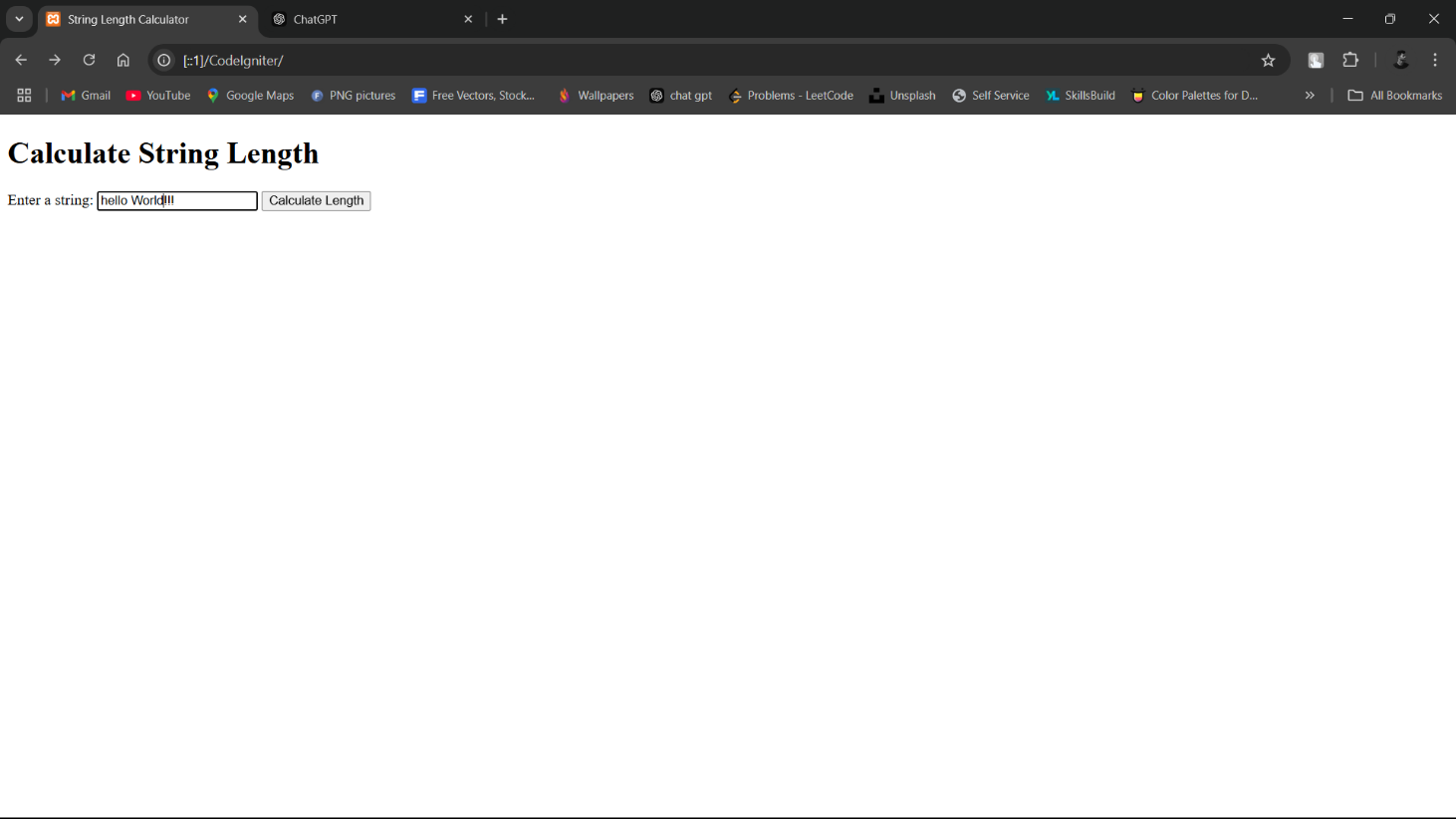
<p>The length of the string "<?php echo $input\_string; ?>" is: <?php echo $length; ?> characters.</p>

<a href="<?php echo site\_url('stringlength'); ?>">Calculate another string</a>

</body>

</html>

**Output**



**Practical NO 12:**

**Write a PHP Program in CodeIgniter to count the number of words in string without using string functions**

1) **Controller** (WordCount.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class WordCount extends CI\_Controller {

public function index() {

$this->load->view('word\_count\_form');

}

public function count\_words() {

$input\_string = $this->input->post('input\_string');

$word\_count = $this->calculate\_word\_count($input\_string);

$data['word\_count'] = $word\_count;

$data['input\_string'] = $input\_string;

$this->load->view('word\_count\_result', $data);

}

private function calculate\_word\_count($string) {

$count = 0;

$in\_word = false;

for ($i = 0; $i < strlen($string); $i++) {

if ($string[$i] != ' ') {

if (!$in\_word) {

$in\_word = true;

$count++;

}

} else {

$in\_word = false;

}

}

return $count;

}

}

?>

2) **View**

* word\_count\_form.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Word Count Calculator</title>

</head>

<body>

<h1>Calculate Word Count</h1>

<form action="<?php echo site\_url('wordcount/count\_words'); ?>" method="post">

<label for="input\_string">Enter a string:</label>

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

<input type="submit" value="Count Words">

</form>

</body>

</html>

* word\_count\_result.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Word Count Result</title>

</head>

<body>

<h1>Word Count Result</h1>

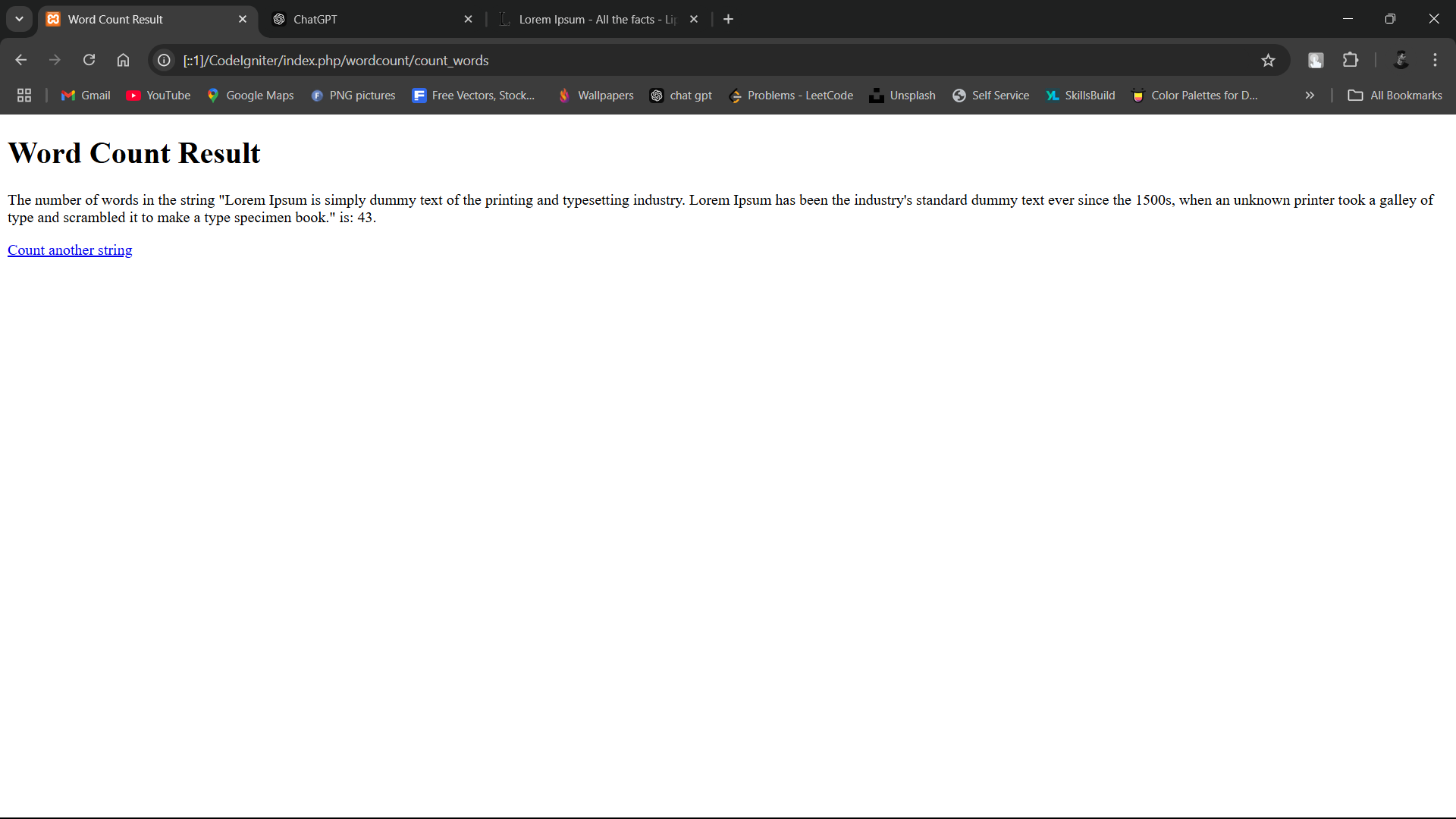
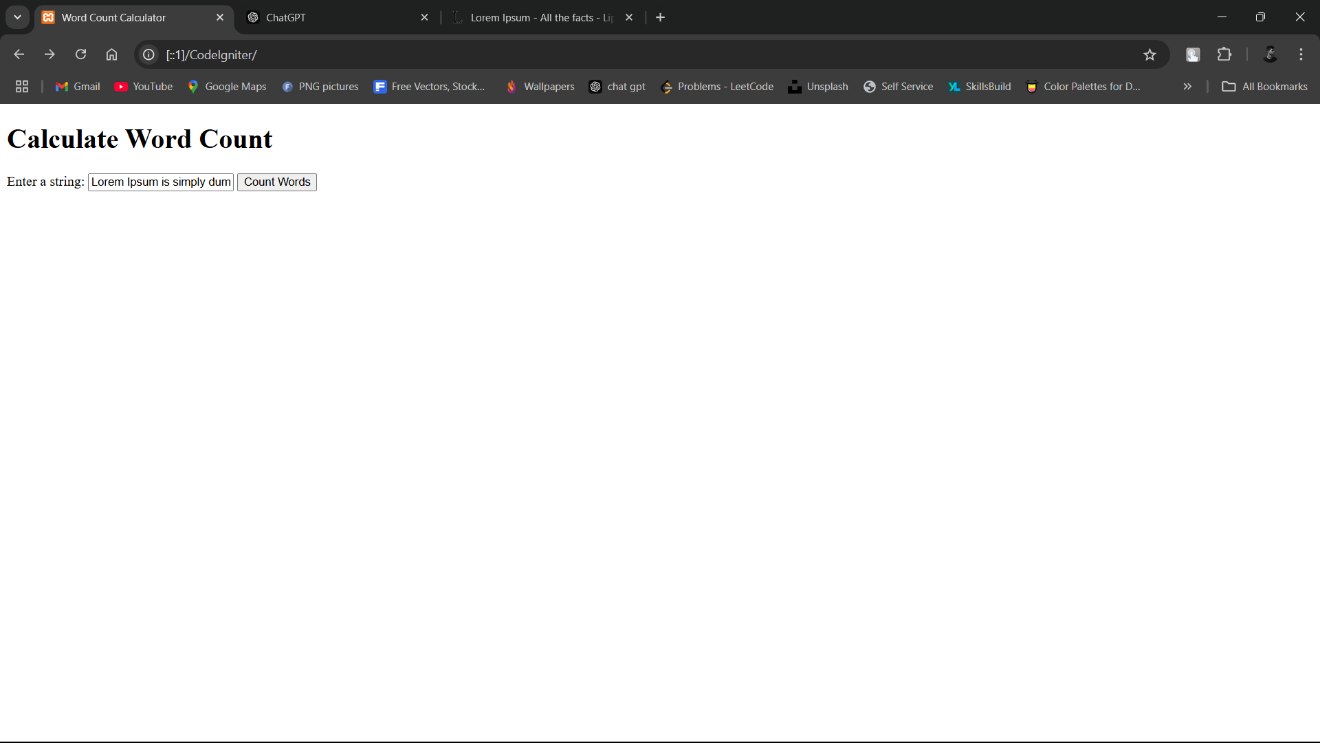
<p>The number of words in the string "<?php echo htmlspecialchars($input\_string); ?>" is: <?php echo $word\_count; ?>.</p>

<a href="<?php echo site\_url('wordcount'); ?>">Count another string</a>

</body>

</html>

**Output**



**Practical NO 13:**

**Write a PHP Program in CodeIgniter to demonstrate use of various built-in string functions.**

1) **Controller** (StringFunctions.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class StringFunctions extends CI\_Controller {

public function index() {

$this->load->view('string\_functions\_form');

}

public function demonstrate() {

$input\_string = $this->input->post('input\_string');

// Demonstrating various string functions

$data['original'] = $input\_string;

$data['length'] = strlen($input\_string);

$data['uppercase'] = strtoupper($input\_string);

$data['lowercase'] = strtolower($input\_string);

$data['reversed'] = strrev($input\_string);

$data['word\_count'] = str\_word\_count($input\_string);

$data['substring'] = substr($input\_string, 0, 5); // First 5 characters

$this->load->view('string\_functions\_result', $data);

}

}

?>

2) **view**

* string\_functions\_form.php:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>String Functions Demonstration</title>

</head>

<body>

<h1>Demonstrate Built-in String Functions</h1>

<form action="<?php echo site\_url('stringfunctions/demonstrate'); ?>" method="post">

<label for="input\_string">Enter a string:</label>

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

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

</form>

</body>

</html>

* string\_functions\_result.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>String Functions Result</title>

</head>

<body>

<h1>String Functions Result</h1>

<p><strong>Original String:</strong> "<?php echo htmlspecialchars($original); ?>"</p>

<p><strong>Length:</strong> <?php echo $length; ?> characters</p>

<p><strong>Uppercase:</strong> <?php echo htmlspecialchars($uppercase); ?></p>

<p><strong>Lowercase:</strong> <?php echo htmlspecialchars($lowercase); ?></p>

<p><strong>Reversed:</strong> <?php echo htmlspecialchars($reversed); ?></p>

<p><strong>Word Count:</strong> <?php echo $word\_count; ?> words</p>

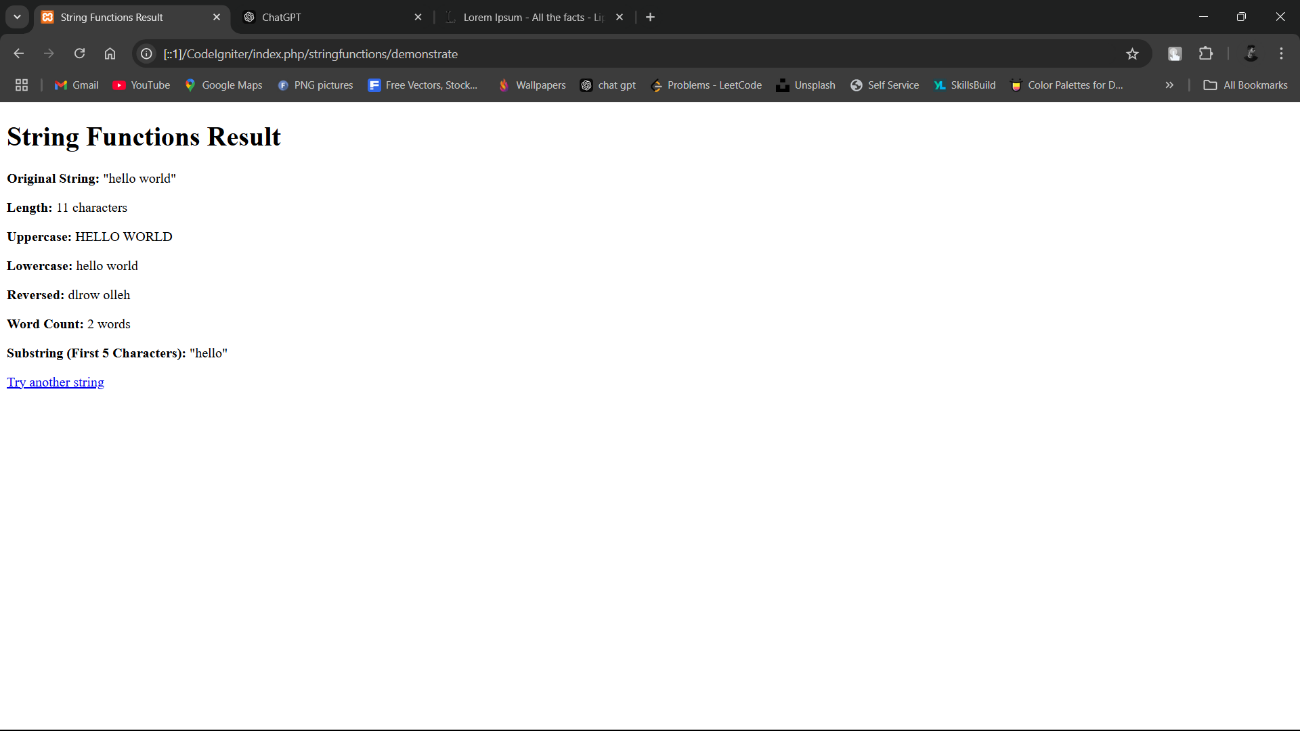
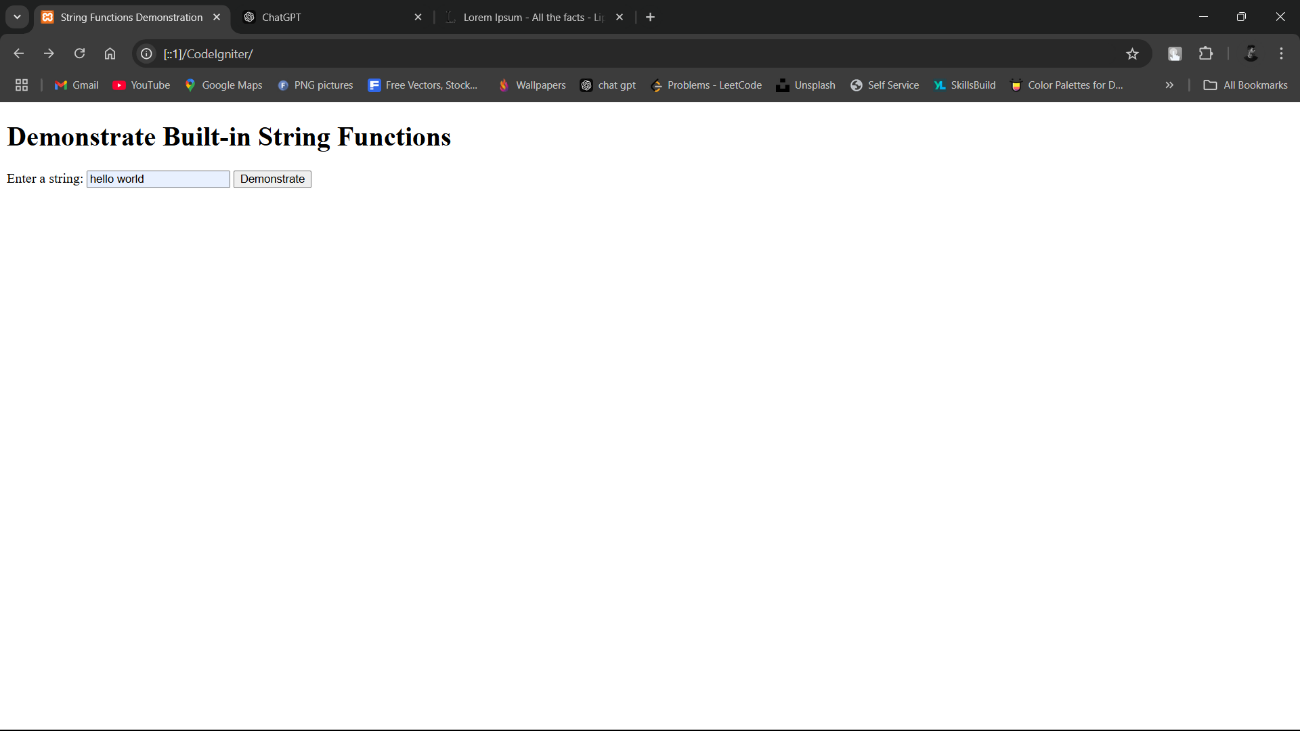
<p><strong>Substring (First 5 Characters):</strong> "<?php echo htmlspecialchars($substring); ?>"</p>

<a href="<?php echo site\_url('stringfunctions'); ?>">Try another string</a>

</body>

</html>

**Output**



**Practical NO 14:**

**Create a CodeIgniter PHP program that demonstrates inheritance with an Animal superclass (with properties name and age and a speak() method ) and a Dog subclass that overrides speak() to include the dog's name and age.**

1) **Super** **Class** (Animal.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class Animal {

protected $name;

protected $age;

public function \_\_construct($name, $age) {

$this->name = $name;

$this->age = $age;

}

public function speak() {

return "I am an animal.";

}

}

?>

2) **Subclass** (Dog.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class Dog extends Animal {

public function speak() {

return "Woof! My name is {$this->name} and I am {$this->age} years old.";

}

}

?>

3) **Controller** (AnimalController.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class AnimalController extends CI\_Controller {

public function index() {

// Create an instance of the Dog subclass

$dog = new Dog("Buddy", 3);

$data['message'] = $dog->speak();

// Load the view

$this->load->view('animal\_view', $data);

}

}

?>

4) **View**

* animal\_view.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Animal Inheritance</title>

</head>

<body>

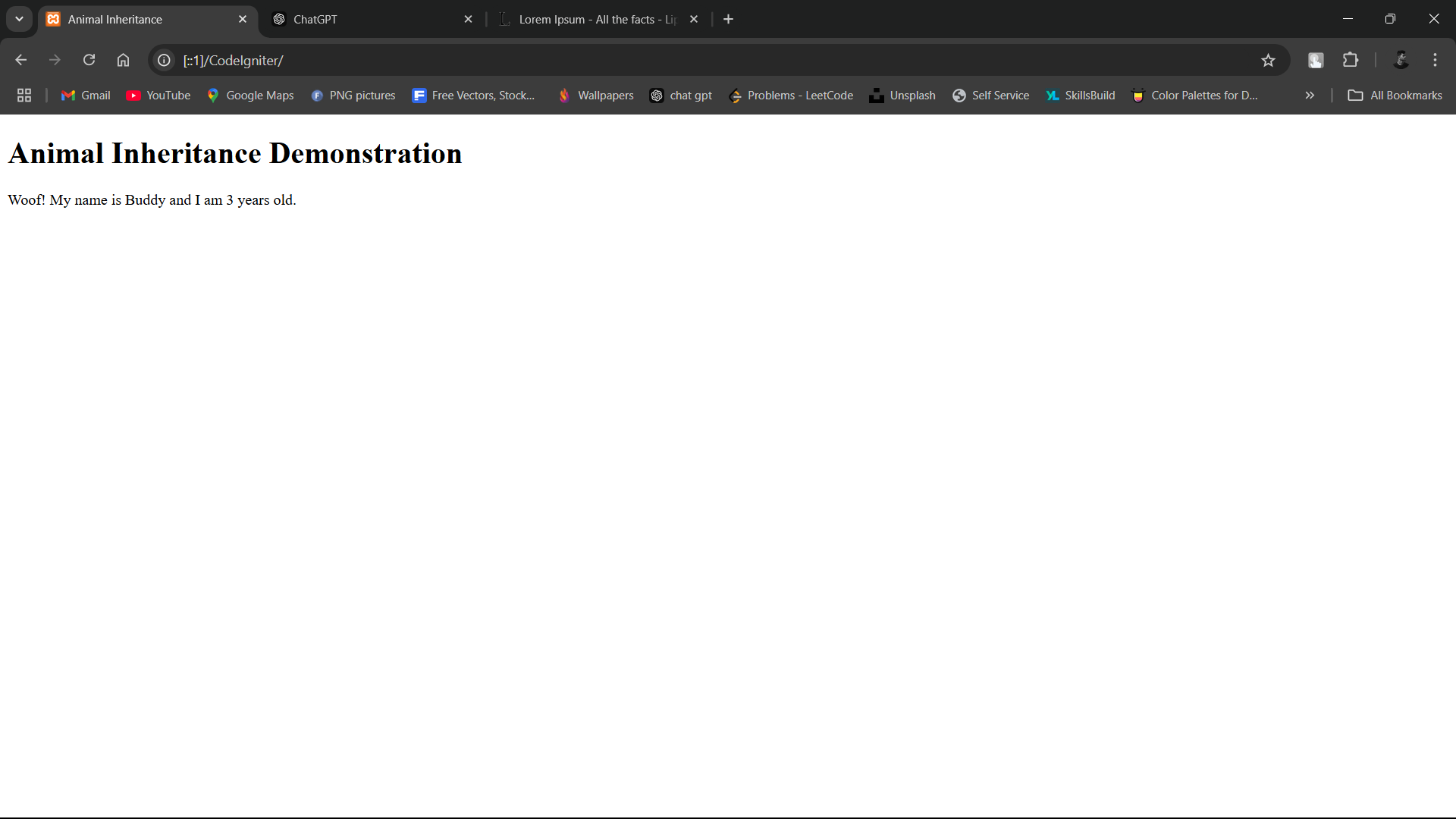
<h1>Animal Inheritance Demonstration</h1>

<p><?php echo $message; ?></p>

</body>

</html>

**Output:**



**Practical NO 15:**

**Write a PHP Program in CodeIgniter to Create a Car\_model class with a constructor to initialize properties like make, model, and year etc**

1) **Model** **Class** (Car\_model.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class Car\_model {

public $make;

public $model;

public $year;

// Constructor to initialize properties

public function \_\_construct($make, $model, $year) {

$this->make = $make;

$this->model = $model;

$this->year = $year;

}

}

?>

2) **Controller** (CarController.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class CarController extends CI\_Controller {

public function index() {

// Create an instance of Car\_model

$car = new Car\_model("Toyota", "Camry", 2022);

// Prepare data for the view

$data['make'] = $car->make;

$data['model'] = $car->model;

$data['year'] = $car->year;

// Load the view

$this->load->view('car\_view', $data);

}

}

?>

3) **view**

* car\_view.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Car Model Details</title>

</head>

<body>

<h1>Car Model Details</h1>

<p><strong>Make:</strong> <?php echo htmlspecialchars($make); ?></p>

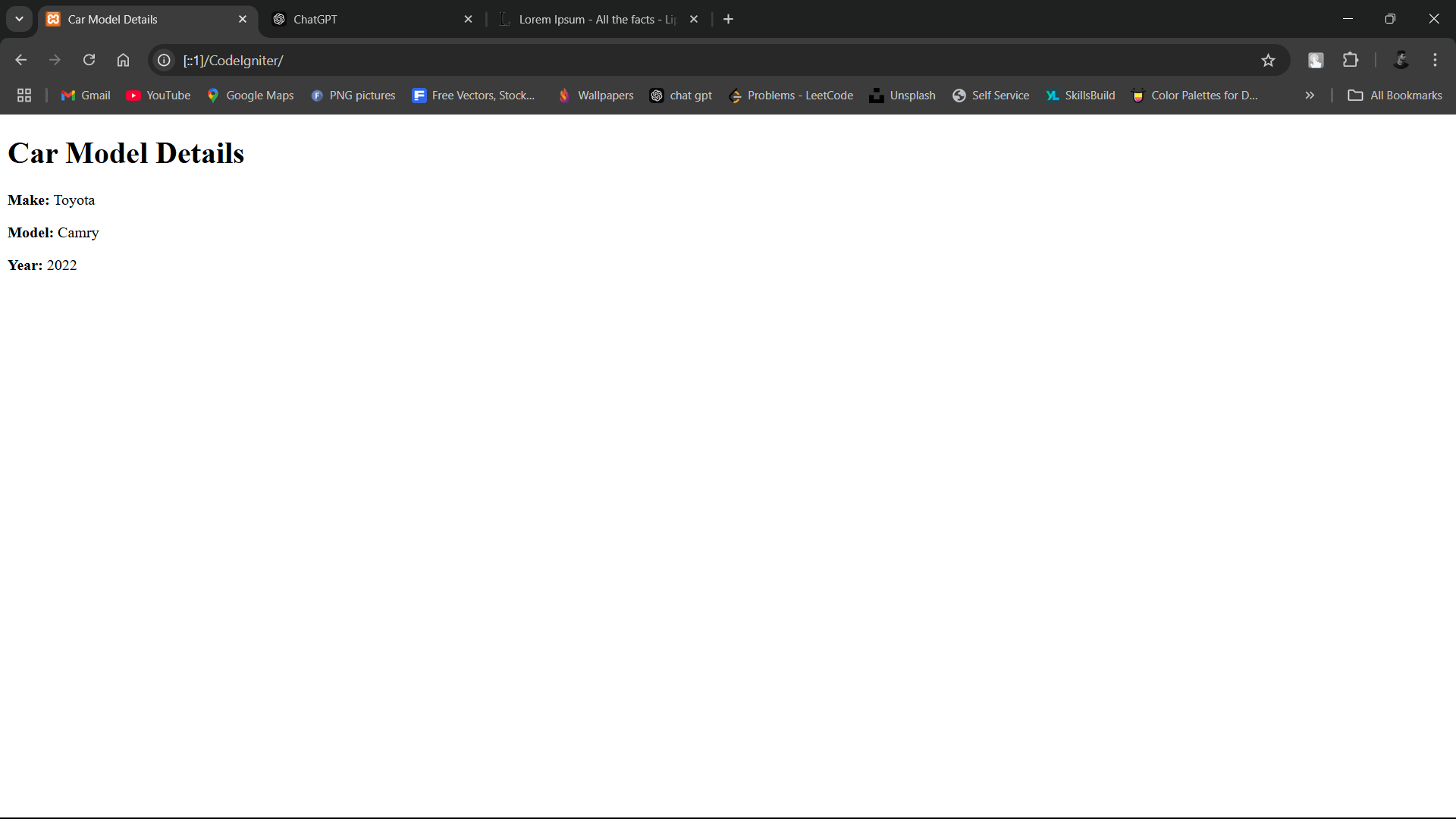
<p><strong>Model:</strong> <?php echo htmlspecialchars($model); ?></p>

<p><strong>Year:</strong> <?php echo htmlspecialchars($year); ?></p>

</body>

</html>

**Output:**



**Practical NO 16:**

**Write a PHP program in CodeIgniter to design a web page featuring a text box for name input, radio buttons for selecting a contact method (Email or Phone), check boxes for choosing interests (Sports, Music, Reading), and buttons for submitting or resetting the form**

1) **Controller** (UserFormController.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class UserFormController extends CI\_Controller {

public function index() {

$this->load->view('user\_form');

}

public function submit() {

// Retrieve input data

$name = $this->input->post('name');

$contact\_method = $this->input->post('contact\_method');

$interests = $this->input->post('interests');

// Prepare data for the view

$data['name'] = $name;

$data['contact\_method'] = $contact\_method;

$data['interests'] = $interests;

// Load the result view

$this->load->view('form\_result', $data);

}

}

?>

2) **View**

* user\_form.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>User Input Form</title>

</head>

<body>

<h1>User Input Form</h1>

<form action="<?php echo site\_url('userformcontroller/submit'); ?>" method="post">

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

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

<label>Contact Method:</label><br>

<input type="radio" name="contact\_method" value="Email" required>Email<br>

<input type="radio" name="contact\_method" value="Phone">Phone<br><br>

<label>Interests:</label><br>

<input type="checkbox" name="interests[]" value="Sports">Sports<br>

<input type="checkbox" name="interests[]" value="Music">Music<br>

<input type="checkbox" name="interests[]" value="Reading">Reading<br><br>

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

<input type="reset" value="Reset">

</form>

</body>

</html>

* form\_result.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Form Submission Result</title>

</head>

<body>

<h1>Submitted Information</h1>

<p><strong>Name:</strong> <?php echo htmlspecialchars($name); ?></p>

<p><strong>Contact Method:</strong> <?php echo htmlspecialchars($contact\_method); ?></p>

<p><strong>Interests:</strong>

<?php

if (!empty($interests)) {

echo implode(", ", $interests);

} else {

echo "None";

}

?>

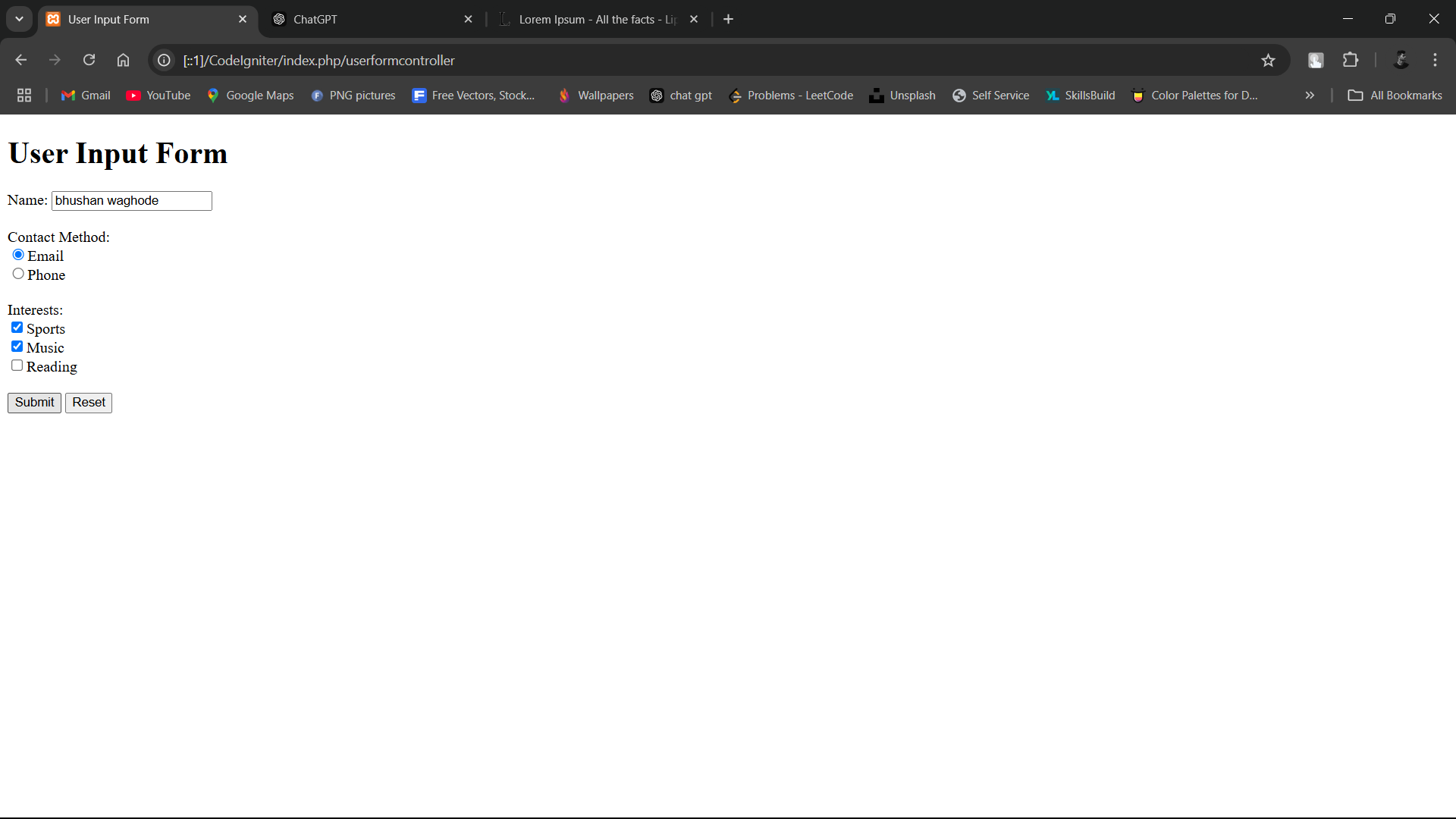
</p>

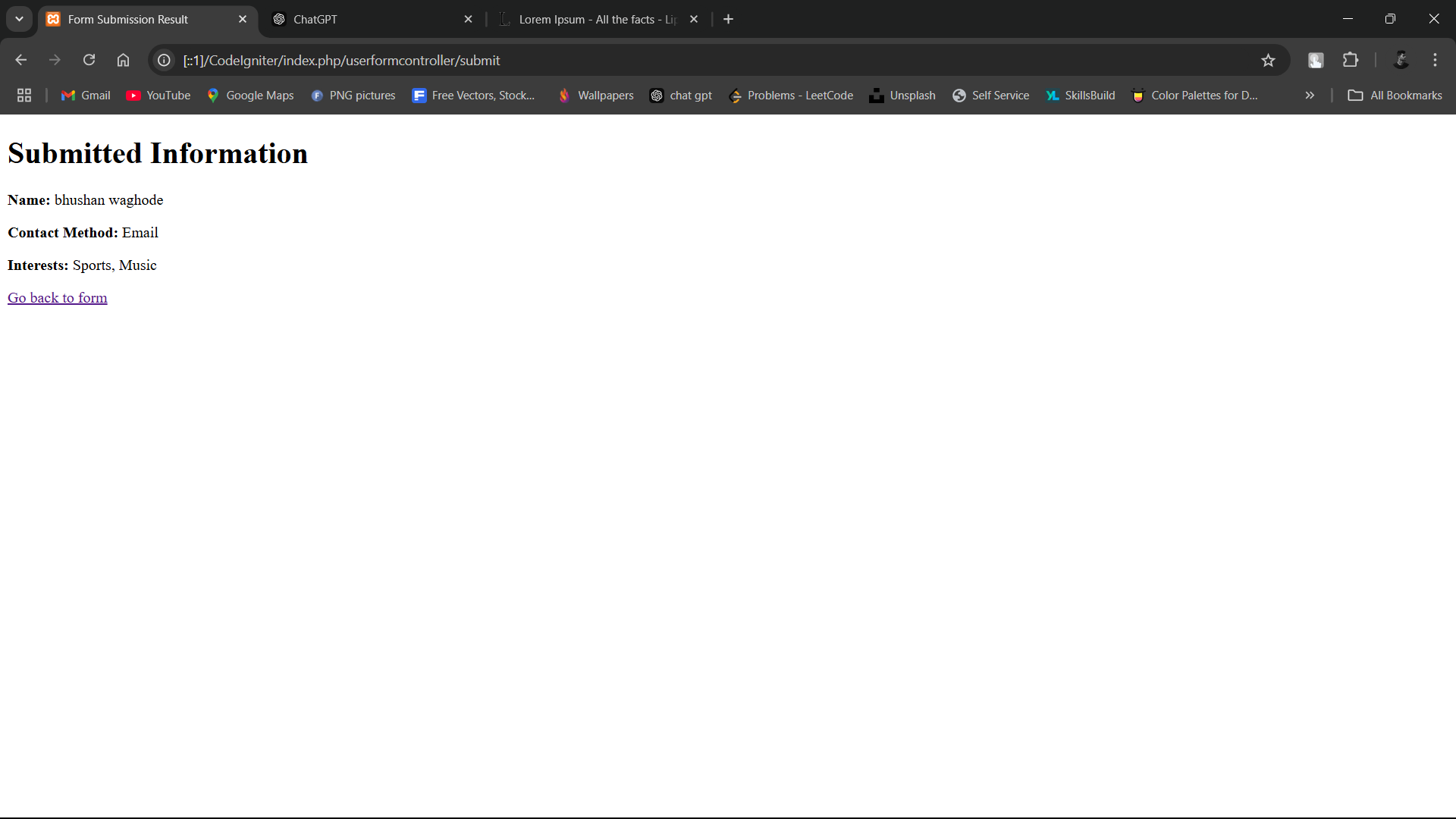
<a href="<?php echo site\_url('userformcontroller'); ?>">Go back to form</a>

</body>

</html>

**Output**





**Practical NO 17:**

**Write a simple PHP program in CodeIgniter that demonstrates introspection and serialization. Use a class to create an object, and then showcase how to inspect its properties and methods using PHP's reflection.**

1) **Class** (Person.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class Person {

public $name;

public $age;

public function \_\_construct($name, $age) {

$this->name = $name;

$this->age = $age;

}

public function greet() {

return "Hello, my name is " . $this->name;

}

}

?>

2) **Controller** (ReflectionController.php)

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class ReflectionController extends CI\_Controller {

public function index() {

// Create an instance of the Person class

$person = new Person("Alice", 30);

// Use Reflection to inspect the Person class

$reflection = new ReflectionClass($person);

// Get properties and methods

$properties = $reflection->getProperties();

$methods = $reflection->getMethods();

// Serialize the object

$serialized\_data = serialize($person);

// Prepare data for the view

$data['properties'] = $properties;

$data['methods'] = $methods;

$data['serialized\_data'] = $serialized\_data;

// Load the view

$this->load->view('reflection\_view', $data);

}

}

?>

3) **View**

* reflection\_view.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Introspection and Serialization</title>

</head>

<body>

<h1>Introspection and Serialization Demo</h1>

<h2>Properties:</h2>

<ul>

<?php foreach ($properties as $property): ?>

<li><?php echo htmlspecialchars($property->getName()); ?></li>

<?php endforeach; ?>

</ul>

<h2>Methods:</h2>

<ul>

<?php foreach ($methods as $method): ?>

<li><?php echo htmlspecialchars($method->getName()); ?></li>

<?php endforeach; ?>

</ul>

<h2>Serialized Object:</h2>

<pre><?php echo htmlspecialchars($serialized\_data); ?></pre>

</body>

</html>

* reflection\_result.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Introspection and Serialization</title>

</head>

<body>

<h1>Introspection and Serialization Demo</h1>

<h2>Properties:</h2>

<ul>

<?php foreach ($properties as $property): ?>

<li><?php echo htmlspecialchars($property->getName()); ?></li>

<?php endforeach; ?>

</ul>

<h2>Methods:</h2>

<ul>

<?php foreach ($methods as $method): ?>

<li><?php echo htmlspecialchars($method->getName()); ?></li>

<?php endforeach; ?>

</ul>

<h2>Serialized Object:</h2>

<pre><?php echo htmlspecialchars($serialized\_data); ?></pre>

</body>

</html>

**Output**

