1.Write HTML to print basic tags.

```
<html>
    <head title="Basic HTML Tags"></head>
    <body>
       <p align="center">Basic HTML Tags</p>
       <h3 align="center"><font color="red">Heading Tags => font size-18pt, colour-RED </font></h3>
       <h1 align="center">Heading H1 - 24pt</h1>
       <h2 align="right">Heading H2 - 18 pt</h2>
       <h3><b>Heading H3 - 12 pt bold</b></h3>
       <h4>Heading H4-12pt</h4>
       <h5>Heading H5 - 10pt</h5>
       <h6>Heading H6 - 7pt</h6>

       <p style="font-size:36pt;">font size-36 pt</p>
       <p style="font-size:24pt;">font size-24 pt</p>
       <p style="font-size:18pt;">font size-18 pt</p>
       <p style="font-size:12pt;"><b>font size-12 pt bold</b></p>
       <p style="font-size:12pt;">font size-12 pt plain</p>
       <p style="font-size:9pt;">font size-9 pt</p>
       <hr>
       <p align="center" style="color:green; font-size:18pt;">Text Elememnts => font size -12pt, colour-green</p>
       <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries.</p>
<h1>Unordered Lists</h1>
       <ul>
           <li>Mango</li>
           <li>Apple</li>
           <li>Banana</li>
       </ul>
```

```
    <h1>Ordered List</h1>
    <ol>
        <li>Mango</li>
        <li>Apple</li>
        <li>Banana</li>
    </ol>
</body></html>
```

OUTPUT:

Basic HTML Tags

Heading Tags => font size-18pt, colour-RED

Heading H1 - 24pt

Heading H2 - 18 pt

Heading H3 - 12 pt bold

Heading H4-12pt

Heading H5 - 10pt

Heading H6 - 7pt

font size-36 pt

font size-24 pt

font size-12 pt plain

font size-9 pt

Text Elememnts => font size -12pt, colour-green

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Unordered Lists

- Mango
- Apple
- Banana

Ordered List

1. Mango
2. Apple
3. Banana

2) Generate class time table using HTML.

```
<!DOCTYPE html>
<html lang="en">
<head title="TIME TABLE SECTION D 6TH SEMESTER">
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initialscale=1.0">
</head>
<body>
    <table border="1" style="border-collapse:collapse" >
        <tr>
            <th></th>
            <th>8:05-9:00</th>
            <th>9:00-9:55</th>
            <th>9:55-10:50</th>
            <th>11:10-12:05</th>
            <th>12:05-1:00</th>
            <th>1:00-1:55</th>
            <th>1:55-2:50</th>
            <th>3:10-4:05</th>
            <th>4:05-5:00</th>
<th>5:00-5:55</th>
        </tr>
        <tr>
            <th>Mon</th>
            <td colspan="4" style="text-align:center">Placement
Preparation classes</td>
            <td>LUNCH</td>
            <td style="background-color:teal;text-align:center">TCS 604</td>
            <td style="background-color:magenta;text-align:center">TCS
693</td>
            <td style="background-color:red;textalign:center">elective</td>
```

```
            <td colspan="2">Compiler Design Lab</td>
        </tr>
        <tr>
            <th>Tues</th>
            <td colspan="2">SWAYAM/Lib</td>
            <td style="background-color:magenta;text-align:center">TCS
693</td>
            <td style="background-color:teal;text-align:center">TCS 604</td>
            <td>LUNCH</td>
            <td style="background-color:grey;text-align:center">XCS 601(soft
skills)</td>
            <td style="background-color:grey;text-align:center">XCS
601(QAR)</td>
            <td style="background-color:red;textalign:center">elective</td>
            <td colspan="2">Software Engineering Lab</td>
        </tr>
        <tr>
            <th>Wed</th>
            <td colspan="4" style="text-align:center">SWAYAM/Lib</td>
            <td>LUNCH</td>
            <td style="background-color:green;text-align:center">TCS
611(Discussion)</td>
            <td style="background-color:magenta;text-align:center">TCS
693</td>
            <td style="background-color:red;textalign:center">elective</td>
            <td style="background-color:grey;text-align:center">XCS
601(verbal)</td>
            <td style="background-color:yellow;text-align:center;">TCS
691</td>
        </tr>
        <tr>
            <th>Thurs</th>
            <td colspan="4" style="text-align:center">Placement
Preparation classes</td>
            <td>LUNCH</td>
```

```
            <td style="background-color:teal;text-align:center">TCS 604</td>
            <td style="background-color:magenta;text-align:center">TCS
693</td>
            <td style="background-color:red;textalign:center">elective</td>
            <td COLSPAN="2" style="background-
color:yellow;textalign:center;">TCS 691</td>
        </tr>
        <tr>
            <th>Fri</th>
            <td></td>
            <td style="background-color:magenta;text-align:center">TCS
693</td>
            <td colspan="2">Web Development Lab</td>
            <td>LUNCH</td>
            <td style="background-color:magenta;text-align:center">TCS
693</td>
            <td style="background-color:teal;text-align:center">TCS 604</td>
            <td style="background-color:red;textalign:center">elective</td>
            <td COLSPAN="2" style="background-
color:yellow;textalign:center;">TCS 691</td>
        </tr>
        <tr>
            <th>Sat</th>
            <td colspan="10" style="text-align:center"></td>

        </tr>
    </table>
<br>
<table border="1" cellspacing="0">
        <tr>
            <th><b> Subject Code</b></th>
            <th><b> Subject Name</b></th>
            <th><b> Faculty</b></th>
        </tr>
```

```
        <tr>
            <td><b> TCS-601</b></td>
            <td> Compiler Design </td>
            <td> Mr. Ashwini Kumar</td>
        </tr>
        <tr>
            <td><b> TCS-611</b></td>
            <td> Software Engineering</td>
            <td> Mr. Prabhdeep Singh</td>
        </tr>
        <tr>
            <td><b> TCS-604</b></td>
            <td> Computer Networks-I</td>
            <td> Mr. Sarvesh Vishwakarma</td>
        </tr>
        <tr>
            <td><b> TCS-693</b></td>
            <td> Full Stack Web Development</td>
            <td> Dr. Parul Madan</td>
        </tr>
        <tr>
            <td><b> XCS-601</b></td>
            <td> Career Skills (QAR/PDP/ Soft Skills -PDP)</td>
            <td> Mr. Saurabh Rawat/Mr. Okesh Chhabra/Dr. Gopal
Krishna</td>
        </tr>
        <tr>

</body>
</html>
```

OUTPUT:

	8:05-9:00	9:00-9:55	9:55-10:50	11:10-12:05	12:05-1:00	1:00-1:55	1:55-2:50	3:10-4:05	4:05-5:00	5:00-5:55
Mon	Placement Preparation classes				LUNCH	TCS 604	TCS 693	elective	Compiler Design Lab	
Tues	SWAYAM/Lib		TCS 693	TCS 604	LUNCH	XCS 601(soft skills)	XCS 601(QAR)	elective	Software Engineering Lab	
Wed	SWAYAM/Lib				LUNCH	TCS 611(Discussion)	TCS 693	elective	XCS 601(verbal)	TCS 691
Thurs	Placement Preparation classes				LUNCH	TCS 604	TCS 693	elective	TCS 691	
Fri		TCS 693	Web Development Lab		LUNCH	TCS 693	TCS 604	elective	TCS 691	
Sat										

Subject Code	Subject Name	Faculty
TCS-601	Compiler Design	Mr. Ashwini Kumar
TCS-611	Software Engineering	Mr. Prabhdeep Singh
TCS-604	Computer Networks-I	Mr. Sarvesh Vishwakarma
TCS-693	Full Stack Web Development	Dr. Parul Madan
XCS-601	Career Skills (QAR/PDP/ Soft Skills -PDP)	Mr. Saurabh Rawat/Mr. Okesh Chhabra/Dr. Gopal Krishna

3)Design a registration form using 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, initialscale=1.0">
    <title>Form Handling</title>
</head>
<body>
    <form>
        <table border style="border:10px solid black;">
            <tr>
                <td>FIRST NAME: <input type="text" name="f_name"></td>
            </tr>
            <tr>
                <td>LAST NAME: <input type="text" name="l_name"></td>
            </tr>
            <tr>
                <td>USER ID: <input type="text" name="user_id"></td>
            </tr>
            <tr><td>PASSWORD: <input type="password" name="pwrd"></td>
            </tr>
            <tr>
                <td>DESCRIPTION:<br>
                <textarea>

                </textarea>
                </td>
            </tr>
            <tr>
```

```
                <td>
                    <input type="checkbox">maths
                    <input type="checkbox">physics
                </td>
            </tr>
            <tr>
                <td>
                    <input type="radio" name="m">maths
                    <input type="radio" name="m">physics
                </td>
            </tr>
            <tr>
                <td>
                <select>
                    <option>Maths</option>
                    <option>Physics</option>
                </select>
                </td>
            </tr>
            <tr>
                <td>
                    EMAIL: <input type="email" placeholder="abc@gmail.com">
                </td>
            </tr>
            <tr>
                <td>
                    <input type="submit" name="subm">
<input type="reset" name="reset">
                    <input  type="button" value="ok">
                    <input type="image">
                </td>
            </tr>
```

```
        </table>
    </form>
</body>
</html>
```

FIRST NAME:

LAST NAME:

USER ID:

PASSWORD:

DESCRIPTION:

maths physics

maths physics

Maths

EMAIL: abc@gmail.com

Submit Reset ok Submit

4) To create an html page with different types of frames such as floating frame, navigation frame & mixed frame. frames.html

```
<!DOCTYPE html>
<html>
<head>
<title>frames1</title>
</head>
<frameset cols="20%,80%">
<frame src="left.html" name="left"></frame>
<frame src="right.html" name="right"></frame>
</frameset>
</html>
```

left.html

```
<!DOCTYPE html>
<html>
<body>
<a href="navigationframe.html" target="right"><h3>navigationframe</h3></a>
<br>
<a href="floatingframe.html" target="right"><h3>floatingframe</h3></a>
<br>
<a href="mixedframe.html" target="right"><h3>mixedframe</h3></a>
<br>
<a href="noframe.html" target="right"><h3>no frame</h3></a>
</body>
</html>
```

navigationFrame.html

```
<!DOCTYPE html>
<html>
<body>
<p><a href="Uttarakhand-MAP.jpg">Uttrakhand map</a></p>
<br>
<p><a href="pic.png">Basic html tag</a></p>
<br>
```

List

</body> </html>

floatingFrame.htm

l

<HTML>

<HEAD>

<TITLE>IFRAME</TITLE></HEAD>

<BODY >

<H2>Inline (Floating) Frames</H2>

<IFRAME SRC="list.html" WIDTH=200 HEIGHT=100% ALIGN=left HSPACE=12></IFRAME>

A floating frame is used to embed another document or page within an existing frame. This frame will be an inline framed region that acts like other embedded objects— meaning

that text can flow around it.

</BODY> </HTML>

mixedFrame.html

<!DOCTYPE html>

<html>

<head>

<title>frames1</title>

</head>

<frameset cols="25%,75%">

<frame src="pic.PNG" name="mixedLeft"/>

<frameset rows="50%,50%">

<frame src="timetable.html" height="100%" width="100%" name="1a"/>

<frame src="video.mp4" type="video/mp4"name="1b"/>

</frameset>

</frameset>

navigationframe

floatingframe

mixedframe

no frame

navigationframe

floatingframe

mixedframe

no frame

Uttrakhand map

Basic html tag

List

navigationframe

floatingframe

mixedframe

no frame

Inline (Floating) Frames

A floating frame is used to embed another document or page within an existing frame. This frame will be an inline framed region that acts like other embedded objects—meaning that text can flow around it.

Here is a nested ordered list:

I. Module 1
II. Module 2
    i. Module 2.1
    ii. Module 2.2
III. Module 3

Disc bullets list

- Keyboard
- Mouse
- Monitor

circle

5) Write a program in HTML to make a clickable image using <map> tag.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <img src="Web capture_28-2-2023_16132_.jpeg" usemap="#numbermap"
height="400px" width="400px" onmousemove="coords(event)">
    <div id="coordinates"></div>
    <map name="numbermap">
        <area shape="circle" coords="96,70,45" onclick="{window.alert('the
number is 1');}">
        <area shape="circle" coords="208,72,45" onclick="{window.alert('the
number is 2');}">
        <area shape="circle" coords="325,66,45" onclick="{window.alert('the
number is 3');}">
        <area shape="circle" coords="96,162,45" onclick="{window.alert('the
number is 4');}">
        <area shape="circle" coords="208,161,45" onclick="{window.alert('the
number is 5');}">
        <area shape="circle" coords="324,163,45" onclick="{window.alert('the
number is 6');}">

        <area shape="circle" coords="94,255,45" onclick="{window.alert('the
number is 7');}">
        <area shape="circle" coords="210,250,45" onclick="{window.alert('the
number is 8');}">
        <area shape="circle" coords="325,255,45" onclick="{window.alert('the
number is 9'    );}">
        <area shape="circle" coords="95,345,45" onclick="{window.alert('the
number is *');}">
```

```
        <area shape="circle" coords="209,347,45" onclick="{window.alert('the number is 0');}">
        <area shape="circle" coords="324,346,45" onclick="{window.alert('the number is #');}">
 </map>     <script>
function coords(t)
        {              var x=t.clientX;              var y=t.clientY;
document.getElementById("coordinates").innerHTML=x+","+y;
        }
    </script>
</body>
</html>
```

OUTPUT:

53,119

6) To create an html file by applying the different styles using inline, external & internal style sheets.

6a.

HTML :

```
<!DOCTYPE html>
<html>
<head>
<title>css ques 1 a.</title>
<link rel="stylesheet" href="stylee.css">

</head>
<body>
<div id="demobox">
<b>This Is My Stylish Website</b>
</div>
<div class="topnav">
<a href="#home">Home</a>
<a href="#background">Background</a>
<a href="#lists">Lists</a>
<a href="#id vs class">Id vs Class</a>
</div>
<div id="descriptionbox">
<b>This website is about me!</b>
</div>
<div id="declaration">
My top 3 favourite this to do
</div>
<div id="favouritebox">
<ol>
<li>Travel</li><br>
<li>Eat ice cream</li><br>
<li>Read a book</li>
</ol>
```

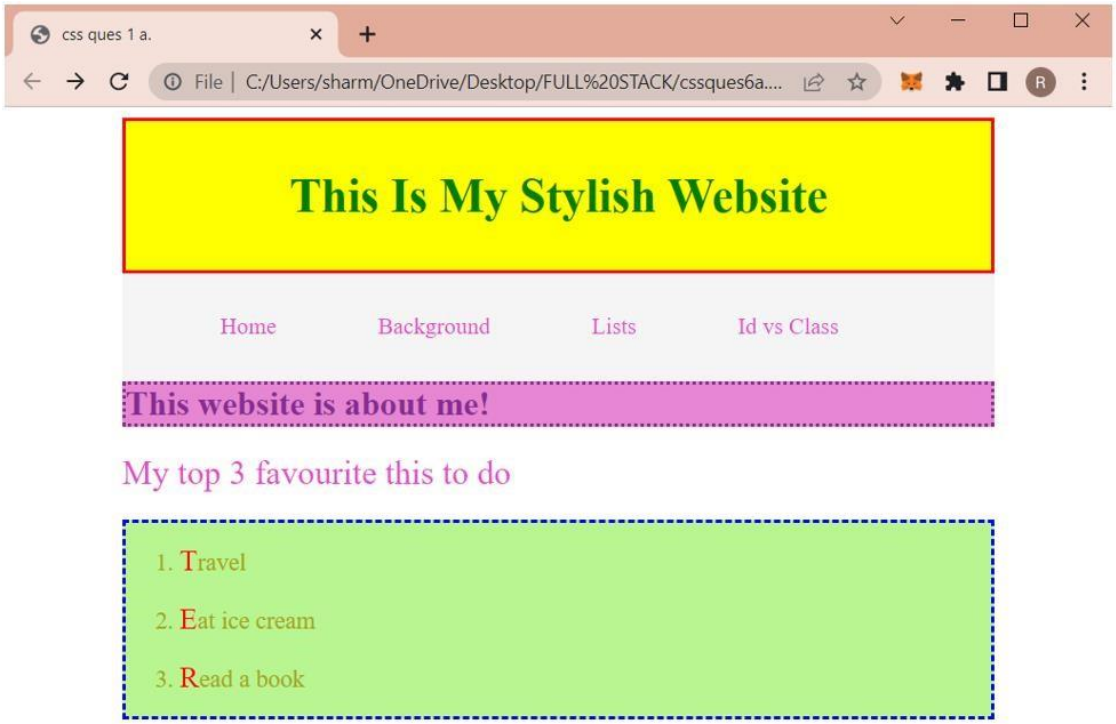
```
</div>
</body>
</html>
```

CSS:

```
#demobox{
background-color: #FFFF00; color :  #008000; padding: 35px; text-align: center;
margin-right:10%; margin-left:10%; border: red; border-style: solid; padding-
right: 30px; font-size: 35px; padding-left: 30px;
}
.topnav { background-color: whitesmoke;
overflow: hidden; margin-right:10%; margin-
left:10%; padding:10px;
}
.topnav a { float: left; color: rgb(233,
73, 207); text-align: left; padding-left:
10%; padding-right:2%; padding-
top:20px; padding-bottom: 20px; text-
decoration: none; font-size: 17px;

}
#descriptionbox { background-color:
#e786d2; color :  #872c95;

text-align: left; margin-right:10%; margin-left:10%; border:#872c95; border-style: dotted; padding-right: 30px;
fontsize: 25px;
}
#declaration
{
background-color:white; color: rgb(233, 73, 207); text-align: left; margin-right:10%; margin-
left:10%; padding-top:20px; padding-bottom:20px; font-size: 25px;
}
#favouritebox{ border:dashed; border-color:blue; margin-left: 10%; margin-right:10%; background-
color: #b9f691;
```

```
} li{
text-transform:lowercase;
color:#a6a60e;  font-
size:large;
}

li::first-letter
{
text-transform: uppercase; font-size: larger; color:red;
}
```

OUTPUT :



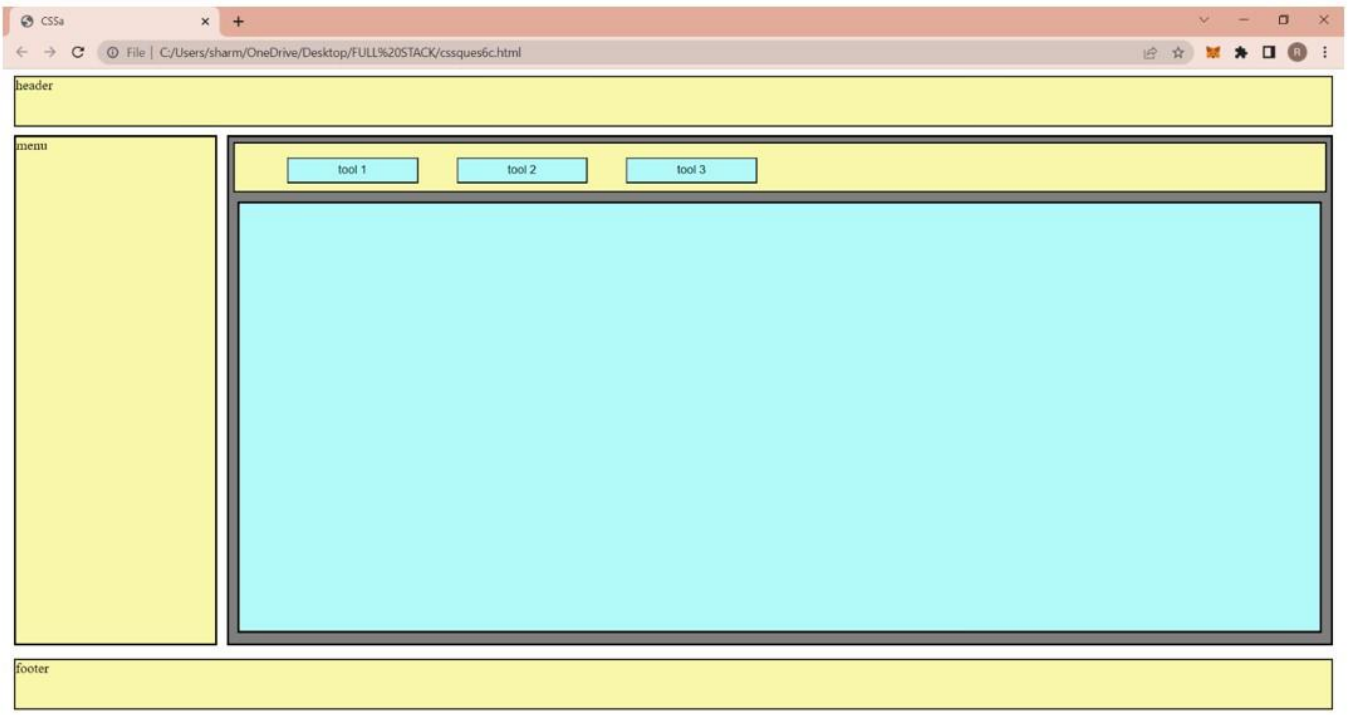
6B.

```
<!DOCTYPE html><html>
<head>
<title>CSSa</title>
<style>
.base
{
background-color: rgb(248, 247, 170); border: solid black ; border-
width: 2px;

}
.height
{
height:55px; margin:
5px;
}
.height2
{
height: 600px;
}
#right{ float:right; width: 83%; margin: 5px; margin-left:
0px; background-color: rgb(126, 126, 126); height: 580px;
border:solid black;
}
#left{margin: 5px; width: 15%; float:
left; border: solid black; height: 580px;
background-color: rgb(248, 247, 170);
}
li{
display: inline; margin-left: 20px; margin-right: 20px;

}
```

```
#content
{
background-color: rgb(177, 250, 250); margin: 10px;
height: 490px; border: solid ;
 }
button{ background-color: rgb(177, 250, 250);
width: 150px; height: 30px; </style>
</head>
<body >

<div class="base height" > header
</div>

<div class="height2">
<div id="right">
<div class="base height">
<ul type="none" >
<li><button type="button">tool 1</button></li>
<li><button type="button">tool 2</button></li>
<li><button type="button">tool 3</button></li>
</ul>
</div>
<div id="content">

</div>
</div>
< div id="left"> menu</div>

</div>
<div class="base height"> footer
</div>
</body>
```

</html>

OUTPUT:


CSSa
File | C:/Users/sharm/OneDrive/Desktop/FULL%20STACK/cssques6c.html
beader
menu
tool 1
tool 2
tool 3
footer

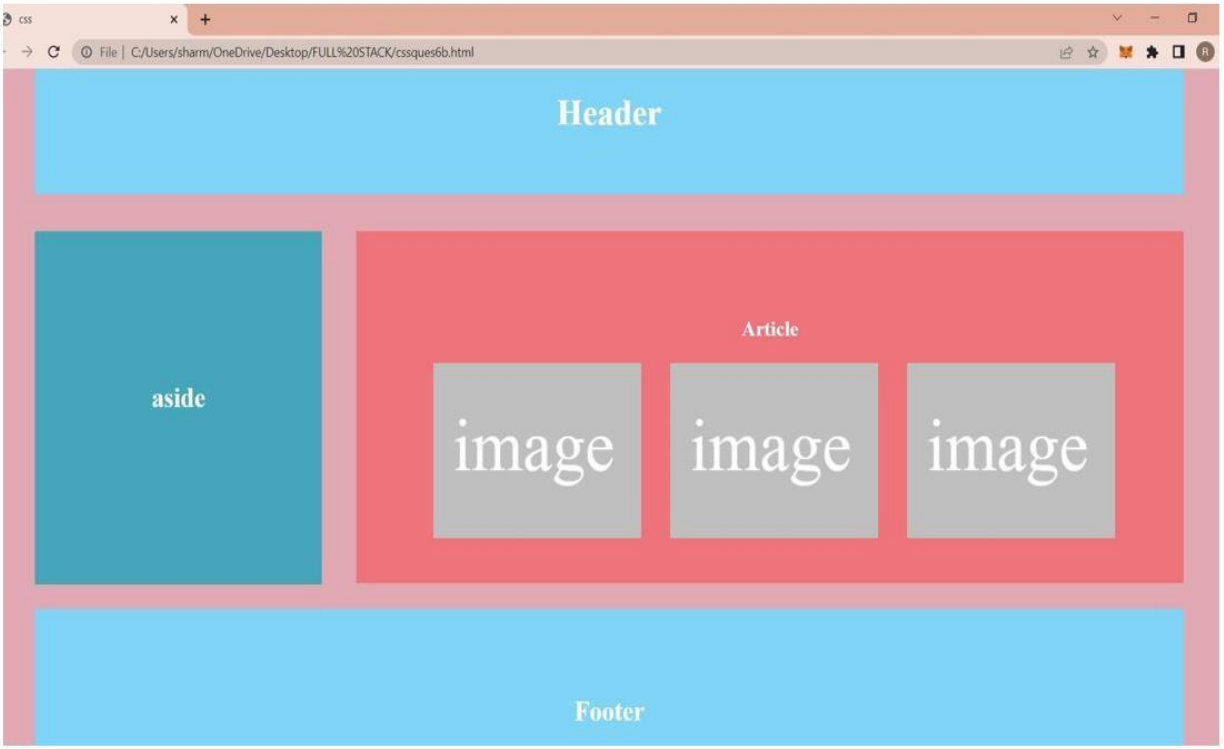
6C.

```
<!DOCTYPE html>
<html>
<head>
<title>css</title>
<style> body
{
background-color: rgb(225, 170, 179);
}
#main
{
margin-left: 55px; margin-right:55px;

}
#head
{
padding-top:40px; padding-
bottom:40px; background-color:
#80d4f5; color: #fff; font-size: 20px; text-
align: center;
}
.div2
{
position: relative; margin-top:40px;
}
div.left
{
float: left;
padding-top:10%; padding-bottom:11.5%; font-size: 20px;
text-align: center;
width: 25%; color: #fff;
background:#44a5bb;
```

```
}
div.right
{
padding-top:5%;  padding-bottom:1%;  margin-left:  10px;  font-size:  20px;  color:
#fff; text-align: center; float: right; width: 72%; background: #ef737b;
}
div.image-section
{ display: inline-block;} } img
{ margin: 0px 10px 30px 20px; }

footer
{
padding-top:5%;        padding-bottom:5%;        margin-
top:450px; background-color: #80d4f5; color: #fff; font-
size: 20px; text-align: center;
}
</style>
</head>
<body>

<div id="main">
<header id="head">
<h1>Header</h1>
</header>
<div class="div2">
<div class="left">
<nav>
<h1>aside</h1>
</nav>
</div>
<div class="right"><section >
<article >
```

```
<h1>Article</h1>
<div class="image-section">
<img src="imagee1.png">
<img src="imagee1.png">
<img src="imagee1.png">
</div>
</article>
</section></div>
</div>

<footer>
<h2>Footer</h2>
</footer>
</div>
</body>

</html>
```

OUTPUT:


css
File | C:/Users/sharm/OneDrive/Desktop/FULL%20STACK/cssques6b.html
Header
aside
Article
image
image
image
Footer

7.Write a program in JavaScript to make a calculator.

Ans) index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<title>Calculator</title>
<link href="styles.css" rel="stylesheet">
<script src="script.js" defer></script>
</head>
<body>
<div class="calculator-grid">
<div class="output">
<div data-previous-operand class="previous-operand"></div>
<div data-current-operand class="current-operand"></div>
</div>
<button data-all-clear class="span-two">AC</button>
<button data-delete>DEL</button>
<button data-operation>÷</button>
<button data-number>1</button>
<button data-number>2</button>
<button data-number>3</button>
<button data-operation>*</button>
<button data-number>4</button>
<button data-number>5</button>
<button data-number>6</button>
<button data-operation>+</button>
<button data-number>7</button>

<button data-number>8</button>
```

```
<button data-number>9</button>
<button data-operation>-</button>
<button data-number>.</button>
<button data-number>0</button>
<button data-equals class="span-two">=</button>
</div>
</body> </html>
```

styles.css

```
*, *::before, *::after { box-sizing: border-box; font-family:
Gotham Rounded, sans-serif; font-weight: normal;
}
body { padding: 0; margin: 0; background: linear-
gradient(to right, #00AAFF, #00FF6C);
}
.calculator-grid { display: grid; justify-content: center; align-
content: center; min-height: 100vh; grid-template-columns:
repeat(4, 100px); grid-template-rows: minmax(120px, auto)
repeat(5, 100px);
}
.calculator-grid > button { cursor: pointer;
font-size: 2rem; border: 1px solid white;
outline: none; background-color: rgba(255,
255, 255, .75);
}


.calculator-grid > button:hover { background-color: rgba(255, 255, 255, .9);
}

 .span-two {
grid-column: span 2;
}
```

```
.output { grid-column: 1 / -1; background-color: rgba(0, 0, 0, .75);
display: flex; align-items: flex-end; justify-content: space-around;
flex-direction: column; padding: 10px; word-wrap: break-word;
word-break: break-all;
}
.output .previous-operand { color: rgba(255, 255, 255, .75); font-size: 1.5rem;
}
.output .current-operand { color: white; font-
size: 2.5rem;
}
```

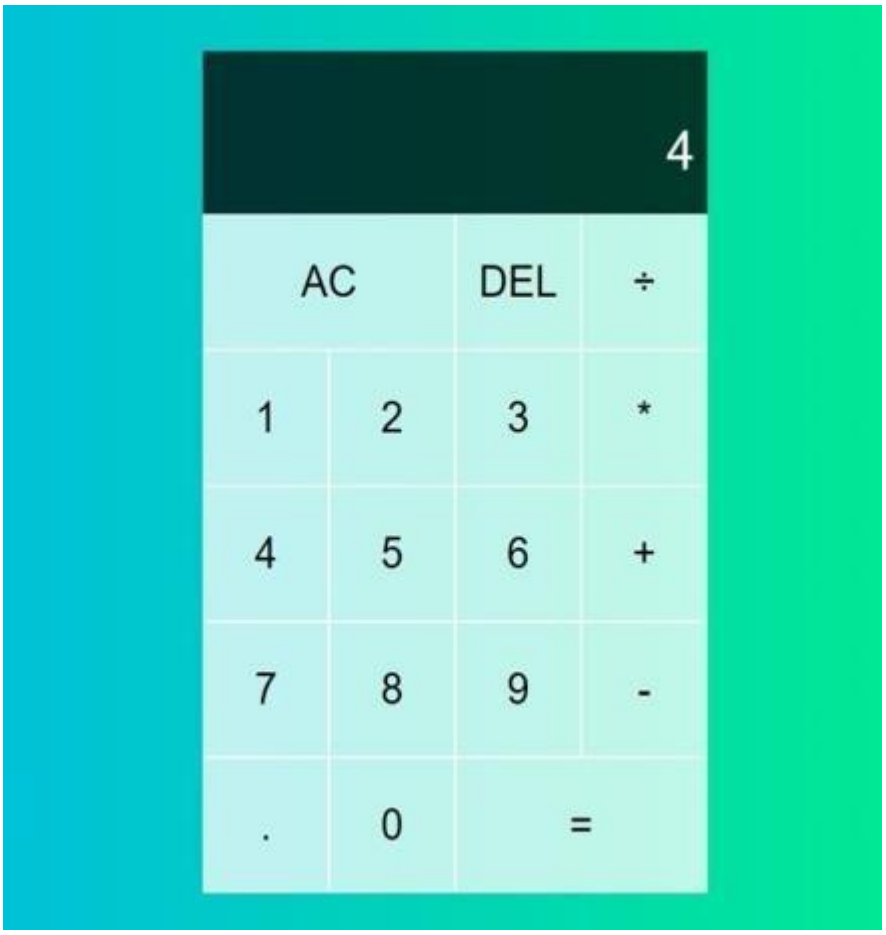
script.js class

```
Calculator {
constructor(previousOperandTextElement, currentOperandTextElement) {
this.previousOperandTextElement = previousOperandTextElement this.currentOperandTextElement =
currentOperandTextElement this.clear()
}
clear() { this.currentOperand = '' this.previousOperand = '' this.operation = undefined
}
delete() { this.currentOperand =
this.currentOperand.toString().slice(0, -1)
}
appendNumber(number) {
if (number === '.' && this.currentOperand.includes('.')) return this.currentOperand =
this.currentOperand.toString() + number.toString()
}
chooseOperation(operation) {
if (this.currentOperand === '') return if (this.previousOperand !== '') { this.compute()
}
this.operation = operation this.previousOperand = this.currentOperand this.currentOperand = ''
}
compute() { let
computation
```

```
const prev = parseFloat(this.previousOperand) const current = parseFloat(this.currentOperand) if (isNaN(prev) || isNaN(current)) return

switch (this.operation) { case '+':

computation = prev + current break

case '-': computation = prev -

current break

case '*': computation = prev *

current break

case '÷':

computation = prev / current break

default: return

}

this.currentOperand = computation this.operation = undefined this.previousOperand = ''

}

getDisplayNumber(number) { const stringNumber = number.toString() const integerDigits = parseFloat(stringNumber.split('.')[0]) const decimalDigits = stringNumber.split('.')[1] let integerDisplay

if (isNaN(integerDigits)) { integerDisplay = '' } else { integerDisplay = integerDigits.toLocaleString('en', {

maximumFractionDigits: 0 })

}

if (decimalDigits != null) { return

`${integerDisplay}.${decimalDigits}`

} else { return

integerDisplay

}

}

updateDisplay() { this.currentOperandTextElement.innerText = this.getDisplayNumber(this.currentOperand)      if (this.operation != null) { this.previousOperandTextElement.innerText = `${this.getDisplayNumber(this.previousOperand)} ${this.operation}`

} else { this.previousOperandTextElement.innerText = ''

}

}

}
```

```
const numberButtons = document.querySelectorAll('[data-number]') const operationButtons =
document.querySelectorAll('[data-operation]') const equalsButton =
document.querySelector('[dataequals]')

const deleteButton = document.querySelector('[data-delete]') const allClearButton =
document.querySelector('[data-all-clear]')

const previousOperandTextElement = document.querySelector('[data-previous-operand]')

const currentOperandTextElement = document.querySelector('[data-current-operand]') const

calculator = new Calculator(previousOperandTextElement, currentOperandTextElement)

numberButtons.forEach(button => { button.addEventListener('click', () => {

calculator.appendNumber(button.innerText) calculator.updateDisplay()

})

})

operationButtons.forEach(button => { button.addEventListener('click', () => {
calculator.chooseOperation(button.innerText) calculator.updateDisplay()

})})

equalsButton.addEventListener('click', button => { calculator.compute()

calculator.updateDisplay() }) allClearButton.addEventListener('click', button => {

calculator.clear() calculator.updateDisplay() }) deleteButton.addEventListener('click', button =>

{ calculator.delete() calculator.updateDisplay()

})
```

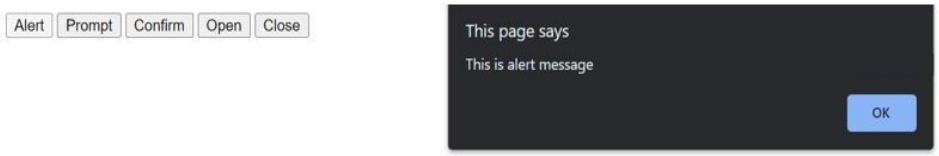
OUTPUT:

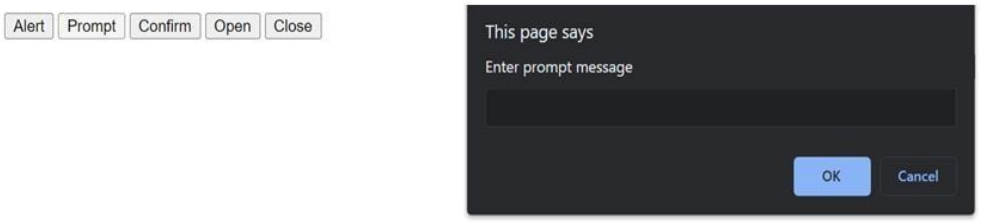


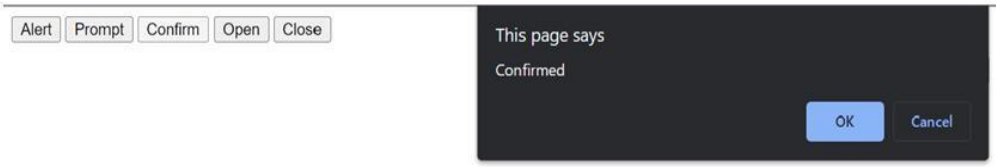
8. Window Object methods alert() , prompt() , confirm(), open(), close() , print().

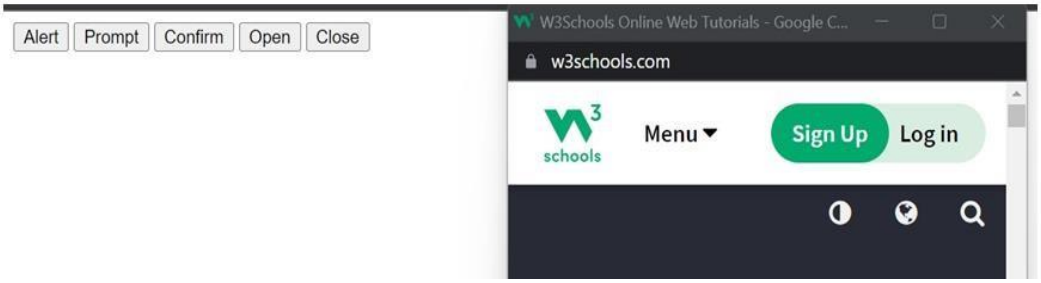
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
</head>
<body>
<input type="button" value="Alert" onclick="fun1(this)">
<input type="button" value="Prompt" onclick="fun2(this)">
<input type="button" value="Confirm" onclick="fun3(this)">
<input type="button" value="Open" onclick="fun4(this)">
<input type="button" value="Close" onclick="fun5(this)">


<script> function fun1(e){
alert("This is alert
message");
}
function fun2(e){ prompt("Enter
prompt message");
}
function fun3(e){ confirm("Confirmed");
}
function fun4(e){ mywindow = open("https://www.w3schools.com", "", "width=400,height=400");
} function fun5(e){ mywindow.close();}
</script>
< /body>
</html>
```

OUTPUT


Alert
Prompt
Confirm
Open
Close
This page says
This is alert message
OK


Alert
Prompt
Confirm
Open
Close
This page says
Enter prompt message
OK
Cancel


Alert
Prompt
Confirm
Open
Close
This page says
Confirmed
OK
Cancel


Alert
Prompt
Confirm
Open
Close
W3Schools Online Web Tutorials - Google C...
w3schools.com
Menu
Sign Up
Log in

9. Event Handling - Background Color Change.

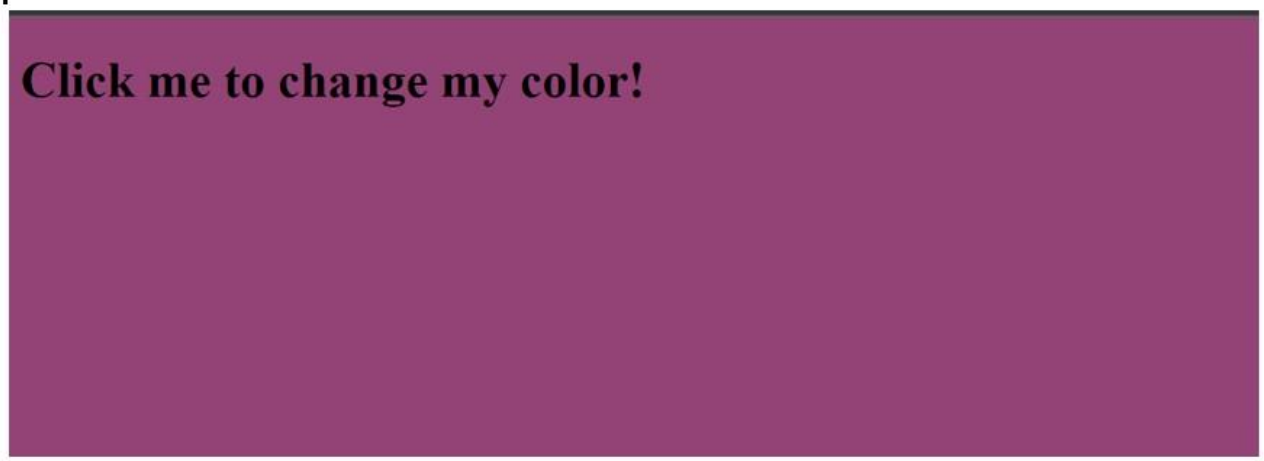
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible"
content="IE=edge">
   <meta name="viewport" content="width=device-
width, initial-scale=1.0">
   <title>Document</title>
</head>
<body>
    <div style="font-size: 50px" class="main-div">
      Click here to change the color
    </div>
</body>
<script src="script.js"></script>
</html>
```

Script.js

```
var clickableDiv=document.body;

clickableDiv.addEventListener('click',function () {
   var randomColor=getRandomColor();
   document.body.style.backgroundColor=randomColor;
})

function getRandomColor() {
  var letters = "0123456789ABCDEF";
  var color = "#";
  for (var i = 0; i < 6; i++) {
    color += letters[Math.floor(Math.random() * 16)];
  }
  return color;
```

```
}
```

OUTPUT



12) Write a program in JavaScript to create an html page with 2 combo box populated with month & year, to display the calendar for the selected month & year from combo box using javascript.

Ans:

```
<!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">
   <style>
      td{
         text-align: center;
      }

      #output {
         margin-top: 2rem;
         display: flex;
         justify-content: center;
      }
   </style>
   <title>Calendar</title>
</head>
<body bgcolor='red'>
   <h1>Calendar</h1>

   <h2>Select Year</h2>
   <select name="year" id="year">
   </select>

   <h2>Select Month</h2>
   <select name="month" id="month">
      <option value="0">Jan</option>
```

```
        <option value="1">Feb</option>
        <option value="2">Mar</option>
        <option value="3">Apr</option>
        <option value="4">May</option>
        <option value="5">Jun</option>
        <option value="6">jul</option>
        <option value="7">Aug</option>
        <option value="8">Sep</option>
        <option value="9">Oct</option>
        <option value="10">Nov</option>
        <option value="11">Dec</option>
    </select>
    <button id="ok">ok</button>
    <div id="output">

    </div>
    <script src="script.js"></script>

</body>
</html>
```

Script.js

```
const year = document.getElementById("year");
const month = document.getElementById("month");
const btn = document.getElementById("ok");
const monthName = [
  "January",
  "February",
  "March",
  "April",
  "May",
  "June",
  "July",
```

```
  "August",
  "September",
  "October",
  "December",
];
const output = document.getElementById("output");
const days = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31];

for (var i = 1990; i < 2050; i++) {
  year.innerHTML += `<option value = ${i}>${i}</option>`;
}

btn.addEventListener("click", (e) => {
  var m = +month.value + 1;
  var D = year.value % 100; //to ge the last two digits of the year value
  var C = Math.floor(year.value / 100); //to get the first two digits of the year value
  var last = days[month.value]; // the last no of the month entitiy

  if (m > 2) m -= 2;
  else {
    m += 10;
    D -= 1;
  }

  var f =
    1 +
    Math.floor((13 * m - 1) / 5) +
    D +
    Math.floor(D / 4) +
    Math.floor(C / 4) -
    2 * C;
  if (f < 0) f = 7000 + f;
```

```
 f = f % 7;


 outputHTML = `<table border='2' cellpadding='20'> <tr><th colspan='7'>${
   monthName[month.value]
 } ${
   year.value
 }</tr>
<tr><th>Sun</th><th>Mon</th><th>Tue</th><th>Wed</th><th>Thur</th><th>Fri</th><th>Sat</th><
/tr>`;


 var count = 1;


 var row = "<tr>";


 for (var temp = 0; temp < f; temp++) row += "<td></td>";


 for (var c = f; c < 7; c++) {
   row += `<td>${count}</td>`;
   count += 1;
 }


 row += "</tr>";


 outputHTML += row;


 while (count <= last) {
   row = "<tr>";
   var tot = 0;


   for (; tot < 7 && count <= last; tot++, count++) row += `<td>${count}</td>`;


   row += "</tr>";
   outputHTML += row;
```

```
  }

  outputHTML += "</table>";

  output.innerHTML = outputHTML;
});
```

OUTPUT

April 2023

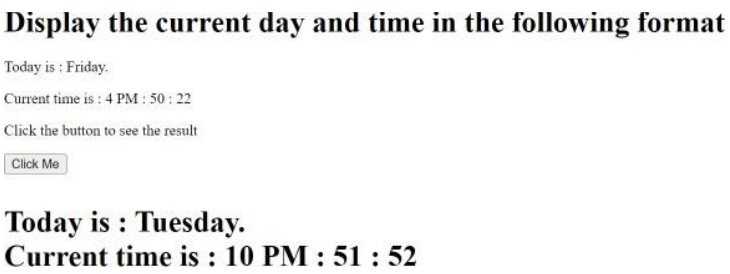
sun	mon	tue	wed	thu	fri	sat
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

13a) Write a JavaScript program to display the current day and time in the following format. Sample Output : Today is : Friday. Current time is : 4 PM : 50 : 22.

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

<h1>Display the current day and time in the following format</h1>
<p>Today is : Friday.</p>
<p>Current time is : 4 PM : 50 : 22</p>
<p>Click the button to see the result</p>
<button id="btn">Click Me</button>
<p id="result"></p>
<style> #result{ font-size: 2rem;
font-weight: bold;
}
</style>

<script> var today = new Date(); var day = today.getDay(); var daylist =
["Sunday","Monday","Tuesday","Wednesday ","Thursday","Friday","Saturday"]; var hour =
today.getHours(); var minute = today.getMinutes(); var second = today.getSeconds(); var
prepand = (hour >= 12)? " PM ":" AM "; hour = (hour >= 12)? hour - 12: hour; if (hour===0
&& prepand===' PM ')
{
if (minute===0 && second===0)
```

```
{
hour=12; prepand=' Noon';
}
else
{
hour=12; prepand=' PM';
}
}
if (hour===0 && prepand===' AM ')
{
if (minute===0 && second===0)
{
hour=12; prepand='
Midnight';
}
else
{
hour=12; prepand=' AM';
}
}
var result = document.querySelector('#result'); var btn = document.querySelector('#btn');
btn.addEventListener('click', function(){
result.innerHTML = 'Today is : ' + daylist[day] + '.' + '<br>' + 'Current time is : ' + hour + prepand + ' : ' +
minute + ' : ' + second;
})
</script>
</body>
</html>
```

OUTPUT:


Display the current day and time in the following format
Today is : Friday.
Current time is : 4 PM : 50 : 22
Click the button to see the result
Click Me
Today is : Tuesday.
Current time is : 10 PM : 51 : 52

13b) Write a JavaScript program to get the current date.

Expected Output : mm-dd-yyyy , mm/dd/yyyy or dd-mm-yyyy, dd/mm/yyyy

Ans:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
</head>
<body>
<h1>Write a JavaScript program to get the current date.</h1>
<p>Expected Output :</p>
<p>mm-dd-yyyy,</p>
<p>mm/dd/yyyy or dd-mm-yyyy,</p>
<p>dd/mm/yyyy</p>
<p>Click the button to see the result</p>
<button id="btn">Click Me</button>
<p id="result"></p>
<style> #result{ font-size: 2rem;
font-weight: bold;
}
</style> <script> var today = new Date(); var dd = today.getDate(); var mm
= today.getMonth()+1; //January is 0! var yyyy = today.getFullYear();
if(dd<10){ dd='0'+dd

}
if(mm<10){ mm='0'+mm
}
var today = mm+'/'+dd+'/'+yyyy;
var result = document.querySelector('#result'); var btn =
document.querySelector('#btn'); btn.addEventListener('click', function(){
result.innerHTML = today;
})
```

```
</script>
</body>
</html>
```

OUTPUT

Write a JavaScript program to get the current date.

Expected Output :

mm-dd-yyyy,

mm/dd/yyyy or dd-mm-yyyy,

dd/mm/yyyy

Click the button to see the result

Click Me

04/11/2023

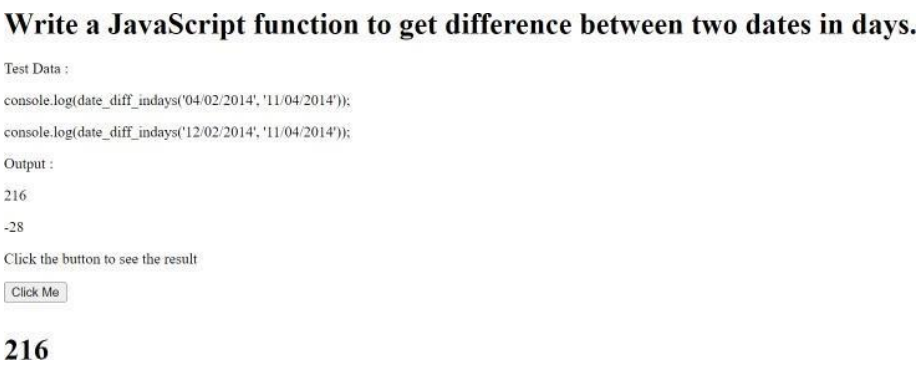
13c) Write a JavaScript function to get difference between two dates in days.

Ans:

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

<h1>Write a JavaScript function to get difference between two dates in days.</h1>
<p>Test Data :</p>
<p>console.log(date_diff_indays('04/02/2014', '11/04/2014'));</p>
<p>console.log(date_diff_indays('12/02/2014', '11/04/2014'));</p>
<p>Output :</p>
<p>216</p>
<p>-28</p>
<p>Click the button to see the result</p>
<button id="btn">Click Me</button>
<p id="result"></p>
<style> #result{ font-size: 2rem;
font-weight: bold;
}
</style> <script> function date_diff_indays(date1, date2) { dt1 =
new Date(date1); dt2 = new Date(date2);

return Math.floor((Date.UTC(dt2.getFullYear(), dt2.getMonth(), dt2.getDate()) -
Date.UTC(dt1.getFullYear(), dt1.getMonth(), dt1.getDate()) ) /(1000 * 60 * 60 * 24));
}
var result = document.querySelector('#result'); var btn =
document.querySelector('#btn'); btn.addEventListener('click', function(){
result.innerHTML = date_diff_indays('04/02/2014', '11/04/2014');
```

```
})
</script>
</body>
</html>
```

OUTPUT


Write a JavaScript function to get difference between two dates in days.
Test Data :
console.log(date_diff_indays('04/02/2014', '11/04/2014'));
console.log(date_diff_indays('12/02/2014', '11/04/2014'));
Output :
216
-28
Click the button to see the result
Click Me
216

13d) Write a JavaScript function to count the number of days passed since beginning of the year.

Ans: <!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<title>Document</title>

</head>

<body>

<h1>Write a JavaScript function to count the number of days passed since beginning of the year.</h1>

Test Data :

console.log(days_passed(new Date(2015, 0, 15)));

15

console.log(days_passed(new Date(2015, 11, 14)));

348

Click the button to see the result

<button id="btn">Click Me</button>

<style> #result{ font-size: 2rem;

font-weight: bold;

}

</style> <script> function days_passed(date1) { var dt1 =

new Date(date1); var dt2 = new Date(dt1.getFullYear(),

0, 0);

return Math.floor((Date.UTC(dt1.getFullYear(), dt1.getMonth(), dt1.getDate()) -
Date.UTC(dt2.getFullYear(), dt2.getMonth(), dt2.getDate())) /(1000 * 60 * 60 * 24));

}

```
var result = document.querySelector('#result'); var btn = document.querySelector('#btn'); btn.addEventListener('click', function(){
result.innerHTML = days_passed(new Date(2015, 0, 15));
})
</script>
</body>
</html>
```

OUTPUT

Write a JavaScript function to count the number of days passed since beginning of the year.

Test Data :

console.log(days_passed(new Date(2015, 0, 15)));

15

console.log(days_passed(new Date(2015, 11, 14)));

348

Click the button to see the result

Click Me

101

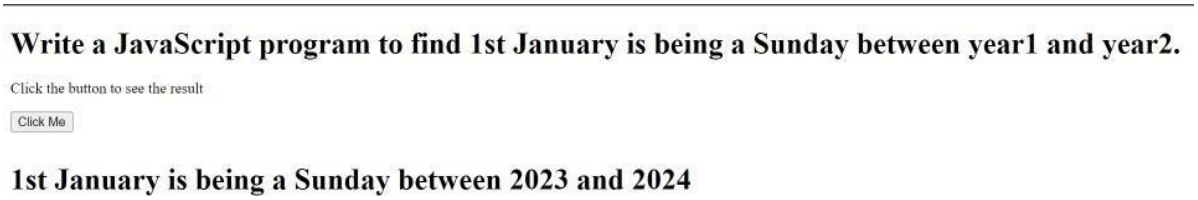
13e) Write a JavaScript program to find 1st January is being a Sunday between year1 and year2.

Ans:

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

<h1>Write a JavaScript program to find 1st January is being a Sunday between year1 and year2.</h1>
<p>Click the button to see the result</p>
<button id="btn">Click Me</button>
<p id="result"></p>
<style> #result{ font-size: 2rem;
font-weight: bold;
}
</style> <script> var year1 = prompt('Enter the first year'); var year2 = prompt('Enter the
second year'); var result = document.querySelector('#result'); var btn =
document.querySelector('#btn'); var flag = 0; btn.addEventListener('click', function(){
for(var i = year1; i <= year2; i++){
var d = new Date(i, 0, 1); if(d.getDay() === 0){
flag = 1;
}
}
        if(flag === 1){
        result.innerHTML = '1st January is being a Sunday between ' + year1 + ' and ' +

year2;
```

```
        }
        else{
        result.innerHTML = '1st January is not being a Sunday between ' + year1 + ' and ' +


year2;

}
})
</script>
</body>
</html>
```

OUTPUT

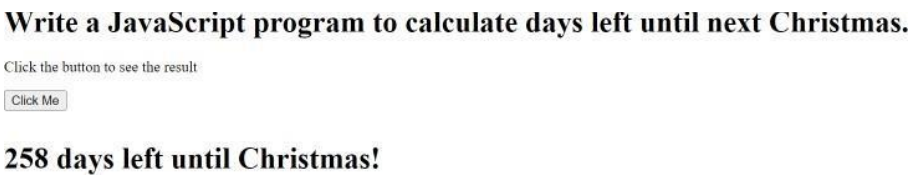


13f) Write a JavaScript program to calculate days left until next Christmas.

Ans:

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

<h1>Write a JavaScript program to calculate days left until next Christmas.</h1>
<p>Click the button to see the result</p>
<button id="btn">Click Me</button>
<p id="result"></p>
<style> #result{ font-size: 2rem;
font-weight: bold;
}
</style> <script> var
today = new Date();
var christmas = new Date(today.getFullYear(), 11, 25); if (today.getMonth() == 11 && today.getDate() > 25) { christmas.setFullYear(christmas.getFullYear() + 1);
}
var one_day=1000*60*60*24; var result = document.querySelector('#result'); var btn =
document.querySelector('#btn'); btn.addEventListener('click', function(){
result.innerHTML = Math.ceil((christmas.getTime()-today.getTime())/(one_day)) + " days left until Christmas!";
})
</script>
</body>
</html>
```

OUTPUT:

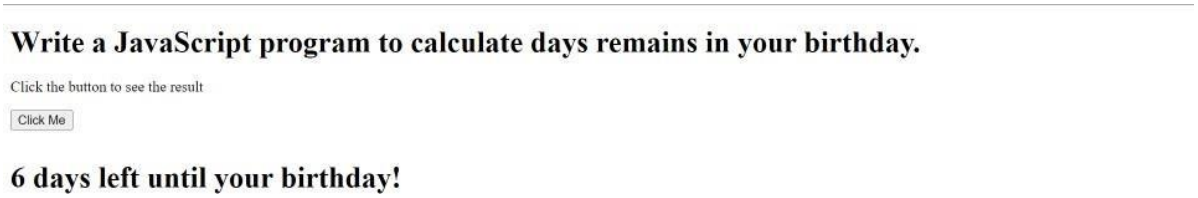

Write a JavaScript program to calculate days left until next Christmas.
Click the button to see the result
Click Me
258 days left until Christmas!

13g) Write a JavaScript program to calculate days remains in your birthday.

Ans:

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

<h1>Write a JavaScript program to calculate days remains in your birthday.</h1>
<p>Click the button to see the result</p>
<button id="btn">Click Me</button>
<p id="result"></p>
<style> #result{ font-size: 2rem;
font-weight: bold;
}
</style> <script> var
today = new Date();
var birthday = prompt("Enter your birthday in the format mm/dd/yyyy"); birthday = new
Date(birthday); birthday.setFullYear(today.getFullYear()); if (today.getMonth() == 11 &&
today.getDate() > 25) { birthday.setFullYear(birthday.getFullYear() + 1);
}
var one_day=1000*60*60*24;
var result = document.querySelector('#result'); var btn = document.querySelector('#btn');
btn.addEventListener('click', function(){
result.innerHTML = Math.ceil((birthday.getTime()-today.getTime())/(one_day)) + " days left until your
birthday!";
})
</script>
</body>
</html>
```

OUTPUT


Write a JavaScript program to calculate days remains in your birthday.
Click the button to see the result
Click Me
6 days left until your birthday!