# INTRODUCTION TO HTML

**Web site:** A set of interconnected web pages, usually including a homepage, generally located on the same server, and prepared and maintained as a collection of information by a person, group, or organization.

**Web Page:** A web page is a document that's created in html that shows up on the internet when you type in or go to the web page's address.

**Types of Web Pages:**

**Static web page:** is delivered exactly as stored, as web content in the web server's file system. Contents cannot be changed.

**Dynamic web page**: is generated by a web application that is driven by server-side software or client-side scripting. Dynamic web pages help the browser (the client) to enhance the web page through user input to the server. Contents can be changed as evolution over time.

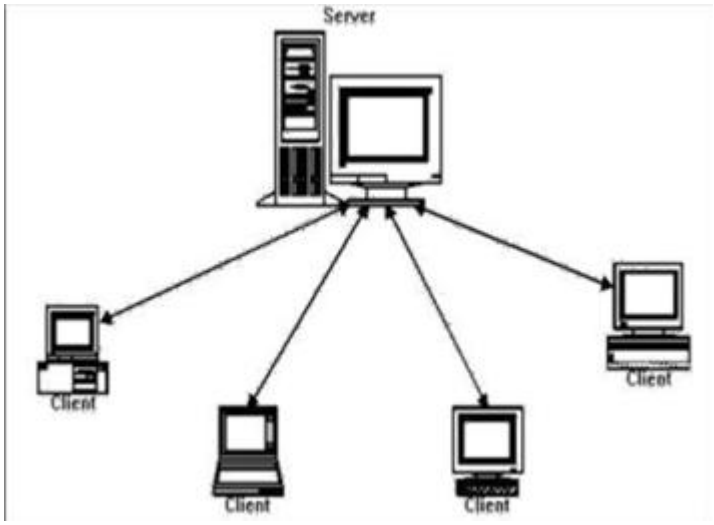
**Browsers & their types:**

A web browser (commonly referred to as a browser) is a software application for retrieving, presenting and traversing information resources on the World Wide Web.

**Client –Server Model**

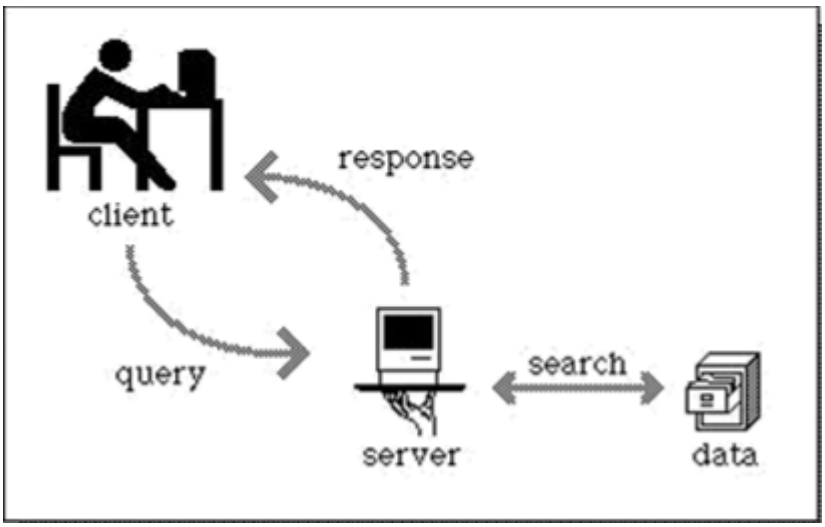
The client–server model is a distributed application structure in computing that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients. Often clients and servers communicate over a computer network. A server is a host that is running one or more server programs which share their resources with clients. A client requests a server's content or service function.

The major web browsers are Google Chrome, Firefox, Internet Explorer, Opera, and Safari.



**Web –Server**

Web server refers to either the hardware (the computer) or the software (the computer application) that helps to deliver web content that can be accessed through the Internet. The most common use of web servers is to host websites, but there are other uses such as gaming, data storage or running enterprise application.




**Working of different types of web pages**

The different types of web pages are :

**Advocacy:** An advocacy web page is one sponsored by an organization to influence opinion. URL ends with .org

**Business and marketing:** It is one sponsored by a commercial enterprise to sell or market their services. URL ends with .com

**News:** It provides timely information about current events and issues. Informational: This includes reports, research findings, schools and college information. URL ends with .edu or .gov.

**Personal:** It is created by an individual for his /her own personal need.URL has tidle(~). General structure of a Web Page A basic HTML page contains a Head section and a Body section. The contents of the head section are normally invisible in a web browser and mainly consists of some Metatags. The Body consist of those HTML elements that you want to have displayed in your browser.

**General structure of a Web Page**

A basic HTML page contains a Head section and a Body section. The contents of the head section are normally invisible in a web browser and mainly consists of some Metatags. The Body consist of those HTML elements that you want to have displayed in your browser.

```
<html>
<head>
</head>
 <body>
 </body>
</html>
```

**Scripting language:**

A scripting language or script language is a programming language that supports the writing of scripts, programs written for a special runtime environment that can interpret and automate the execution of tasks which could alternatively be executed one- by-one by a human operator.

**URL:**

A uniform resource locator (URL), also known as web address, is a specific character string that constitutes a reference to a resource. In most web browsers, the URL of a web page is displayed on top inside an address bar. An example of a typical URL would be "http://en.example.org/wiki/Main_Page".

Popular Search Engines
Yahoo Search
Google Search
Bing
Info.com
Search.com
Infospace

WWW: The World Wide Web (WWW) is a system of interlinked hypertext documents accessed via the Internet. With a web browser, one can view web pages that may contain text, images, videos, and other multimedia, and navigate between them via hyperlinks.

ILLUSTRATING HTML TAGS AND THEIR ATTRIBUTES

**HTML:**

Hyper Text Markup Language is the most widely used language to write web pages.it is a markup language.

Hypertext :Refers to the way in which web pages are linked together.

Markup Language: The user simply markups a text document with tags that tell a web browser how to structure it to display.

Creating HTML document :To begin coding HTML user needs only three things:

1. A simple text editor (notepad). 2. A web browser. 3. VS Code

**HTML document structure:** An HTML document starts and ends with <html> and </html> tags.These tags tell the browser that the entire document is composed in HTML.
1. The <head>……</head> elements contains information about the document such as title of the document etc.
2. The <body>…. </body> elements contains the real content of the document that you see on your screen.

**ATTRIBUTES:** An attribute is used to define the characteristics of an element and is placed inside the element's opening tag. All attributes are made up of 2 parts: a name and a value. -The name is the property you want to set. -The value is what you want the value of the property to be. Example: <font face="arial" color="red">

**1. Develop a HTML program to display**
**(i) Welcome Full Stack Web Development & Own paragraph about why learning of Full stack web development course.**
**(ii) Create a Course Registration form using HTML**

Solution

O/P

prg:

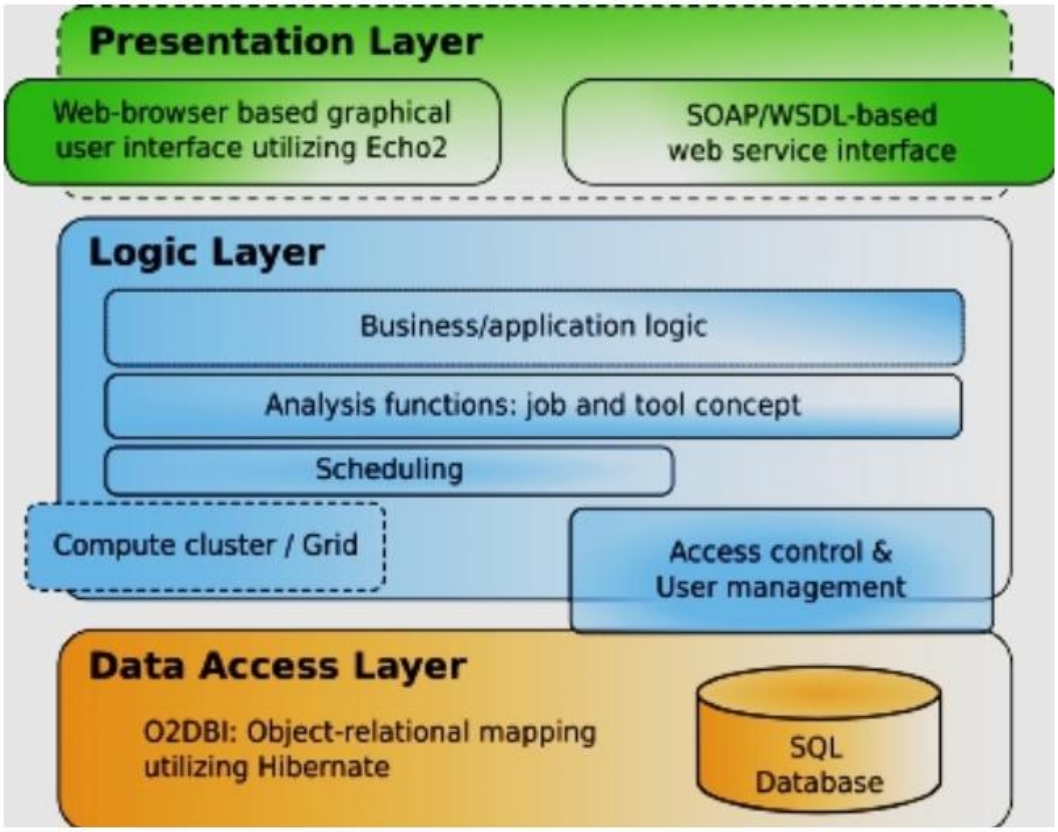
```
<!DOCTYPE html>
<title> Full Stack  Web  Development </title>
<h1> Introduction to Full Stack Web Development </h1>
<p style="font-size:150%;">
    Activities of both front end and web development  works.<br> what are the
Activities ? <br>
    <ol>
    <ul style="font-size:20px">
        <li > Presenatation Layer or UI or front end apperance layer <br> which
consists of language  HTML,CSS,JavaScript  </li>
        <li> Logic Layer:Place/bring/reterieve  the data  to consumption of
client <br>
             Involves in auth/auth API design, logic to implement business logic
  </li>
        <li> Data Base Layer ; Store/reterive CRUD operations of Database </li>

    </ul>
   </ol>
<img src="./Full_Stack_3_layers.jpg" alt="3 layers" >
<h1> Concepts of Web developers needs to know  </h1>
<p>
    <ul style="font-size:20px">
```

```
        <li> Web </li>
        <li> url </li>
        <li> HTTPS </li>
        <li> WebSite </li>
        <li> Domain Name </li>
        <li> Hosting </li>
        <li> Server </li>
    </ul>
</p>
</p>
```

# Introduction to Full Stack Web Development

Activities of both front end and web development works.
what are the Activities ?

- Presenatation Layer or UI or front end apperance layer
  which consists of language HTML,CSS,JavaScript
- Logic Layer:Place/bring/reterieve the data to consumption of client
  Involves in auth/auth API design, logic to implement business logic
- Data Base Layer ; Store/reterive CRUD operations of Database




# Concepts of Web developers needs to know

- Web
- url
- HTTPS
- WebSite
- Domain Name
- Hosting
- Server

**(ii) Create a Course Registration form using HTML**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Course Registration Form </title>
    <img  src="./Heading_Sagar.jpg" alt="header Dayananda Sagar">
    <h1  style="text-align:center;">   Even  Semester Course  Registration Form
</h1>

</head>
<style>
    table,th,td{
        border:1px solid black;
    }
</style>
<body>
    <table>
        <tr>
        <th> Sl No </th>
        <th> Description </th>
        <th> Course Details </th>
    </tr>
    <tr>
        <td> 1 </td>
        <td> Program </td>
        <td style="color:red";> ISE </td>
    </tr>
    <tr>
        <td> 2 </td>
        <td> Name of the Candidate </td>
        <td style="color:green";> Jagatha </td>

    </tr>
    <tr>
        <td> 3 </td>
        <td> USN  </td>
        <td> 1DS21IS091 </td>
    </tr>
    <tr>
        <td> 4 </td>
        <td> Semester & Section  </td>
        <td> 6B </td>
```

```
    </tr>
    <tr>
        <td> 5 </td>
        <td> Contact No  </td>
        <td> 9916988325 </td>
    </tr>
    <tr>
        <td> 8   </td>
        <td> Course reg for even <br> the semester    </td>
        <td>
         <table>
            <tr>
            <th> Course </th>
            <th> Course Code </th>
            <th> Tick </th>
        </tr>
             <tr>
                <td> Big Data Analytics  </td>
                <td> PCC211S61 </td>
                <td> Yes <br> No  </td>
             </tr>
             <tr>
                <td> Full Stack Development   </td>
                <td> IPCC21IS62  </td>
                <td> Yes <br> No  </td>
             </tr>
             <tr>
                <td> Full Stack Development Lab  </td>
                <td> IPCC21IS62  </td>
                <td> Yes <br> No  </td>
             </tr>
             <tr>
                <td> Web Technology  Lab  </td>
                <td> PCC21IS66  </td>
                <td> Yes <br> No  </td>
             </tr>
         </table>
        </td>

    </tr>
    <tr>
        <td> 9 </td>
        <td colspan="2" style="color: brown;"> * I am aware of the University
Regulation  (Autonomous) govering the UG and I abide
           rules and regulations of the Institution.
```

```
          </td>

    </tr>
    <tr>
        <td> 10 </td>
        <td> Signature  (Candidate full name and Sign) and Date </td>
        <td> Jahatha SC  </td>
    </tr>
     </table>
</body>
</html>
```

o/P

DAYANANDA SAGAR COLLEGE OF ENGINEERING

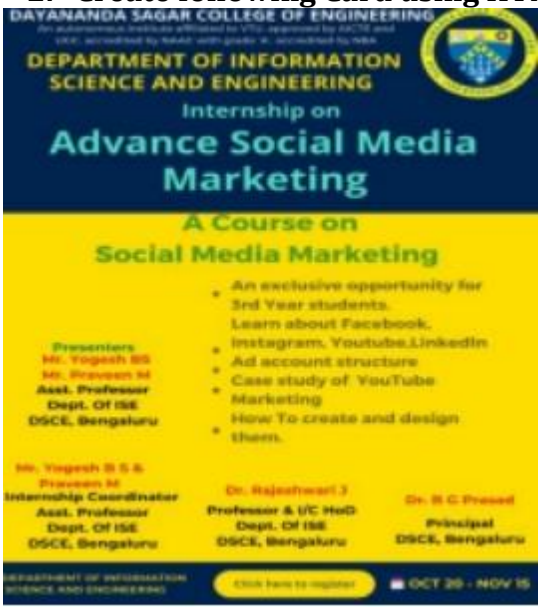
(An Autonomous Institute affiliated to Visvesvaraya Technological University (VTU), Belagavi, Approved by AICTE and UGC, Accredited by NAAC with 'A' grade & ISO 9001-2015 Certified Institution)

Department of Information Science and Engineering

# Even Semester Course Registration Form

| Sl No | Description | Course Details |
|---|---|---|
| 1 | Program | ISE |
| 2 | Name of the Candidate | Jagatha |
| 3 | USN | 1DS21IS091 |
| 4 | Semester & Section | 6B |
| 5 | Contact No | 9916988325 |
| 8 | Course reg for even the semester | Big Data Analytics — PCC211S61 — Yes No<br>Full Stack Development — IPCC21IS62 — Yes No<br>Full Stack Development Lab — IPCC21IS62 — Yes No<br>Web Technology Lab — PCC21IS66 — Yes No |
| 9 | * I am aware of the University Regulation (Autonomous) govering the UG and I abide rules and regulations of the Institution. | |
| 10 | Signature (Candidate full name and Sign) and Date | Jahatha SC |

## 2. Create following Card using HTML




```
<!DOCTYPE html>
<html lang="en">
<head>
    <img src="/2.1+Heading+Element/1.1+Full_Stack_development/Sagar_logo.jpg" alt="Sagar
Logo">
</head>
<style>
.blueclass{
    background-color: #002868;
      height: 300px;
      width: 650px;
      top: 401px;
      text-align: center;
      font-size: xx-large;
      color:rgb(247, 255, 2)
  }
  .subheading{
    font-size: 75%;
    font-weight:bold;
    color:#90ccee


  }
  .subheading1{
    font-family:"Times New Roman", sans-serif
    font-weight:bold;
    font-size: 100%;
    color:#90ccee


  }
.yellowclass{
    background-color: yellow;
      height: 400px;
      width: 650px;
      top: 401px;
```

```
      text-align: center;
      font-size: xx-large;
      color:rgb(243, 82, 232)


  }
  .ymainheading{

    font-size: 50%;
    font-weight:small;
    color:black;
    text-align: left;

  }

 .lastclass
 {
    background-color: blue;
      height: 100px;
      width: 650px;
      top: 401px;
      font-size: 100%;
      color: rgb(255, 242, 0);
  }
  p { margin:0 }

</style>


<body text="white">
    <div class="blueclass"> DEPARTMENT OF INFORMATION <br> SCIENCE AND ENGINERING
    <p class="subheading"> Internship on </p>
    <p class="subheading1"> Advance Social Media  <br> Marketing </p>
    </div>
    <div class="yellowclass">
      <p > A Course on   </p>
      <p>   Social Media Marketing</p>
      <ul class="ymainheading"> <li> An exclusive opportunity for <br> 3rd year students </li>
      <li> Introduction to facebook,<br> Instagram,Youtube,Linkedln</li>
      <li> Ad account structure </li>
      <li> Case study of Youtube, Marketing</li>
      <li> Architecture of above social media</li>
      </ul>
      <p> Presenters </p>
      <p> Mr XXXX BS & <br>
          Mr XXXXX M </p>
    </div>
    <div class="lastclass"> <p> DEPARTMENT OF INFORMATION <br> SCIENCE AND ENGINEERING

                     
      <input type="button" value="Click here to register " onclick=<a
href = "https://www.html.com"> </a>
        Date  Oct 20 -Nov 15th
    </p>

    </div>"

</body>
```

**3. For the above Card display/any other card perform formatting using CSS.**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="Cache-Control" content="no-cache, no-store, must-
revalidate" />
    <meta http-equiv="Pragma" content="no-cache" />
    <meta http-equiv="Expires" content="0" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Course Registration Form </title>
    <img
 src="/2.1+Heading+Element/1.1+Full_Stack_development/Heading_Sagar.jpg"
alt="header Dayananda Sagar">
    <h1  style="text-align:center;">   Even  Semester Course  Registration
Form </h1>

</head>
<style>
    table,th,td{
        border:1px solid black;
    }
</style>
<body>
    <table>
        <tr>
        <th> Sl No </th>
        <th> Description </th>
        <th> Course Details </th>
    </tr>
    <tr>
        <td> 1 </td>
        <td> Program </td>
        <td style="color:blueviolet";> ISE </td>
    </tr>
    <tr>
        <td> 2 </td>
        <td> Name of the Candidate </td>
        <td  ><input type="text" name="name" id="nme"></td>
```

```
    </tr>
    <tr>
        <td> 3 </td>
        <td> USN  </td>
        <td> <input type="text" name="usn" id="usno"> </td>
    </tr>
    <tr>
        <td> 4 </td>
        <td> Mobile No  </td>
        <td> <input type="text" name="cno" id="conno"> </td>

    </tr>
     </table>
</body>
</html>
```

**4. Develop a JavaScript program that implements a "form" validation that displays an error message if a required field is left empty when**
**(i) While moving to next filed**
**(ii) Submitting the form {if it is the last field}**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="Cache-Control" content="no-cache, no-store, must-
revalidate" />
    <meta http-equiv="Pragma" content="no-cache" />
    <meta http-equiv="Expires" content="0" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Course Registration Form </title>
    <img
 src="/2.1+Heading+Element/1.1+Full_Stack_development/Heading_Sagar.jpg"
alt="header Dayananda Sagar">
    <h1  style="text-align:center;">   Even  Semester Course  Registration
Form </h1>
</head>
<style>
    table,th,td{
        border:1px solid black;
    }
</style>
<body>
    <table>
        <tr>
        <th> Sl No </th>
        <th> Description </th>
        <th> Course Details </th>
    </tr>
    <tr>
        <td> 1 </td>
        <td> Program </td>
        <td style="color:blueviolet";> ISE </td>
    </tr>
    <tr>
        <td> 2 </td>
        <td> Name of the Candidate </td>
```

```
        <td  ><input type="text" name="name" id="nme"></td>

    </tr>
    <tr>
        <td> 3 </td>
        <td> USN  </td>
        <td> <input type="text" name="usn" id="usno"> </td>
    </tr>
    <tr>
        <td> 4 </td>
        <td> Mobile No  </td>
        <td> <input type="text" name="cno" id="conno"> </td>

    </tr>
     </table>
     <br>
     <form  onsubmit="return dataVal()"  action="redirect.html" >
    <input type="submit" name= "onclick" value="Submit Your Data"  >

      <script>
        function dataVal()
        {
             a=document.getElementById("nme").value;
             b=document.getElementById("usno").value;
             c=document.getElementById("conno").value;

            if (a=="")
            {
               alert("Enter valid name");
               return false;

            }
            else if (isNaN(b))
            {
                var x=b.toLowerCase();
                var str=x.search(/^[0-1][a-z][a-z][0-9][0-9][a-z][a-z][0-
9][0-9][0-9]/);
                if(str!=0)
                alert(" Invalid CSN No, Enter valid CSN No,");
                return false;
            }
```

```
            else if(c.length<10||c.length>10)
            {

                alert("Enter valid Mobile No, MobNo lenght cannot be less
or greater than 10 digits");

                return false;
            }
            else
            {
               true;
            }
        }
        </script>
</form>
</body>
</html>
```

**5. Create a Git repository and execute Git commands or commits.**
**Publish a website in a git repository and access from different locations. Modify, Update, Delete contents of the website.**

1. Create an Git Hub account as student of Dayananda Sagar College of Engineering.

**6. Develop a JavaScript program to sort a list of elements using the Alpha Numerical sorting algorithm.**

```
<!DOCTYPE html>
<html>
<head>
<title>Alpha Numerical Sorting Algorithm</title>
</head>
<body>
<script>
// Create an array of elements to sort
var elements = ["Mango", "TutiFruity", "Bella", "1DS1020", "2300", "KadaleKai",
"HOD", "ootadabbi" "hod"];

// Define the Alpha Numerical sorting algorithm
function alphaNumericalSort(a, b) {
  // Convert the elements to strings
  var aString = a.toString();
  var bString = b.toString();

  // Compare the strings using the localeCompare() method
  var result = aString.localeCompare(bString);

  // Return the result
  return result;
}

// Sort the array using the Alpha Numerical sorting algorithm
elements.sort(alphaNumericalSort);

// Display the sorted array
console.log(elements);
</script>
</body>
</html>
```

**7. Design and develop an Online Voting website using HTML CSS, JavaScript and reactjs**

<APP.js>

```
import React,{Component} from 'react';
import './App.css';

class App extends Component{
	constructor(props){
		super(props);
		this.state = {
			languages : [
				{name: "Javascript", votes: 0},
				{name: "Python", votes: 0},
				{name: "C++", votes: 0},
				{name: "Java", votes: 0}
			]
		}
	}

	vote (i) {
		let newLanguages = [...this.state.languages];
		newLanguages[i].votes++;
		function swap(array, i, j) {
			var temp = array[i];
			array[i] = array[j];
			array[j] = temp;
		}
		this.setState({languages: newLanguages});

	}

	render(){
		return(
			<>
<h1>Vote Your Language!</h1>
<div className="languages">
                                        {
                                                this.state.languages.map((lang, i) =>
		<div key={i} className="language">
		<div className="voteCount">
                                                                {lang.votes}
```

```
                                                    </div>
<div className="languageName">
                                                    {lang.name}
                                                    </div>
                                                    <button
onClick={this.vote.bind(this, i)}>Click Here</button>
          </div>
          )}
          </div>
        </>
          );
          }
}
export default App;


*{
  margin: 0;
  padding: 0;
}

body {
  text-align: center;
  color: #222;
  font-size: 24px;
  font-family: sans-serif;
}

h1 {
  margin: 30px;
}

.languages {
  height: 400px;
  width: 400px;
  margin: 10px auto;
  display: flex;
  flex-direction: column;
}

.language {
  flex: 1;
  display: flex;
```

```
  justify-content: space-between;
  align-items: center;
  padding: 10px 40px;
  background-color: blanchedalmond;
  border: 1px solid #222;
  margin: 2px;
}

.voteCount {
  border-radius: 50%;
  display: flex;
  justify-content: center;
  align-items: center;
}

.language button {
  color: blueviolet;
  background-color: #0000;
  border: none;
  font-size: 30px;
  outline: none;
  cursor: pointer;
}
```

Result

**8. Design and develop a login web page with strong password validation using react js.**

```
function checkStrength(password) {
var strength = 0
if (password.length < 6) {
$('#strengthMessage').removeClass()
$('#strengthMessage').addClass('Short')
return 'Too short'
}
 if (password.length > 7) strength += 1
// If password contains both lower and uppercase characters,
increase strength value.
if (password.match(/([a-z].*[A-Z])|([A-Z].*[a
z])/)) strength += 1
// If it has numbers and characters, increase strength value
.
if (password.match(/([a-zA-Z])/) && password.match(/([0-
9])/)) strength += 1
// If it has one special character, increase strength value.
if (password.match(/([!,%,&,@,#,$,^,*,?,_,~])/)) strength +=
1
// If it has two special characters, increase strength value
.
if (password.match(/(.*[!,%,&,@,#,$,^,*,?,_,~].*[!,%,&,@,#,$
,^,*,?,_,~])/)) strength += 1
// Calculated strength value, we can return messages
// If value is less than 2
if (strength < 2) {
$('#strengthMessage').removeClass()
$('#strengthMessage').addClass('Weak')
return 'Weak'
} else if (strength == 2) {
$('#strengthMessage').removeClass()
$('#strengthMessage').addClass('Good')
return 'Good'
} else {
$('#strengthMessage').removeClass()
$('#strengthMessage').addClass('Strong')
return 'Strong'
}
}
});
.Short {
width: 100%;
background-color: #dc3545;
```

```
margin-top: 5px;
height: 3px;
color: #dc3545;
font-weight: 500;
font-size: 12px;
}
.Weak {
 width: 100%;
background-color: #ffc107;
margin-top: 5px;
height: 3px;
color: #ffc107;
font-weight: 500;
font-size: 12px;
}
.Good {
width: 100%;
background-color: #28a745;
margin-top: 5px;
height: 3px;
color: #28a745;
font-weight: 500;
font-size: 12px;
}
.Strong {
width: 100%;
background-color: #d39e00;
margin-top: 5px;
height: 3px;
color: #d39e00;
font-weight: 500;
font-size: 12px;
}
<body>
<form id="form1" runat="server">
<div class="container py-3">
<h4 class="text-center text
uppercase">How to check password strength in jquery</h4>
<div class="row">
<div class="col-md-12">
<div class="row">
<div class="col-md-6 mx-auto">
<div class="card border-secondary">
<div class="card-header">
<h3 class="mb-0 my-
```

```
2">Sign Up</h3>
</div>
<div class="card-body">
<div class="form-group">
<label>Name</label>
 <div class="input-group">
<div class="input-group
prepend">
<span class="input
group-text"><i class="fa fa-user"></i></span>
</div>
<asp:TextBox ID="txtFirst
Name" runat="server" CssClass="form-control"></asp:TextBox>
</div>
</div>
<div class="form-group">
<label>Phone Number</label>
<div class="input-group">
<div class="input-group
prepend">
<span class="input
group-text"><i class="fa fa-phone"></i></span>
</div>
<asp:TextBox ID="txtPhone
Number" runat="server" CssClass="form-control"></asp:TextBox>
</div>
</div>
<div class="form-group">
<label>Email</label>
<div class="input-group">
<div class="input-group
prepend">
<span class="input
group-text"><i class="fa fa-envelope"></i></span>
</div>
<asp:TextBox ID="txtEmail
" runat="server" CssClass="form-control"></asp:TextBox>
</div>
</div>
<div class="form-group">
<label>Password</label>
<div class="input-group">
<div class="input-group
prepend">
<span class="input
```

```
group-text"><i class="fa fa-lock"></i></span>
</div>
<asp:TextBox ID="txtPassw
ord" runat="server" TextMode="Password" CssClass="form
 control"></asp:TextBox>

</div>
<div id="strengthMessage"></d
iv>
</div>
<div class="form-group">
<label>Confirm Password</labe
l>
<div class="input-group">
<div class="input-group
prepend">
<span class="input
group-text"><i class="fa fa-lock"></i></span>
</div>
<asp:TextBox ID="txtConfi
rmPassword" runat="server" TextMode="Password" CssClass="form
control"></asp:TextBox>
</div>
</div>
<div class="form-group">
<button type="submit" class="
btn btn-success float-right rounded-0">Register</button>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</form>
 </body>
```

**9. Develop a node.js program to get files or directories of a directory in JSON format.**

```
const fs = require('fs');

const getFilesAndDirectories = (directory) => {
  const filesAndDirectories = [];

  fs.readdirSync(directory).forEach((fileOrDirectory) => {
    const filePath = `${directory}/${fileOrDirectory}`;
    const stats = fs.statSync(filePath);

    if (stats.isFile()) {
      filesAndDirectories.push({
        type: 'file',
        name: fileOrDirectory,
        path: filePath,
      });
    } else if (stats.isDirectory()) {
      filesAndDirectories.push({
        type: 'directory',
        name: fileOrDirectory,
        path: filePath,
      });

      filesAndDirectories.push(...getFilesAndDirectories(filePath));
    }
  });

  return filesAndDirectories;
};

const directoryPath = './my-directory';

const filesAndDirectories = getFilesAndDirectories(directoryPath);

console.log(JSON.stringify(filesAndDirectories, null, 2));
```

O/P:


```
  {
    "type": "file",
    "name": "file1.txt",
    "path": "./my-directory/file1.txt"
  },
  {
    "type": "directory",
    "name": "subdirectory1",
    "path": "./my-directory/subdirectory1"
  },
  {
    "type": "file",
    "name": "file2.txt",
    "path": "./my-directory/subdirectory1/file2.txt"
  },
  {
    "type": "directory",
    "name": "subdirectory2",
    "path": "./my-directory/subdirectory2"
  },
  {
    "type": "file",
    "name": "file3.txt",
    "path": "./my-directory/subdirectory2/file3.txt"
  }
]
```

**10. Design and Develop Screen Shot Generator Web application and Debug a website using REST API.**

To start I'm going to create a local node project and install the puppeteer package.

```
npm init
npm install puppeteer
```

```
const puppeteer = require('puppeteer');

    takeScreenshot()
        .then(() => {
            console.log("Screenshot taken");
        })
        .catch((err) => {
            console.log("Error occured!");
            console.dir(err);
        });

    async function takeScreenshot() {
        const browser = await puppeteer.launch();
        const page = await browser.newPage();
        await page.goto("https://medium.com", {waitUntil:
'networkidle2'});


        const buffer = await page.screenshot({
               path: './screenshot.png'
        });

        await page.close();
        await browser.close();
    }
```