**\* Project report \***

(Based on industrial training)

ON

**Pomato- A mobile selling website**

*Submitted By* –

AKASH PATIAL(03)

AKSHIT SHARMA(06)

KRITIKA RANA(19)

(Diploma in Computer Engineering)

5th *semester*

*Under the guidance of*

*-Miss Sukhveer Kaur*



Department of Computer Science Engineering

**Government Polytechnic Kangra**

October, 2021

**DECLARATION**

We hereby declare that the Industrial Training Report entitled is an authentic record of our own work of 3 weeks Industrial Training during the period from 01-september-2021 to 21-september-2021 as per the requirement of the institute and for partial fulfillment for the award of diploma in computer engineering, under the guidance of Miss Sukhveer Kaur.

**Dated** :

**(Signature of students)**

**(Diploma in cse- 5tSemester)**

**ACKNOWLEDGEMENT**

“**Choose what you are passionate about & what you love to do. The success will be yours!** “

The experience of making the project had been productive and enjoyable. It proved an opportunity for us to upgrade our skills and upgraded our professional knowledge. In course of present work it has been our privilege to receive help and assistance from many quarters. Before we get into the thick of things, we would like to share few heartfelt words with the people who were the part of project in numerous ways.

My primary thanks goes to Miss.sukhveer Kaur ……..our project guide who poured over every inch of the manuscript with painstaking attention to detail and made a semi- infinite number of helpful suggestions..

Akash Patial (03)

Akshit Sharma (06)

Kritika Rana (19)

***SYNOPSIS***

1. **TITLE OF THE PROJECT:**

POMATO- a mobile selling website

1. **OBJECTIVE OF THE PROJECT:**

The main objective of our project is to sell mobile phones online moreover, to deliver them in urban as well as remote areas.

In India, people of different backgrounds exist and have different living backgrounds like rural, urban, metropolitan cities. So the ratio varies among population and Pomato has made now its accessibility almost in every dimension

As the spread of COVID-19 has affected the educational system of our country and the online classes was introduced. A lot of students was unable to attend online classes so, our website provide you the best brands of

Phones with affordable prices and easily accessible even in remote areas through our website.

1. **PROJECT CATEGORY:**

Web Designing

1. **LANGUAGE AND SOFTWARE USED:**

Language used: HTML and CSS

Software used: Sublime Text

Operating System: window 10

1. **FUTURE SCOPE OF THE PROJECT:**

This project will enhance online marketing

Social networking facilitates the easy logins and purchase of product.

Pomato is working round the clock with innovativeness for delivery of products with intention of reducing costs, time and efforts. To easily buying of mobiles on a single click only.  
The buying and selling of goods online have become quite easy due to E-commerce.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **SNO.** | **CHAPTER NAME** | **TOPICS** |
| **1.** | **Introduction to HTML** | **Html, basic structure, lists, table, forms,** |
| **2.** | **Html tables & forms** | **table, table row, table head , table attributes, forms, input , label, radio button checkbox, submit button** |
| **3.** | **Introduction to CSS** | **Style tag, padding, border, bg color, types of CSS** |
| **4.** | **Animation** | **Marquee, display, transition, position, Navigation bar, text decoration, letter spacing, text index opacity.** |
| **5.** | **Java script** | **Introduction to JavaScript, Id, class, tags, For loop, If condition, if Else condition, else if Else condition, while loop, do-while loop, AND, OR, NOT,** |
| **6.** | **Functions** | **Anonymous function, alert function, Object constructor function, arrays: push, pop, unshift, buttons, onclicks, Onmouseover, onmouseout,** |
| **7.** | **Php** | **Xampp installation, php variables, php function declaration,** |

**Chapter 1**

## What is HTML?

* HTML stands for Hyper Text Markup Language
* HTML is the standard markup language for creating Web pages
* HTML describes the structure of a Web page
* HTML consists of a series of elements
* HTML elements tell the browser how to display the content
* HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

**What is an HTML Element?**

An HTML element is defined by a start tag, some content, and an end tag:<tagname>Content goes here...</tagname>

The HTML **element** is everything from the start tag to the end tag:

<h1>My First Heading</h1>

<p>My first paragraph.</p>

***Basic structure-***

Example-

<!DOCTYPE html>  
<html>  
<head>  
<title>Page Title</title>  
</head>  
<body>  
<h1>My First Heading</h1>  
<p>My first paragraph.</p>  
</body>  
</html>

# *HTML Lists:*

HTML lists allow web developers to group a set of related items in lists.

## Unordered HTML List

An unordered list starts with the <ul> tag. Each list item starts with the <li> tag.The list items will be marked with bullets (small black circles) by default

**Ordered HTML List**

An ordered list starts with the <ol> tag. Each list item starts with the <li> tag.The list items

**HTML Description Lists**

A description list is a list of terms, with a description of each term.

The <dl> tag defines the description list, the <dt> tag defines the term (name), and the <dd> tag describes each term

Chapter 2

## *HTML Table -*

A table in HTML consists of table cells inside rows and columns

## *Table Cells*

Each table cell is defined by a <td> and a </td> tag. Td stands for table data. Everything between <td> and </td> are the content of the table cell.

## *Table Rows*

Each table row starts with a <tr> and end with a </tr> tag. Tr stands for table row.

## *Table Headers*

Sometimes you want your cells to be headers, in those cases use the <th> tag instead of the <td> tag

## HTML Table Tags

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<table>](https://www.w3schools.com/tags/tag_table.asp) | Defines a table |
| [<th>](https://www.w3schools.com/tags/tag_th.asp) | Defines a header cell in a table |
| [<tr>](https://www.w3schools.com/tags/tag_tr.asp) | Defines a row in a table |
| [<td>](https://www.w3schools.com/tags/tag_td.asp) | Defines a cell in a table |
| [<caption>](https://www.w3schools.com/tags/tag_caption.asp) | Defines a table caption |
| [<col>](https://www.w3schools.com/tags/tag_col.asp) | Specifies column properties for each column within a <colgroup> element |
| [<thead>](https://www.w3schools.com/tags/tag_thead.asp) | Groups the header content in a table |
| [<tbody>](https://www.w3schools.com/tags/tag_tbody.asp) | Groups the body content in a table |

# *HTML Forms -*

An HTML form is used to collect user input. The user input is most often sent to a server for processing.

First name:  
  
Last name:  
  
  


## *The <form> Element*

The HTML <form> element is used to create an HTML form for user input

The <form> element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc

## *The <input> Element*

The HTML <input> element is the most used form element.

An <input> element can be displayed in many ways, depending on the type attribute.

## *The <label> Element*

Notice the use of the <label> element in the example above.

The <label> element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.

The <label> element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the <label> element, it toggles the radio button/checkbox.

## *Radio Buttons*

The <input type="radio"> defines a radio button.

Radio buttons let a user select ONE of a limited number of choices.

## *Checkboxes*

The <input type="checkbox"> defines a **checkbox**.

Checkboxes let a user select ZERO or MORE options of a limited number of choices.

## *The Submit Button*

The <input type="submit"> defines a button for submitting the form data to a form-handler.

The form-handler is typically a file on the server with a script for processing input data.

Chapter 3

### CSS stands for “cascade style sheets”.

CSS is the language we use to style an HTML document.

CSS describes how HTML elements should be displayed.

This tutorial will teach you CSS from basic to advanced.

It is used to style webpages. We use css to change look and feel of a website.

There are three types of css:

🡪**Inline CSS**(can style an element)

🡪**Internal CSS**(can style an entire webpage)

🡪**External CSS**(can style an entire website)

## CSS Padding

The CSS padding properties are used to generate space around an element's content, inside of any defined borders.

With CSS, you have full control over the padding. There are properties for setting the padding for each side of an element (top, right, bottom, and left).

# CSS Borders

## The CSS border properties allow you to specify the style, width, and color of an element's border.

## CSS Border Style

The border-style property specifies what kind of border to display.

The border-style property can have from one to four values (for the top border, right border, bottom border, and the left border).

Chapter 4

**ANIMATIONS**

**MARQUEE**

Marquee is an animation effect for web pages used to create horizontal or vertical scrolling text and images. The <marquee> element of HTML is not a standard-compliant.

## Syntax

<marquee attribute\_name = "attribute\_value"....more attributes>

One or more lines or text message or image

</marquee>

### Example

<!DOCTYPE html>

<html>

<body>

<marquee>This is basic example of marquee</marquee>

</body>

</html>

## The position Property

The position property specifies the type of positioning method used for an element. There are five different position values:

* static
* relative
* fixed
* absolute
* sticky

**Navigation Bar**

Having easy-to-use navigation is important for any web site.

With CSS you can transform boring HTML menus into good-looking navigation bars.

# CSS text-decoration

The text-decoration property specifies the decoration added to text, and is a shorthand property for:

* text-decoration-line (required)
* text-decoration-color
* text-decoration-style

## *Syntax*

text-decoration: text-decoration-line text-decoration-color text-decoration-style|initial|inherit;

# CSS letter-spacing

The letter-spacing property increases or decreases the space between characters in a text.

## *Syntax*

letter-spacing: normal|*length*|initial|inherit;

# CSS opacity

The opacity property sets the opacity level for an element.

The opacity-level describes the transparency-level, where 1 is not transparent at all, 0.5 is 50% see-through, and 0 is completely transparent.

## *Syntax*

opacity: *number*|initial|inherit;

Chapter 5

**INTRODUCTION TO JAVASCRIPT**

JavaScript is a lightweight, cross-platform, and interpreted scripting language. It is well-known for the development of web pages, many non-browser environments also use it. JavaScript can be used for Client-side developments as well as Server-side developments. JavaScript contains a standard library of objects, like Array, Date, and Math, and a core set of language elements like operators, control structures, and statements.

One of many JavaScript HTML methods is getElementById().

*Example-*

document.getElementById("demo").innerHTML = "Hello JavaScript";

# JavaScript For Loop

Loops can execute a block of code a number of times.

The for loop has the following **syntax**:

for (*statement 1*;*statement 2*;*statement 3*) {  
  // *code block to be executed*  
}

**Statement 1** is executed (one time) before the execution of the code block.

**Statement 2** defines the condition for executing the code block.

**Statement 3** is executed (every time) after the code block has been executed.

### *Example*

for (let i = 0; i < 5; i++) {  
  text += "The number is " + i + "<br>";  
}

## Conditional Statements

In JavaScript we have the following conditional statements:

* if
* else
* else
* switch

## The if Statement

Use the if statement to specify a block of JavaScript code to be executed if a condition is true.

### *Syntax*

if (*condition*) {  
  //  block of code to be executed if the condition is true}

## The else Statement

Use the else statement to specify a block of code to be executed if the condition is false.

### *Syntax*

if (*condition*) {  
  //  block of code to be executed if the condition is true} else {  
  //  block of code to be executed if the condition is false}

## The else if Statement

### *Syntax*

if (*condition1*) {  
  //  block of code to be executed if condition1 is true} else if (*condition2*) {  
  //  block of code to be executed if the condition1 is false and condition2 is true  
} else {  
  //  block of code to be executed if the condition1 is false and condition2 is false}

## The Switch Statement

The switch statement is used to perform different actions based on different conditions. Use the switch statement to select one of many code blocks to be executed.

### *Syntax*

switch(expression) {  
  case x:  
    *// code block*    break;  
  case y:  
    *// code block*    break;  
  default:  
    // code block  
}

# While Loop

The while loop loops through a block of code as long as a specified condition is true.

### Syntax

while (condition) {  
*// code block to be executed*  
}

## The Do While Loop

The do while loop is a variant of the while loop. This loop will execute the code block once, before checking if the condition is true, then it will repeat the loop as long as the condition is true.

### *Syntax*

do {  
*// code block to be executed*}  
while (condition);

CHAPTER 6

# Function

JavaScript functions are **defined** with the function keyword.

You can use a function **declaration** or a function **expression**.

function functionName(parameters) {  
  // code to be executed  
}

## Anonymous functions

An anonymous function is a [function](https://www.javascripttutorial.net/javascript-function/) without a name. An anonymous function is often not accessible after its initial creation.

***Example-***

**let show = function() {**

**console.log(‘Anonymous function’);**

**};**

**Show();**

# Alert() Method

The alert() method displays an alert box with a specified message and an OK button. An alert box is often used if you want to make sure information comes through to the user.

## *Syntax*

alert(message)

### *Example-*

### alert("Hello! I am an alert box!!");

## Object constructor function

Sometimes we need a "**blueprint**" for creating many objects of the same "type".

The way to create an "object type", is to use an **object constructor function**.

In the example below, function Person() is an object constructor function.

### *Example-*

function Person(first, last, age, eye) {  
  this.firstName = first;  
  this.lastName = last;  
  this.age = age;  
  this.eyeColor = eye;  
}

# Array methods in JavaScript

# JavaScript gives us four methods to add or remove items from the beginning or end of arrays:

🡪Push()

🡪pop()

🡪shift

🡪unshift

### **pop()**: Remove an item from the end of an array

let cats = ['Bob', 'Willy', 'Mini'];

cats.pop(); // ['Bob', 'Willy'] | pop() returns the removed item.

### **push()**: Add items to the end of an array

*Example-*

let cats = ['Bob'];

cats.push('Willy'); // ['Bob', 'Willy']

cats.push('Puff', 'George'); // ['Bob', 'Willy', 'Puff', 'George']

push() returns the new array length.

### ****shift()****: Remove an item from the beginning of an array

let cats = ['Bob', 'Willy', 'Mini'];

cats.shift(); // ['Willy', 'Mini']

shift() returns the removed item.

### ****unshift()****: Add items to the beginning of an array

let cats = ['Bob'];

cats.unshift('Willy'); // ['Willy', 'Bob']

cats.unshift('Puff', 'George'); // ['Puff', 'George', 'Willy', 'Bob']

unshift() returns the new array length.

# onclick Event

# The onclick event occurs when the user clicks on an element.

## *Syntax*

In html :

<*element* onclick="*myScript*">

In javascript:

object.onclick = function(){*myScript*};

# Onmouseover Event

The Onmouseover event occurs when the mouse pointer is moved onto an element, or onto one of its children.

## *Syntax*

In html:

<*element* onmouseover="*myScript*">

**In jacascript;**

object.onmouseover = function(){*myScript*};

# Onmouseout Event

The onmouseout event occurs when the mouse pointer is moved out of an element, or out of one of its children.

## *Syntax*

**In html**

<*element* onmouseout="*myScript*">

In javascript

object.onmouseout = function(){*myScript*};

**Chapter 7**

**Php- hypertext preprocessor**

* PHP is an acronym for "PHP: Hypertext Preprocessor"
* PHP is a widely-used, open source scripting language
* PHP scripts are executed on the server
* PHP can generate dynamic page content
* PHP can create, open, read, write, delete, and close files on the server
* PHP can collect form data
* PHP can send and receive cookies
* PHP can add, delete, modify data in your database
* PHP can be used to control user-access
* PHP can encrypt data

# *PHP Syntax*

A PHP script can be placed anywhere in the document.A PHP script starts with <?php and ends with ?>

<?php  
// PHP code goes here  
?>

The default file extension for PHP files is ".php".

PHP statements end with a semicolon (;)

*Example-*

<!DOCTYPE html>  
<html>  
<body>  
  
<h1>My first PHP page</h1>  
  
<?php  
echo "Hello World!";  
?>  
  
</body>  
</html>

# PHP Variables

In PHP, a variable starts with the $ sign, followed by the name of the variable

* A variable starts with the $ sign, followed by the name of the variable
* A variable name must start with a letter or the underscore character
* A variable name cannot start with a number
* A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and \_ )

*Example*-

<?php  
$txt = "Hello world!";  
$x = 5;  
$y = 10.5;  
?>

After the execution of the statements above, the variable $txt will hold the value Hello world!, the variable $x will hold the value 5, and the variable $y will hold the value 10.5

# PHP Functions

## PHP Built-in Functions

PHP has over 1000 built-in functions that can be called directly, from within a script, to perform a specific task.

## PHP User Defined Functions

Besides the built-in PHP functions, it is possible to create your own functions.

A user-defined function declaration starts with the word function:

### Syntax

function *functionName*() {  
*code to be executed*;  
}

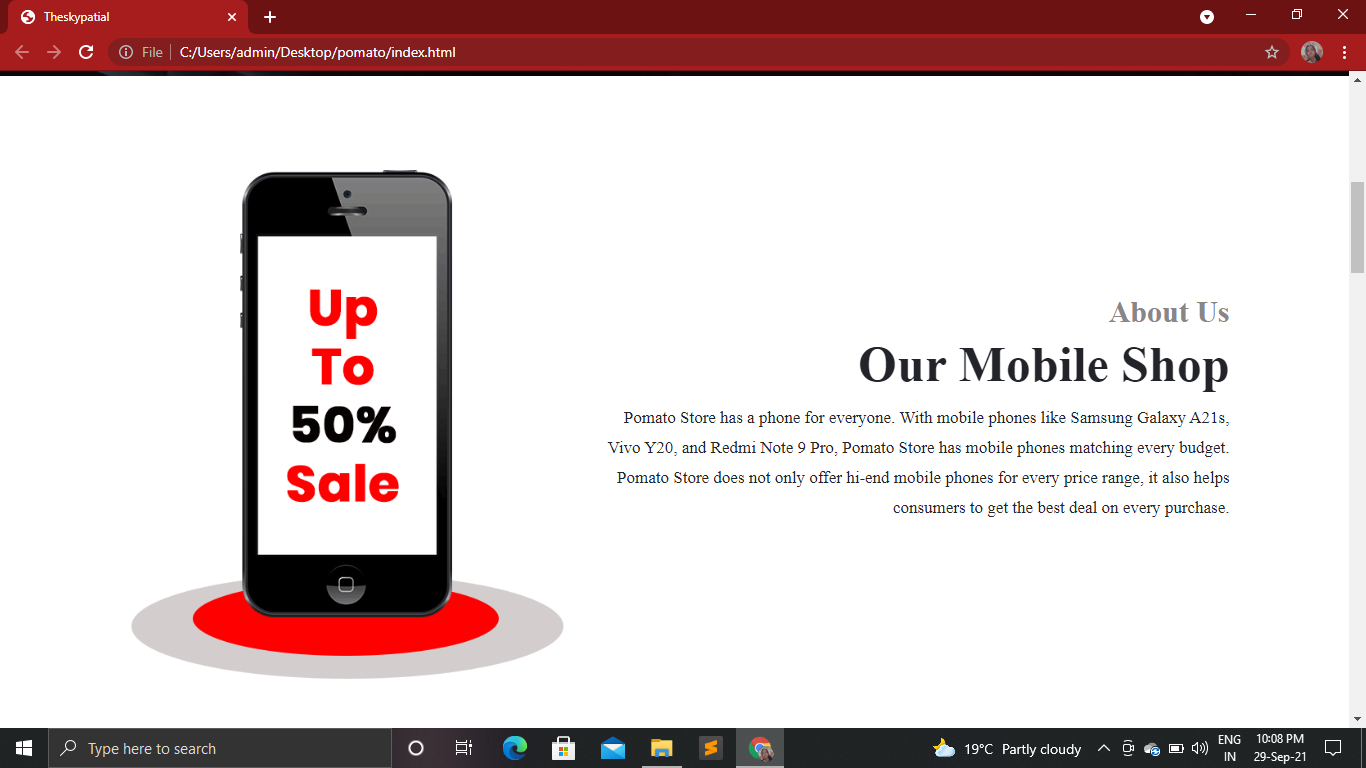
## PHP Function Arguments

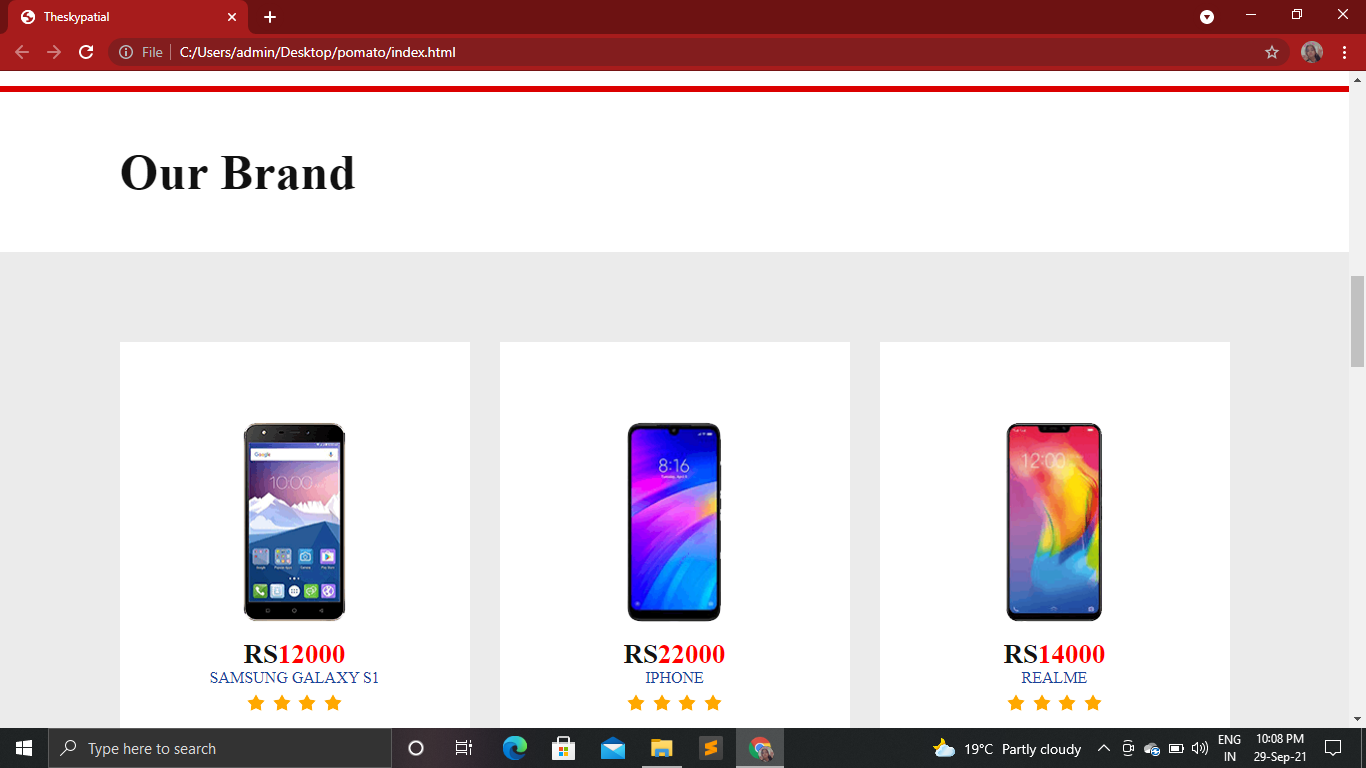
Information can be passed to functions through arguments. An argument is just like a variable.

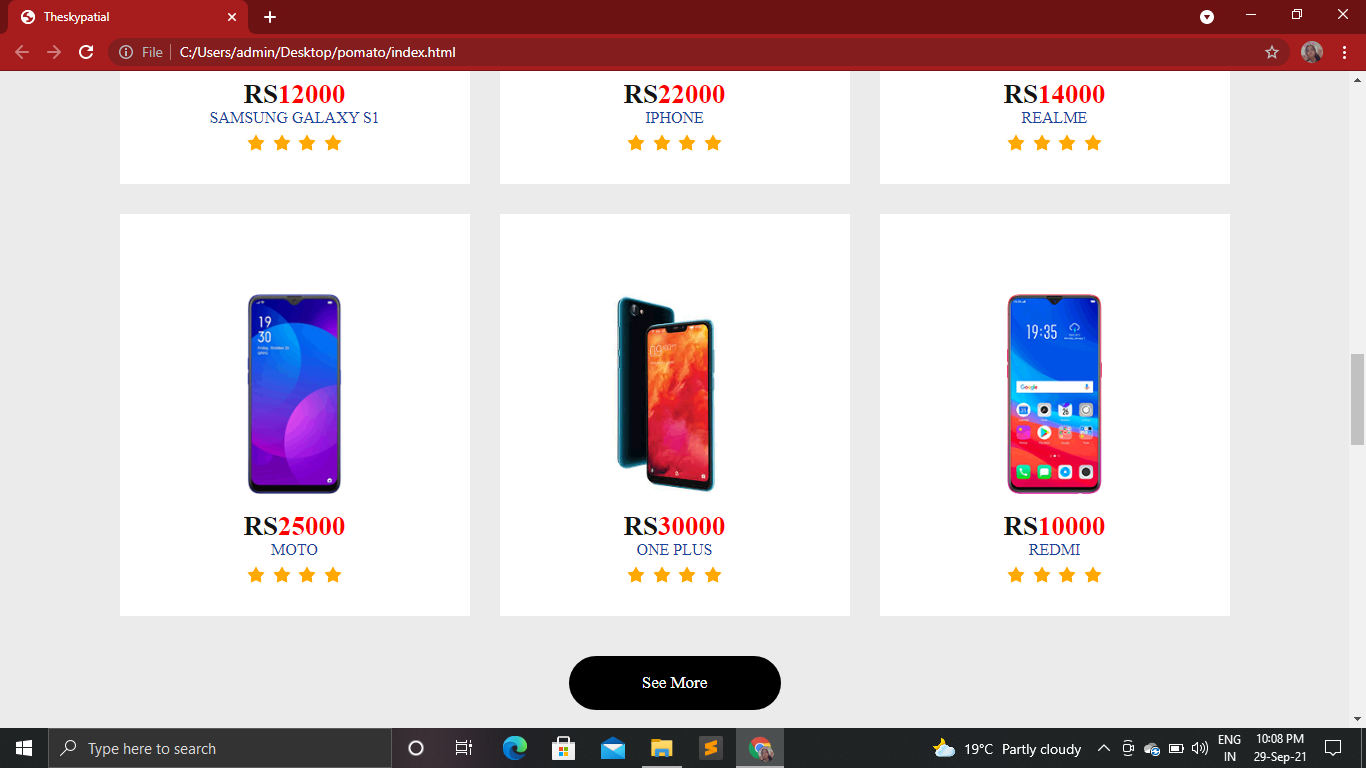
Arguments are specified after the function name, inside the parentheses. You can add as many arguments as you want, just separate them with a comma.

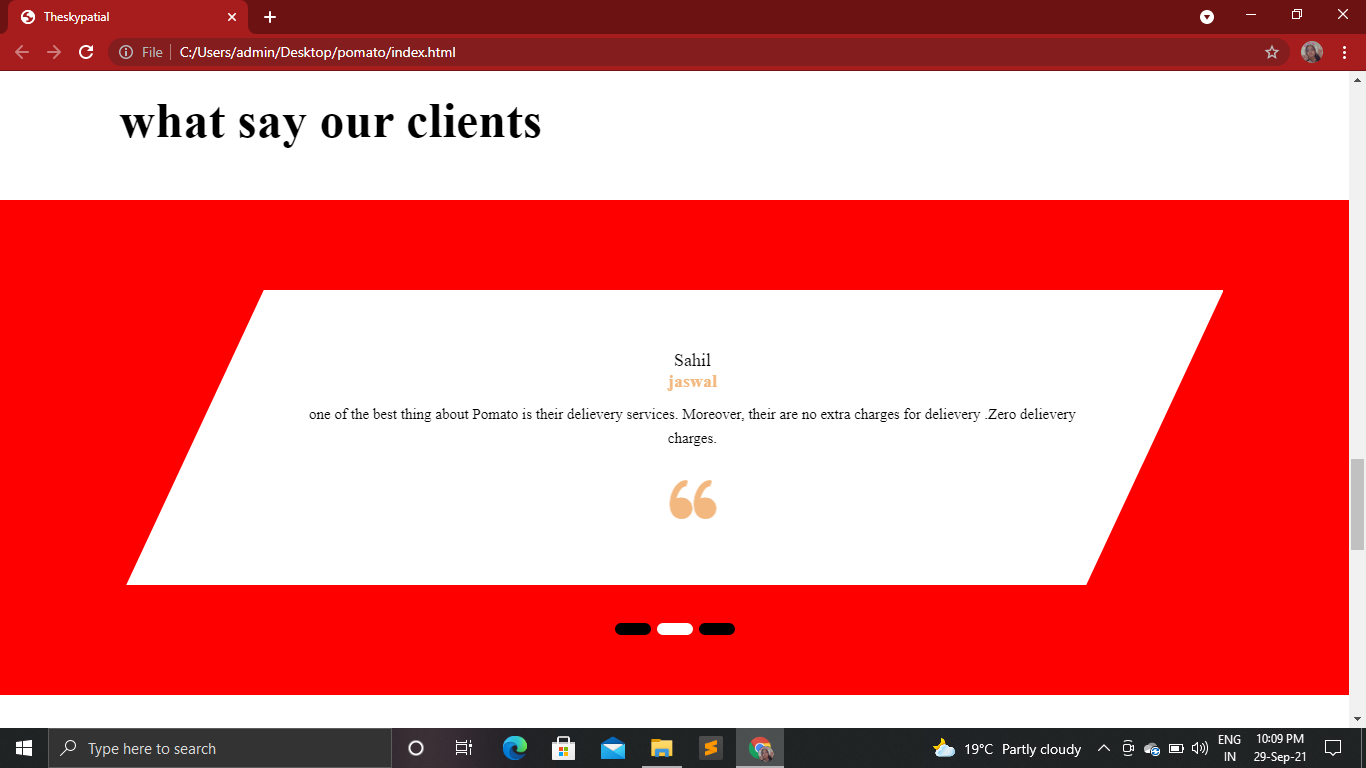
**SNAPSHOTS –**

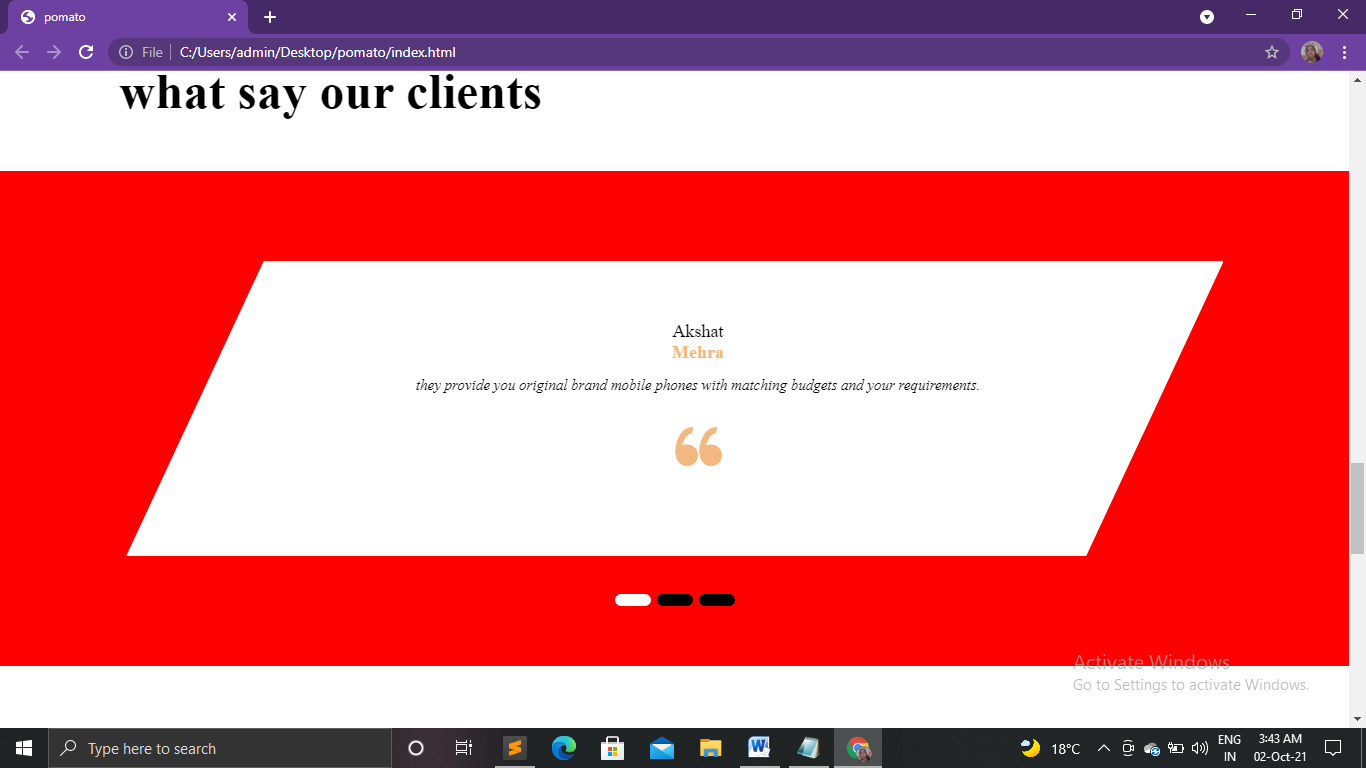
# 

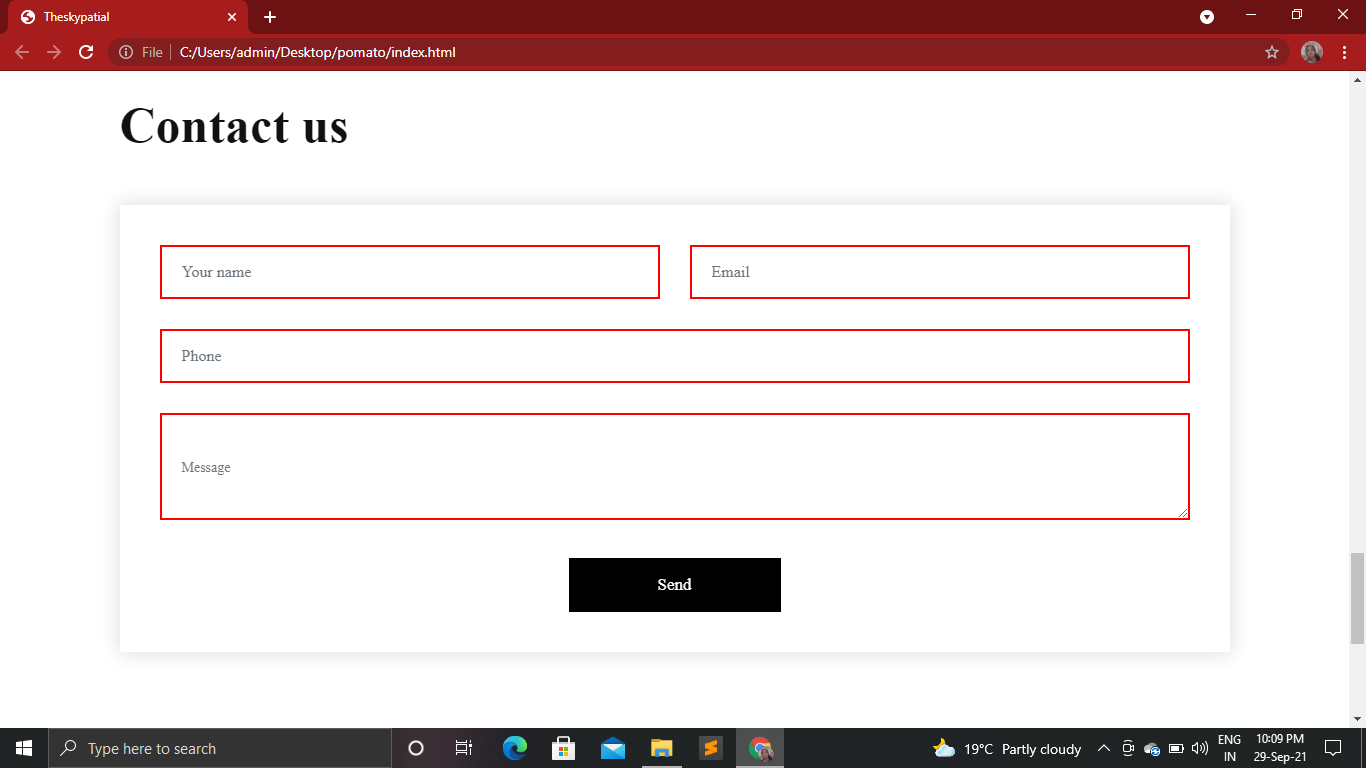
****

****

****

****

****

****

