Web Programming Laboratory SET - 1

1) Create an html page named as “cse.html” Add the following tags detail.
   (i) Set the title of the page as “Basic Html Tags”
   (ii)Within the body perform the following a) Moving text = “Basic HTML Tags” b) Different heading tags (h1 to h6) c) Paragraph d) Horizontal line e) Line Break f) div tag.

1.Basic html tags

Cse.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Basic Html Tags</title>
    </head>
    <body>
        <div>
            <marquee>Basic Html Tags</marquee>
            <h1>Heading 1</h1>
            <h2>Heading 2</h2>
            <h3>Heading 3</h3>
            <h4>Heading 4</h4>
            <h5>Heading 5</h5>
            <h6>Heading 6</h6>
            <hr>
            <br>
            <p>Paragraph</p>
        </div>
    </body>
</html>
```

Output :

2) Create a static web page for Cafeteria Menu using order, unordered and nested List tag.

2.Cafeteria menu

Menu.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Web page for Cafeteria Menu</title>
    </head>
    <body>
        <h1>Cafeteria Menu</h1>
        <p>Dessert</p>
        <ol>
            <li>Chocolate cake</li>
            <li>venilla cake</li>
            <li>Cheese cake</li>
        </ol>
        <p>Sandwich</p>
        <ul>
            <li>Egg & ham</li>
            <li>Hamburger</li>
            <li>cheese burger</li>
        </ul>
        <p>Drinks</p>
        <ul>
            <li> Mojitos drink
                <ul>
```

```
                    <li>Blue lagoon</li>
                    <li>Cocktail</li>
                    <li>very berry</li>
                </ul>
            </li>
            <li>cool drinks
                <ul>
                    <li>Thumps up</li>
                    <li>Pepsi</li>
                    <li>Sprite</li>
                </ul>
            </li>
        </ul>
    </body>
</html>
```

Output:

Cafeteria Menu

Dessert

1. Chocolate cake
2. venilla cake
3. Cheese cake

Sandwich

- Egg & ham
- Hamburger
- cheese burger

Drinks

- Mojitos drink
  - Blue lagoon
  - Cocktail
  - very berry
- cool drinks
  - Thumps up
  - Pepsi
  - Sprite

3) Create an html page named “image.html” to display the image, when the image is clicked, description about the image should be displayed.

3. Image description

Image.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Image description</title>
    </head>
    <body>
        <div class="image center">
            <a href="#" onClick="showStuff()">
                <img src="***path of the image file***" /></a>
       </div>
        <div id="hidden" style="display: none"><p>Some text about the
image eg. Captain america the first avenger</p> </div>

        <script type="text/javascript">
            function showStuff() {
                let hidden = document.getElementById('hidden');
                if (hidden.style.display == "none") {
                    hidden.style.display = "block"
                    } else {
                        hidden.style.display = "none"
                    }
            }
        </script>
    </body>
</html>
```

Output:

Some text about the image eg. Captain america the first avenger

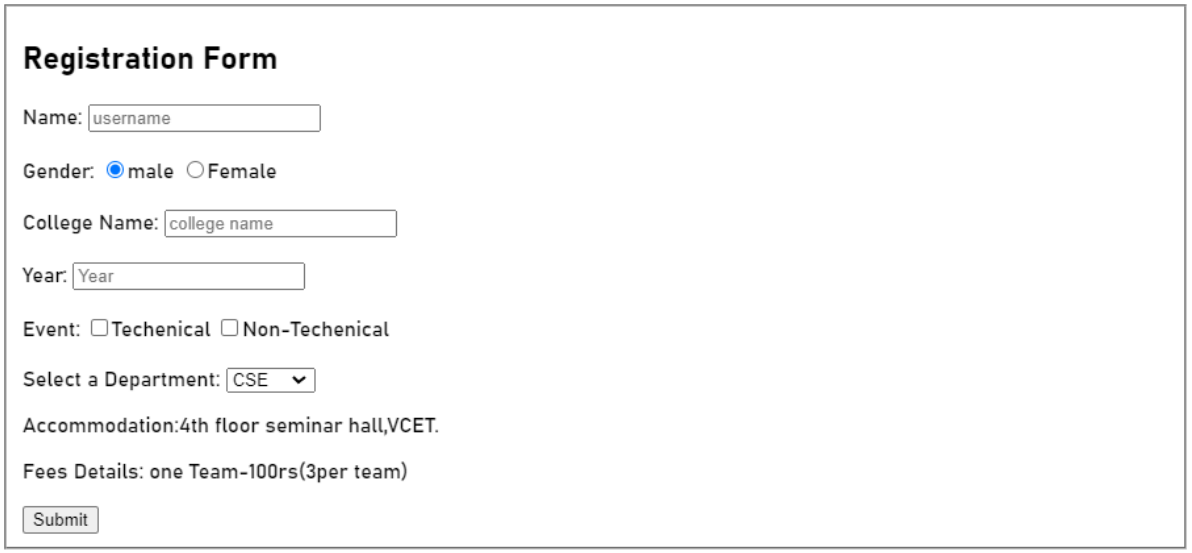
4) Create a Registration Form for Symposium event prescribing participant details as mentioned below: Name, College Name, Year, Department, Technical /Non-Technical event, Accommodation and fee details using TextBox, RadioButton, CheckBox and Dropdown menu.

4. Registration Form for Symposium

Form.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Registration Form</title>
</head>
<body>
    <fieldset>
    <h2>Registration Form</h2>
    <label>Name:</label>
    <input type="text" name="Name" placeholder="username" ><br><br>
    <label>Gender:</label>
    <input type="radio" name="gender">male
    <input type="radio" name="gender">Female <br><br>
    <label>College Name:</label>
    <input type="text" name="College Name" placeholder="college
name"><br><br>
    <label>Year:</label>
    <input type="number" name="Year" placeholder="Year"><br><br>
    <label>Event:</label>
    <input type="checkbox" name="event">Techenical
    <input type="checkbox" name="event">Non-Techenical <br><br>
    <label for="Department">Select a Department:</label>
    <select name="Department" id="Department">
    <option value="CSE">CSE</option>
    <option value="IT">IT</option>
    <option value="MECH">MECH</option>
    <option value="CIVIL">CIVIL</option></select><br>
    <p>Accommodation:4th floor seminar hall,VCET.</p>
    <p>Fees Details: one Team-100rs(3per team)</p>
    <button>Submit</button>
    </feildset>
</body>
</html>
```

Output:



5) Create an html page named as "TimeTable.html" to display your class time table. a) Provide the title as Time Table. b) Provide various color options to different subjects (Highlight the lab hours and elective hours with different colors.) c) Include spanning of rows and columns d) Include cell spacing and padding.

5.Timetable

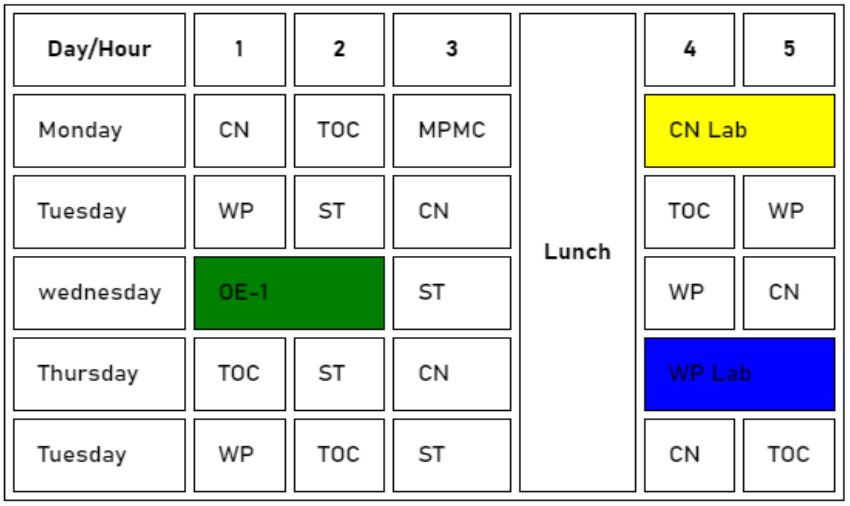
Timetable.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Time table</title>
        <style>
            table, th, td {
                border: 1px solid black;
                border-spacing: 5px;
               }
           th, td {
               padding: 15px;
               }
        </style>
    </head>
    <body>
        <table>
            <tr>
```

```
                <th>Day/Hour</th>
                <th>1</th>
                <th>2</th>
                <th>3</th>
                <th rowspan="6">Lunch</th>
                <th>4</th>
                <th>5</th>
            </tr>
            <tr>
                <td>Monday</td>
                <td>CN</td>
                <td>TOC</td>
                <td>MPMC</td>
                <td colspan="2" style="background-color:yellow">CN
Lab</td>
            </tr>
            <tr>
                <td>Tuesday</td>
                <td>WP</td>
                <td>ST</td>
                <td>CN</td>
                <td>TOC</td>
                <td>WP</td>
            </tr>
            <tr>
                <td>wednesday</td>
                <td colspan="2"
style="background-color:green">OE-1</td>
                <td>ST</td>
                <td>WP</td>
                <td>CN</td>
            </tr>
            <tr>
                <td>Thursday</td>
                <td>TOC</td>
                <td>ST</td>
                <td>CN</td>
                <td colspan="2" style="background-color:blue">WP
Lab</td>
            </tr>
            <tr>
                <td>Tuesday</td>
                <td>WP</td>
```

```
                <td>TOC</td>
                <td>ST</td>
                <td>CN</td>
                <td>TOC</td>
            </tr>
        </table>
    </body>
</html>
```

Output:



Day/Hour	1	2	3		4	5
Monday	CN	TOC	MPMC		CN Lab	
Tuesday	WP	ST	CN		TOC	WP
wednesday	OE-1		ST	Lunch	WP	CN
Thursday	TOC	ST	CN		WP Lab	
Tuesday	WP	TOC	ST		CN	TOC

6) Write an Angular JS framework that allows users to design a order form with a total price updated in real time.

6.order form

Orderform.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Order form</title>
        <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.
js"></script>
        <style>
        table {
            border-collapse: collapse;
            }
            th, td {
                padding: 10px;
```

```
                text-align: left;
                }
        </style>
    </head>
    <body>
        <div ng-app="myApp" ng-controller="myCtrl">
            <table border="1">
                <tr>
                    <th>Name</th>
                    <th>Quantity</th>
                    <th>Price Per Item</th>
                    <th>Total Price</th>
                    <th>Add Item</th>
                    <th>Remove Item</th>
                    </tr>
                    <tr ng-repeat="product in products">
                        <td>{{product.name}}</td>
                        <td>{{product.quantity}}</td>
                        </td>
                        <td>Rs.{{product.price}}</td>
                        <td>Rs.{{product.quantity*product.price}}</td>
                        <td><button class="btnbtn-primary"
ng-click="add(product)">ADD</button></td>
                        <td><button class="btnbtn-danger"
ng-click="remove(product)">REMOVE</button></td>
                        </tr>
            </table>
            <h2>Total Price: {{totalPrice}}</h2>
        </div>
        <script>
        var app = angular.module('myApp', []);
        app.controller('myCtrl', function ($scope) {
            $scope.products = [
            {
            name: "Miranda",
            quantity: 0,
            price: 100
            },
        {
            name: "Pepsi",
            quantity: 0,
            price: 75
            },
```

```
        {
            name: "Fanta",
            quantity: 0,
            price: 85
            },
        {
            name: "Fizz",
            quantity: 0,
            price: 80
            },
        ]
        $scope.totalPrice = 0
        $scope.add = (product) => {
            if(product.quantity >= 0){
                product.quantity++;
                $scope.totalPrice = $scope.totalPrice + product.price
                }
                }
            $scope.remove = (product) => {
                if(product.quantity > 0){
                    product.quantity--;
                    $scope.totalPrice = $scope.totalPrice -
product.price
                    }
                }
            });
        </script>
    </body>
</html>
```

Output:

Name	Quantity	Price Per Item	Total Price	Add Item	Remove Item
Miranda	2	Rs.100	Rs.200	ADD	REMOVE
Pepsi	1	Rs.75	Rs.75	ADD	REMOVE
Fanta	1	Rs.85	Rs.85	ADD	REMOVE
Fizz	3	Rs.80	Rs.240	ADD	REMOVE

Total Price: 600

7) Write an Angular JS framework that allows users to switching between different layout modes (grid or list) with a click of a button.

7.Layouts

Layouts.html

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>Layouts</title>
        <style type="text/css">
            .bar{
            background-color: #5c9bb7;
            background-image: -webkit-linear-gradient(top, #5c9bb7,
#5392ad);
            background-image: -moz-linear-gradient(top, #5c9bb7,
#5392ad);
            background-image: linear-gradient(top, #5c9bb7, #5392ad);
            box-shadow: 0 1px 1px #ccc;
            border-radius: 2px;
            width: 580px;
            padding: 10px;
            margin: 45px auto 25px;
            position: relative;
            text-align: right;
            line-height: 1;
            }
            .bar a{
            background: #4987a1 center center no-repeat;
            width: 32px;
            height: 32px;
            display: inline-block;
            text-decoration: none !important;
            margin-right: 5px;
            border-radius: 2px;
            cursor: pointer;
            }
            .bar a.active{
            background-color: #c14694;
            }
 /*------List layout------*/
            ul.list
```

```
            {
            list-style: none;
            width: 500px;
            margin: 0 auto;
            text-align: left;
            }
            ul.list li
            {
            padding: 10px;
            overflow: hidden;
            }
            ul.list li img
            {
            float: left;
            border: none;
            }
            ul.list li p
            {
            margin-left: 135px;
            font-weight: bold;
            color: black;
            }
 /*------Grid layout------*/
            ul.grid
            {
            list-style: none;
            width: 570px;
            margin: 0 auto;
            text-align: left;
            }
            ul.grid li
            {
            padding: 10px;
            float: left;
            color: black;
            font-weight: bold;
            }
        </style>
        <script type="text/javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.7.8/angular.mi
n.js"></script>
            <script type="text/javascript">
            var app = angular.module("MyApp", []);
```

```
            app.controller("MyController", function ($scope) {
            $scope.layout = 'grid';
            $scope.Data = [
            { Name: "Desert" },
            { Name: "Hydrangeas" },
            { Name: "Jellyfish" },
            { Name: "Koala" },
            { Name: "Lighthouse" },
            { Name: "Penguins" },
            { Name: "Tulips"}];
            });
        </script>
    </head>
    <body>
        <div ng-app="MyApp" ng-controller="MyController">
        <div class="bar">
            <a class="list-icon" ng-class="{active: layout == 'list'}"
ng-click="layout = 'list'"></a>
            <a class="grid-icon" ng-class="{active: layout == 'grid'}"
ng-click="layout = 'grid'"></a>
        </div>
        <ul ng-show="layout == 'grid'" class="grid">
        <li ng-repeat="data in Data"><p>{{data.Name}}</p></li>
        </ul>
        <ul ng-show="layout == 'list'" class="list">
        <li ng-repeat="data in Data">
        <p>{{data.Name}}</p>
        </li>
        </ul>
        </div>
    </body>
</html>
```

Output:

Desert

Hydrangeas

Jellyfish

Koala

Lighthouse

Penguins

Tulips

8) Create an external stylesheet using different subset of the style rules with the following specification
body{font-family:arial, Helvetica, Times new roman} li{font-weight:bold} h1{text-decoration:underline} ul{margin-left:20px}

8.External stylesheet

External.html

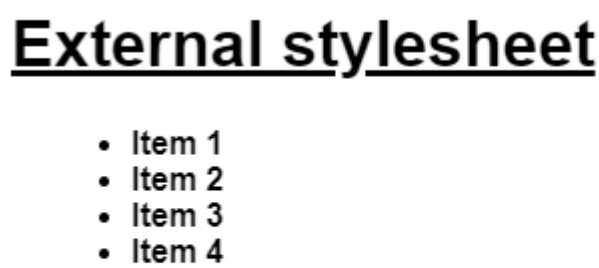
```
<!DOCTYPE html>
<html>
    <head>
        <title>External stylesheet</title>
        <link rel="stylesheet" href="style.css">
    </head>
    <body>
```

```
        <h1>External stylesheet</h1>
        <ul>
            <li>Item 1</li>
            <li>Item 2</li>
            <li>Item 3</li>
            <li>Item 4</li>
        </ul>
    </body>
</html>
```

Style.css

```
body{
      font-family: arial, Helvetica, Times new roman;
      }
li{
      font-weight: bold;
}
h1{
      text-decoration: underline;
}
ul{
      margin-left: 20px;
}
```

Output:



9) Design a given form that includes the Form elements as shown below

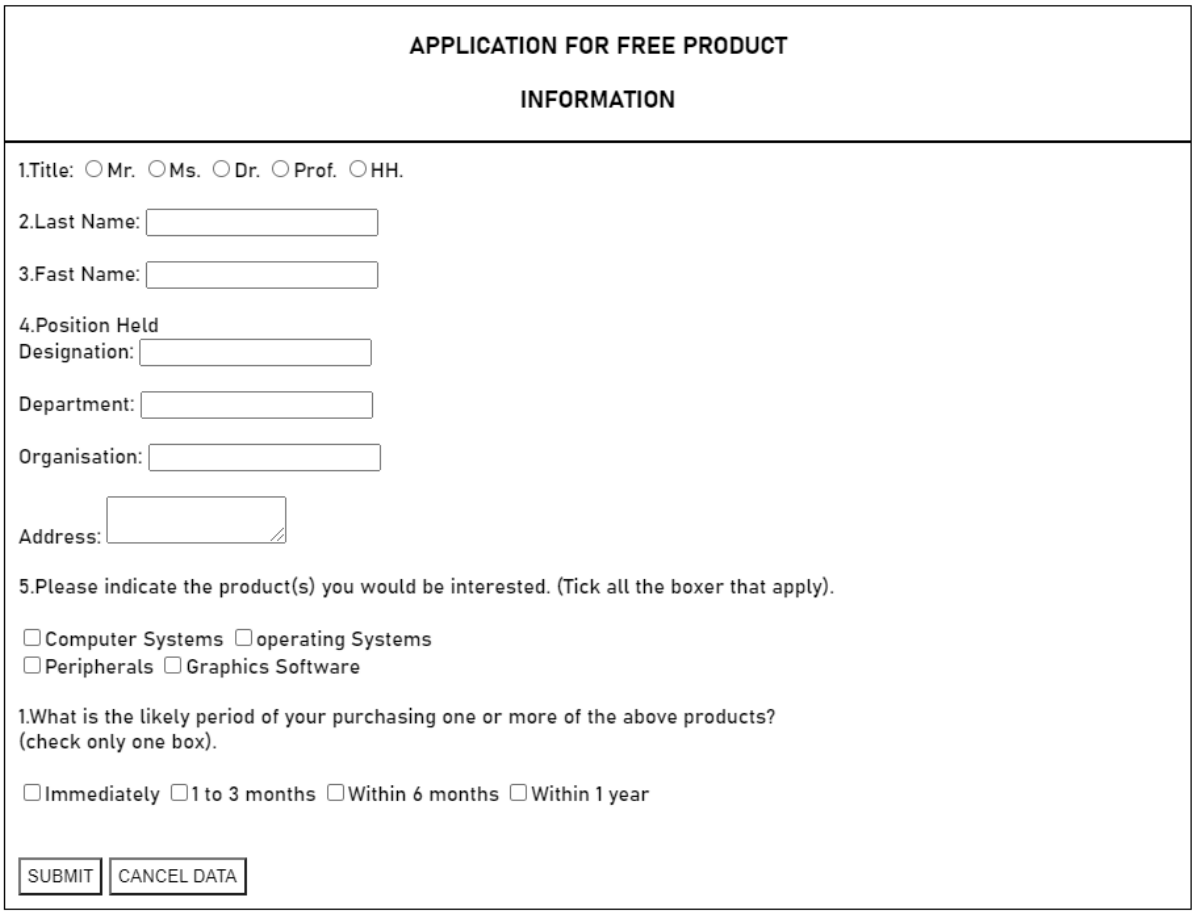
9. Design form

Form.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Form</title>
        <style>
            .title{
                border: 1px solid black;
                text-align: center;
            }
            .formbody{
                border: 1px solid black;
                padding: 10px;
            }
            button{
                padding: 5px;
                text-align: center;
                background-color: white;
            }
        </style>
    </head>
    <body>
        <div class="title">
            <h3>APPLICATION FOR FREE PRODUCT</h3>
            <h3>INFORMATION</h3>
        </div>
        <div class="formbody">
            <label>1.Title:</label>
            <input type="radio" name="title">Mr.
            <input type="radio" name="title">Ms.
            <input type="radio" name="title">Dr.
            <input type="radio" name="title">Prof.
            <input type="radio" name="title">HH. <br><br>
            <label>2.Last Name:</label>
            <input type="text" name="name"><br><br>
            <label>3.Fast Name:</label>
            <input type="text" name="name"><br><br>
```

```
            <label>4.Position Held </label><br>
            <label>Designation:</label>
            <input type="text" name="name"><br><br>
            <label>Department:</label>
            <input type="text" name="name"><br><br>
            <label>Organisation:</label>
            <input type="text" name="name"><br><br>
            <label>Address:</label>
            <textarea cols="15" rows="2"></textarea><br><br>
            <label>5.Please indicate the product(s) you would be
interested. (Tick all the boxer that apply).</label><br><br>
            <input type="checkbox" name="product" >Computer Systems
            <input type="checkbox" name="product">operating Systems
<br>
            <input type="checkbox" name="product">Peripherals
            <input type="checkbox" name="product">Graphics Software
<br><br>
            <label>1.What is the likely period of your purchasing one
or more of the above products?<br>(check only one box).</label><br><br>
            <input type="checkbox" name="period">Immediately
            <input type="checkbox" name="period">1 to 3 months
            <input type="checkbox" name="period">Within 6 months
            <input type="checkbox" name="period">Within 1 year <br>
            <br><br>
            <button>SUBMIT</button>
            <button>CANCEL DATA</button><br>
        </div>
    </body>
</html>
```

Output:



10) Create a web page with the following for the Amazon website. i) Cascading style sheets. ii) Embedded style sheets. iii) Inline style sheets.

10. Amazon website

Amazon.html

```
<!DOCTYPE html>

<html>

<head>
    <title>Amazon</title>
    <link rel="stylesheet" href="./amazon.css" >
    <style>
        * {
            margin: 0;
```

```
            padding: 0;
        }
        .products{
            position: relative;
            width: 100%;
            height: 400px;
            padding: 10px;
            display: flex;
            justify-content: space-evenly;
            align-items: center;
        }
        .items{
            background-color: rgb(206, 201, 201);
            padding: 10px;
            text-align: center;
        }
        .items img{
            height: 300px;
            width: 300px;
            object-fit: cover;
        }
    </style>
</head>

<body>
    <nav class="navbar background">
        <ul class="nav-list">
            <div class="logo">
                <img src=
"https://kfinancial.com/wp-content/uploads/2019/02/amazon-logo-vector-p
ng-vector-png-free-amazon-logos-705.jpg">
            </div>
            <li><a href="#">Today's deals</a></li>
            <li><a href="#">Customer service</a></li>
            <li><a href="#">My order</a></li>
        </ul>

        <div class="rightNav">
            <input type="text" name="search" id="search">
            <button style="padding: 6px 10px;vertical-align:
middle;color: black;">Search</button>
        </div>
    </nav>
```

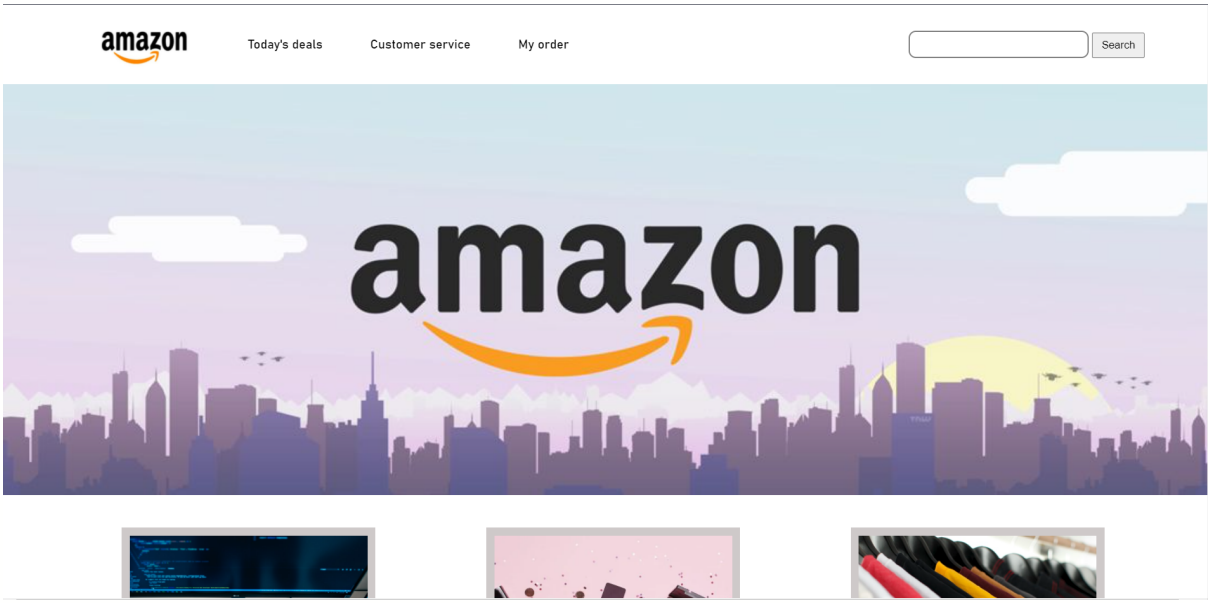
```
    <div>
        <img
src="https://images.freekaamaal.com/store_desc_images/1511854212.jpg"
width="100%">
    </div>
    <div class="products">
        <div class="items">
            <img
src="https://images.unsplash.com/photo-1577375729078-820d5283031c?ixlib
=rb-1.2.1&ixid=MnwxMjA3fDB8MHxzZWFyY2h8MXx8Y29tcHV0ZXIlMjBhY2Nlc3Nvcmll
c3xlbnwwfHwwfHw%3D&w=1000&q=80">
            <h3>Computers & Accessories</h3>
        </div>
        <div class="items">
            <img
src="https://img.ti-media.net/wp/uploads/sites/46/2021/06/GettyImages-1
249466095-1-920x518.jpg">
            <h3>Beauty</h3>
        </div>
        <div class="items">
            <img
src="https://images.unsplash.com/photo-1489987707025-afc232f7ea0f?ixlib
=rb-1.2.1&ixid=MnwxMjA3fDB8MHxzZWFyY2h8Mnx8Y2xvdGhzfGVufDB8fDB8fA%3D%3D
&w=1000&q=80">
            <h3>Clothes</h3>
        </div>
    </div>
</body>

</html>
```

Amazon.css

```
.navbar {
    display: flex;
    align-items: center;
    justify-content: center;
    position: sticky;
    top: 0;
    height: 100px;
    cursor: pointer;
}
```

```css
.background {
    background: white;
    background-blend-mode: darken;
    background-size: cover;
}

.nav-list {
    width: 70%;
    display: flex;
    align-items: center;
}

.logo {
    display: flex;
    justify-content: center;
    align-items: center;
}

.logo img {
    width: 200px;
}

.nav-list li {
    list-style: none;
    padding: 26px 30px;
}

.nav-list li a {
    text-decoration: none;
    color: black;
}
.rightnav {
    width: 30%;
    text-align: right;
}

#search {
    padding: 5px;
    font-size: 17px;
    border: 2px solid grey;
    border-radius: 9px;
}
```

Output:



11) Design a form using Ajax that contains a text field and a pushbutton. The user should enter a customer ID and press the button, and should see (in the same page) either: • A bulleted list of the id, first name, last name, and balance of the person with that ID. • An error message for an unknown/missing id

11. Ajax

Index.html

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

<head>
    <title>Document</title>
</head>

<body>
    <div class="container">
        <h2>User Details</h2>
```

```
        <input type="text" id="uname" placeholder="Enter User
Id"><br><br> <button id="btn"
            onclick="submit()">Submit</button>
    </div>
    <br>
    <div>
        <ul id="result"></ul>
    </div>
</body>
<script src="script.js"></script>
</html>
```

Data.xml

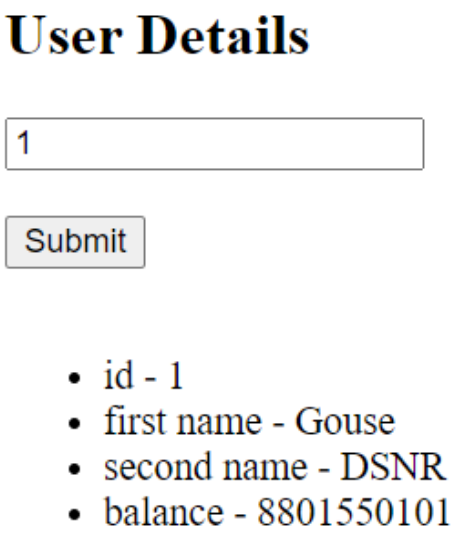
```
<?xml version="1.0" encoding="UTF-8"?>
<userlist>
    <user>
        <userid>1</userid>
        <fname>Gouse</fname>
        <sname>DSNR</sname>
        <balance>8801550101</balance>
    </user>
    <user>
        <userid>2</userid>
        <fname>D Divakar</fname>
        <sname>Ameerpet</sname>
        <balance>9888888888</balance>
    </user>
    <user>
        <userid>3</userid>
        <fname>Rajinth</fname>
        <sname>SR Nagar</sname>
        <balance>9866666666</balance>
    </user>
</userlist>
```

Script.js

```
function submit() {
    let flag = 0;
```

```
    const id = document.getElementById("uname").value;
    const xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function () {
        if (this.readyState == 4 && this.status == 200) {
            const data = xhttp.responseText;
            let parser = new DOMParser();
            let xml = parser.parseFromString(data, "application/xml");
            let userid = xml.getElementsByTagName("userid");
            let fname = xml.getElementsByTagName("fname");
            let sname = xml.getElementsByTagName("sname");
            let balance = xml.getElementsByTagName("balance");

            for (let i = 0; i < userid.length; i++) {
                if(userid[i].firstChild.nodeValue == id){
                    flag = 1;
                    const html = `<li>id -
${userid[i].firstChild.nodeValue}</li>
                    <li>first name -
${fname[i].firstChild.nodeValue}</li>
                    <li>second name -
${sname[i].firstChild.nodeValue}</li>
                    <li> balance -
${balance[i].firstChild.nodeValue}</li>`
                    document.getElementById("result").innerHTML = html
                }
            }
            if(flag === 0){
                document.getElementById("result").innerHTML = "Invalid
User";
            }
        };
    }
    xhttp.open("GET", "data.xml", true);
    xhttp.send();
```

Output:



12) Design a web page using CSS that includes • Different font styles. • Set background image for the web pages. • Define styles for links as link, visited, active, hover.

12.Css style

cssstyle.html

```
<!DOCTYPE html>
<html>
<head>
<title>Css style</title>
<style type="text/css">
body{
    font-family: arial;
    font-style: italic;
    background-image:url('***image path***');
    background-repeat: none;
    background-position:0 0px;
}
 a:link{
    text-decoration: none;
    font-size: 28px;
 }
```

```
 a:hover {
  color: yellow;
}
a:visited{
    color: white;
}
a:active{
    color: red;
}
</style>

</head>
<body>
<div>
    <a href="#1">Link 1</a> <br>
    <a href="#2">Link 2</a> <br>
    <a href="#3">Link 3</a> <br>
    <a href="#4">Link 4</a> <br>
</div>
</body>
</html>
```

Output:

13) Create an XML document, which contains 10 users' information. Design a webpage which takes User Id as input and returns the user details from the XML Document.

13.xml document (10 user)

Index.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">
    <link rel="stylesheet" href="style.css">
    <title>Document</title>
</head>
<body>
    <div class="container">
        <h2>User Details</h2>
        <input type="text" id="uname" placeholder="Enter User
Id"><br><br>
        <button id="btn" onclick="okie()">Submit</button>
    </div>
    <br>
    <div>
        <h3 id="result"></h3>
    </div>
</body>
<script src="script.js"></script>
</html>
```

style.css

```
.container {
    position: relative;
    top: 10%;
    left: 40%;
   }
   #btn {
```

```
 margin-left: 65px;
 padding: 10px;
 padding-left: 25px;
 padding-right: 25px;
 font-size: small;
 border: 1px solid black;
 background-color: black;
 color: white;
 transition: 0.2s;
}
#btn:hover {
 background-color: white;
 border: 1px solid black;
 color: black;
}
h2 {
 margin-left: 45px;
}
input {
 padding: 25px;
 padding-left: 35px;
 padding-right: 35px;
 border: 0px;
 box-shadow: 0px 0px 10px 8px rgb(238, 238, 238);
}
input:focus {
 background-color: white;
 outline: none;
}
#result {
 margin-left: 30%;
 margin-top: 80px;
}
```

script.js

```
 function okie() {
 let id = document.getElementById("uname").value;
 let flag = 0;
 let url = "data.xml";
 fetch(url)
 .then((response) => response.text())
 .then((data) => {
```

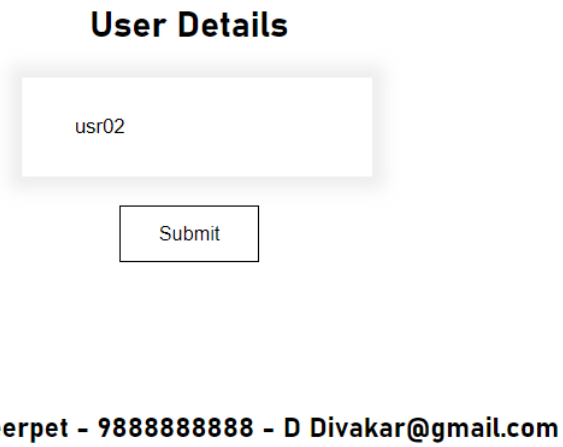
```
    let parser = new DOMParser();
    let xml = parser.parseFromString(data, "application/xml");
    let userid = xml.getElementsByTagName("userid");
    let username = xml.getElementsByTagName("username");
    let address = xml.getElementsByTagName("address");
    let phone = xml.getElementsByTagName("phone");
    let email = xml.getElementsByTagName("email");
    for (let i = 0; i < userid.length; i++) {
    if (id == userid[i].firstChild.nodeValue) {
    let user =
    username[i].firstChild.nodeValue +
    " - " +
    address[i].firstChild.nodeValue +
    " - " +
    phone[i].firstChild.nodeValue +
    " - " +
    email[i].firstChild.nodeValue;
    document.getElementById("result").innerHTML = user;
    flag = 1;
    }
    }
    if (flag === 0) {
    document.getElementById("result").innerHTML = "Invalid User";
    }
    });
}
okie()
```

data.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<userlist>
 <user>
    <userid>usr01</userid>
    <username>Gouse</username>
    <address>DSNR</address>
    <phone>8801550101</phone>
    <email>Gouse.sheikh@gmail.com</email>
 </user>
 <user>
    <userid>usr02</userid>
    <username>D Divakar</username>
    <address>Ameerpet</address>
```

```
    <phone>9888888888</phone>
    <email>D Divakar@gmail.com</email>
 </user>
 <user>
    <userid>usr03</userid>
    <username>Rajinth</username>
    <address>SR Nagar</address>
    <phone>9866666666</phone>
    <email>Rajinth@yahoo.com</email>
 </user>
</userlist>
```

Output:



14) Create a registration form with the following fields • Name (Text field) • Password (Password field) • Email id (Text field) • Phone Number (Text field) • Date of birth (3 select boxes) • Languages known (checkboxes – English, Tamil, Hindi, Telugu) • Address (Text area) Write JavaScript to validate the following fields • Name should contain alphabets and the length should not be less than 15 characters. • Passwords should not be less than 8 characters. • Email-Id should not contain any invalid characters and must follow the standard pattern (name@domain.com) • Phone number should contain 10 digits only.

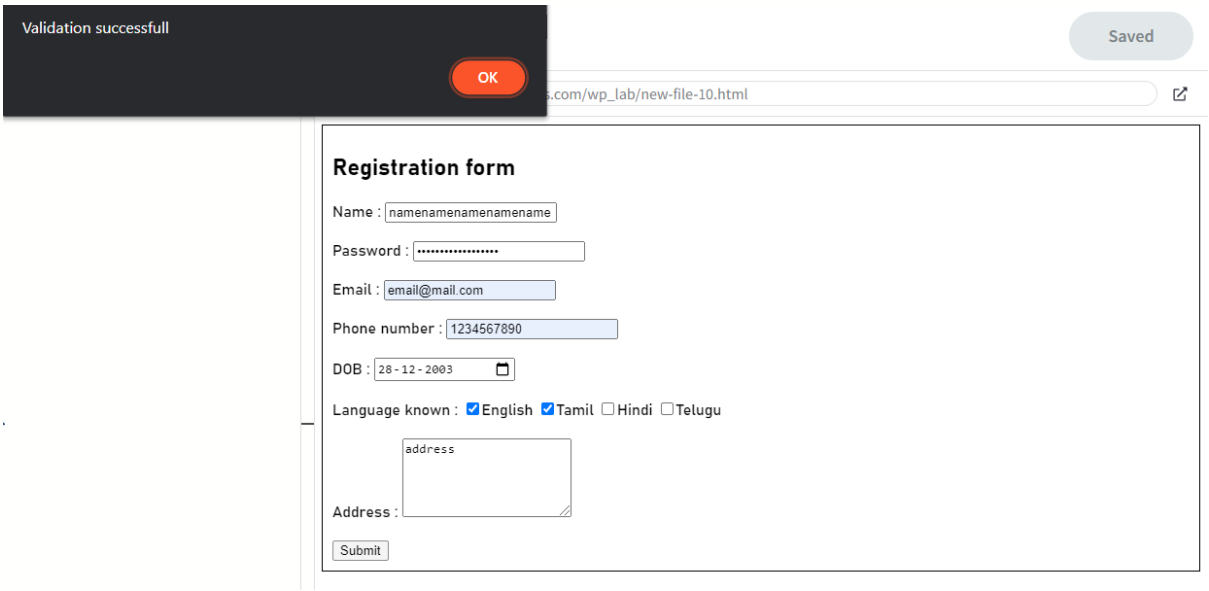
14. Form validation using js

Form.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Form validation</title>
    </head>
    <body>
        <form name="regForm" style="border: 1px solid black; padding:
10px">
            <h2>Registration form</h2>
            <label>Name :</label>
            <input type="text" id="name" required><br><br>
            <label>Password :</label>
            <input type="password" id="password" required><br><br>
            <label>Email :</label>
            <input type="email" id="email" required><br><br>
            <label>Phone number :</label>
            <input type="text" id="phone" required><br><br>
            <label>DOB :</label>
            <input type="date" id="dob"><br><br>
            <label>Language known :</label>
            <input type="checkbox" id="language">English
            <input type="checkbox" id="language">Tamil
            <input type="checkbox" id="language">Hindi
            <input type="checkbox" id="language">Telugu<br><br>
            <label>Address :</label>
            <textarea rows="5" id="address"></textarea><br><br>
            <button id="submit" onclick="validForm()">Submit</button>
        </form>

        <script>
            function validForm(){

                var name = document.regForm.name.value;
                var password = document.regForm.password.value;
                var email = document.regForm.email.value;
                var number = document.regForm.phone.value;

                if(!name.match(/^[a-zA-Z\s]+$/)||name.length<15){
                    alert('Please enter a valid name');
                }else if(password.length < 8){
                    alert('Please enter a valid password');
                }else if(!email.match(/^\S+@\S+\.\S+$/)){
                    alert('Please enter a valid email');
```

```
                }else if(!number.match(/^[1-9]\d{9}$/)){
                    alert('Please enter a valid phone number');
                }else{
                    alert('Validation successfull');
                }
            }
        </script>
    </body>
</html>
```

Output:



15) Write an HTML page including required JavaScript that takes a number from one text field in the range of 0 to 999 and shows it in a another text field in words. If the number is out of range, it should show “out of range” and if it is not a number, it should show “not a number” message in the result box.

15.Number range

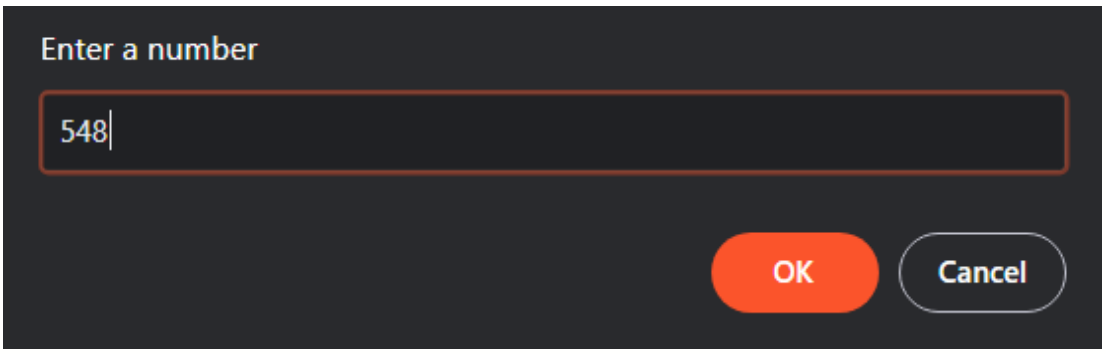
Numberrange.html

```
<!DOCTYPE html>
```

```
<html>
    <head>
        <title>Number range</title>
    </head>
    <body style="background-color: lightblue; color:dongerblue">
        <script language="javascript">
            function numtowords(number)
            {
            hyphen = '-';
            conjunction = ' and ';
            separator = ', ';
            dictionary = {
            0 : 'zero',
            1 : 'one',
            2 : 'two',
            3 : 'three',
            4 : 'four',
            5 : 'five',
            6 : 'six',
            7 : 'seven',
            8 : 'eight',
            9 : 'nine',
            10 : 'ten',
            11 : 'eleven',
            12 : 'twelve',
            13 : 'thirteen',
            14 : 'fourteen',
            15 : 'fifteen',
            16 : 'sixteen',
            17 : 'seventeen',
            18 : 'eighteen',
            19 : 'nineteen',
            20 : 'twenty',
            30 : 'thirty',
            40 : 'fourty',
            50 : 'fifty',
            60 : 'sixty',
            70 : 'seventy',
            80 : 'eighty',
            90 : 'ninety',
            100 : 'hundred',
            };
            if (number < 0 || number > 999)
```

```
            {
            alert("Enter a number range between 0 and 999");
            return "";
            }
            switch (true)
            {
            case (number < 21):
            string = dictionary[number];
            break;
            case (number < 100):
            tens = parseInt(number / 10) * 10;
            units = number % 10;
            string = dictionary[tens];
            if (units)
            {
            string += hyphen + dictionary[units];
            }
            break;
            case (number < 1000):
            hundreds = parseInt(number / 100);
            remainder = number % 100;
            string = dictionary[hundreds] + ' ' + dictionary[100];
            if (remainder)
            {
            string += conjunction + numtowords(remainder);
            }
            break;
            default:
            break;
            }
            return string;
            }
            a=prompt("Enter a number");
            num=parseInt(a);
            document.write(numtowords(num));
        </script>
    </body>
</html>
```

Output:



five hundred and fourty-eight