WEB PROGRAMMING LAB WORK

Submitted by:

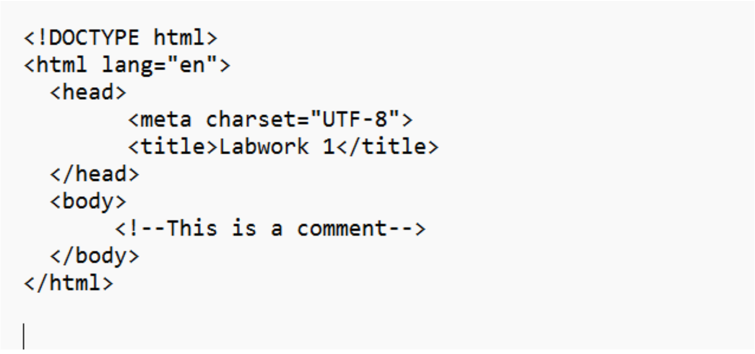
Name: Rishika Hazarika

Roll No: 210710007037

Date of Submission: 06.12.2023

1. Create a Web Page which shows how to use *Comments* in HTML.

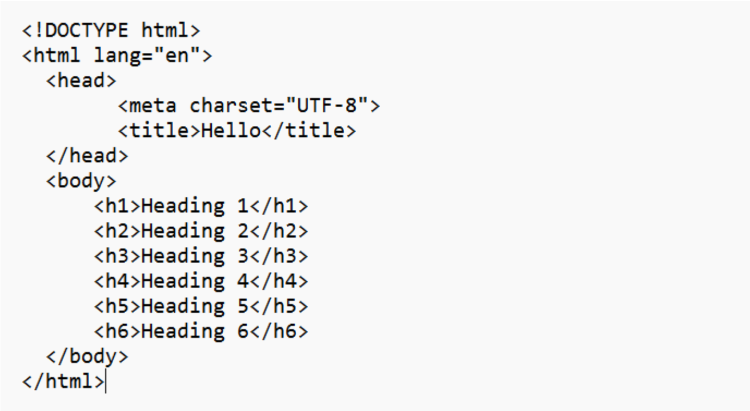
Code:



```
<!DOCTYPE html>
<html lang="en">
  <head>
        <meta charset="UTF-8">
        <title>Labwork 1</title>
  </head>
  <body>
       <!--This is a comment-->
  </body>
</html>
```

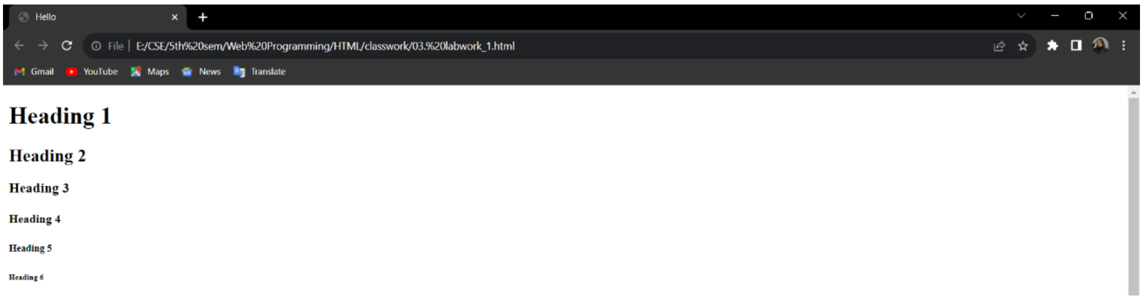
2. Create a Web Page to demonstrate the use of all *Heading tags.*

Code:



```
<!DOCTYPE html>
<html lang="en">
  <head>
        <meta charset="UTF-8">
        <title>Hello</title>
  </head>
  <body>
      <h1>Heading 1</h1>
      <h2>Heading 2</h2>
      <h3>Heading 3</h3>
      <h4>Heading 4</h4>
      <h5>Heading 5</h5>
      <h6>Heading 6</h6>
  </body>
</html>
```

Output:



Hello
File | E:/CSE/5th%20sem/Web%20Programming/HTML/classwork/03.%20labwork_1.html
Gmail
YouTube
Maps
News
Translate

Heading 1

Heading 2

Heading 3

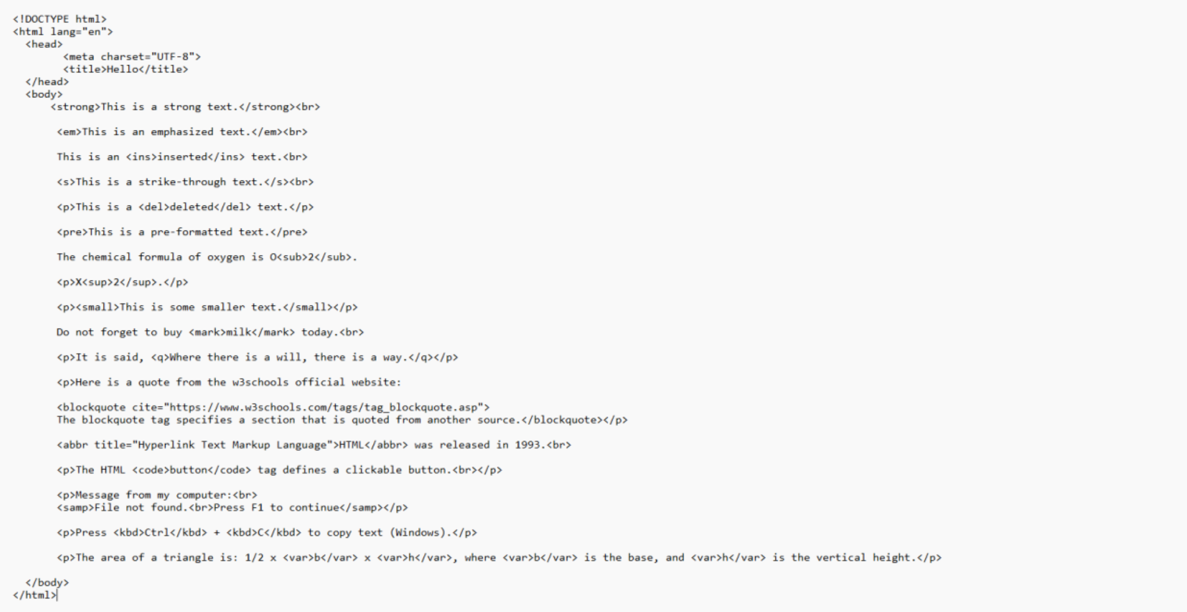
Heading 4

Heading 5

Heading 6

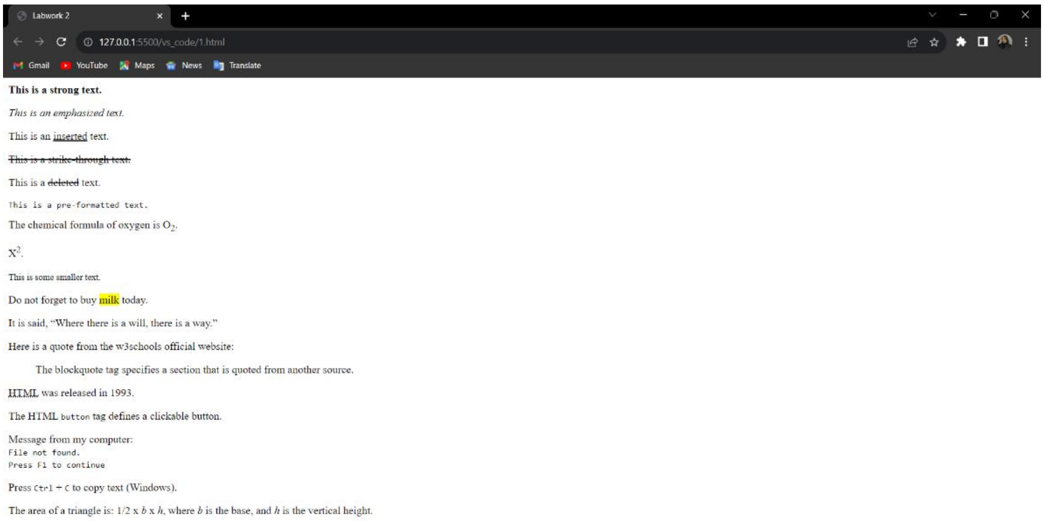
3. Check how to use the following tags, and create a Web Page to show their output: `<strong>, <em>, <ins>, <s>, <del>, <pre>, <sub>, <sup>, <small>, <mark>, <q>, <blockquote>, <abbr>, <code>, <samp>, <kbd>, <var>`

Code:



```
<!DOCTYPE html>
<html lang="en">
  <head>
        <meta charset="UTF-8">
        <title>Hello</title>
  </head>
  <body>
      <strong>This is a strong text.</strong><br>

       <em>This is an emphasized text.</em><br>

       This is an <ins>inserted</ins> text.<br>

       <s>This is a strike-through text.</s><br>

       <p>This is a <del>deleted</del> text.</p>

       <pre>This is a pre-formatted text.</pre>

       The chemical formula of oxygen is O<sub>2</sub>.

       <p>X<sup>2</sup>.</p>

       <p><small>This is some smaller text.</small></p>

       Do not forget to buy <mark>milk</mark> today.<br>

       <p>It is said, <q>Where there is a will, there is a way.</q></p>

       <p>Here is a quote from the w3schools official website:

       <blockquote cite="https://www.w3schools.com/tags/tag_blockquote.asp">
       The blockquote tag specifies a section that is quoted from another source.</blockquote></p>

       <abbr title="Hyperlink Text Markup Language">HTML</abbr> was released in 1993.<br>

       <p>The HTML <code>button</code> tag defines a clickable button.<br></p>

       <p>Message from my computer:<br>
       <samp>File not found.<br>Press F1 to continue</samp></p>

       <p>Press <kbd>Ctrl</kbd> + <kbd>C</kbd> to copy text (Windows).</p>

       <p>The area of a triangle is: 1/2 x <var>b</var> x <var>h</var>, where <var>b</var> is the base, and <var>h</var> is the vertical height.</p>

  </body>
</html>
```

Output:



This is a strong text.

This is an emphasized text.

This is an inserted text.

~~This is a strike-through text.~~

This is a ~~deleted~~ text.

```
This is a pre-formatted text.
```

The chemical formula of oxygen is O_2.

X^2.

This is some smaller text.

Do not forget to buy milk today.

It is said, “Where there is a will, there is a way.”

Here is a quote from the w3schools official website:

> The blockquote tag specifies a section that is quoted from another source.

HTML was released in 1993.

The HTML `button` tag defines a clickable button.

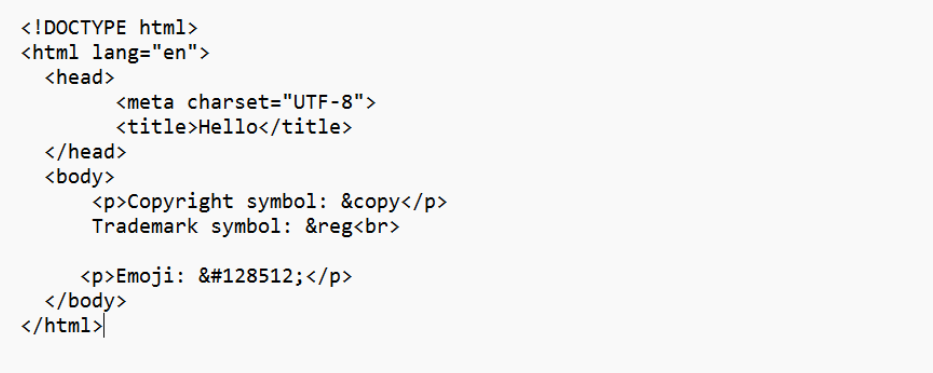
Message from my computer:
```
File not found.
Press F1 to continue
```

Press `Ctrl` + `C` to copy text (Windows).

The area of a triangle is: 1/2 x *b* x *h*, where *b* is the base, and *h* is the vertical height.

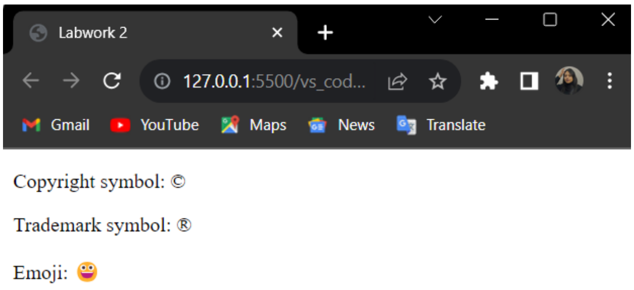
4. Check how to insert special characters (eg. Copyright symbol, Trademark Symbol etc.) and Emojis in a Web Page.

Code:



```
<!DOCTYPE html>
<html lang="en">
  <head>
        <meta charset="UTF-8">
        <title>Hello</title>
  </head>
  <body>
      <p>Copyright symbol: &copy</p>
      Trademark symbol: &reg<br>

     <p>Emoji: &#128512;</p>
  </body>
</html>
```

Output:



Copyright symbol: ©

Trademark symbol: ®

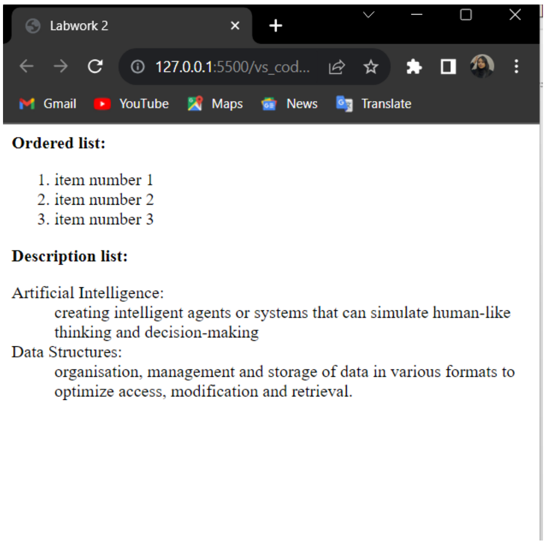
Emoji: 😃

5. Create a Web Page to show how to use *Ordered Lists* and *Description Lists.*

Code:

```
<!DOCTYPE html>
<html lang="en">
  <head>
        <meta charset="UTF-8">
        <title>Labwork 2</title>
  </head>
  <body>
      <b>Ordered list:</b>
<ol>
<li>item number 1</li>
<li>item number 2</li>
<li>item number 3</li>
</ol>

<b>Description list:</b>
<dl>
<dt>Artificial Intelligence:</dt>
<dd>creating intelligent agents or systems that can simulate human-like thinking and decision-making</dd>
<dt>Data Structures:</dt>
<dd>organisation, management and storage of data in various formats to optimize access, modification and retrieval.</dd>
</dl>

  </body>
</html>
```

Output:



Ordered list:

1. item number 1
2. item number 2
3. item number 3

Description list:

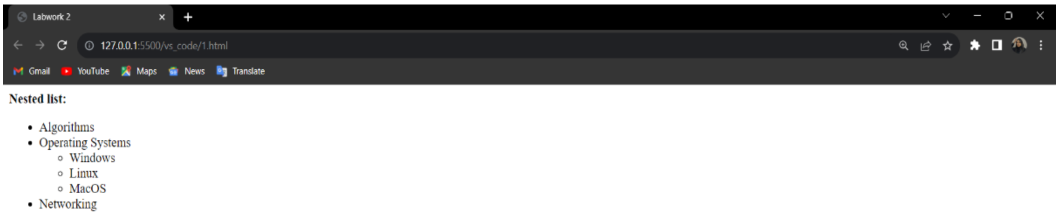
Artificial Intelligence:
creating intelligent agents or systems that can simulate human-like thinking and decision-making
Data Structures:
organisation, management and storage of data in various formats to optimize access, modification and retrieval.

6. Create a Web Page to demonstrate the use of *Nested Lists.*

Code:

```
<!DOCTYPE html>
<html lang="en">
  <head>
        <meta charset="UTF-8">
        <title>Labwork 2</title>
  </head>
  <body>
      <b>Nested list:</b>
      <ul>
      <li>Algorithms</li>
      <li>Operating Systems
      <ul>
      <li>Windows</li>
      <li>Linux</li>
      <li>MacOS</li>
      </ul>
      </li>
      <li>Networking</li>
      </ul>
  </body>
</html>
```

Output:



Nested list:

- Algorithms
- Operating Systems
  - Windows
  - Linux
  - MacOS
- Networking

7. Create a Web Page to show the use of *image as a link.*

Code:

```
<!DOCTYPE html>
<html lang="en">
  <head>
        <meta charset="UTF-8">
        <title>Labwork 2</title>
  </head>
  <body>
      <p><b>Image as a link:</b></p>
      <a href="https://www.google.com"><img src="C:\Users\rishi\Downloads\google.png" width="200"></a><br>
  </body>
</html>
```

Output:

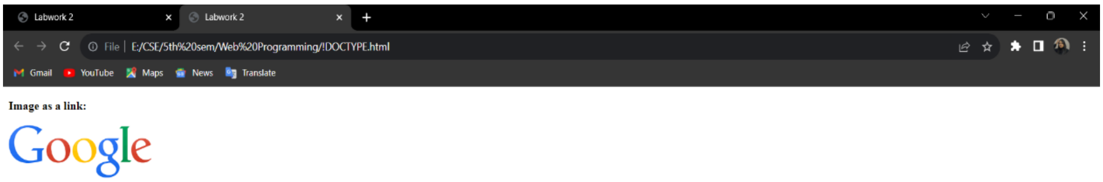


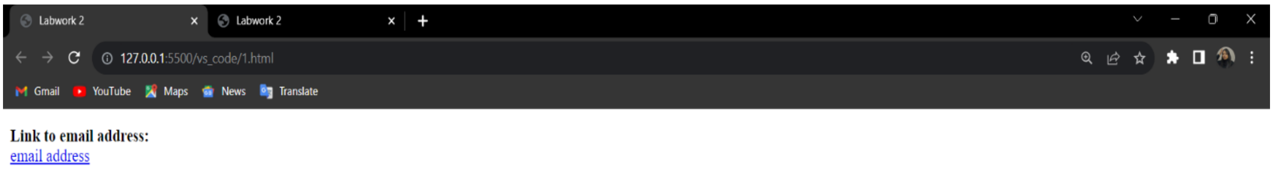
Image as a link:

8. Create a Web Page to create a *link to an e-mail address.*

Code:

```
<!DOCTYPE html>
<html lang="en">
  <head>
        <meta charset="UTF-8">
        <title>Labwork 2</title>
  </head>
  <body>
      <p><b>Link to email address:</b><br>
            <a href= "mailto: rishikahz@gmail.com">email address</a><br></p>
  </body>
</html>
```

Output:



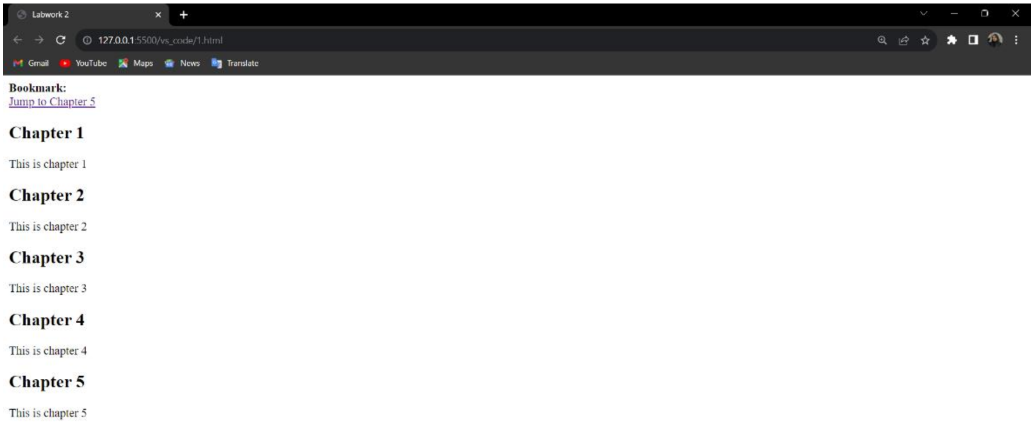
Link to email address:
email address

9. Create a Web Page to show the use of *bookmark.*

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
      <meta charset="UTF-8">
      <title>Labwork 2</title>
</head>
<body>

      <b>Bookmark:</b><br>
      <a href="#C5">Jump to Chapter 5</a>
      <h2>Chapter 1</h2>
      <p>This is chapter 1</p>
      <h2>Chapter 2</h2>
      <p>This is chapter 2</p>
      <h2>Chapter 3</h2>
      <p>This is chapter 3</p>
      <h2>Chapter 4</h2>
      <p>This is chapter 4</p>
      <h2 id="C5">Chapter 5</h2>
      <p>This is chapter 5</p>

</body>
</html>
```

Output:



Bookmark:
Jump to Chapter 5

Chapter 1

This is chapter 1

Chapter 2

This is chapter 2

Chapter 3

This is chapter 3

Chapter 4

This is chapter 4

Chapter 5

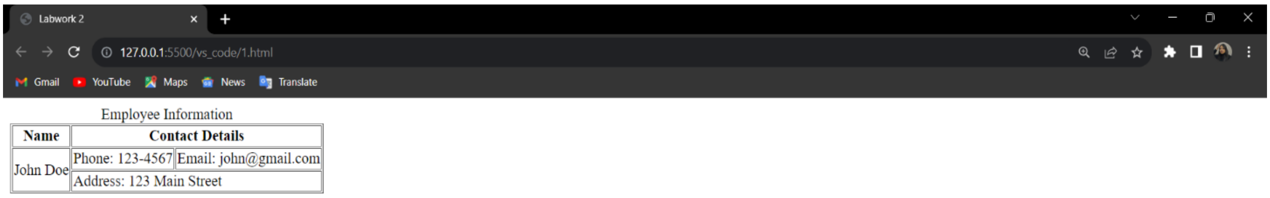
This is chapter 5

10. Create a table in a Web Page to show the usage of `<caption>` tag, and `colspan`, `rowspan` properties.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
      <meta charset="UTF-8">
      <title>Labwork 2</title>
</head>
<body>

    <table border="lpk">
         <caption>Employee Information</caption>
         <tr>
         <th>Name</th>
         <th colspan="2">Contact Details</th>
         </tr>
         <tr>
         <td rowspan="2">John Doe</td>
         <td>Phone: 123-4567</td>
         <td>Email: john@gmail.com</td>
         </tr>
         <tr>
         <td colspan="2">Address: 123 Main Street</td>
         </tr>
     </table>

</body>
</html>
```

Output:



Employee Information

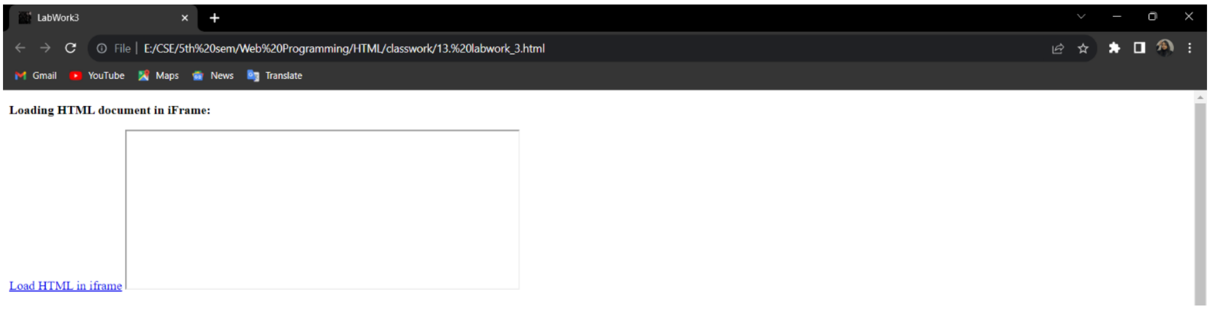
Name	Contact Details	
John Doe	Phone: 123-4567	Email: john@gmail.com
	Address: 123 Main Street	

11. Create a Web Page to show how to load a HTML document in an iFrame, when the user clicks on a link (i.e. iFrame as the target of a link).

Code:

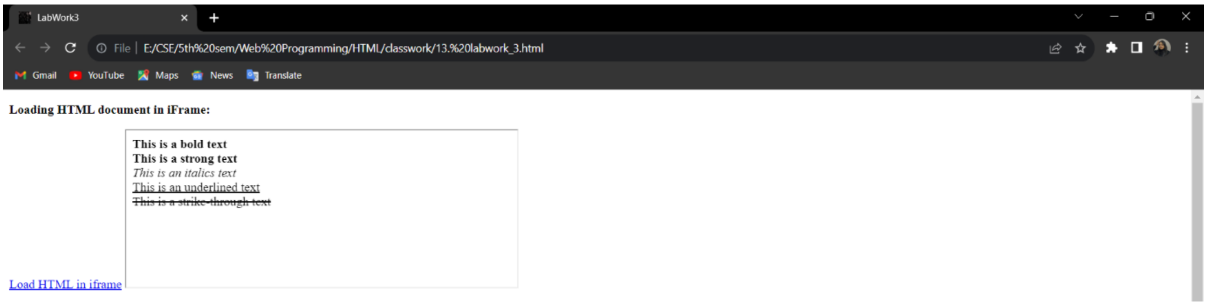
```
<!DOCTYPE html>
<html lang="en">
<head>
      <meta charset="UTF-8">
      <title>Labwork 3</title>
</head>
<body>
      <p><b>Loading HTML document in iFrame:</b></p>
      <a href="02. text_formatting.html" target="iframe">Load HTML in iframe</a>
      <iframe name= "iframe" width="500" height="200">
      </iframe>
</body>
</html>
```

Output:


LabWork3
File | E:/CSE/5th%20sem/Web%20Programming/HTML/classwork/13.%20labwork_3.html
Gmail
YouTube
Maps
News
Translate

Loading HTML document in iFrame:

Load HTML in iframe


LabWork3
File | E:/CSE/5th%20sem/Web%20Programming/HTML/classwork/13.%20labwork_3.html
Gmail
YouTube
Maps
News
Translate

Loading HTML document in iFrame:

This is a bold text
This is a strong text
This is an italics text
This is an underlined text
~~This is a strike-through text~~

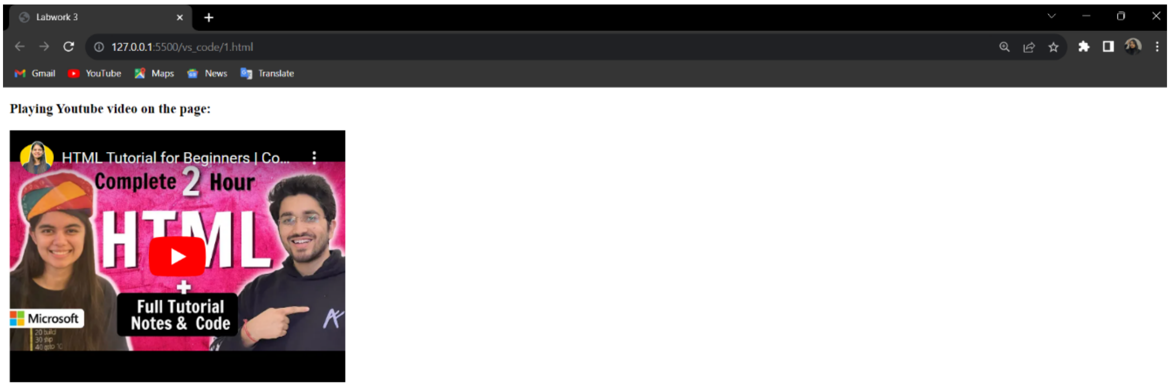
Load HTML in iframe

12. Create a Web Page to show how to play a Youtube video in the page.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
      <meta charset="UTF-8">
      <title>Labwork 3</title>
</head>
<body>
      <p><b>Playing Youtube video on the page:</b></p>
      <iframe width="400" height="300" src="https://www.youtube.com/embed/HcOc7P5BMi4?si=raTj1VzqCWEk38wS"
            frameborder="0">
      </iframe>
</body>
</html>
```

Output:



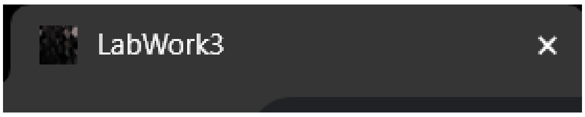
Playing Youtube video on the page:

13. Create a Web Page to show how to include a *Favicon.*

Code:

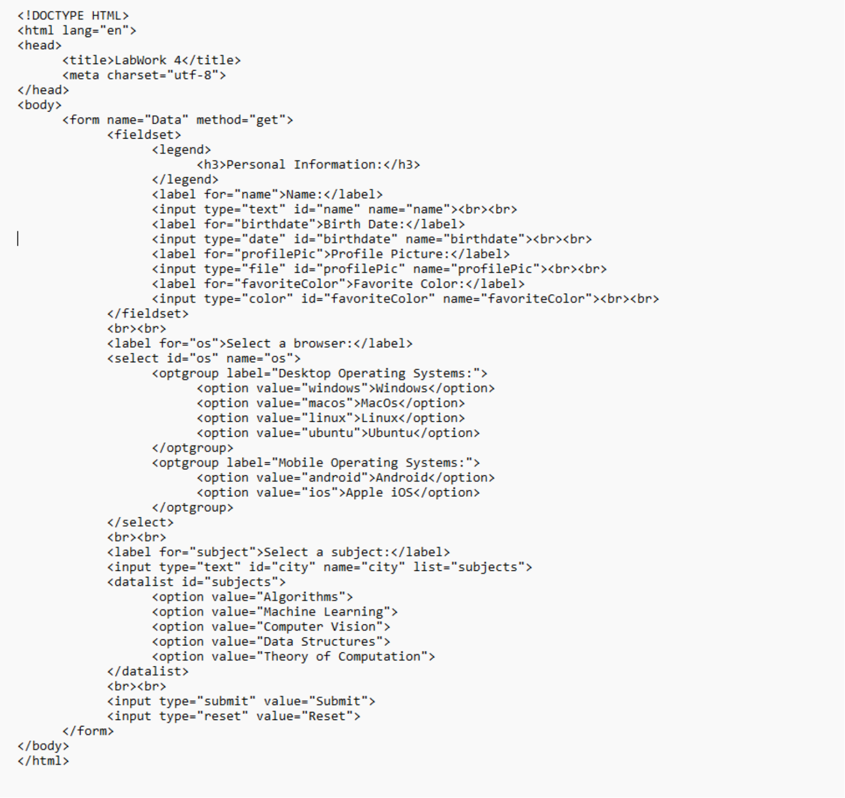
```
<!DOCTYPE html>
<html lang="en">
<head>
      <meta charset="UTF-8">
      <title>Labwork 3</title>
      <link rel="icon" href="C:\Users\rishi\Downloads\favicon_io\favicon.ico" type="image/x-icon">
      <link rel="shortcut icon" href="C:\Users\rishi\Downloads\favicon_io\favicon.ico" type="image/x-icon">
</head>
<body>
</body>
</html>
```

Output:



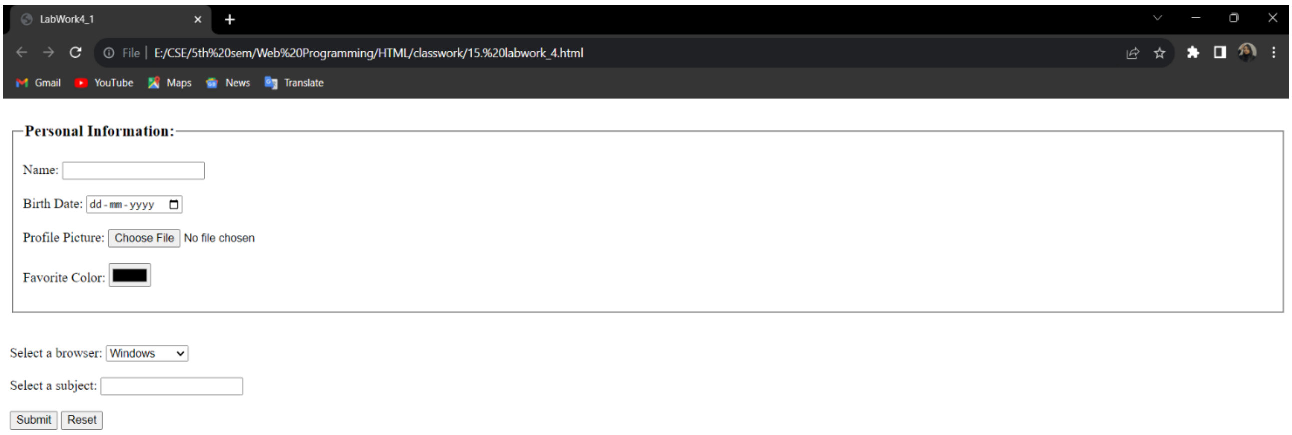
14. Create a Web Page to demonstrate the following `<form>` elements:
    - `<fieldset>` and `<legend>`
    - Input types: `date`, `file` and `color`
    - `<datalist>`
    - `<optgroup>`

Code:



```
<!DOCTYPE HTML>
<html lang="en">
<head>
      <title>LabWork 4</title>
      <meta charset="utf-8">
</head>
<body>
      <form name="Data" method="get">
            <fieldset>
                  <legend>
                        <h3>Personal Information:</h3>
                  </legend>
                  <label for="name">Name:</label>
                  <input type="text" id="name" name="name"><br><br>
                  <label for="birthdate">Birth Date:</label>
                  <input type="date" id="birthdate" name="birthdate"><br><br>
                  <label for="profilePic">Profile Picture:</label>
                  <input type="file" id="profilePic" name="profilePic"><br><br>
                  <label for="favoriteColor">Favorite Color:</label>
                  <input type="color" id="favoriteColor" name="favoriteColor"><br><br>
            </fieldset>
            <br><br>
            <label for="os">Select a browser:</label>
            <select id="os" name="os">
                  <optgroup label="Desktop Operating Systems:">
                        <option value="windows">Windows</option>
                        <option value="macos">MacOs</option>
                        <option value="linux">Linux</option>
                        <option value="ubuntu">Ubuntu</option>
                  </optgroup>
                  <optgroup label="Mobile Operating Systems:">
                        <option value="android">Android</option>
                        <option value="ios">Apple iOS</option>
                  </optgroup>
            </select>
            <br><br>
            <label for="subject">Select a subject:</label>
            <input type="text" id="city" name="city" list="subjects">
            <datalist id="subjects">
                  <option value="Algorithms">
                  <option value="Machine Learning">
                  <option value="Computer Vision">
                  <option value="Data Structures">
                  <option value="Theory of Computation">
            </datalist>
            <br><br>
            <input type="submit" value="Submit">
            <input type="reset" value="Reset">
      </form>
</body>
</html>
```

Output:


LabWork4_1
File | E:/CSE/5th%20sem/Web%20Programming/HTML/classwork/15.%20labwork_4.html
Gmail
YouTube
Maps
News
Translate

Personal Information:

Name:

Birth Date: dd - mm - yyyy

Profile Picture: Choose File No file chosen

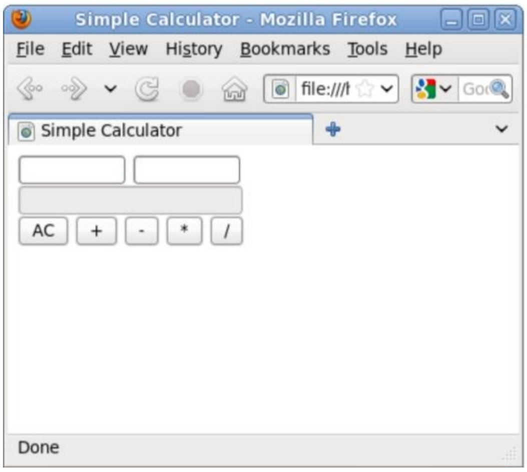
Favorite Color:

Select a browser: Windows

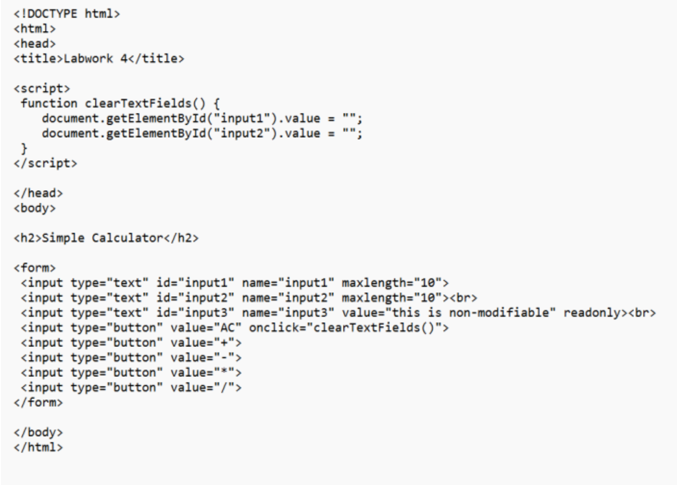
Select a subject:

Submit Reset

15. Create a Web Page to generate a form as shown below. The form should have the following features:
    - The first two boxes must not allow input of more than 10 characters.
    - The contents of the third text box should not be modifiable
    - Clicking on the 'AC' button should clear the contents of the first two text boxes.

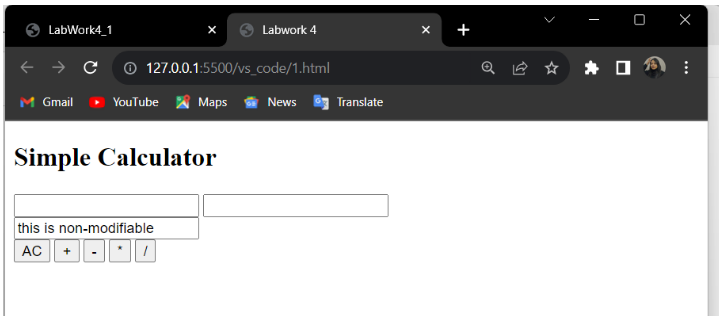

Simple Calculator - Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///t
Goo
Simple Calculator
AC
+
-
*
/
Done

Code:



```
<!DOCTYPE html>
<html>
<head>
<title>Labwork 4</title>

<script>
 function clearTextFields() {
    document.getElementById("input1").value = "";
    document.getElementById("input2").value = "";
 }
</script>

</head>
<body>

<h2>Simple Calculator</h2>

<form>
 <input type="text" id="input1" name="input1" maxlength="10">
 <input type="text" id="input2" name="input2" maxlength="10"><br>
 <input type="text" id="input3" name="input3" value="this is non-modifiable" readonly><br>
 <input type="button" value="AC" onclick="clearTextFields()">
 <input type="button" value="+">
 <input type="button" value="-">
 <input type="button" value="*">
 <input type="button" value="/">
</form>

</body>
</html>
```

Output:

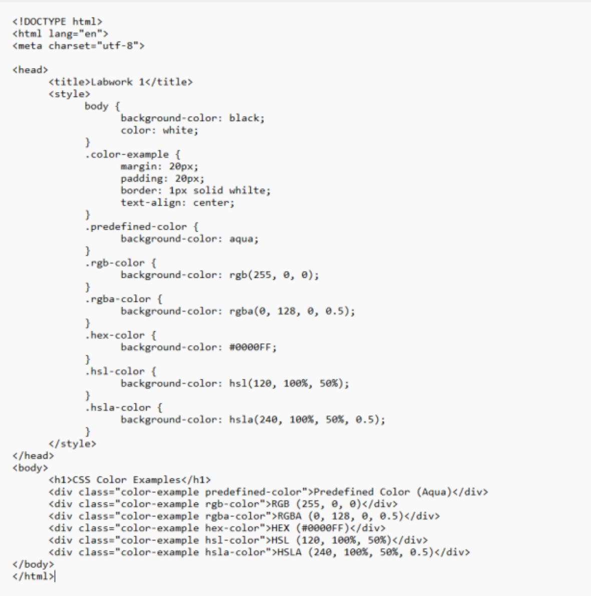


Simple Calculator

this is non-modifiable

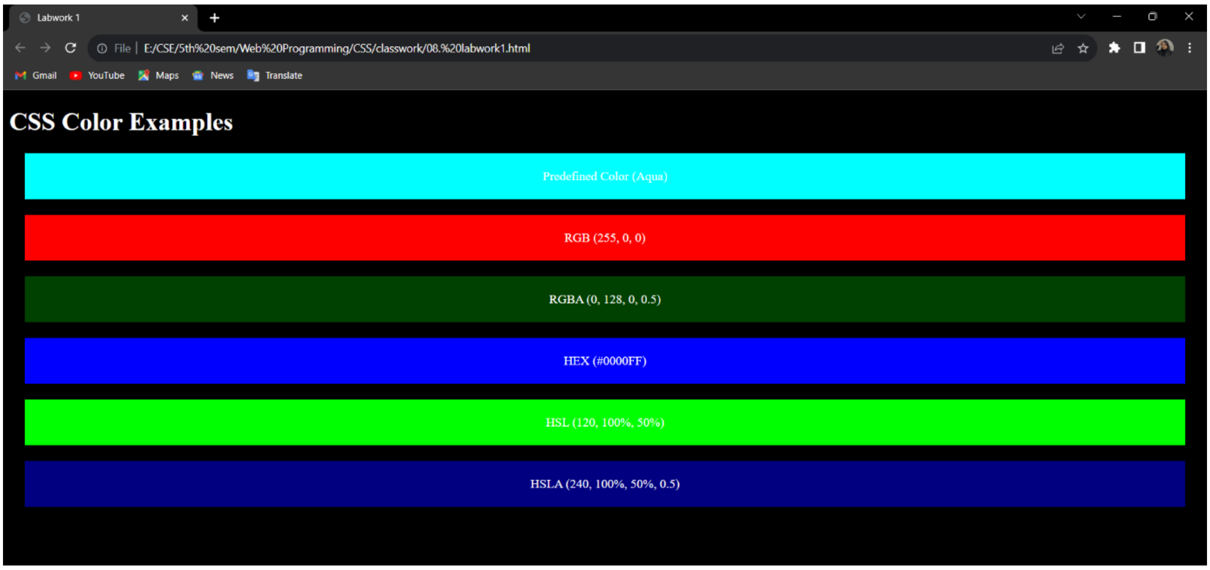
16. Create a web page to demonstrate different ways to specify colors in CSS (*Predefined color names, RGB, RGBA, Hex, HSL, HSLA*).

Code:



```
<!DOCTYPE html>
<html lang="en">
<meta charset="utf-8">

<head>
      <title>Labwork 1</title>
      <style>
            body {
                  background-color: black;
                  color: white;
            }
            .color-example {
                  margin: 20px;
                  padding: 20px;
                  border: 1px solid whilte;
                  text-align: center;
            }
            .predefined-color {
                  background-color: aqua;
            }
            .rgb-color {
                  background-color: rgb(255, 0, 0);
            }
            .rgba-color {
                  background-color: rgba(0, 128, 0, 0.5);
            }
            .hex-color {
                  background-color: #0000FF;
            }
            .hsl-color {
                  background-color: hsl(120, 100%, 50%);
            }
            .hsla-color {
                  background-color: hsla(240, 100%, 50%, 0.5);
            }
      </style>
</head>
<body>
      <h1>CSS Color Examples</h1>
      <div class="color-example predefined-color">Predefined Color (Aqua)</div>
      <div class="color-example rgb-color">RGB (255, 0, 0)</div>
      <div class="color-example rgba-color">RGBA (0, 128, 0, 0.5)</div>
      <div class="color-example hex-color">HEX (#0000FF)</div>
      <div class="color-example hsl-color">HSL (120, 100%, 50%)</div>
      <div class="color-example hsla-color">HSLA (240, 100%, 50%, 0.5)</div>
</body>
</html>
```

Output:



E:/CSE/5th%20sem/Web%20Programming/CSS/classwork/08.%20labwork1.html

CSS Color Examples

Predefined Color (Aqua)

RGB (255, 0, 0)

RGBA (0, 128, 0, 0.5)

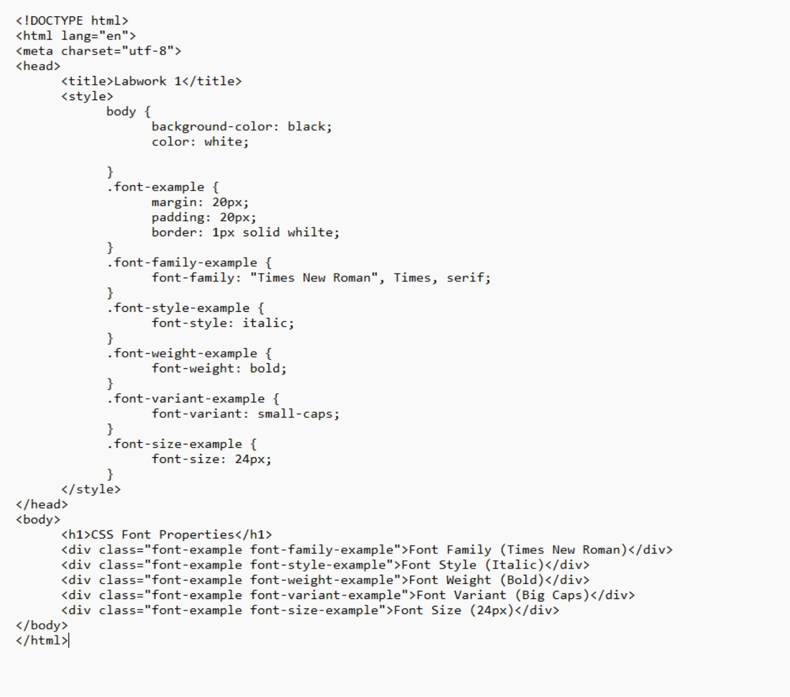
HEX (#0000FF)

HSL (120, 100%, 50%)

HSLA (240, 100%, 50%, 0.5)

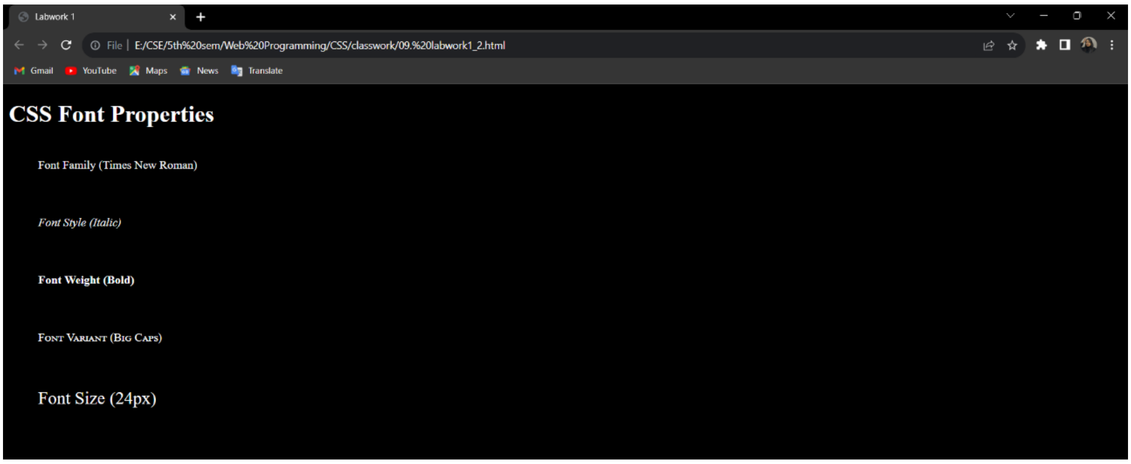
17. Create a web page to demonstrate some of the comon font properties in CSS (eg. `font-family, font-style, font-weight, font-variant, font-size)`.

Code:



```
<!DOCTYPE html>
<html lang="en">
<meta charset="utf-8">
<head>
      <title>Labwork 1</title>
      <style>
            body {
                  background-color: black;
                  color: white;

            }
            .font-example {
                  margin: 20px;
                  padding: 20px;
                  border: 1px solid whilte;
            }
            .font-family-example {
                  font-family: "Times New Roman", Times, serif;
            }
            .font-style-example {
                  font-style: italic;
            }
            .font-weight-example {
                  font-weight: bold;
            }
            .font-variant-example {
                  font-variant: small-caps;
            }
            .font-size-example {
                  font-size: 24px;
            }
      </style>
</head>
<body>
      <h1>CSS Font Properties</h1>
      <div class="font-example font-family-example">Font Family (Times New Roman)</div>
      <div class="font-example font-style-example">Font Style (Italic)</div>
      <div class="font-example font-weight-example">Font Weight (Bold)</div>
      <div class="font-example font-variant-example">Font Variant (Big Caps)</div>
      <div class="font-example font-size-example">Font Size (24px)</div>
</body>
</html>
```

Output:


Labwork 1
File | E:/CSE/5th%20sem/Web%20Programming/CSS/classwork/09.%20labwork1_2.html
Gmail
YouTube
Maps
News
Translate

CSS Font Properties

Font Family (Times New Roman)

Font Style (Italic)

Font Weight (Bold)

Font Variant (Big Caps)

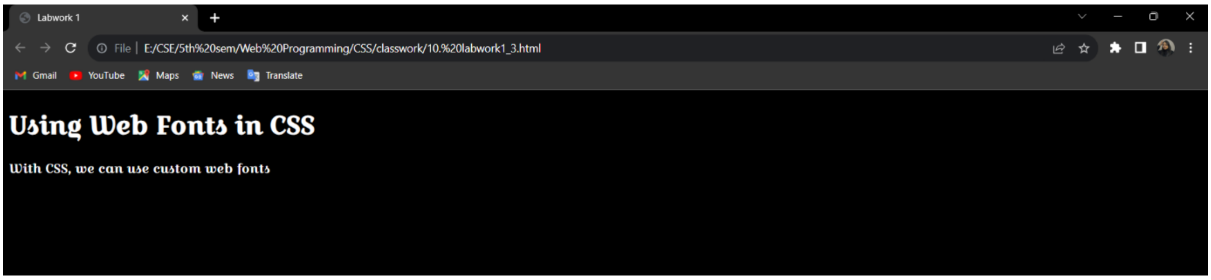
Font Size (24px)

18. Create a web page to demonstrate how to use *Web Fonts* in CSS.

Code:

```
<!DOCTYPE html>
<html lang="en">
<meta charset="utf-8">
<head>
      <title>Labwork 1</title>
      <style>
            body {
                  background-color: black;
                  color: white;
            }
            @font-face {
                  font-family: myFont;
                  src: url(CroissantOne-Regular.ttf);
            }
            * {
                  font-family: myFont;
            }
      </style>
</head>
<body>
      <h1>Using Web Fonts in CSS</h1>
      <div>With CSS, we can use custom web fonts</div>
</body>
</html>
```

Output:


Labwork 1
File | E:/CSE/5th%20sem/Web%20Programming/CSS/classwork/10.%20labwork1_3.html
Gmail
YouTube
Maps
News
Translate

Using Web Fonts in CSS

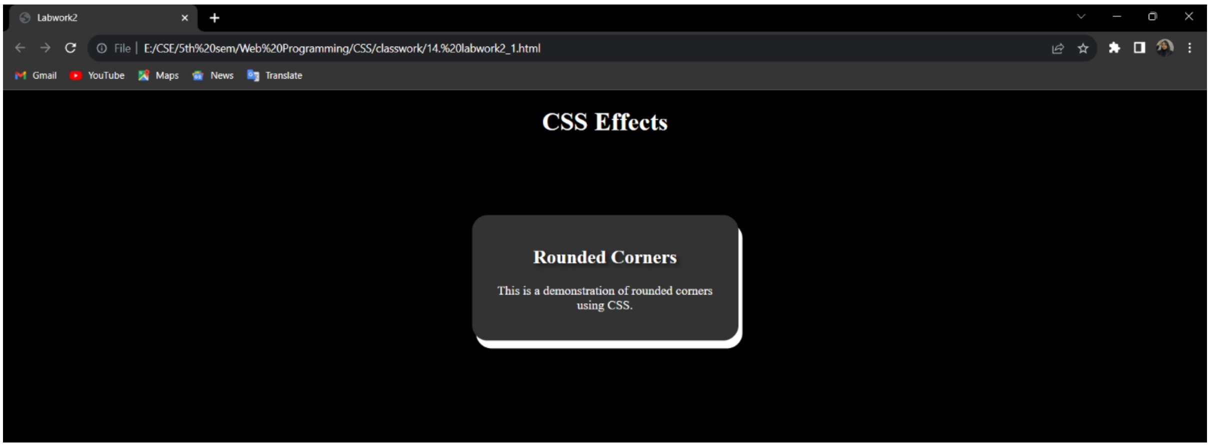
With CSS, we can use custom web fonts

19. Create a web page to demonstrate how to create *rounded corners*, *box shadow* and *text shadow* in CSS.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
      <title>Labwork2</title>
      <meta charset="UTF-8">
      <style>
            body {
                  background-color: black;
                  color: white;
                  text-align: center;
                  font-family: Times New Roman, sans-serif;
            }
            .container {
                  width: 300px;
                  margin: 100px auto;
                  padding: 20px;
                  background-color: rgba(255, 255, 255, 0.2);
                  border-radius: 20px;
                  box-shadow: 5px 10px;
            }
            h2 {
                  text-shadow: 3px 3px 5px rgba(0, 0, 0, 0.5);
            }
      </style>
</head>
<h1>CSS Effects</h1>
<div class="container">
      <h2>Rounded Corners</h2>
      <p>This is a demonstration of rounded corners using CSS.</p>
      <div class="rounded-box"></div>
</div>
</body>
</html>
```

Output:


Labwork2
File | E:/CSE/5th%20sem/Web%20Programming/CSS/classwork/14.%20labwork2_1.html
Gmail
YouTube
Maps
News
Translate

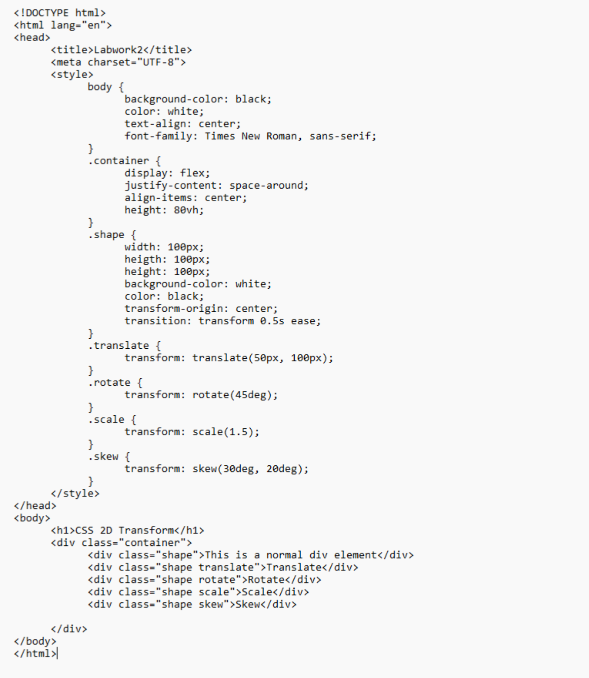
CSS Effects

Rounded Corners

This is a demonstration of rounded corners using CSS.

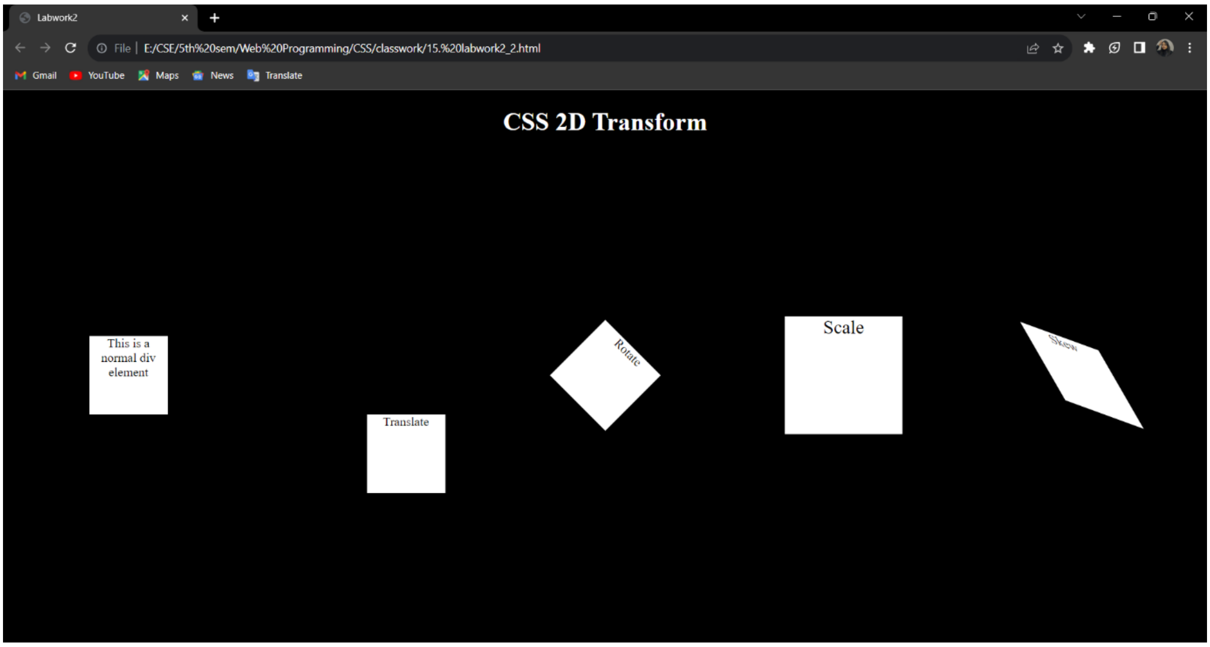
20. Create a webpage to demonstrate some of the 2-D *Transforms* in CSS (eg. `translate(),rotate(),scale(),skew()`).

Code:



```
<!DOCTYPE html>
<html lang="en">
<head>
      <title>Labwork2</title>
      <meta charset="UTF-8">
      <style>
            body {
                  background-color: black;
                  color: white;
                  text-align: center;
                  font-family: Times New Roman, sans-serif;
            }
            .container {
                  display: flex;
                  justify-content: space-around;
                  align-items: center;
                  height: 80vh;
            }
            .shape {
                  width: 100px;
                  heigth: 100px;
                  height: 100px;
                  background-color: white;
                  color: black;
                  transform-origin: center;
                  transition: transform 0.5s ease;
            }
            .translate {
                  transform: translate(50px, 100px);
            }
            .rotate {
                  transform: rotate(45deg);
            }
            .scale {
                  transform: scale(1.5);
            }
            .skew {
                  transform: skew(30deg, 20deg);
            }
      </style>
</head>
<body>
      <h1>CSS 2D Transform</h1>
      <div class="container">
            <div class="shape">This is a normal div element</div>
            <div class="shape translate">Translate</div>
            <div class="shape rotate">Rotate</div>
            <div class="shape scale">Scale</div>
            <div class="shape skew">Skew</div>

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

Output:



Labwork2
File | E:/CSE/5th%20sem/Web%20Programming/CSS/classwork/15.%20labwork2_2.html
Gmail
YouTube
Maps
News
Translate

CSS 2D Transform

This is a normal div element

Translate

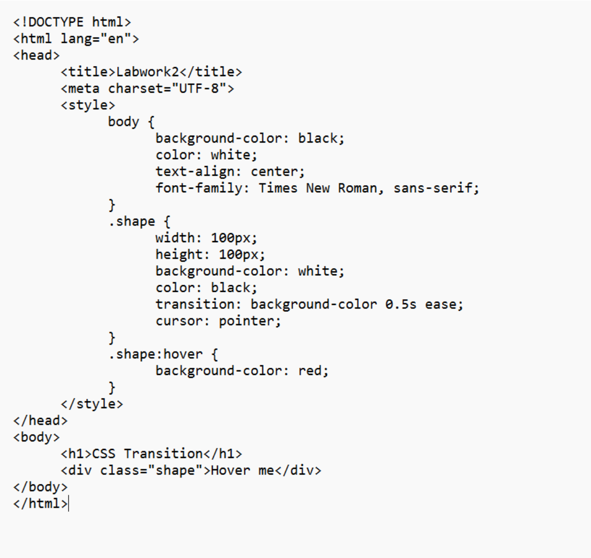
Rotate

Scale

Skew

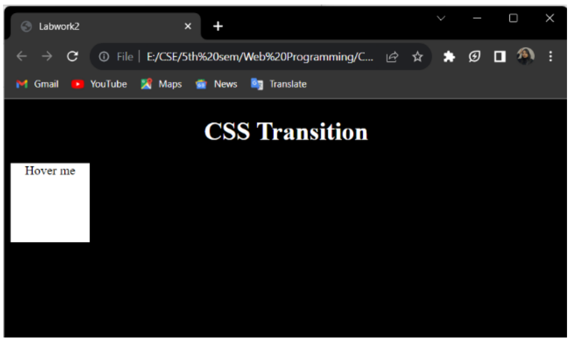
21. Create a web page to demonstrate transitions in CSS.

Code:



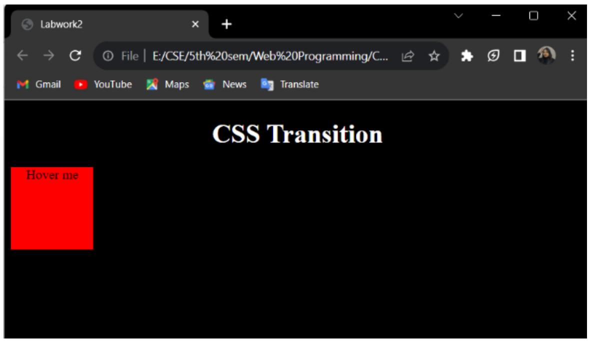
```
<!DOCTYPE html>
<html lang="en">
<head>
      <title>Labwork2</title>
      <meta charset="UTF-8">
      <style>
            body {
                  background-color: black;
                  color: white;
                  text-align: center;
                  font-family: Times New Roman, sans-serif;
            }
            .shape {
                  width: 100px;
                  height: 100px;
                  background-color: white;
                  color: black;
                  transition: background-color 0.5s ease;
                  cursor: pointer;
            }
            .shape:hover {
                  background-color: red;
            }
      </style>
</head>
<body>
      <h1>CSS Transition</h1>
      <div class="shape">Hover me</div>
</body>
</html>
```

Output:



CSS Transition

Hover me

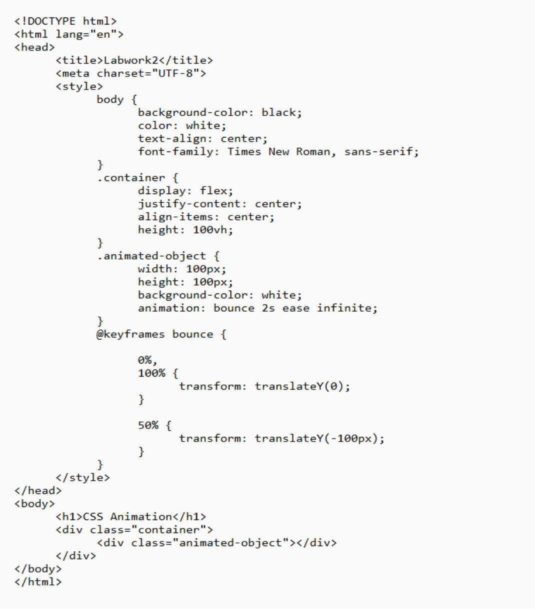


CSS Transition

Hover me

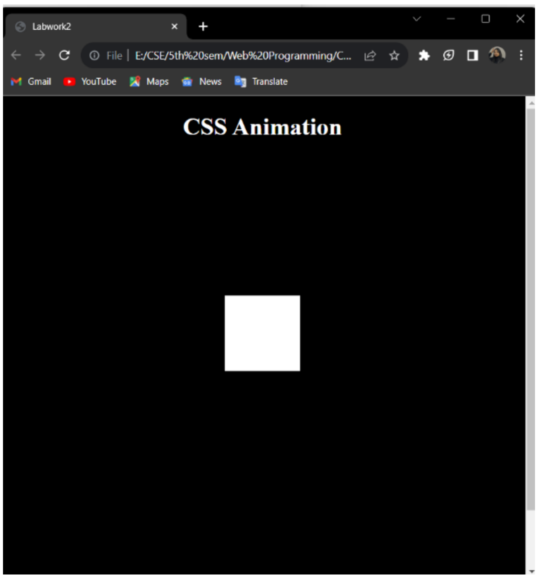
22. Create a web page to demonstrate how to *animate* objects using `@keyframes` in CSS.

Code:

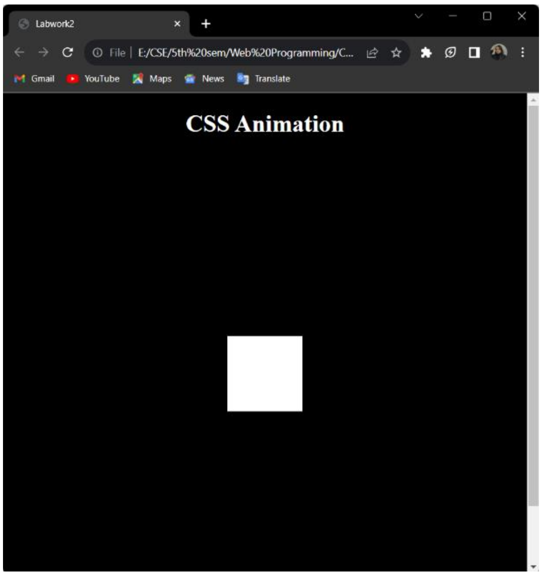


```
<!DOCTYPE html>
<html lang="en">
<head>
      <title>Labwork2</title>
      <meta charset="UTF-8">
      <style>
            body {
                  background-color: black;
                  color: white;
                  text-align: center;
                  font-family: Times New Roman, sans-serif;
            }
            .container {
                  display: flex;
                  justify-content: center;
                  align-items: center;
                  height: 100vh;
            }
            .animated-object {
                  width: 100px;
                  height: 100px;
                  background-color: white;
                  animation: bounce 2s ease infinite;
            }
            @keyframes bounce {

                  0%,
                  100% {
                        transform: translateY(0);
                  }

                  50% {
                        transform: translateY(-100px);
                  }
            }
      </style>
</head>
<body>
      <h1>CSS Animation</h1>
      <div class="container">
            <div class="animated-object"></div>
      </div>
</body>
</html>
```

Output:



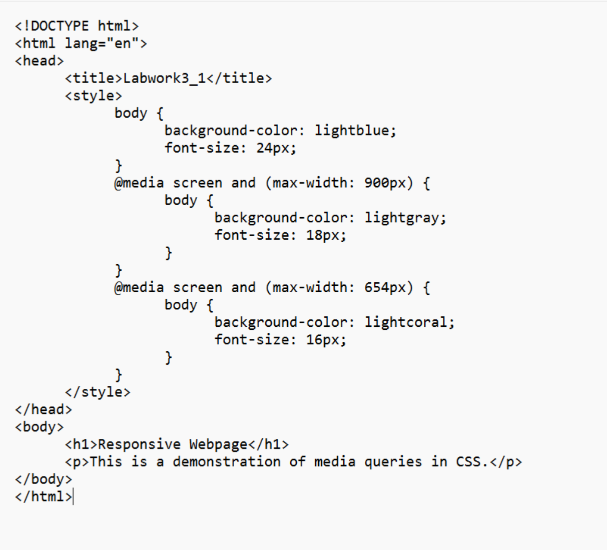
CSS Animation



CSS Animation

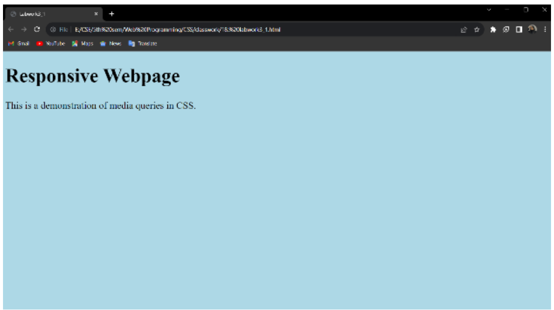
23. Create a web page to demonstrate the use of *Media Queries* in CSS.

Code:



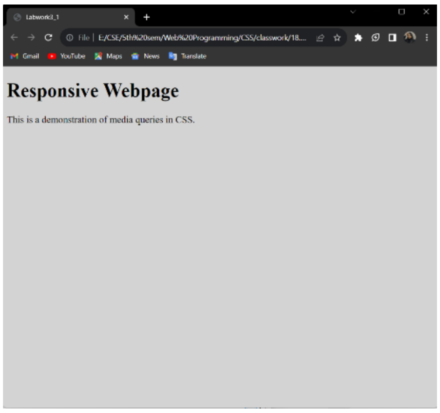
```
<!DOCTYPE html>
<html lang="en">
<head>
      <title>Labwork3_1</title>
      <style>
            body {
                  background-color: lightblue;
                  font-size: 24px;
            }
            @media screen and (max-width: 900px) {
                  body {
                        background-color: lightgray;
                        font-size: 18px;
                  }
            }
            @media screen and (max-width: 654px) {
                  body {
                        background-color: lightcoral;
                        font-size: 16px;
                  }
            }
      </style>
</head>
<body>
      <h1>Responsive Webpage</h1>
      <p>This is a demonstration of media queries in CSS.</p>
</body>
</html>
```

Output:



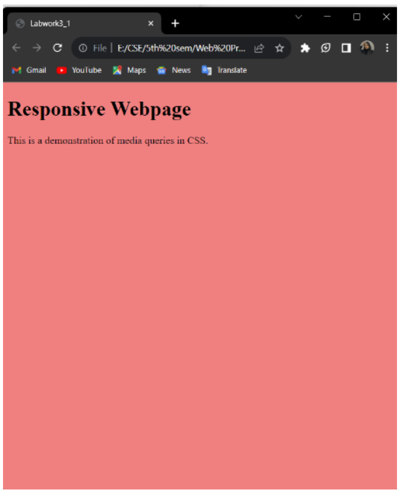
Responsive Webpage

This is a demonstration of media queries in CSS.


Labwork3_1
File | E:/CSE/5th%20sem/Web%20Programming/CSS/classwork/18...
Gmail
YouTube
Maps
News
Translate

Responsive Webpage

This is a demonstration of media queries in CSS.

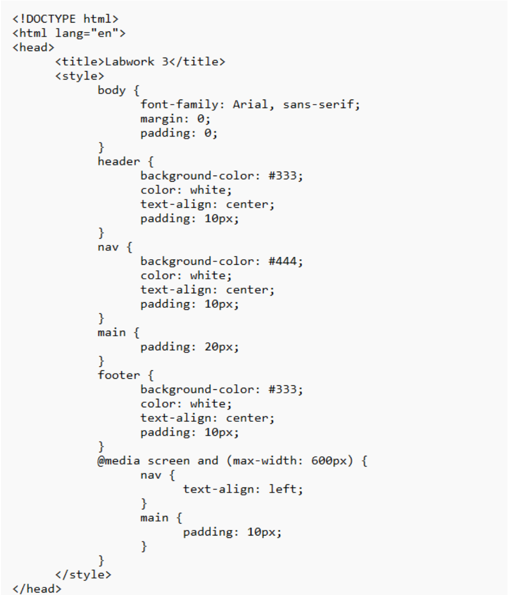


Responsive Webpage

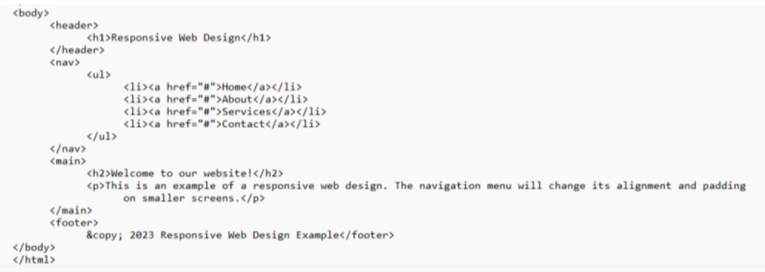
This is a demonstration of media queries in CSS.

24. Create a web page to demonstrate *Responsive Web Design* in CSS.

Code:

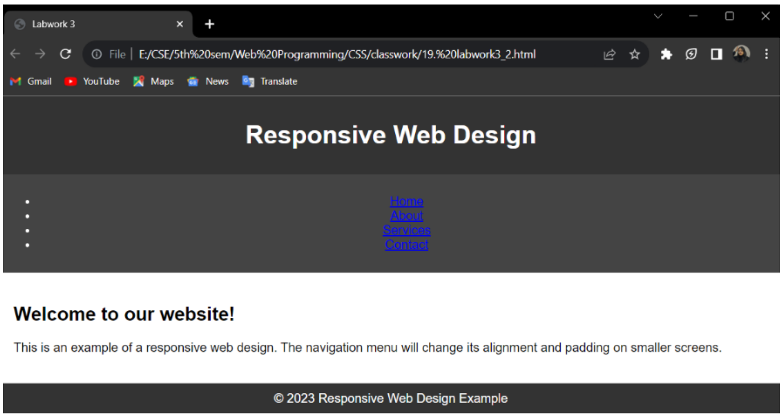


```
<!DOCTYPE html>
<html lang="en">
<head>
      <title>Labwork 3</title>
      <style>
            body {
                  font-family: Arial, sans-serif;
                  margin: 0;
                  padding: 0;
            }
            header {
                  background-color: #333;
                  color: white;
                  text-align: center;
                  padding: 10px;
            }
            nav {
                  background-color: #444;
                  color: white;
                  text-align: center;
                  padding: 10px;
            }
            main {
                  padding: 20px;
            }
            footer {
                  background-color: #333;
                  color: white;
                  text-align: center;
                  padding: 10px;
            }
            @media screen and (max-width: 600px) {
                  nav {
                        text-align: left;
                  }
                  main {
                        padding: 10px;
                  }
            }
      </style>
</head>
```



```
<body>
      <header>
            <h1>Responsive Web Design</h1>
      </header>
      <nav>
            <ul>
                  <li><a href="#">Home</a></li>
                  <li><a href="#">About</a></li>
                  <li><a href="#">Services</a></li>
                  <li><a href="#">Contact</a></li>
            </ul>
      </nav>
      <main>
            <h2>Welcome to our website!</h2>
            <p>This is an example of a responsive web design. The navigation menu will change its alignment and padding
                  on smaller screens.</p>
      </main>
      <footer>
            &copy; 2023 Responsive Web Design Example</footer>
</body>
</html>
```

Output:


Labwork 3
File | E:/CSE/5th%20sem/Web%20Programming/CSS/classwork/19.%20labwork3_2.html
Gmail
YouTube
Maps
News
Translate

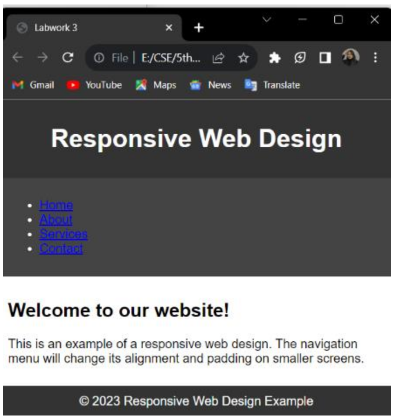
Responsive Web Design

- Home
- About
- Services
- Contact

Welcome to our website!

This is an example of a responsive web design. The navigation menu will change its alignment and padding on smaller screens.

© 2023 Responsive Web Design Example



Responsive Web Design

- Home
- About
- Services
- Contact

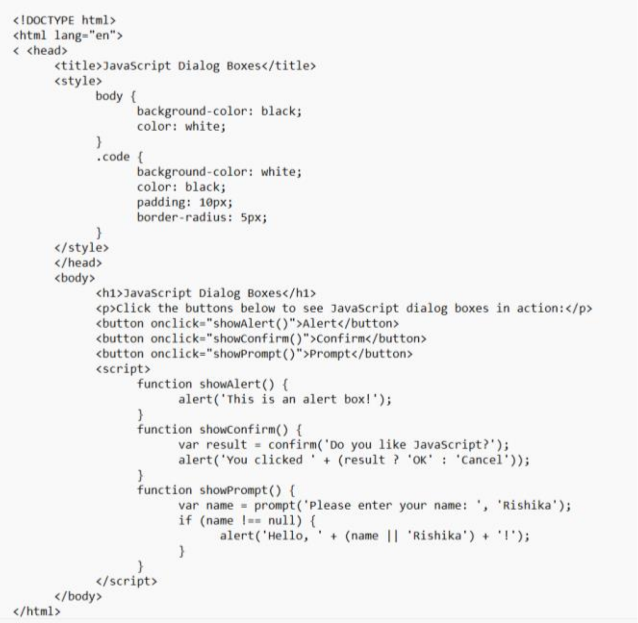
Welcome to our website!

This is an example of a responsive web design. The navigation menu will change its alignment and padding on smaller screens.

© 2023 Responsive Web Design Example

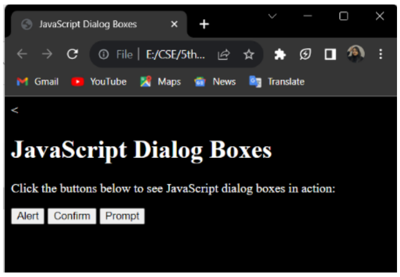
25. Create a web page to demonstrate the use of JavaScript `alert()`, `confirm()`, and `prompt()`.

Code:



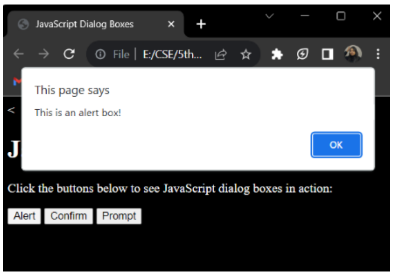
```
<!DOCTYPE html>
<html lang="en">
< <head>
      <title>JavaScript Dialog Boxes</title>
      <style>
            body {
                  background-color: black;
                  color: white;
            }
            .code {
                  background-color: white;
                  color: black;
                  padding: 10px;
                  border-radius: 5px;
            }
      </style>
      </head>
      <body>
            <h1>JavaScript Dialog Boxes</h1>
            <p>Click the buttons below to see JavaScript dialog boxes in action:</p>
            <button onclick="showAlert()">Alert</button>
            <button onclick="showConfirm()">Confirm</button>
            <button onclick="showPrompt()">Prompt</button>
            <script>
                  function showAlert() {
                        alert('This is an alert box!');
                  }
                  function showConfirm() {
                        var result = confirm('Do you like JavaScript?');
                        alert('You clicked ' + (result ? 'OK' : 'Cancel'));
                  }
                  function showPrompt() {
                        var name = prompt('Please enter your name: ', 'Rishika');
                        if (name !== null) {
                              alert('Hello, ' + (name || 'Rishika') + '!');
                        }
                  }
            </script>
      </body>
</html>
```

Output:



JavaScript Dialog Boxes

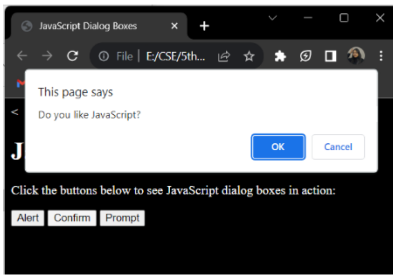
Click the buttons below to see JavaScript dialog boxes in action:



This page says

This is an alert box!

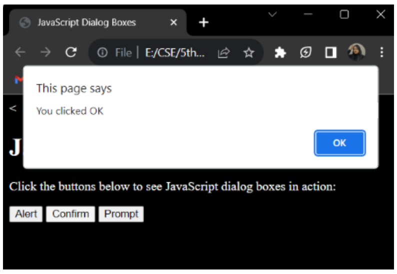
Click the buttons below to see JavaScript dialog boxes in action:



This page says

Do you like JavaScript?

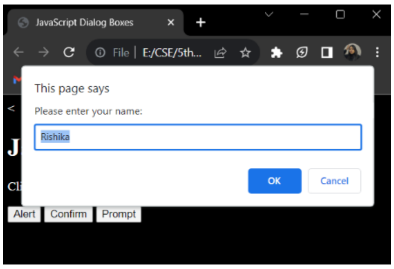
Click the buttons below to see JavaScript dialog boxes in action:



This page says

You clicked OK

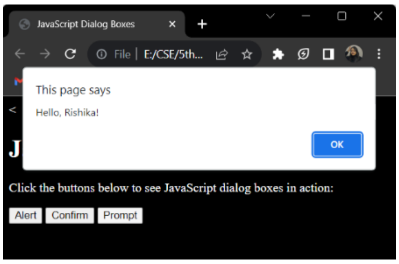
Click the buttons below to see JavaScript dialog boxes in action:



This page says

Please enter your name:

Rishika



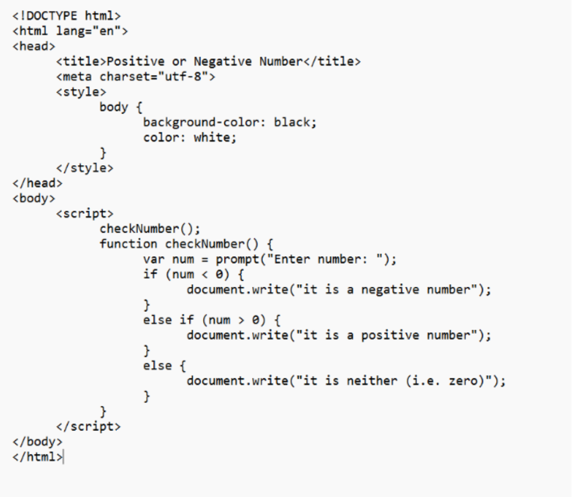
This page says

Hello, Rishika!

Click the buttons below to see JavaScript dialog boxes in action:

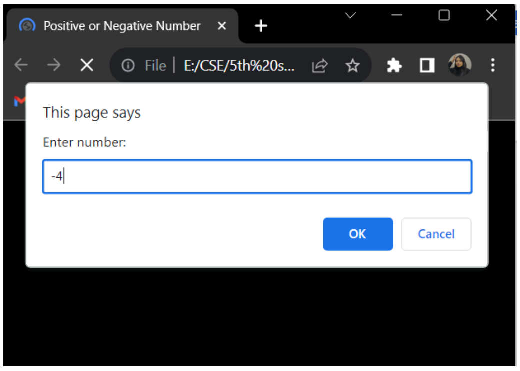
26. Using JavaScript, check if a given integer is positive, negative, or neither (i.e.zero).

Code:



```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Positive or Negative Number</title>
    <meta charset="utf-8">
    <style>
        body {
            background-color: black;
            color: white;
        }
    </style>
</head>
<body>
    <script>
        checkNumber();
        function checkNumber() {
            var num = prompt("Enter number: ");
            if (num < 0) {
                document.write("it is a negative number");
            }
            else if (num > 0) {
                document.write("it is a positive number");
            }
            else {
                document.write("it is neither (i.e. zero)");
            }
        }
    </script>
</body>
</html>
```

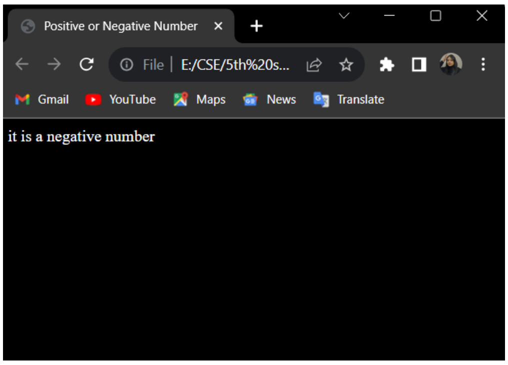
Output:



This page says

Enter number:

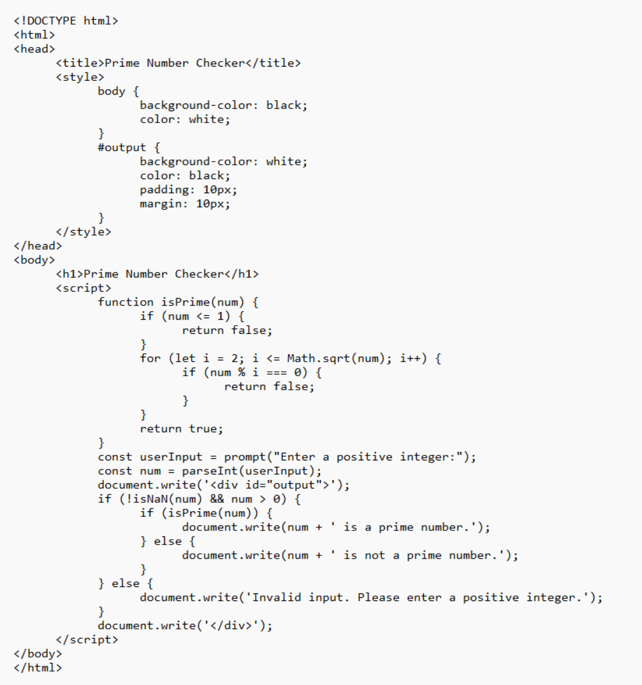
-4



```
it is a negative number
```

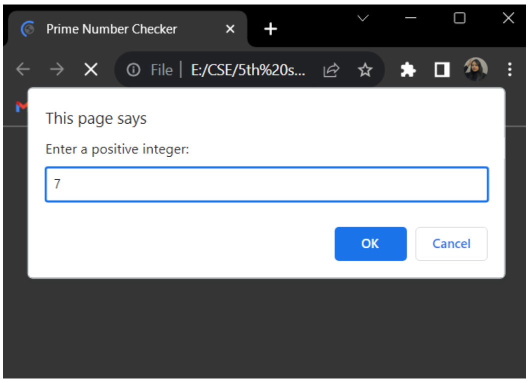
27. Using JavaScript, check if a given positive integer is prime or not.

Code:



```
<!DOCTYPE html>
<html>
<head>
    <title>Prime Number Checker</title>
    <style>
        body {
            background-color: black;
            color: white;
        }
        #output {
            background-color: white;
            color: black;
            padding: 10px;
            margin: 10px;
        }
    </style>
</head>
<body>
    <h1>Prime Number Checker</h1>
    <script>
        function isPrime(num) {
            if (num <= 1) {
                return false;
            }
            for (let i = 2; i <= Math.sqrt(num); i++) {
                if (num % i === 0) {
                    return false;
                }
            }
            return true;
        }
        const userInput = prompt("Enter a positive integer:");
        const num = parseInt(userInput);
        document.write('<div id="output">');
        if (!isNaN(num) && num > 0) {
            if (isPrime(num)) {
                document.write(num + ' is a prime number.');
            } else {
                document.write(num + ' is not a prime number.');
            }
        } else {
            document.write('Invalid input. Please enter a positive integer.');
        }
        document.write('</div>');
    </script>
</body>
</html>
```

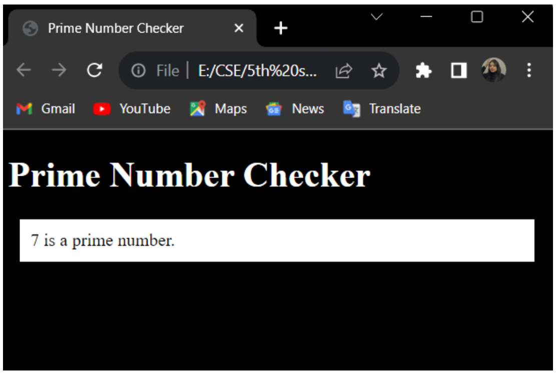
Output:



This page says

Enter a positive integer:

7

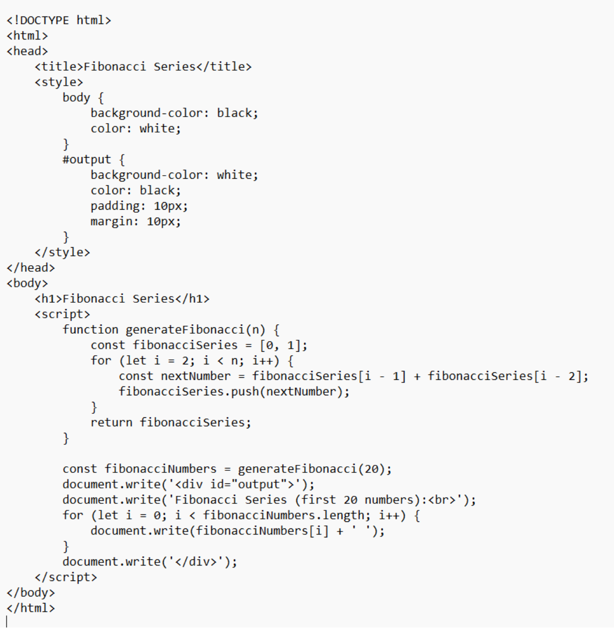


Prime Number Checker

7 is a prime number.

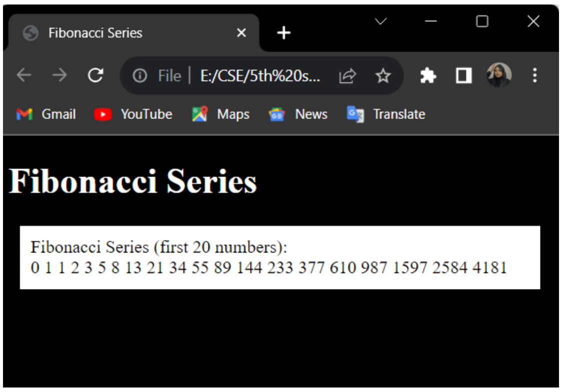
28. Using JavaScript, calculate and print the first 20 numbers of the Fibonacci series.

Code:



```
<!DOCTYPE html>
<html>
<head>
    <title>Fibonacci Series</title>
    <style>
        body {
            background-color: black;
            color: white;
        }
        #output {
            background-color: white;
            color: black;
            padding: 10px;
            margin: 10px;
        }
    </style>
</head>
<body>
    <h1>Fibonacci Series</h1>
    <script>
        function generateFibonacci(n) {
            const fibonacciSeries = [0, 1];
            for (let i = 2; i < n; i++) {
                const nextNumber = fibonacciSeries[i - 1] + fibonacciSeries[i - 2];
                fibonacciSeries.push(nextNumber);
            }
            return fibonacciSeries;
        }

        const fibonacciNumbers = generateFibonacci(20);
        document.write('<div id="output">');
        document.write('Fibonacci Series (first 20 numbers):<br>');
        for (let i = 0; i < fibonacciNumbers.length; i++) {
            document.write(fibonacciNumbers[i] + ' ');
        }
        document.write('</div>');
    </script>
</body>
</html>
```

Output:

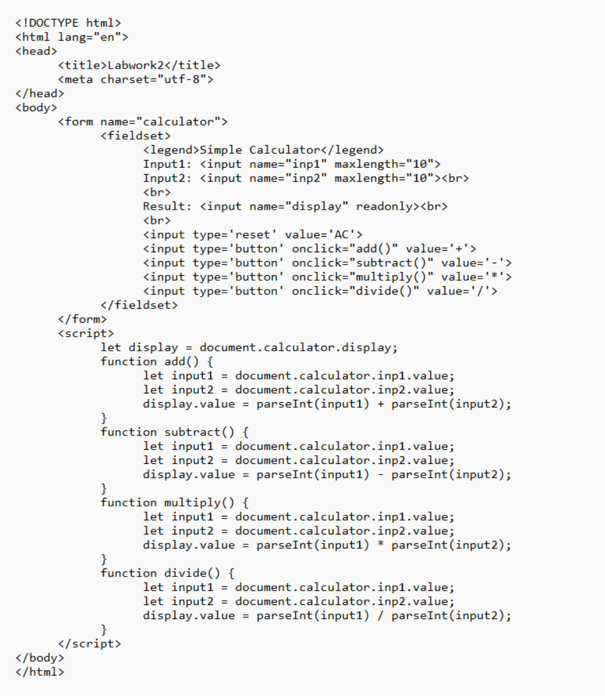


Fibonacci Series

```
Fibonacci Series (first 20 numbers):
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181
```

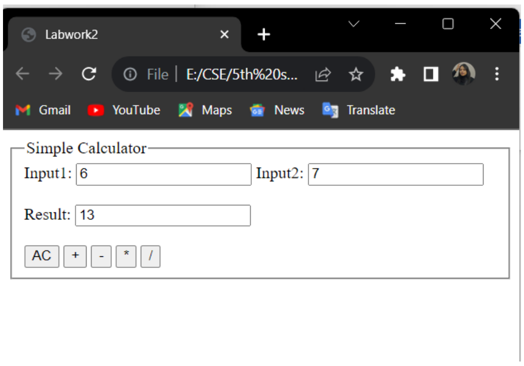
29. Add functionality to the ‘Simple Calculator’ created in Day 5, so that the arithmatic operations can be carried out. You may also add some styling so that the numbers and the components are properly aligned, and are of proper size.

Code:



```
<!DOCTYPE html>
<html lang="en">
<head>
      <title>Labwork2</title>
      <meta charset="utf-8">
</head>
<body>
      <form name="calculator">
            <fieldset>
                  <legend>Simple Calculator</legend>
                  Input1: <input name="inp1" maxlength="10">
                  Input2: <input name="inp2" maxlength="10"><br>
                  <br>
                  Result: <input name="display" readonly><br>
                  <br>
                  <input type='reset' value='AC'>
                  <input type='button' onclick="add()" value='+'>
                  <input type='button' onclick="subtract()" value='-'>
                  <input type='button' onclick="multiply()" value='*'>
                  <input type='button' onclick="divide()" value='/'>
            </fieldset>
      </form>
      <script>
            let display = document.calculator.display;
            function add() {
                  let input1 = document.calculator.inp1.value;
                  let input2 = document.calculator.inp2.value;
                  display.value = parseInt(input1) + parseInt(input2);
            }
            function subtract() {
                  let input1 = document.calculator.inp1.value;
                  let input2 = document.calculator.inp2.value;
                  display.value = parseInt(input1) - parseInt(input2);
            }
            function multiply() {
                  let input1 = document.calculator.inp1.value;
                  let input2 = document.calculator.inp2.value;
                  display.value = parseInt(input1) * parseInt(input2);
            }
            function divide() {
                  let input1 = document.calculator.inp1.value;
                  let input2 = document.calculator.inp2.value;
                  display.value = parseInt(input1) / parseInt(input2);
            }
      </script>
</body>
</html>
```

Output:



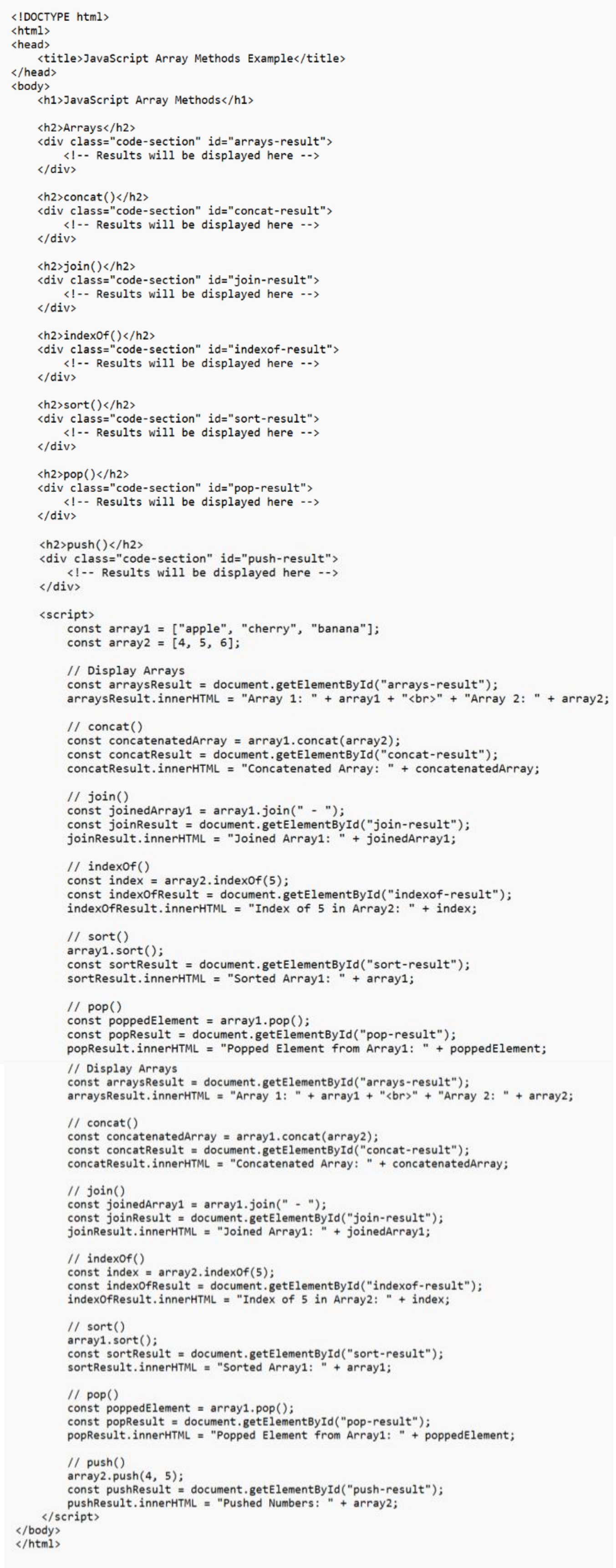
Simple Calculator

Input1: 6 Input2: 7

Result: 13

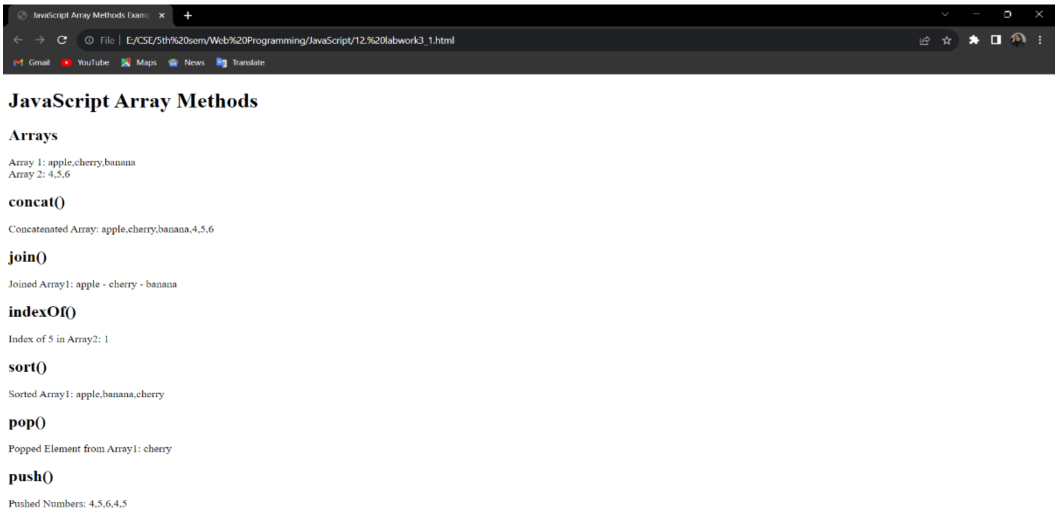
30. Create a web page to demonstrate some of the methods of JavaScript arrays (eg. `concat()`, `join()`, `indexOf()`, `sort()`, `pop()`, `push()` etc.)

Code:



```
<!DOCTYPE html>
<html>
<head>
    <title>JavaScript Array Methods Example</title>
</head>
<body>
    <h1>JavaScript Array Methods</h1>

    <h2>Arrays</h2>
    <div class="code-section" id="arrays-result">
        <!-- Results will be displayed here -->
    </div>

    <h2>concat()</h2>
    <div class="code-section" id="concat-result">
        <!-- Results will be displayed here -->
    </div>

    <h2>join()</h2>
    <div class="code-section" id="join-result">
        <!-- Results will be displayed here -->
    </div>

    <h2>indexOf()</h2>
    <div class="code-section" id="indexof-result">
        <!-- Results will be displayed here -->
    </div>

    <h2>sort()</h2>
    <div class="code-section" id="sort-result">
        <!-- Results will be displayed here -->
    </div>

    <h2>pop()</h2>
    <div class="code-section" id="pop-result">
        <!-- Results will be displayed here -->
    </div>

    <h2>push()</h2>
    <div class="code-section" id="push-result">
        <!-- Results will be displayed here -->
    </div>

    <script>
        const array1 = ["apple", "cherry", "banana"];
        const array2 = [4, 5, 6];

        // Display Arrays
        const arraysResult = document.getElementById("arrays-result");
        arraysResult.innerHTML = "Array 1: " + array1 + "<br>" + "Array 2: " + array2;

        // concat()
        const concatenatedArray = array1.concat(array2);
        const concatResult = document.getElementById("concat-result");
        concatResult.innerHTML = "Concatenated Array: " + concatenatedArray;

        // join()
        const joinedArray1 = array1.join(" - ");
        const joinResult = document.getElementById("join-result");
        joinResult.innerHTML = "Joined Array1: " + joinedArray1;

        // indexOf()
        const index = array2.indexOf(5);
        const indexOfResult = document.getElementById("indexof-result");
        indexOfResult.innerHTML = "Index of 5 in Array2: " + index;

        // sort()
        array1.sort();
        const sortResult = document.getElementById("sort-result");
        sortResult.innerHTML = "Sorted Array1: " + array1;

        // pop()
        const poppedElement = array1.pop();
        const popResult = document.getElementById("pop-result");
        popResult.innerHTML = "Popped Element from Array1: " + poppedElement;
        // Display Arrays
        const arraysResult = document.getElementById("arrays-result");
        arraysResult.innerHTML = "Array 1: " + array1 + "<br>" + "Array 2: " + array2;

        // concat()
        const concatenatedArray = array1.concat(array2);
        const concatResult = document.getElementById("concat-result");
        concatResult.innerHTML = "Concatenated Array: " + concatenatedArray;

        // join()
        const joinedArray1 = array1.join(" - ");
        const joinResult = document.getElementById("join-result");
        joinResult.innerHTML = "Joined Array1: " + joinedArray1;

        // indexOf()
        const index = array2.indexOf(5);
        const indexOfResult = document.getElementById("indexof-result");
        indexOfResult.innerHTML = "Index of 5 in Array2: " + index;

        // sort()
        array1.sort();
        const sortResult = document.getElementById("sort-result");
        sortResult.innerHTML = "Sorted Array1: " + array1;

        // pop()
        const poppedElement = array1.pop();
        const popResult = document.getElementById("pop-result");
        popResult.innerHTML = "Popped Element from Array1: " + poppedElement;

        // push()
        array2.push(4, 5);
        const pushResult = document.getElementById("push-result");
        pushResult.innerHTML = "Pushed Numbers: " + array2;
    </script>
</body>
</html>
```

Output:


JavaScript Array Methods Examp
File | E:/CSE/5th%20sem/Web%20Programming/JavaScript/12.%20labwork3_1.html
Gmail
YouTube
Maps
News
Translate

JavaScript Array Methods

Arrays

Array 1: apple,cherry,banana
Array 2: 4,5,6

concat()

Concatenated Array: apple,cherry,banana,4,5,6

join()

Joined Array1: apple - cherry - banana

indexOf()

Index of 5 in Array2: 1

sort()

Sorted Array1: apple,banana,cherry

pop()

Popped Element from Array1: cherry

push()

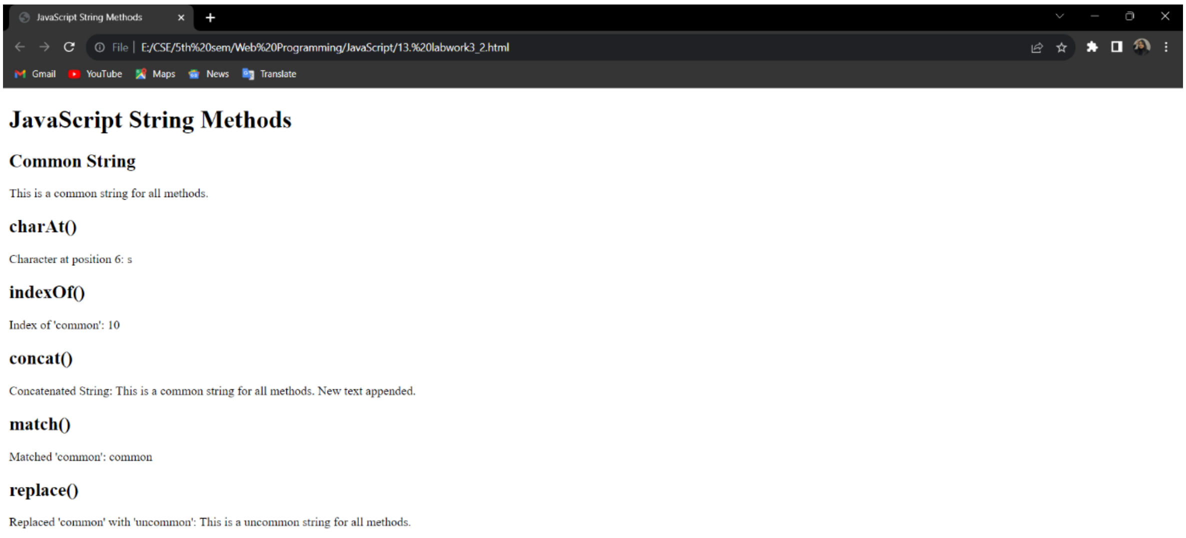
Pushed Numbers: 4,5,6,4,5

31. Create a web page to demonstrate some of the methods of JavaScript strings (eg. `charAt(), indexOf(), concat(), match(), replace()` etc.)

Code:

```
<!DOCTYPE html>
<html>
<head>
    <title>JavaScript String Methods</title>
</head>
<body>
    <h1>JavaScript String Methods</h1>

    <h2>Common String</h2>
    <div class="code-section" id="string-result">
        <!-- Common string will be displayed here -->
    </div>

    <h2>charAt()</h2>
    <div class="code-section" id="charat-result">
        <!-- Result will be displayed here -->
    </div>

    <h2>indexOf()</h2>
    <div class="code-section" id="indexof-result">
        <!-- Result will be displayed here -->
    </div>

    <h2>concat()</h2>
    <div class="code-section" id="concat-result">
        <!-- Result will be displayed here -->
    </div>

    <h2>match()</h2>
    <div class="code-section" id="match-result">
        <!-- Result will be displayed here -->
    </div>

    <h2>replace()</h2>
    <div class="code-section" id="replace-result">
        <!-- Result will be displayed here -->
    </div>
    <script>
        // Common string for all methods
        const commonString = "This is a common string for all methods.";

        // Display the common string
        const stringResult = document.getElementById("string-result");
        stringResult.textContent = commonString;

        // charAt()
        const charAtResult = document.getElementById("charat-result");
        charAtResult.textContent = `Character at position 6: ${commonString.charAt(6)}`;

        // indexOf()
        const indexOfResult = document.getElementById("indexof-result");
        indexOfResult.textContent = `Index of 'common': ${commonString.indexOf("common")}`;

        // concat()
        const concatResult = document.getElementById("concat-result");
        concatResult.textContent = `Concatenated String: ${commonString.concat(" New text appended.")}`;

        // match()
        const matchResult = document.getElementById("match-result");
        const match = commonString.match(/common/);
        matchResult.textContent = `Matched 'common': ${match[0]}`;

        // replace()
        const replaceResult = document.getElementById("replace-result");
        const replacedString = commonString.replace("common", "uncommon");
        replaceResult.textContent = `Replaced 'common' with 'uncommon': ${replacedString}`;
    </script>
</body>
</html>
```

Output:


JavaScript String Methods
File | E:/CSE/5th%20sem/Web%20Programming/JavaScript/13.%20labwork3_2.html
Gmail
YouTube
Maps
News
Translate

JavaScript String Methods

Common String

This is a common string for all methods.

charAt()

Character at position 6: s

indexOf()

Index of 'common': 10

concat()

Concatenated String: This is a common string for all methods. New text appended.

match()

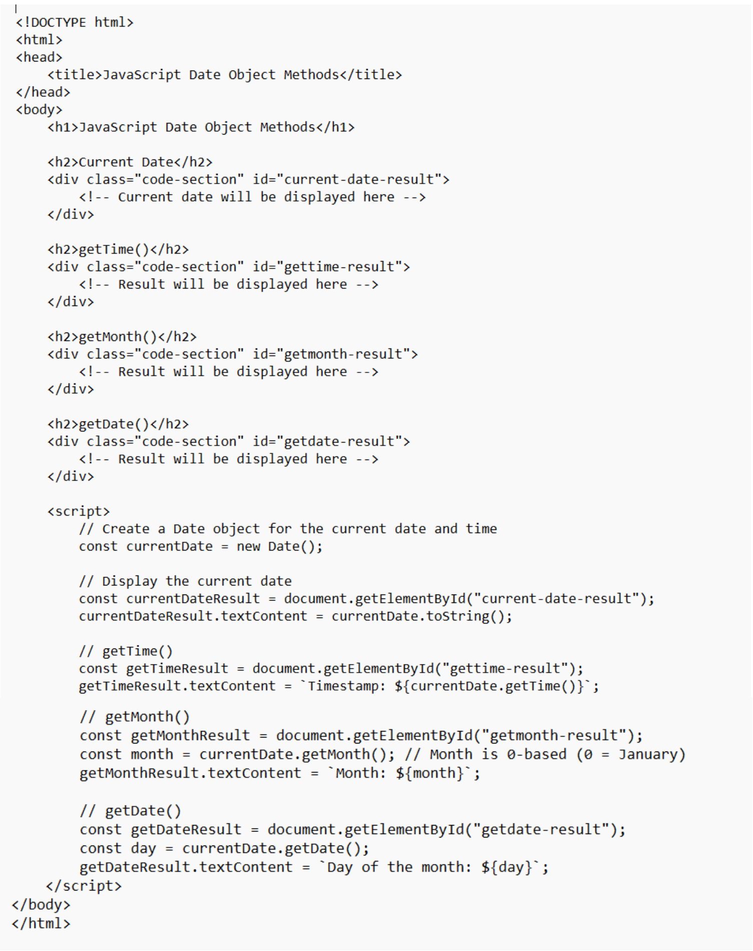
Matched 'common': common

replace()

Replaced 'common' with 'uncommon': This is a uncommon string for all methods.

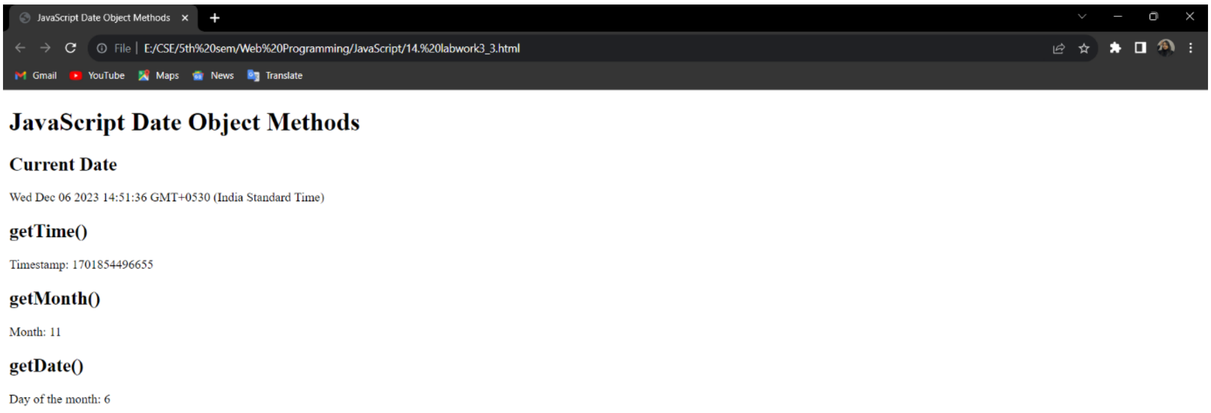
32. Create a web page to demonstrate some of the methods of the Date object of JavaScript (eg. `getTime()`, `getMonth()`, `getDate()`, etc.)

Code:



```
<!DOCTYPE html>
<html>
<head>
    <title>JavaScript Date Object Methods</title>
</head>
<body>
    <h1>JavaScript Date Object Methods</h1>

    <h2>Current Date</h2>
    <div class="code-section" id="current-date-result">
        <!-- Current date will be displayed here -->
    </div>

    <h2>getTime()</h2>
    <div class="code-section" id="gettime-result">
        <!-- Result will be displayed here -->
    </div>

    <h2>getMonth()</h2>
    <div class="code-section" id="getmonth-result">
        <!-- Result will be displayed here -->
    </div>

    <h2>getDate()</h2>
    <div class="code-section" id="getdate-result">
        <!-- Result will be displayed here -->
    </div>

    <script>
        // Create a Date object for the current date and time
        const currentDate = new Date();

        // Display the current date
        const currentDateResult = document.getElementById("current-date-result");
        currentDateResult.textContent = currentDate.toString();

        // getTime()
        const getTimeResult = document.getElementById("gettime-result");
        getTimeResult.textContent = `Timestamp: ${currentDate.getTime()}`;

        // getMonth()
        const getMonthResult = document.getElementById("getmonth-result");
        const month = currentDate.getMonth(); // Month is 0-based (0 = January)
        getMonthResult.textContent = `Month: ${month}`;

        // getDate()
        const getDateResult = document.getElementById("getdate-result");
        const day = currentDate.getDate();
        getDateResult.textContent = `Day of the month: ${day}`;
    </script>
</body>
</html>
```

Output:


JavaScript Date Object Methods
File | E:/CSE/5th%20sem/Web%20Programming/JavaScript/14.%20labwork3_3.html
Gmail
YouTube
Maps
News
Translate

JavaScript Date Object Methods

Current Date

Wed Dec 06 2023 14:51:36 GMT+0530 (India Standard Time)

getTime()

Timestamp: 1701854496655

getMonth()

Month: 11

getDate()

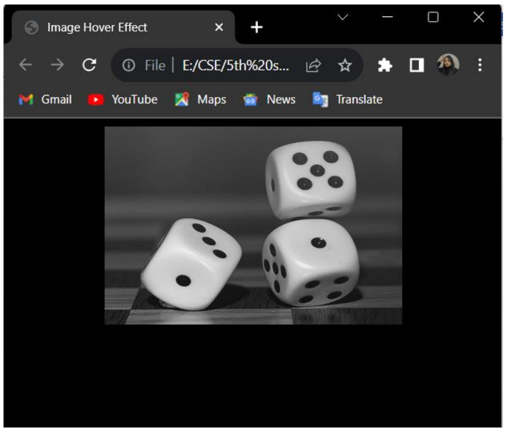
Day of the month: 6

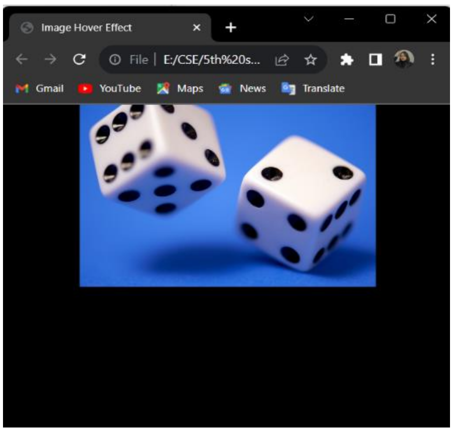
33. Create a web page containing an image. When the user moves the mouse pointer over the image, the image should change to a new image. When the mouse pointer is moved away from the image, the original image should be shown. Use JavaScript for the implementation.

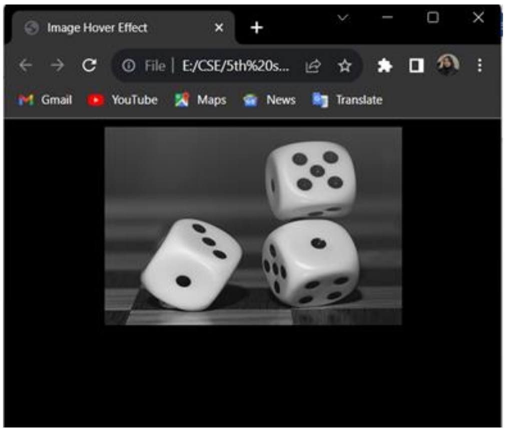
Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Image Hover Effect</title>
    <style>
        body {
            background-color: black;
            text-align: center;
        }

        img {
            width: 300px;
            height: 200px;
            transition: transform 0.3s;
        }

        img:hover {
            transform: scale(1.1);
        }
    </style>
</head>
<body>
    <img src="image1.png" alt="Original Image" onmouseover="this.src='image2.jpeg'" onmouseout="this.src='image1.png'">

</body>
</html>
```

Output:

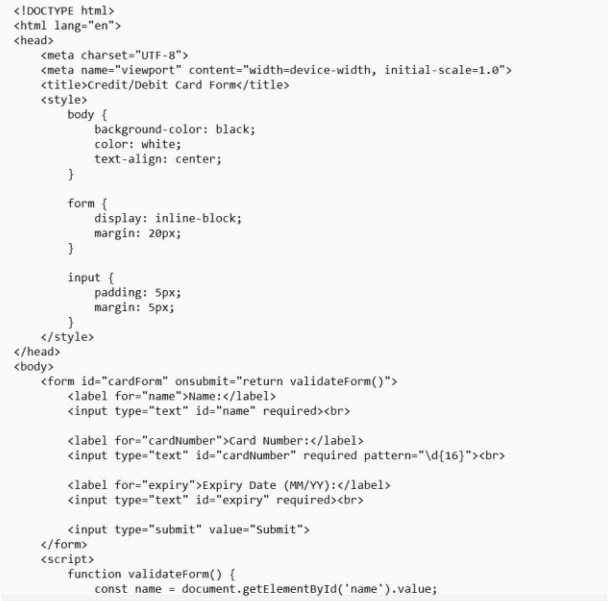




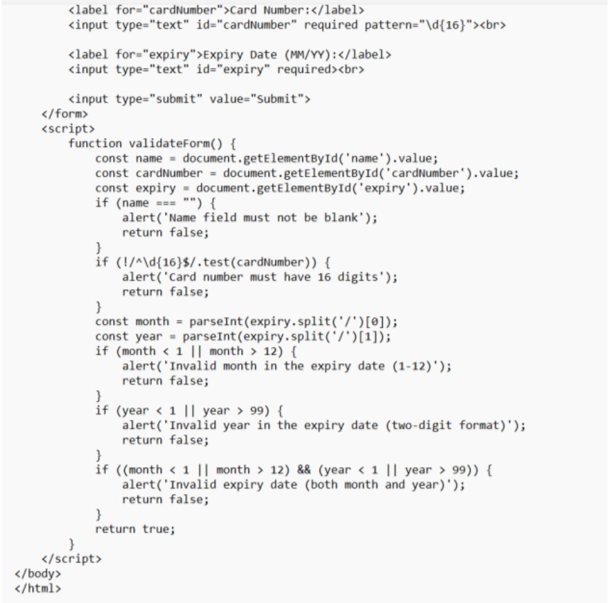


34. Create a Web Page to do form validation using JavaScript. Create a form for entering credit/debit card details. It should have a name field, card number field, and an expiry date (month & year). Use JavaScript to do the following:
    - The name field must not be blank
    - The card number field must have 16 digits
    - The month in the expiry date field must only take values from 1to 12.
    - The year in the expiry date field must only accept two digits.

Code:



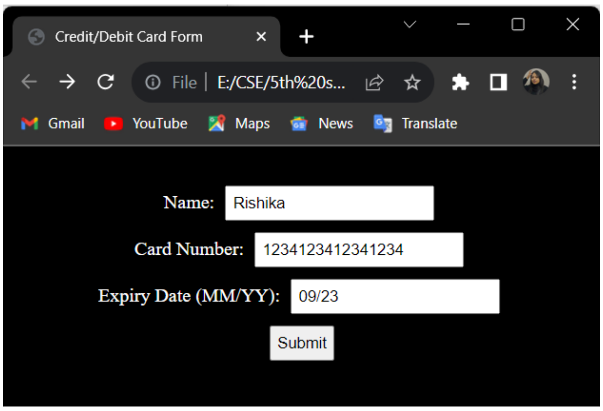
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Credit/Debit Card Form</title>
    <style>
        body {
            background-color: black;
            color: white;
            text-align: center;
        }

        form {
            display: inline-block;
            margin: 20px;
        }

        input {
            padding: 5px;
            margin: 5px;
        }
    </style>
</head>
<body>
    <form id="cardForm" onsubmit="return validateForm()">
        <label for="name">Name:</label>
        <input type="text" id="name" required><br>

        <label for="cardNumber">Card Number:</label>
        <input type="text" id="cardNumber" required pattern="\d{16}"><br>

        <label for="expiry">Expiry Date (MM/YY):</label>
        <input type="text" id="expiry" required><br>

        <input type="submit" value="Submit">
    </form>
    <script>
        function validateForm() {
            const name = document.getElementById('name').value;
```



```
        <label for="cardNumber">Card Number:</label>
        <input type="text" id="cardNumber" required pattern="\d{16}"><br>

        <label for="expiry">Expiry Date (MM/YY):</label>
        <input type="text" id="expiry" required><br>

        <input type="submit" value="Submit">
    </form>
    <script>
        function validateForm() {
            const name = document.getElementById('name').value;
```

```
            const cardNumber = document.getElementById('cardNumber').value;
            const expiry = document.getElementById('expiry').value;
            if (name === "") {
                alert('Name field must not be blank');
                return false;
            }
            if (!/^\d{16}$/.test(cardNumber)) {
                alert('Card number must have 16 digits');
                return false;
            }
            const month = parseInt(expiry.split('/')[0]);
            const year = parseInt(expiry.split('/')[1]);
            if (month < 1 || month > 12) {
                alert('Invalid month in the expiry date (1-12)');
                return false;
            }
            if (year < 1 || year > 99) {
                alert('Invalid year in the expiry date (two-digit format)');
                return false;
            }
            if ((month < 1 || month > 12) && (year < 1 || year > 99)) {
                alert('Invalid expiry date (both month and year)');
                return false;
            }
            return true;
        }
    </script>
</body>
</html>
```

Output:



Name: Rishika

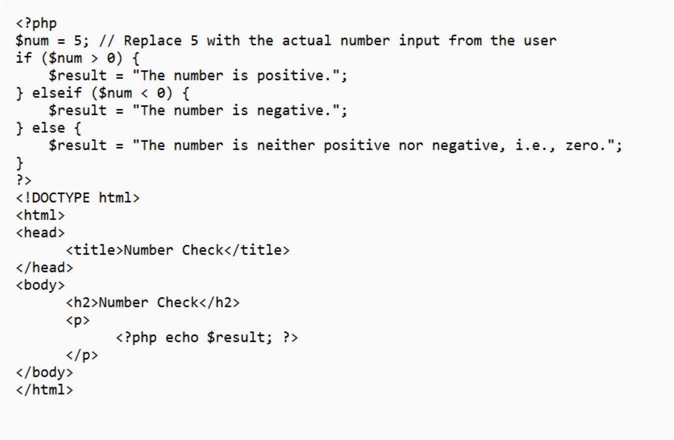
Card Number: 1234123412341234

Expiry Date (MM/YY): 09/23

Submit

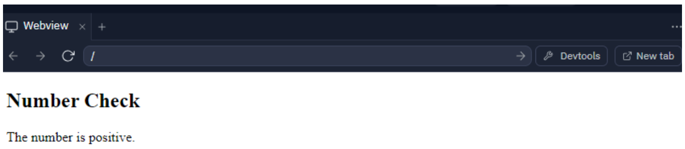
35. Using PHP, check if a given integer is positive, negative or neither (i.e.zero)

Code:



```
<?php
$num = 5; // Replace 5 with the actual number input from the user
if ($num > 0) {
    $result = "The number is positive.";
} elseif ($num < 0) {
    $result = "The number is negative.";
} else {
    $result = "The number is neither positive nor negative, i.e., zero.";
}
?>
<!DOCTYPE html>
<html>
<head>
      <title>Number Check</title>
</head>
<body>
      <h2>Number Check</h2>
      <p>
            <?php echo $result; ?>
      </p>
</body>
</html>
```

Output:

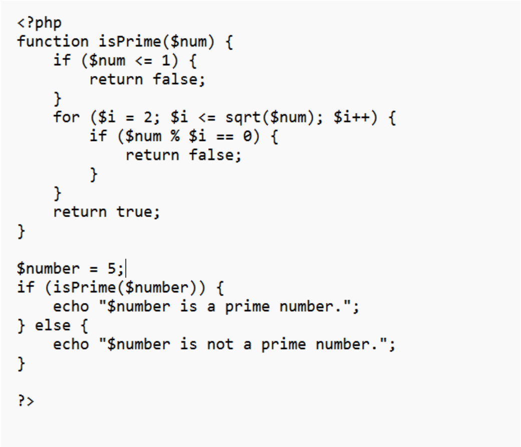


Number Check

The number is positive.

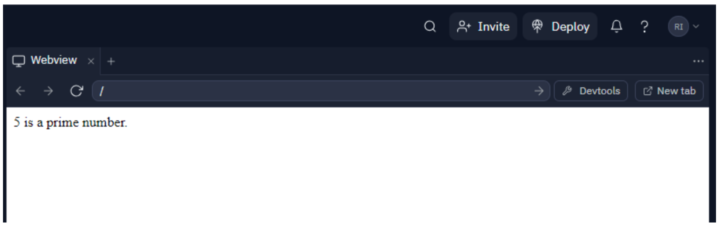
36. Using PHP, check if a given positive integer is prime or not.

Code:



```
<?php
function isPrime($num) {
    if ($num <= 1) {
        return false;
    }
    for ($i = 2; $i <= sqrt($num); $i++) {
        if ($num % $i == 0) {
            return false;
        }
    }
    return true;
}

$number = 5;
if (isPrime($number)) {
    echo "$number is a prime number.";
} else {
    echo "$number is not a prime number.";
}

?>
```

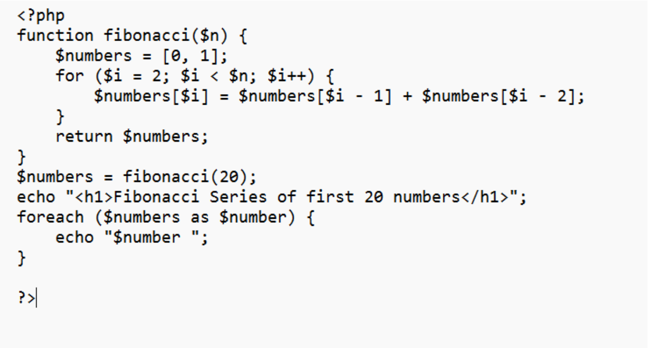
Output:



5 is a prime number.

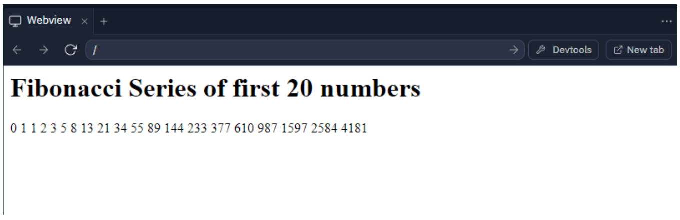
37. Using PHP, calculate and print the first 20 numbers of the Fibonacci series.

Code:



```php
<?php
function fibonacci($n) {
    $numbers = [0, 1];
    for ($i = 2; $i < $n; $i++) {
        $numbers[$i] = $numbers[$i - 1] + $numbers[$i - 2];
    }
    return $numbers;
}
$numbers = fibonacci(20);
echo "<h1>Fibonacci Series of first 20 numbers</h1>";
foreach ($numbers as $number) {
    echo "$number ";
}

?>
```

Output:



Fibonacci Series of first 20 numbers

```
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181
```